



HAL
open science

Unified matheuristic for solving rich vehicle routing problems

Rahma Lahyani

► **To cite this version:**

Rahma Lahyani. Unified matheuristic for solving rich vehicle routing problems. Signal and Image processing. Ecole Centrale de Lille; Université de Sfax. Faculté des sciences économiques et de gestion, 2014. English. NNT: 2014ECLI0011 . tel-01124339

HAL Id: tel-01124339

<https://theses.hal.science/tel-01124339>

Submitted on 6 Mar 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre: 2 4 5

ÉCOLE CENTRALE DE LILLE

THÈSE

présentée en vue d'obtenir le grade de

Docteur

en

Automatique, Génie Informatique, Traitement du Signal et Image

par

Rahma Lahyani

Doctorat délivré simultanément par l'École Centrale de Lille et la Faculté Des Sciences Économiques et de Gestion de Sfax dans le cadre d'une cotutelle internationale de thèse

Une matheuristique unifiée pour résoudre des problèmes de tournées de véhicules riches

Unified matheuristic for solving rich vehicle routing problems

Soutenue le 13 juin 2014 devant le jury d'examen:

Rapporteur	Bernard Gendron	Université de Montréal
Rapporteur	Saoussen Krichen	Université de Jendouba
Président	Saïd Hanafi	Université de Valenciennes
Examineur	Jorge E. Mendoza	Université Catholique de l'Ouest
Examineur	Nenad Mladenovic	Brunel University
Directeurs	Habib Chabchoub	Université de Sfax
	Mahdi Khemakhem	Université de Sfax
	Frédéric Semet	Ecole Centrale de Lille

Thèse préparée au sein des laboratoires LAGIS et LOGIQ
École Doctorale SPI 072 (EC Lille)
PRES Université Lille Nord-de-France

*for me,
for my parents,
and for my brother and sisters.*

Acknowledgments

This research project would not have been possible without the help and support of many people, in particular my three very complimentary advisers. I would like to express my gratitude to Pr. Frédéric Semet. He taught me to define clear research objectives, to mark the boundaries of what can be achieved in limited time, and to keep trying until things are perfect. I am also thankful to him for opening the doors of the Ecole Centrale de Lille, and giving me the opportunity to join the LAGIS team. I would like also to thank Dr. Mahdi Khemakhem for giving me the taste of operations research in his excellent master degree course on linear programming and metaheuristics. With him I learned that research was not exclusively about brilliant ideas, but also about methodology, attention to details, and perseverance. A special thank goes to him for his training lessons in the programming language *C*, for his patience and for introducing me to the world of researchers and showing me that fun and science can go well together, but most importantly for the simple fact of being a friend and believing in me. I would also like to thank Pr. Habib Chabchoub for believing in me from the first months, for his valuable advice before taking the decision to start this dissertation, and for his support during the moments of doubts.

I would like to thank Pr. Saoussen Krichen for being part of my committee, for reviewing the present dissertation and for her valuable advice on research and life. I also would like to thank Pr. Bernard Gendron for accepting the invitation to be part of my committee and reviewing the present dissertation, and also for offering me the opportunity to pursue my investigation with his team. I would thank him for receiving me in CIRRELT and for the fruitful (and fun) discussions we have made. In addition, I would like to thank Pr. Saïd Hanafi for his participation in the jury, his advices along the way and our always interesting discussions in congresses and conferences. I also thank Dr. Jorge Mendoza for being part of the jury and for his inspiring determination. Finally, I would like to thank Pr. Nenad Mladenovic not only for being part of my jury but also for sharing his experience in different research areas, which gave me a broader vision for my research career choices. I would like to thank Pr. Lenadro C. Coelho and Pr. Gilbert Laporte whom I visited in Montreal, and had the opportunity to continue working with after I returned to France. I thank Leandro for sharing his expertise on routing and metaheuristics, for always being here to discuss ideas even when he was busy and also for offering me the opportunity to pursue my investigation with his team.

Acknowledgments

Going through this three years journey would not have been possible without the help and support of friends and colleagues on both sides of the Mediterranean. A special thank goes to Yang, Julie, Baisi, Henri, Fanny who kindly accepted me in their office and with whom I share a good meal and a good laugh. All my gratitude goes to Ramla, Manel, Manel, Maissa, Ryma, Abir, Fattoum who made me feel at home when missing Tunisia and for their good friendship inside and outside the University. Life would not have been possible without the support of my family, my sisters Fatma, Amina, Hanen, my brother Mohamed, my nephews Abdou, Doudi and Maymou who bring me much joy and happiness in my life. Finally, I would like to dedicate this thesis to my parents Wassila and Abdelkarim for their love, confidence, prayers and support throughout my studies and life.

Contents

Acknowledgments	iii
Introduction	1
1 Rich vehicle routing problems: from a taxonomy to a definition	11
1.1 Introduction	12
1.2 RVRP Taxonomy	13
1.2.1 Taxonomy	14
1.2.1.1 Scenario characteristics (SCs)	16
1.2.1.2 Problem physical characteristics (PPCs)	20
1.3 Taxonomy analysis	26
1.4 Conclusions	35
2 Models for a rich vehicle routing problem with compartments	37
2.1 Introduction	38
2.2 Mixed integer linear program for the MDMC MCm-VRPTW	39
2.2.1 Problem description	39
2.2.2 Mathematical model	44
2.3 Reformulation based on Dantzig-Wolfe decomposition	49
2.3.1 Decomposition principle	49
2.3.2 DW reformulation for the MDMCMCm-VRPTW	52
2.4 Pricing subproblem formulation	55
2.5 Conclusions	57
3 Matheuristic design for a multi-constrained TSP with profits	59
3.1 Introduction	60
3.2 Describing available unified methods for VRPs	61
3.3 Matheuristic approach	62
3.3.1 Perturbation phase	64
3.3.2 Improvement phase	66
3.4 Main features of the matheuristic approach	68
3.4.1 Multi-start constructive heuristic	68
3.4.2 Neighborhoods	69
3.4.2.1 Routing neighborhoods	70
3.4.2.2 Loading neighborhoods	77

3.4.3	Route feasibility check	79
3.4.3.1	Loading feasibility check	79
3.4.3.2	Temporal feasibility check and optimization	80
3.5	Conclusions	81
4	From methods to implementation and results	83
4.1	Design of data structure classes	84
4.1.1	Challenges	84
4.1.2	Data structures classes	85
4.2	Computational experiments	89
4.2.1	OP instances	90
4.2.2	OPTW instances	91
4.2.3	Experimental results for the RPTP	96
4.2.3.1	New testbed	96
4.2.3.2	Sensitivity analysis	101
4.3	Conclusions	103
5	Column generation heuristic	105
5.1	Introduction	105
5.2	Column generation heuristic components	106
5.2.1	Initial solution heuristic	106
5.2.2	Accelerating strategies for solving the pricing problem	109
5.2.3	Post-processing procedure	110
5.3	Computational experiments	111
5.3.1	CVRP instances	111
5.3.2	MCMCm-VRPTW instances	113
5.4	Conclusions	115
6	Real case study: The collection of olive oil in Tunisia	117
6.1	Introduction	118
6.2	Mathematical description of the problem	120
6.2.1	Notation	120
6.2.2	Mathematical model	121
6.2.3	Valid inequalities	122
6.3	Branch-and-cut algorithm	124
6.4	Computational experiments	125
6.4.1	Instance generation	125
6.4.2	Computational results	128
6.5	Conclusions	131
	Conclusions	133
	References	137
	Résumé	167

Abstract	171
Electronic appendix	173

List of Figures

1	Operations research cycle (Rardin, 1998)	2
2	Contributions summary	7
1.1	Cluster analysis	31
2.1	The MDMCMCm-VRPTW assumptions	41
2.2	The related problems to the MDMCMCm-VRPTW and their inter-connections	42
2.3	Example of solution for the MDMCMCm-VRPTW	44
2.4	Block-angular matrices for K blocks	50
3.1	Transformation process for solving rich and basic VRPs with a unified matheuristic, adapted from Røpke and Pisinger (2006a)	63
3.2	Representing the VNS* general behavior	66
3.3	Similarity removal heuristic	72
3.4	Spatio-temporal insertion heuristic	76
3.5	Example of a solution based on the Quadratic Multiple Knapsack Problem with Conflicts	78
4.1	Solution representation using the order class and the compartment class	88
5.1	Column generation based heuristic	107
6.1	Map of Tunisia pinpointing producers and regional offices locations (Source: Google Maps, accessed March 2014)	119
6.2	Geographical locations of the producers around the region of Sfax (Source: Google Maps, accessed March 2014)	127
6.3	Geographical locations of the producers around the region of Kairouan (Source: Google Maps, accessed March 2014)	127

List of Algorithms

1	Variable Neighborhood Search* (VNS*)	65
2	Generic structure of removal heuristics	71
3	Generic structure of insertion heuristics	74
4	Get_Compartment heuristic	80
5	Heuristic to construct an initial set of columns	108
6	Branch-and-cut algorithm	126

List of Tables

1.1	A Taxonomy of RVRPs	15
1.2	Selected papers devoted to pure routing problems	28
1.3	Selected papers devoted to combined routing problems	29
2.1	Notations used in the description of the MDMCMCm-VRPTW	45
4.1	Details of the order class	86
4.2	Details of the compartment class	87
4.3	Details of the solution class	87
4.4	Comparison of VNS* to the state-of-the-art methods on Tsiligirides (1984) data sets	91
4.5	Comparison of VNS* to the state-of-the-art methods on Chao et al. (1996) data sets	91
4.6	Average results of the slow methods on Solomon’s data sets	95
4.7	Average results of the fast methods on Solomon’s data sets	95
4.8	Average results of the state-of-the-art methods on Cordeau’s data sets	95
4.9	Incompatibility configurations in generated data sets	97
4.10	Average results of VNS* on RPTP data sets	99
4.11	Average results of VNS* variants on RPTP data sets	100
4.12	Effectiveness of the insertion and removal heuristics	103
5.1	Performance of the column generation heuristic on the CVRP instances	112
5.2	Classes description	113
5.3	Results of the column generation heuristic on the MCMCm-VRPTW instances	114
6.1	Producers locations in Figures 6.2 and 6.3	128
6.2	Configurations of the real-world instances	129
6.3	Summary of computational results of the test instances with respect to several configurations	130
6.4	Relative savings compared to existing situation	131

Introduction

Once upon a time ...or back in 2009 to be more precise, I discovered thoroughly the transportation management system in the courses "*Modeling and Optimization*" and "*Logistics*" held respectively by Frédéric Semet and Habib Chabchoub as part of my master program "*Operations Research and Production Management*" at the University of Sfax. I was immediately interested in supply chain management, a subject that influences considerably the performance of the whole system. I was particularly fascinated by the possibility of translating the complexity of real-life transport systems into mathematical models, often easy to state and difficult to solve. Through these courses, I discovered *the traveling salesman problem* and *the vehicle routing problem*, which have attracted the interest of many researchers over more than 50 years and are still among the most prominent combinatorial optimization problems studied until today. This hands-on, I followed later on, over the period winter 2009 - autumn 2010, an intensive training entitled "*Algorithms and Data Structures applied to Operational Research*" held by Mahdi Khemakhem. This training was part of the Ph.D. preparation courses organized by the *LOGIQ* laboratory. The training offered insight into metaheuristics solution approaches applied to several combinatorial optimization problems. We considered and applied efficient algorithms combining neighborhoods and strategy guidelines to obtain high quality solutions with a little computational effort. These courses enabled me to obtain a more mature perception of transportation problems and to express my interest for these problems for the first time through my master research project in 2010. In this project, I proposed a generic mathematical model covering different scenarios for a multi-constrained vehicle routing problem with compartments. Then, I mostly applied an exact branch-and-bound algorithm to solve it. I succeeded to apply this approach only for instances of limited sizes because of the combinatorial aspect of the problem. Therefore, with some knowledge about exact and approximate solution approaches and with a lot of motivation, I decided to explore more thoroughly this research area in my Ph.D. thesis, since January 2011.

Rich vehicle routing problems as research field

Satisfying efficiently requirements for transportation systems is a central aspect in logistics. It typically affects strategic and long term horizon decisions involving large capital investments. But it conditions also the success of medium and short term horizon operations. Modeling real-life transportation problems as mathematical programs has a deep impact on the optimization process. Over simplifications may omit relevant features of the problem or may even generate solutions and recommendations which cannot be converted into feasible real-life decisions. Similarly, when all details with respect to the transportation problem are included, the resulting model may turn to be intractable. The proposed mathematical models should then maintain a balance between generality and exhaustiveness. Figure 1 illustrates the operations research cycle gathering the problem modeling, the model solution and the solution interpretation so as to change the original problem through decisions. Finally, one has to ensure that a final solution addresses the core problem properly.

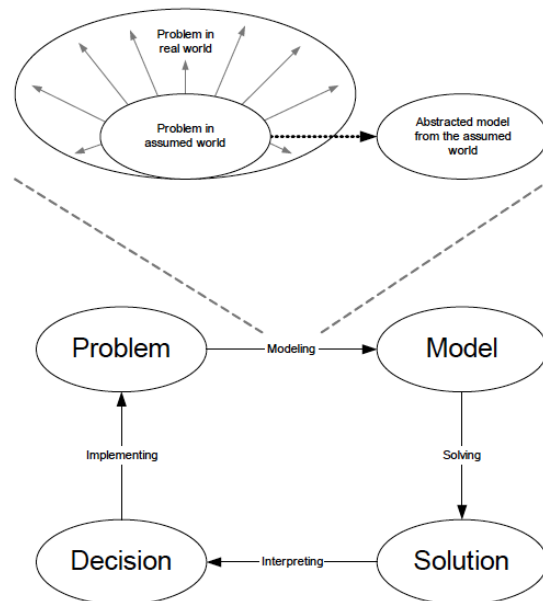


Figure 1: Operations research cycle ([Rardin, 1998](#))

The problems considered in this thesis arise in the field of freight road transportation in many real-life industrial contexts. They correspond often to complex combinations of assignment, loading, scheduling and routing decisions. All the decisions are inter-correlated with the objective of optimizing the transportation resources while satisfying the customers requirements. Because many features have

to be considered as well as different types of decision, such transportation problems are usually considered as "*rich*". A rich transportation problem may be considered as a multi-attribute problem reflecting the complexities of the real-life transportation systems by combining various challenges revealed daily. The most general case of transportation problems can be modeled as a capacitated vehicle routing problem (CVRP). Introduced more than fifty years ago (Laporte, 2009), the Vehicle Routing Problem (VRP) is one of the most prominent and widely studied combinatorial optimization problem. We conduct a simple query on *Scopus* using "*vehicle routing*" as search words over a period of 39 months, from January 2010 to March 2014. This exact phrase was searched in "*Article Title, Abstract, Keywords*" field options to remove irrelevant studies. The search resulted in 3570 publications with a peak in 2013 with up to 1000 publications. These publications include 1858 conferences papers, 1692 journal articles, 13 book chapters and 7 books. The growth rate of the VRP literature requires necessarily a systematic way to classify the emergent and new variants of VRPs. Many surveys exist, describing specific VRP variants and displaying the similarities and the differences among the contributions. We refer, for example, to (Ritzinger and Puchinger, 2013) for the stochastic VRP, (Pillac et al., 2013) for the dynamic VRP, (Archetti and Speranza, 2012) for the split VRP, (Campbell and Wilson, 2014) for the periodic VRP. Nonetheless, this observation enables us to identify existing gaps in the literature dealing with VRPs. There is a lack of studies dealing with the emergent research area, often known as Rich VRPs (RVRPs), in spite of the several surveys for single-variant VRPs. The class of RVRPs requires a taxonomy providing a framework to identify those which can be considered as rich among the numerous variants of the VRPs. The purpose of the first part of the thesis is to provide a generic taxonomy for the RVRPs literature with respect to relevant real-life issues.

Unified framework methodology as a research incentive

Many solution methods have been designed for the VRPs: exact methods, which provide the optimal solution of a model, and approximate approaches, which produce high quality solutions within a reasonable amount of time at the expense of optimality. Since most of the problems tackled are *NP*-hard, exact methods are seldom applicable in real-life contexts. Such cases require often a solution of a large and multifaceted VRPs within a short computation time. An alternative to exact

methods would be to combine exact and metaheuristic algorithms. This emergent approach, referred to as *matheuristics* or *model-based metaheuristics*, seems to be a very promising path towards the solution of rich combinatorial optimization problems. Matheuristics take advantage from synergy between approximate and exact solution approaches and often lead to considerably higher performance with respect to solution quality and running time. Puchinger and Raidl (2005) classify these methods into two main categories: *collaborative combinations* and *integrative combinations*. The collaboration consists in executing exact and heuristic algorithms sequentially or in parallel while exchanging information between both. However, no algorithm is contained in another. At the opposite, the integration requires the definition of a subproblem embedded into a master problem. The *column generation* algorithm based on metaheuristics is a good example for this category. It consists in using the value of the dual bounds for assessing the quality of admissible solutions obtained by approximate approaches. We follow this approach in this thesis.

Previously proposed solution methods for VRPs may often be adapted to other problem to a certain degree while the real-life problems bring new methodological challenges because of their variety. Such variety may affect both the problem model and the proposed solution method. In addition, RVRPs always combine conflicting decisions and objectives. Solving these problems iteratively by considering subset of features and/or decisions may lead to unfeasible decisions. Therefore, proposing new solution approaches for each specific real-life problem independently is no longer a reasonable choice neither for researchers nor for practitioners. Indeed, commercial softwares for transportation systems must be able to face a variety of real-life problems and respond to the multiple combinatorial issues that may arise. These statements motivate the development of a unified method to provide good quality solution approaches for a wide class of rich and basic VRPs. The number of published studies that propose unified solution approaches addressing several variants of routing problems has increased significantly, see e.g. (Røpke and Pisinger, 2006b, Subramanian et al., 2011, Derigs and Vogel, 2014, Vidal et al., 2013b, 2014). As stated, the methodological purpose of this thesis is to propose a flexible column generation based matheuristic to solve RVRPs. The proposed method may be used to solve a set of basic and rich VRPs. The matheuristic proposed relies on the variable neighborhood search (VNS), (Mladenovic and Hansen, 1997), since it includes the desirable features of a metaheuristic (Cordeau et al., 2002b). The simple version of the VNS is based on exploiting neighborhood structures iteratively. This feature offers a considerable flexibility when it comes to extension as well as specialization of the matheuristic.

In this thesis several problems were selected as testbed for reasons explained at the beginning of each corresponding chapter. All these variants can be viewed as extensions of the Capacitated VRP (CVRP) or/and of the traveling salesman problem. The general problem we tackle is the multi-depot multi-compartment multi-commodity vehicle routing problem with time windows (MDMCMCm-VRPTW). The main variants addressed are: the multi-compartment profitable tour problem with time windows and incompatibility constraints denoted by (RPTP), the orienteering problem (OP), the orienteering problem with time window (OPTW), the multi-product, multi-period and multi-compartment vehicle routing problem (MPPC-VRP) and the multi-compartment vehicle routing problem (MC-VRP).

Contributions details

Motivated by a case study dealing with heterogeneous goods, we examine a variant of the CVRP arising when multi-compartment vehicles are required to ensure the transport of multiple products under physical and temporal restrictions. In this problem, a customer may have different orders, each corresponding to one single product. Since orders may be delivered by different vehicles, this introduces some restricted splitting policies and leads to increase the problem difficulty. We propose to design and implement a unified matheuristic framework to solve this challenging VRP, and show how it can be used to solve related rich VRPs. The main contributions of this thesis are sketched by Figure 2.

Taxonomy : Our first contribution consists in defining the hierarchical attributes considered when building the taxonomy and identifying the related recent and relevant surveys and studies. We survey and classify 41 papers describing practical cases and addressing different issues related to RVRPs according to the taxonomy built. Finally, we conduct a cluster analysis on the selected papers that leads us to provide a discriminating definition for this class of problems.

Flexible and unified method : This thesis contributes to the solution of a rich VRP and some of its variants through a column generation based matheuristic. The column generation resolution requires the definition of a master problem and sub-problems, one for each vehicle. The overall performance relies on the algorithm used for them. A first methodological contribution of the thesis is the design of generic data structures which facilitates the flexibility of the code. We design four main data structures describing respectively the problem, the customer order, the

compartment and the solution. Next, we propose a VNS matheuristic, based on approximate routing neighborhoods and exact loading neighborhoods to solve the MDMCMCm-VRPTW with a single vehicle. Routing neighborhoods are based on the *Ruin and Recreate* paradigm, (Schrimph et al., 2000). The current solution is partially destroyed, and then repaired using adapted insertion and removal heuristics. The performance of the VNS matheuristic is assessed on instances including up to 288 customers for the OP and the OPTW literature and on more complex generated instances. It produces high quality solutions (less than 3% from the optimal solution values) within very short computational time (few seconds). The column generation based method (detailed in chapter 5) solves the master problem heuristically thanks to the embedded VNS matheuristic. At each iteration, single vehicle routes generated by the VNS are included in the pool of routes. Then a new solution combining some of these routes is identified, which leads to another call to the VNS procedure.

A post-processing step solving to optimality an NP -hard assignment problem is executed to eliminate dominated routes. The efficiency of the algorithm is assessed on instances for the CVRP and the MCVRP. The obtained results are competitive. By validating the proposed framework, we contribute to the research in the RVRP area where such solution algorithms have not been considered before.

Compartment assignment problem : Considering multi-compartment vehicles in a routing problem further increases the difficulty of the problem addressed. In such cases, the routing and the loading problems have to be optimized jointly. However, we examine separately the loading problem associated with MC-VRP and referred to the *compartment assignment problem* (CAP). We propose and solve to optimality two mathematical models for the CAP: a *Quadratic Multiple Knapsack Problem with Conflicts* and a *Linear Multiple Knapsack Problem with Conflicts*. These models are coupled as exact loading neighborhood to the VNS matheuristic to optimize the loading problem of the current solution.

Modeling and solving a real-life case study : This thesis was initiated through a cooperation with industrial partners facing the problem of collecting olive oil in Tunisia. The harvest must be picked up from different producers, geographically scattered over the center of Tunisia, during a set of periods and with respect to incompatibility restrictions. The relatively small size of the real-life instances encouraged us to solve the problem to optimality. We propose a mathematical

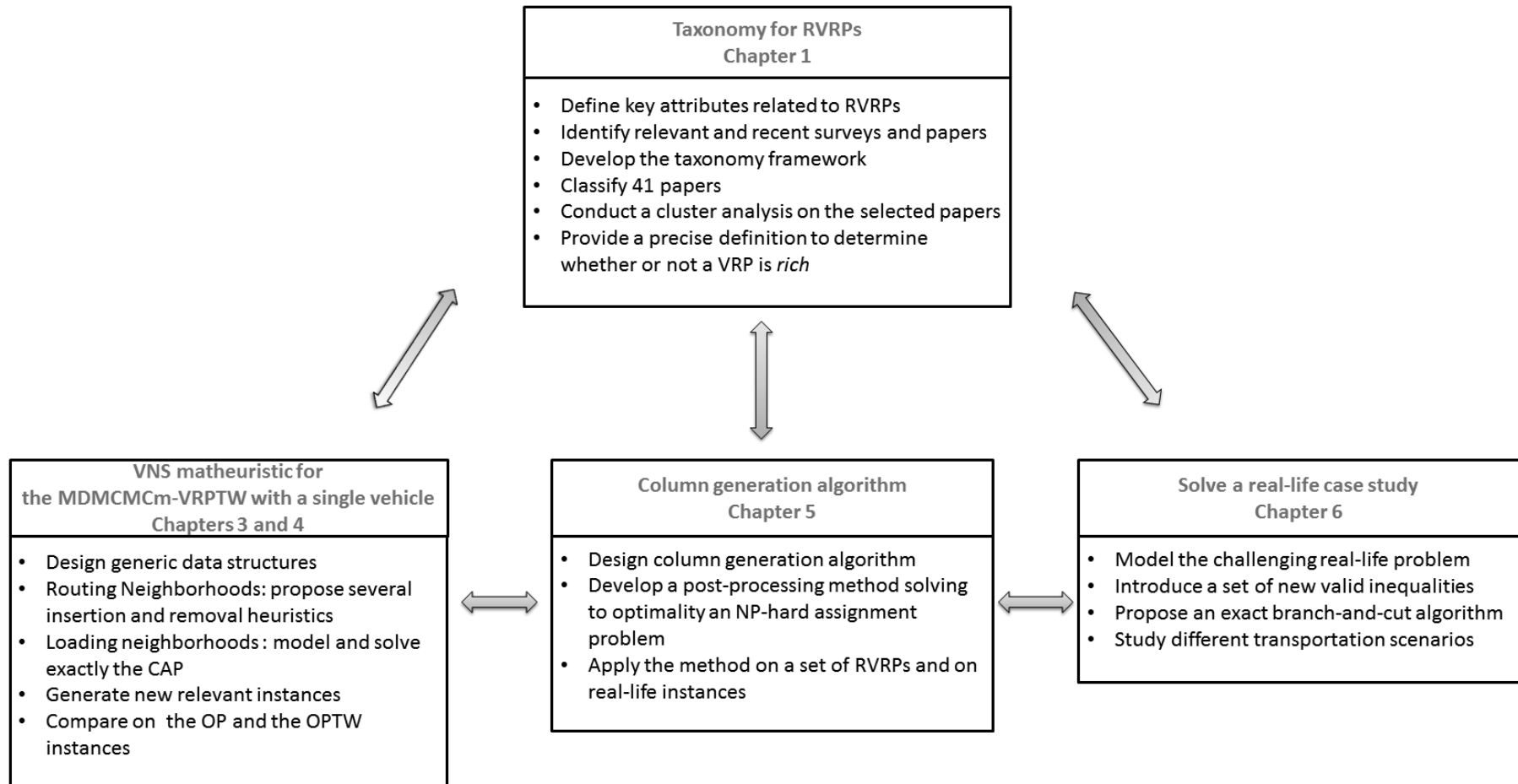


Figure 2: Contributions summary

formulation for the problem along with a set of new and known valid inequalities and routing and assignment cuts. Based on the work of (Coelho and Laporte, 2013a), we propose an exact branch-and-cut algorithm. We obtain considerable improvements when compared to the current plans designed manually by the industrial partners and we provide an automated way of making routing and allocation decisions more efficiently. Additionally, we generate realistic data sets and conduct further experiments to evaluate different transportation scenarios and to highlight the advantages of buying multi-compartments vehicles.

Outline

Being divided into six parts, the structure of this document matches with the research work described above. Chapter 1 presents an overview for many relevant VRP classes, then provides a definition for RVRPs. The remaining chapters describe the methodological contributions of this thesis. Chapter 2 is about the MDMCMCm-VRPTW. We formally introduce the problem. Then, we propose a mixed integer linear programming formulation and an integer model based on the Dantzig-Wolfe decomposition. Chapter 3 is devoted to the description of hybrid solution approaches to solve the MDMCMCm-VRPTW with a single vehicle. We give a general presentation of the VNS method forming the core of our solution approach. Sequential and collaborative combinations of a set of routing and loading neighborhoods are presented next. We focus on the description of efficient removal heuristics for shaking. For customers re-insertion and route improvement we propose several insertion heuristics as well as exact loading neighborhoods. In Chapter 4 we move from the method description to the method implementation. We analyze the challenges encountered while developing the VNS method and provide a simple and effective data structure which may be easily generalized for other VRP classes. Later, we assess the efficiency of the proposed matheuristic on a set of single-vehicle routing instances with profits from the literature and on a set of new generated instances. We provide a sensitivity analysis distinguishing the main components for increasing the performance of the VNS. The column generation based algorithm is described in Chapter 5. We present its different components and further provide computational results on a selection of RVRPs including the instances from the real-life application. The latter is described in Chapter 6. The formulation is provided and strengthened by the introduction of a set of known and new valid inequalities. We examine their impact on the algorithm efficiency. To better support the managerial process of the industrial partner which is currently on the verge of buying new vehicles, we gener-

ate a set of instances with different fleet composition describing the possible future scenarios. Finally, overall conclusions and perspectives are drawn in the last chapter of the thesis. In Appendix A, we report detailed computational experiments carried out in this thesis.

Chapter 1

Rich vehicle routing problems: from a taxonomy to a definition

The purpose of this chapter is twofold: i) to provide a comprehensive and relevant taxonomy for the RVRP literature; ii) to propose an elaborate definition of RVRPs. To this end, selected papers addressing various cases are classified using the proposed taxonomy. Once the articles have been classified, a cluster analysis based on two discriminating criteria is performed and leads to the definition of RVRPs.

Part of this work was presented at the *First IEEE International Conference on Logistics Operations Management (GOL'2012)*, (Lahyani et al., 2012). An article based on this chapter was submitted for publication: R. Lahyani, M. Khemakehm, F. Semet. Rich vehicle routing problems: From a taxonomy to a definition. *European Journal of Operational Research*, submitted for publication.

1.1 Introduction

The Vehicle Routing Problem (VRP), introduced by [Dantzig and Ramser \(1959\)](#), is a central problem in operations research applied to transportation sciences. Over the last three decades, the number of academic publications on the numerous variants of the VRP has increased extensively (see ([Eksioglu et al., 2009](#))). These studies can be roughly divided into theoretical papers providing mathematical formulations and exact or approximate solution methods for academic problems and case-oriented papers. Several taxonomies and surveys devoted to the VRP have appeared, e.g., ([Bodin, 1975](#), [Bodin and Golden, 1981](#), [Desrochers et al., 1990](#), [Laporte and Osman, 1995](#)) who provided a bibliography of 500 studies. More recently, [Laporte \(2009\)](#) reported on the last fifty years of academic vehicle routing from a historical perspective and [Eksioglu et al. \(2009\)](#) presented a taxonomy for the VRP literature. Many books or book chapters have been devoted to the VRP, its variants, and to exact and heuristic algorithms, see, e.g., ([Toth and Vigo, 2002](#), [Cordeau et al., 2007](#), [Golden et al., 2008](#)).

The most elementary VRP considered in the literature is the CVRP. Geographically scattered customers have demands for a homogeneous product. They have to be served by identical vehicles with a limited capacity based at one depot. The CVRP aims to determine a set of vehicles routes of minimum total cost over a single period such that: i) each route starts and ends at the depot; ii) each customer is served by only one vehicle; iii) the total demand on each route does not exceed the vehicle capacity. Most papers devoted to classical problems focus on idealized models and are motivated by unsolved theoretical problems. Nevertheless, in recent years methodological progress and the development of computer technologies has led to an increasing academic attention to new variants including more complex constraints and objectives. This trend is stimulated by the complex characteristics of real-life VRPs. The families of these extended problems are often called Rich Vehicle Routing Problems (RVRPs). Several works focusing on RVRPs have been published. In particular, two special issues were dedicated to works on rich combinatorial optimization problems ([Hartl et al., 2006](#), [Hasle et al., 2006](#)). Papers by ([Sörensen et al., 2008](#), [Drexler, 2012a](#)) compare the VRPs in academic research versus the VRPs in the real-life and delineate the complexity of real-life VRPs. Based on identified gaps, they emphasize on the necessity of adapting commercial software systems to the evolution of customer needs, and of incorporating more intricate constraints. [Doerner and Schmid \(2010\)](#) present a survey devoted to hybrid matheuristics for RVRPs and identify promising future avenues.

In most papers devoted to RVRPs, the authors claim that the problem addressed is rich, and then focus on the mathematical modeling and on the solution methods. Thus, the definitions of the RVRP are rather vague and not significantly different. For instance, (Pellegrini, 2005, Reyes et al., 2008, Rieck and Zimmermann, 2010, Drexl, 2012a) suggest that the term rich vehicle routing is associated with problems that represent some or all aspects of a real-world application including optimization criteria, constraints, and preferences. Recently, some attempts have been made to propose unified models and algorithms tackling different classes of routing problems, see e.g. (Røpke and Pisinger, 2006b, Subramanian et al., 2011, Derigs and Vogel, 2014, Vidal et al., 2013b, 2014).

There is no precise definition nor criterion that leads to determine whether or not a VRP is rich. Such definition has to rely on a relevant taxonomy which can help to distinguish among the numerous variants of the VRP. Therefore, the objective of this chapter is twofold: i) to provide a generic taxonomy for the RVRP literature with respect to relevant real-life issues; ii) to propose a discriminating definition of the RVRP.

The remainder of this chapter is organized as follows. Section 1.2 describes the taxonomy and introduces the key characteristics considered when it was built. Definitions of the hierarchical taxonomy attributes are provided. In Section 1.3, we survey several papers describing practical cases and addressing different issues related to RVRPs. They are classified on the basis of the taxonomy attributes. A cluster analysis of the selected papers is provided and discussed. Last, a RVRP definition is proposed. Section 1.4 concludes this chapter by discussing some future research avenues.

1.2 RVRP Taxonomy

Creating a taxonomy is an efficient and effective way of consolidating knowledge (Reisman, 1992). It enables not only efficient and effective storage, sorting, and statistical analysis but also knowledge expansion and building (Eksioglu et al., 2009). Several surveys and classifications of the VRP have been used as guidelines for the RVRP taxonomy developed in this chapter. This taxonomy aims to build a relevant framework to classify any RVRP study without going into unnecessary details. It attempts also to highlight the different facets of richness encountered in the literature, and to distinguish RVRPs from standard VRPs.

To "validate" this taxonomy, we have selected papers devoted to RVRPs published since 2006. Real-life and academic works using as benchmarks randomly

generated instances or real data have been considered. Surveys or theoretical articles without testbed have been omitted. Only papers devoted to node routing problems for road transportation have been retained. More than a half of them are based on real-life applications. We also have paid attention to take papers emanating from different countries. Indeed, each country has its geographical and political specificities and its own industrial practices. This may lead to introduce specific constraints on the routing plan. As a result, 41 papers published in different journals and conferences are examined attempting to be as exhaustive as possible. However, we apologize for any unintended omission of some relevant articles.

1.2.1 Taxonomy

In this section, we focus on the description of the taxonomy (see Table 1.1) and on the presentation of its main attributes. The taxonomy was iteratively built, due to the complexity of the distribution planning process. The taxonomy does not intend to highlight all differences between variants of the VRP in order to maintain its comprehensibility and its size. It is instead designed according to central concepts in routing that are often present in industrial applications. More precisely, the attributes mentioned are not necessarily the basic VRP features but are related to characteristics which alter the nature of the problem significantly. The purpose of the taxonomy is not to classify the papers according to all the details but rather to focus on relevant features. Indeed, we face the following dilemma. The omission of relevant variants of problems studied in the literature introduces some bias in the classification. Similarly, deepening the level of details may lead to an unmanageable taxonomy. Hence, we try to maintain a moderate level of granularity for the proposed RVRP taxonomy.

The taxonomy is constructed hierarchically with at most four subclasses. Problems are considered according to the Scenario Characteristics (SCs) and to the Problem Physical Characteristics (PPCs). Under each of these two classes, the most discriminating attributes are listed. They determine whether or not the problem under study can be classified as rich. The taxonomy is organized in an arborescent way with three levels associated with the strategic level, the tactical level and the operational level. Each of them is divided into sublevels. The difference between the three levels depends on the types of decision involved. The strategic and tactical levels are associated with the first branch of the taxonomy, i.e. the SCs. They correspond to the transportation strategy which describes the distribution system and designs its main components. At the strategic level, the company has to decide if the operational plan deals simultaneously with decisions related to different

functions of the supply chain or if transportation planning issues are addressed. For instance, the strategic planning could include decisions related to the locations and the number of depots used. At the tactical level, the order type and the visit frequencies at customers over a given time horizon could be considered. The multi-use of vehicles or the data type leads to other extensions. Although these decisions are not related to daily transport activities, they affect the routing plan significantly.

The operational level is associated with the PPCs. It describes the distribution planning including the vehicle and the driver schedules. At this level, short-term and daily decisions are handled considering each vehicle route. These decisions relate to the routing of goods using the distribution system designed at the strategic and tactical levels. These decisions are based on the characteristics of vehicles, and on specific constraints faced daily. These constraints are specified for a customer, a vehicle, a driver or a road.

Table 1.1: A Taxonomy of RVRPs

1 Scenario characteristics	2 Problem physical characteristics
1.1 Input data	2.1 Vehicles
1.1.1 Static	2.1.1 Type
1.1.2 Dynamic	2.1.1.1 Homogeneous
1.1.3 Deterministic	2.1.1.2 Heterogeneous
1.1.4 Stochastic	2.1.2 Number
1.2 Decision management components	2.1.2.1 Fixed
1.2.1 Routing	2.1.2.2 Unlimited
1.2.2 Inventory and routing	2.1.3 Structure
1.2.3 Location and routing	2.1.3.1 Compartmentalized
1.2.4 Routing and driver scheduling	2.1.3.2 Not compartmentalized
1.2.5 Production and distribution planning	2.1.4 Capacity constraints
1.3 Number of depots	2.1.5 Loading Policy
1.3.1 Single	2.1.5.1 Chronological order
1.3.2 Multiple	2.1.5.2 No policy
1.4 Operation type	2.1.6 Drivers regulations
1.4.1 Pickup or delivery	2.2 Time constraints
1.4.2 Pickup and delivery	2.2.1 Restriction on customer
1.4.3 Backhauls	2.2.2 Restriction on road access
1.4.4 Dial-a-ride	2.2.3 Restriction on depot
1.5 Load splitting constraints	2.2.4 Service time
1.5.1 Splitting allowed	2.2.5 Waiting time
1.5.2 Splitting not allowed	2.3 Time window structure
1.6 Planning period	2.3.1 Single time window
1.6.1 Single period	2.3.2 Multiple time windows
1.6.2 Multi-period	2.4 Incompatibility constraints
1.7 Multiple use of vehicles	2.5 Specific constraints
1.7.1 Single trip	2.6 Objective function
1.7.2 Multi-trip	2.6.1 Single objective
	2.6.2 Multiple objectives

1.2.1.1 Scenario characteristics (SCs)

In this section, we describe the sublevels of the strategic and tactical levels presented in Table 1.1. We briefly define the characteristics of each sublevel and provide some relevant references.

1.2.1.1.a Input data

The uncertainty and the variability of the data over the planning period are key factors for a classification of VRPs. Data can be subdivided into four classes: deterministic, stochastic, static and dynamic.

The deterministic routing problem assumes that the problem parameters are known with certainty while the stochastic data assumes that probability distributions are associated with them. In the Stochastic Vehicle Routing Problem (SVRP), the routes may not be followed as planned. The three most common stochastic parameters studied in the literature are: customers demands, service times and travel times (Hasle and Kloster, 2007). We refer to (Gendreau et al., 1996, Flatberg et al., 2005, Cordeau and Laporte, 2007, Louveaux and Laporte, 2009, Ritzinger and Puchinger, 2013) for focused surveys.

A seminal work on the Dynamic Vehicle Routing problem (DVRP) is due to Psaraftis (1988). In the DVRP, the scheduling plan established at the beginning of the planning period may be adjusted. It allows the possibility of receiving additional information and changing some problems parameters. Then, the problem is resolved. For example, new customer requests may occur during the planning period and must be considered while the vehicles routes are being executed. For recent literature reviews, we refer to (Psaraftis, 1995, Powell et al., 2001, 2003, Larsen et al., 2008, Berbeglia et al., 2010, Pillac et al., 2013).

1.2.1.1.b Decision management components

Supply chain management is a set of approaches utilized to integrate efficiently customers, manufacturers, warehouses and stores. It ensures producing the products and distributing them at the right quantities, to the right location and at the right time in order to minimize system wide costs while satisfying service level requirements (Kaminsky and Simchi-Levi, 2003). Hence, integrating decisions of different functions such as purchasing, inventory control, outsourcing, locating depots, production planning, and distribution management are a practice increasingly followed by many companies (Armentano et al., 2011). Nevertheless, the complexity of these functions may inhibit defining and solving a complete model including all decision

variables. The related extensions of the VRP in the literature are inventory-routing, location-routing, production-routing and vehicle and driver scheduling. [Schmid et al. \(2013\)](#) propose an elaborated survey on relevant extensions of routing problems arising in the context of supply chain management. The authors describe a new family of integrated models considering the interdependencies between different types of decisions.

The service type decision is a strategic level of decision. Going back to the paper by [Dantzig et al. \(1954\)](#), each customer has to be served once and only once. However, profits may be associated with the customer service. In such cases, a subset of served customers must be determined to maximize an objective function which includes the total profit collected. This VRP variant is known as the VRP with profits, see ([Feillet et al., 2005](#)).

[Boudia et al. \(2007\)](#) states that, ideally, any cost reduction effort should encompass a production planning. First introduced by [Glover et al. \(1979\)](#), production and distribution planning aims to determine the quantity produced for each item, the distribution plans and the quantities of each item delivered to each customer. This variant was reviewed by ([Vidal and Goetschalckx, 1997](#), [Sarmiento and Nagi, 1999](#), [Chen, 2004](#), [Adulyasak et al., 2013](#)) while recent works are those of ([Chandra and Fisher, 1994](#), [Armentano et al., 2011](#), [Adulyasak et al., 2012](#)).

In Inventory Routing Problems (IRP), the supplier defines, in addition to the routing plans, the quantities to deliver using inventory levels at consumers to avoid stock shortages. [Fisher et al. \(1982\)](#) and [Bell et al. \(1983\)](#) pioneered this approach when they studied the IRP at Air Products, a producer of industrial gases. Elaborate surveys are provided by ([Campbell et al., 1998](#), [Cordeau et al., 2007](#), [Bertazzi et al., 2008](#)). [Andersson et al. \(2010\)](#) also describe industrial aspects of combined inventory management and routing in maritime and road transportation. They propose a classification and a comprehensive literature review of the current state of the research. [Coelho et al. \(2013\)](#) describe this research area over the last thirty years. They categorize the IRP literature with respect to the structure of the problem and to the availability of information on customer demand. When production management, inventory management and transportation management are combined, the objective is to determine the quantities produced, the quantities delivered to customers and the vehicle schedules while minimizing the total costs. These costs include setup costs, holding costs and distribution costs. Some relevant references are ([Fumero and Vercellis, 1999](#), [Boudia et al., 2007](#), [Boudia and Prins, 2009](#), [Bard and Nananukul, 2009](#), [Coelho et al., 2012a,b](#), [Coelho, 2013](#), [Coelho and Laporte, 2013b,c](#), [2014](#)).

The Location Routing Problem (LRP) aims to determine the location of depots serving customers and the routes rooted at those depots simultaneously. Surveys on the LRP have been proposed by (Balakrishnan et al., 1987, Laporte, 1988, 1989, Berman et al., 1995, Min et al., 1998) and recently by (Nagy and Salhi, 2007).

The simultaneous determination of the vehicle and driver schedules represents a significant trend in transportation management. It implies to elaborate a crew assignment and the associated routes concurrently. The literature devoted to this challenging variant is scarce (Wen et al., 2011). Nevertheless, recent papers have been published e.g., (Haase et al., 2001, Freling et al., 2003, Xu et al., 2003, Goel, 2009, Zäpfel and Bögl, 2008, Wen et al., 2011). Note that the regulation of working hours can be modeled as additional constraints (see the section dedicated to driver regulations).

1.2.1.1.c Depots

In the classic VRP, a single depot is in use which may be quite restrictive in practice. Thus, in real-life applications, there are often multiple depots and vehicles may have different starting and final locations. It may be required to allocate the customers to the appropriate depot. The depots may have different characteristics, regarding of their number, locations and capacities, which may affect the overall costs. Over the last four decades, many papers studied the multi-depots vehicle routing problem, for instance, (Tillman, 1969, Laporte et al., 1984, 1988, Salhi et al., 1998, Surekha and Sumathi, 2011, Vidal et al., 2012, Hemmelmayr et al., 2013, Rahimi-Vahed et al., 2013, Muter et al., 2014).

1.2.1.1.d Operation type

Four classes of routing problems can be distinguished: problems where goods are either delivered or picked-up, problems where goods are loaded and unloaded, problems where goods are loaded on board when the delivery part of the route is completed, and last dial-a-ride problems. The first class corresponds to the classical VRP. The goods are loaded at the depot and then unloaded at customer locations, or pickup tasks are performed at the customer sites and the unloading at the depot. The second class, where goods are transported between pickup and delivery locations, is named Vehicle Routing Problems with Pickups and Deliveries (VRPPD). In the VRPPD, goods can be loaded or unloaded at each customer. In the standard case, the pickup point and the delivery point must be served on the same route. There are many possible extensions to the VRPPD. The VRPPD was divided into

other subclasses in [Parragh et al. \(2008b\)](#). A survey on this research area is due to [Desaulniers et al. \(2002\)](#).

The Vehicle Routing Problems with Backhauls (VRPB) was introduced by [Goetschalckx and Jacobs-Blecha \(1989\)](#). It consists in transporting goods from the depot to linehaul customers and from backhaul customers to the depot. The VRPB arises in various applications like in the grocery industry where supermarkets and shops are considered as the linehaul customers, and grocery suppliers are the backhaul customers ([Toth and Vigo, 2002](#)). Four subtypes of the VRPB were considered and detailed in [Parragh et al. \(2008a\)](#).

The Dial-A-Ride Problem (DARP) refers to the situation where a shipment has to be transported between pre-specified pickup and delivery locations under service restrictions. The DARP arises frequently in health related transportation like patient transportation or blood transportation. Recent solution methods and surveys on the DARP can be found in ([Berbeglia et al., 2007](#), [Cordeau and Laporte, 2007](#), [Parragh et al., 2008b](#), [Paquette et al., 2009](#), [Kirchler and Wolfer Calvo, 2013](#), [Paquette et al., 2013](#)).

The difference between the above described problems is usually expressed through additional constraints. For the last three classes, there are always precedence constraints between the different types of nodes.

1.2.1.1.e Load splitting

In the classical VRP, each customer is served by only one vehicle. The possibility of multiple visits to the same customer characterizes the VRP with Split Deliveries (SDVRP) introduced by [Dror and Trudeau \(1989\)](#). An intermediate level of splitting should also be identified when several products have to be delivered to a customer. In this case, several visits to the same customer may occur, each product being delivered during a unique visit. [Archetti and Speranza \(2007, 2012\)](#) proposed a state of the art for the SDVRP.

1.2.1.1.f Planning period

The distribution plan may be computed over a single period or over several periods. In the Periodic Vehicle Routing problem (PVRP), all the input data is available at the beginning of the planning period. The customer requests are known in advance as well as the possible combinations for the visiting days. At each period, one has to decide which customers are served in this period and which orders are postponed to the next periods. The PVRP was first introduced in the paper of [Beltrami](#)

and Bodin (1974). Since then it was extensively studied and enriched with several variants. A vast literature is dedicated to the PVRP, see e.g., (Cordeau et al., 1997, Francis and Smilowitz, 2006, Hemmelmayr et al., 2009, Wen et al., 2010, Baldacci et al., 2011, Gulczynski et al., 2011, Vidal et al., 2012, Michallet et al., 2014). (Francis et al., 2008, Campbell and Wilson, 2014) have proposed a focused survey on modeling and solution methods.

1.2.1.1.g Multiple use of vehicles

In the VRP with multiple use of vehicles, the same vehicle may perform several trips during the planning period while respecting additional temporal precedence constraints. Introduced by Fleischmann (1990), the multiple use of vehicles was addressed recently. Most papers appear in the last decade, see, e.g., (Petch and Salhi, 2003, Olivera and Viera, 2007, Salhi and Petch, 2007, Alonso et al., 2008, Azi et al., 2010, 2012, Macedo et al., 2011, Cattaruzza et al., 2014). In this variant, it is common to add additional restrictions related to the vehicle use and to decompose the operating costs in loading/unloading costs and variable costs. The variable costs may be related to the vehicle cleaning service between tours as in (Oppen et al., 2010, Oppen and Løkketangen, 2008).

1.2.1.2 Problem physical characteristics (PPCs)

1.2.1.2.a Vehicles

Vehicles with different characteristics better match customer needs related to physical restrictions, to environmental concerns, to specific logistic equipments or to demand variations, see, for example, (Semet and Taillard, 1993, Tarantilis et al., 2003, 2004, Bräysy et al., 2008). The first characteristics of the vehicles considered here are related to the types of vehicle available at the planning period. There are also physical characteristics related to the loading/unloading process. Last, social driver regulations may significantly affect the vehicle routes and are considered then as vehicle characteristics.

Dealing with the fleet composition goes back to the seminal paper by Kirby (1959). Since then, several variants of the Heterogeneous Fleet Vehicle Routing Problem (HFVRP) have been addressed in the literature. Recent surveys have been proposed by Baldacci et al. (2008), Hoff et al. (2010). The primary decision related to the fleet consists to determine the size of the fleet, which is always limited in real life applications.

The most common capacity constraints in freight transportation are expressed in terms of weight, volume, or number of pallets. Several capacity restrictions may be taken into account simultaneously. The vehicles may differ according to their capacities and the same vehicle may have several compartments with different capacities. [Derigs et al. \(2011a\)](#) have proposed a survey on VRP with multiple compartments. The use of multiple compartments is relevant when several products, which must remain separated during transportation, have to be loaded on the vehicles. For instance, multi-compartment vehicles are used, to perform selective waste collection ([Muyldermans and Pang, 2010b](#), [Reed et al., 2014](#)), to distribute food requiring different levels of refrigeration ([Chajakis and Guignard, 2003](#)), to distribute various types of fuel to petroleum companies ([Brown and Graves, 1981](#), [Cornillier et al., 2008b,a](#)) and to transport animals from farms to slaughterhouses ([Oppen and Løkketangen, 2008](#)). The optimization of routes to transport animals from farms to slaughterhouse is known as the livestock collection problem. It is a complex IRP combining animal welfare regulations and production and inventory constraints, and requiring vehicles with compartments.

Recently, loading and routing problems have been addressed jointly. Their combined optimization leads to the Loading Vehicle Routing Problem (LoVRP) dealing with more complex loading constraints than a simple weight or volume restriction. Routes are obtained thanks to the chronological order induced by the loading policy. The most common loading/unloading methods are based on the LIFO (Last-In-First-Out) and FIFO (First-In-First-Out) rules. If the vehicle picks up request i before request j , then it must deliver request j (i) before delivering request i (j) respectively ([Carrabs et al., 2007](#)). Some references devoted to different variants of the LoVRP are ([Doerner et al., 2007](#), [Fuellerer et al., 2009, 2010](#), [Tricoire et al., 2011](#)). In other studies, the loading problem and the routing problem are solved separately. The loading problem is viewed as a Bin Packing Problem ([Cruz Reyes et al., 2007](#)) or as a Tank Truck Loading Problem ([Cornillier et al., 2008b,a, 2009, 2012](#)). [Iori and Martello \(2010\)](#) propose a recent survey on the VRP with loading constraints.

Although a vast literature is devoted to the VRP, constraints related to the legislation on driving and working hours were included infrequently. Such rules are either imposed by the company or legal regulations. Some examples are: daily working periods, number and duration of daily and weekly rest periods, maximum period of driving hours, overtime working hours and wages, etc. Driver regulations aim to provide safe driving, to reduce the number of accidents and to reduce the total costs. In recent years, there has been an increasing number of papers describing VRPs where a limited set of the European Union or United States regulations is

taken into account (e.g., (Zäpfel and Bögl, 2008, Wen et al., 2011, Beaudry et al., 2010, Derigs et al., 2011b, Parragh et al., 2012)). Other studies include all parts of the mandatory legislation and the legal requirements, e.g., (Rancourt et al., 2013, Vidit, 2008) who studied the US regulation in his thesis. (Goel, 2009, Kok et al., 2010, Prescott-Gagnon et al., 2010) consider a VRP with time window respecting all driver rules imposed by the European Union. Recently, Goel and Vidal (2013) consider several sets of driver regulations in the United States, Canada, the European Union and Australia to provide an international evaluation of the impact of different rules on the road safety and on the minimization of transportation costs.

1.2.1.2.b Time related constraints

The VRP with time window (VRPTW) was studied first in case studies by (Pullen and Webb, 1967, Knight and Hofer, 1968, Madsen, 1976). Time window constraints impose that the service at every customer must start and end within a given time window. In the case of hard time window, the vehicle is allowed to arrive before the defined time window and waits until the customer becomes available, but it is not allowed to arrive late. In the case of soft time window, penalties are given for services starting after the allowed time windows. It leads to take into account the travel times between customer locations, the service times at customers and the loading/unloading times at the depot. Defining time window at the depot implies that the earliest departure time and the latest arrival time for each vehicle must lie within the interval time associated with the depot. A variant is the multiple time windows in which one of the time intervals associated with each customer has to be selected. In the last decades, time constrained routing problems have been extensively studied. Bräysy and Gendreau (2005a,b), Kallehaug et al. (2005), Kallehaug (2008) and Gendreau and Tarantilis (2010) propose recent surveys on this problem.

Often, the route duration is limited to a predefined parameter which may be equal to the total driver working hours or to the route access time. Indeed, a road segment may have a limited access given by an interval of time specifying when it can be entered. The time restriction can also be set by the number of customers visited on a route.

1.2.1.2.c Incompatibility constraints

In real-life applications, many (in)compatibility constraints may occur between the problem entities specifically customer, depot, vehicle compartment, products and driver, as it was discussed in Hasle and Kloster (2007). These

(in)compatibilities can be classified in two types depending on the causing factor: physical (in)compatibilities and temporal (in)compatibilities. Some of these constraints are described in what follows.

An extension of the VRPTW is the VRP with temporal dependencies, which includes synchronization and sequencing constraints between customers, vehicles, visits or depots. As [Dohn et al. \(2011\)](#) point out, many practical applications including such constraints were studied. Some examples are: (i) sequencing constraints related to technicians in the Port of Singapore, (e.g., ([Lim et al., 2004](#), [Li et al., 2005](#))); (ii) synchronization between visits in ground handling at airports, (e.g., [Dohn et al. \(2009\)](#)); (iii) coupled time windows for vehicles and schools in the design of school bus transportation services, (e.g., [Fügenschuh \(2006\)](#)).

The temporal incompatibilities are also present as precedence constraints requiring synchronization between tours. The vehicle must visit some pick-up customers before visiting some delivery customers. This chronological order is a key point in the DARP, in the VRPPD, in the VRP with backhauls and sometimes in VRP. For example, in the transportation of livestock from farms to slaughterhouses, ([Oppen and Løkketangen, 2008](#), [Oppen et al., 2010](#)), the health status associated with the farms may enforce an order for the visits. An elaborate and recent survey on vehicle routing problems with multiple synchronization constraints is due to [Drexel \(2012b\)](#).

There are also physical inclusion and exclusion restrictions as discussed in ([Desrochers et al., 1990](#)). It may occur that a customer must (not) be served from a given depot, by a given vehicle (e.g., [Pellegrini et al. \(2007\)](#)) or a specific driver (e.g., [Rieck and Zimmermann \(2010\)](#)). Indeed, the vehicle requirements and the driver qualifications, such as licensing, training for transporting specific materials and knowledge about a geographical area, may limit the compatibility between driver and vehicle. The VRP including incompatibilities between customers and vehicles types is known as the site dependent VRP (see e.g., [Baldacci et al. \(2008\)](#)).

Similarly, the vehicle characteristics or the product specificity may cause vehicle-request incompatibility, see e.g., ([Goel and Gruhn, 2008](#), [Goel, 2010](#), [Ceselli et al., 2009](#)). In the distribution of multi-commodity loads, each vehicle can be used to handle specific types of cargos. Distribution to groceries is a relevant example where different types of food to deliver require vehicles with different temperature levels. In animal transportation, different animal types cannot be transported together in the same compartment (e.g., ([Oppen and Løkketangen, 2008](#), [Oppen et al., 2010](#))). Similarly in a health application, the patient condition may prohibit sharing the ambulance with other patients to avoid the spread of diseases or to allow a medical staff assistance, see e.g., ([Beaudry et al., 2010](#), [Parragh et al., 2012](#)).

1.2.1.2.d Specific constraints

In industrial real-life applications, managers face continuously various and non-standard challenging constraints coming from the problem specificities. For instance, decisions may be constrained by outsourcing resources, by environmental issues, by the prioritization of customers or by cross docking related restrictions. Other constraints are logistic restrictions encountered daily when the vehicle routes are designed. They are described in what follows.

- Outsourcing decisions

In some cases, decision has to be made whether a route is performed using the company resources or outsourcing services. The constraint set is then expanded, and the cost function includes cost terms for outsourced resources. Recent papers taking into account such decisions are due to (Ileri et al., 2006, Zäpfel and Bögl, 2008, Moon et al., 2012, Kovacs et al., 2012, Stenger et al., 2013).

- Environmental protection

During the last years, environmental concerns have been addressed in VRP studies. This research avenue is known as green routing problems in the literature. More precisely, the use of vehicles may be constrained by the release of gas and toxic effluents into the atmosphere. Reducing noise may also be a restriction on the types of vehicles used. Erdoğan and Miller-Hooks (2012) formulated and proposed solution methods for a green VRP arising in the refueling industry. Xiao et al. (2012) extend classical works on CVRP with the objective of minimizing fuel consumption and identified factors causing the variation in fuel consumption. The reader may refer to the book of McKinnon et al. (2012) and to the survey papers by (Sbihi and Eglese, 2007a,b, Demir et al., 2014, Lin et al., 2014) who examined the environmental issues related to vehicle routing and scheduling problem.

- Prioritization of customers

Additional transportation requests may have to be planned to face some unexpected events such as out of stock situations at some customers or vehicle breakdowns. In such a case, priority tags are put on orders to determine which goods must be delivered first (see e.g., Cornillier et al. (2008b, 2009)).

- Cross docking strategy

A cross dock terminal can be considered as a consolidation center having a short term inventory holding capacity. During a short time period incoming

cargos are unloaded and sorted at a distribution center, and then goods are loaded on outgoing vehicles serving given areas, see e.g. , (Lee et al., 2006, Liao et al., 2010, Wen et al., 2009, Santos et al., 2011a,b, Vahdani et al., 2012, Van Belle et al., 2012). Such delivery systems are of interest for industries facing large distribution costs like pharmaceutical and food companies, see e.g., Boysen (2010). Decisions related to the management of cross docking terminals are constrained by the design of the pickup and delivery routes.

- Open routes

A few papers have been devoted to the Open Vehicle Routing Problem (OVRP) introduced by Schrage (1981). In the OVRP, the driver must not return to the depot once the last customer on the route has been served. The route may terminate at a car park or at the driver home. Some relevant applications of the OVRP can be found in the home delivery of packages and newspapers (e.g., Russell et al. (2008)), when deliveries are outsourced to independent contractors or when drivers use their own vehicles. Some recent studies on the OVRP are due to (Tarantilis and Kiranoudis, 2002, Brandão, 2004, Fu et al., 2005, Letchford et al., 2007, Ceselli et al., 2009, Rieck and Zimmermann, 2010).

- Accessibility constraints

Accessibility constraints differ from incompatibility constraints between vehicles and customer locations by imposing the use of different transportation means to serve customers. This corresponds to different real-life cases based on multimodal transportation systems. For example, in Semet and Taillard (1993) some routes are covered by trucks and trailers which have to be uncoupled to serve a subset of customers. Some references describing different truck and trailer routing applications are Gerdessen (1996), Hoff and Løkketangen (2007), Caramia and Guerriero (2010), Derigs et al. (2013), Villegas et al. (2011, 2013).

- Simultaneous vehicles on site

In some industry sectors, many orders per day may have to be delivered at the same customer separately. Since queues may occur, a limit is set on the number of vehicles that are present simultaneously at the same location, e.g., Cruz Reyes et al. (2007). This restriction occurs in the production and the delivery of newspapers where a fixed number of trucks may start their routes at the same time for the state edition, e.g., Russell et al. (2008).

- Cleaning

To satisfy sanitary rules and to prevent contamination while transporting oil, chemicals products, lubricants, cattle food and animals, vehicle cleaning is an essential daily operation. Cleaning may be not necessary for some sequencing of products. Hence, in the VRP model we have to decide when cleaning operations are necessary for the proposed scheduling plan and how much it costs. For example, in (Oppen and Løkketangen, 2008, Oppen et al., 2010), the authors impose the visits of the disinfected farms at the end of the route to avoid unnecessary cleaning.

1.2.1.2.e Objective function

The objectives can be multiple and diverse. The most common objectives include minimizing some or all of these criteria: the total traveled distance, the total time, the total tour cost, the fleet size, and/or maximizing the service quality, the collected profit. When multiple objectives are identified, the different objectives are often in conflict. Hence, adequate algorithms have to ensure some trade-off between them. For a survey on the multi-objective VRPs, we refer to Jozefowicz et al. (2008).

1.3 Taxonomy analysis

Considering the selected papers, we have to mention that some authors first consider a simplified version of a VRP and then tackle a more complicated variant by considering additional constraints. For instance, Parragh (2011) first solves a restricted and simplified version of the heterogeneous DARP. Then Parragh et al. (2012) take into consideration the medical requirements of patients and modeled them as incompatibilities constraints. Similarly, Pellegrini (2005) focuses on a VRP with heterogeneous vehicles and multiple time windows for customers. Later on, in Pellegrini et al. (2007) the authors consider multiple visits to some customers over a periodic horizon, hierarchical objectives, and additional customer requirements.

In Tables 1.2 and 1.3, papers addressing different variants of the VRP are listed. Table 1.2 is devoted to papers addressing pure routing problems. Therefore, the sublevels of the level decision management component (1.2) are omitted. In Table 1.3, we consider papers that combine routing vehicles with other strategic or tactic facets. Each article is classified according to the attributes defined in the above taxonomy (see Table 1.1). The subclasses and branching classes are shaded. When the attribute is present in the corresponding paper, 'X' is reported in the associated cell. The column headed Number provides the numbers, we assigned, to each se-

lected paper. In what follows, we give the number of the selected paper into square brackets. The columns headed Formulation, Real case study indicate whether or not the article includes a mathematical formulation and/or a set of instances based on a real-life application. In the column headed Rich, we report whether or not the authors consider the problem under study as a RVRP. The last column headed Method refers to the types of designed method(s).

In Tables 1.2 and 1.3, all described attributes are present in one paper at least. This shows that no unnecessary node has been introduced in the taxonomy. In selected papers, the routing based problems with pickups or deliveries were studied more deeply than the other variants. Indeed, 61% of the papers selected addressed routing issue only (1.2.1), and 70.7% of them are devoted to pickup or delivery (1.4.1) problems. At the opposite, little attention has been paid to routing problems dealing with other decisions, such as location (1.2.3), driver scheduling (1.2.4) and production (1.2.5). This suggests some promising future research avenues. Even if stochastic and dynamic problems are quite challenging, they constitute almost 15% of the selected papers which illustrate the key role played by these problem characteristics. Different types of temporal and physical incompatibilities between the basic entities (2.4) are present in the considered papers (63.4%) as well as specific restrictions (2.5) related to real-life applications (53.7%). Moreover, the time constrained routing problems have been studied intensively. In more than half of the selected papers, the VRPs tackled take into account time window restrictions for customers and/or road accesses (2.2.2). Opening hours for depots (2.2.3) are defined in almost 40% of the papers. Last, it is noteworthy to mention that the social constraints such as legislation rules on driving and working hours are considered in 22% of the papers (2.1.6). This illustrates that social driver regulations increasingly are taken into account in the VRP literature since 2006.

In 56% of the papers, mathematical formulations are provided even if approximate methods are then designed to solve the problem under study. The models typically aim to provide a complete description of the problem. With respect to the solution methods, exact algorithms are proposed in two papers while heuristics and metaheuristics are described in the remaining 39 papers. This is due to the ability of approximate methods to find near-optimal solutions for large instances of complex problems.

Table 1.3: Selected papers devoted to combined routing problems

Number	Paper	1.1	1.12	1.14	1.2	1.21	1.22	1.23	1.24	1.25	1.3	1.32	1.4	1.41	1.42	1.43	1.44	1.5	1.51	1.6	1.62	1.7	1.72	2
1	Armentano <i>et al.</i> 2011																							
2	Bard and Nananukul 2009					x																		
4	Boulic <i>et al.</i> 2006																							
5	Boudia and Prins 2009																							
6	Boudia <i>et al.</i> 2007																							
8	Cornillier <i>et al.</i> 2009																							
9	Cornillier <i>et al.</i> 2008																							
17	Hemmelmayr <i>et al.</i> 2010																							
20	Kok <i>et al.</i> 2010																							
21	Lin and Kwok 2006																							
26	Oppen and Løkketangen 2008																							
27	Oppen <i>et al.</i> 2010																							
30	Prodrom and Prins 2008																							
31	Rancourt <i>et al.</i> 2012																							
38	Wen <i>et al.</i> 2011																							
41	Zäpfel and Bögl 2008																							

Formulation	Rich	Methods
x	x	Tabu Search algorithm with Path Relinking heuristic
x	x	Tabu Search algorithm with Path Relinking heuristic
x	x	Classical heuristics
x		Memetic algorithm
x		Greedy Randomized Adaptive Search algorithm with Path Relinking heuristic
x		Classical heuristics
x		Multi-phase heuristic
x		Variable Neighborhood Search algorithm
x		Dynamic Programming heuristic
x		Tabu Search algorithm and metaheuristics
x		Tabu Search algorithm
x		Branch and Price algorithm
x		Memetic algorithm
x		Tabu Search algorithm
x		Variable Neighborhood Search algorithm
x		Genetic algorithm and Tabu Search algorithm

To obtain a more accurate partition of the selected papers, we classify them using a cluster analysis. The cluster analysis consists in gathering the observed data having many similarities into significant structures. The formed groups must be as different as possible while the degree of similarity of the data clustered in the same group must be maximal. The clustering method used is the K-means algorithm. The clustering is based on the paper score according to 2 criteria: the number of scenario characteristics and the number of the operational characteristics. To calculate the scores, we follow 2 main principles. First, we count at least 1 for each paper according to the attributes (1.2) and (1.4). These attributes are necessary to describe the problem under study. Second, we add one to the score only if the problem addressed has one additional feature compared to the basic variant of the deterministic VRP (see below). Each of these attributes may be considered as dichotomous either the score increases by one or zero. The basic variant of VRP, we defined, is a single depot pickup (or delivery) routing problem. The planning horizon is one period. All vehicles are identical and covered one route only. Their number is unlimited. All data are known a priori. This basic variant has a score of 2 for the SCs and of 0 for the PPCs. For the clustering analysis, we include standard VRPs (see [Toth and Vigo \(2002\)](#)) in addition to the VRPs described in the 41 selected papers. The standard versions are: the CVRP [90], the Distance Constrained VRP (DCVRP) [91], the VRPTW [92], the VRPB [93] and the VRPPD [94]. The greatest dissimilarity between clusters is obtained for 3 clusters using the K-means method. These clusters are depicted in [Figure 1.1](#).

Cluster C1 contains papers having a significant degree of similarity. It includes 13 papers with the largest scores for the PPCs: [3, 7, 8, 9, 11, 12, 26, 27, 28, 29, 31, 33, 38]. All papers except the article by [Rieck and Zimmermann \(2010\)](#) [33] are devoted to variants of the VRP with many complicating optimization criteria and constraints coming from different real-life applications. In all papers of C1, a capacitated heterogeneous fleet is used, and (in)compatibility constraints (2.4) are present. In most of them, time window constraints are imposed and/or a restriction on the number of vehicles (2.1.2.1) is present. Nine papers address variants with a wide variety of specific constraints (2.5).

The papers of C1 are devoted to VRPs with eight operational characteristics at least. They are considered by their authors as RVRP studies. For example, in the papers of ([Oppen and Løkketangen, 2008](#), [Oppen et al., 2010](#)) [26, 27], an inventory routing problem is tackled. It consists in collecting animals from farms for slaughterhouses. This real-world case extends standard routing problems by defining multiple trips, multiple periods and split deliveries. Several PPCs are considered.

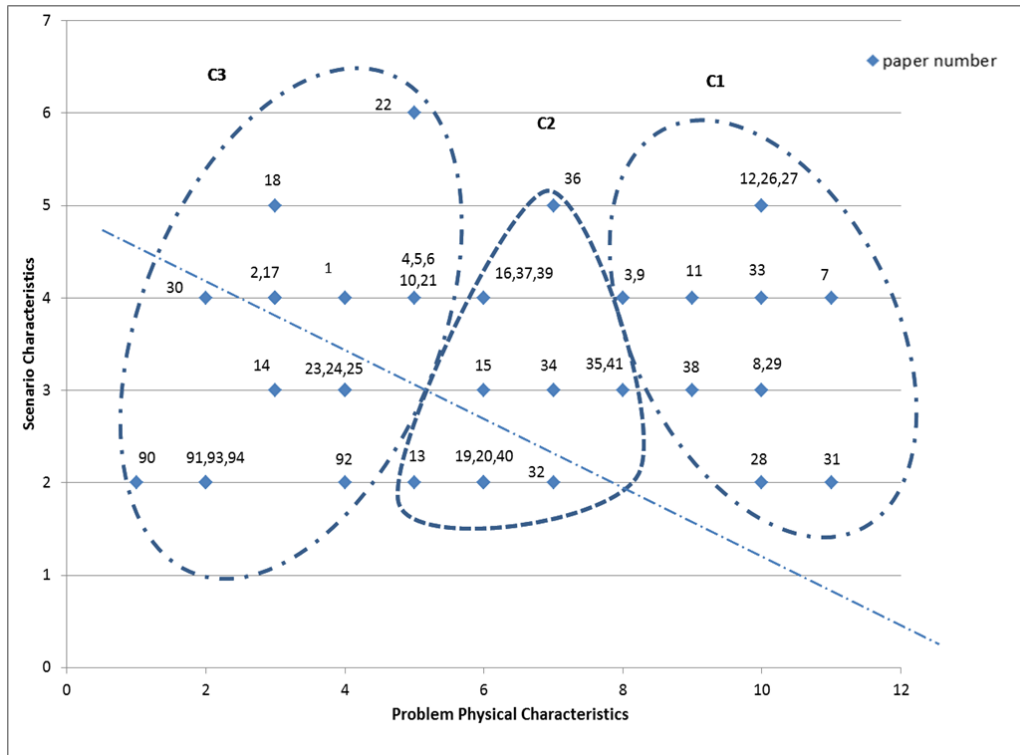


Figure 1.1: Cluster analysis

They are related to the heterogeneous capacitated and multi-compartment vehicles, to loading rules for animals, to cleaning constraints, to incompatibilities constraints between animal types, farms and animals, etc. The variety of the real-life features considered assesses the richness of the problem addressed in these papers.

The papers by [Cruz Reyes et al. \(2007, 2008\)](#) [11, 12] describe the distribution of bottled products for a Mexican company. The problem tackled includes 6 variants of the VRP in [11] and 11 variants in [12]. Many practical aspects are considered such as a heterogeneous capacitated and limited fleet, the assignment of several trips to vehicles, the loading plan, time window restrictions and incompatibilities between products. Additional constraints are included. First, the positions of the loaded products are optimized according to their weights. Second, the number of vehicles simultaneously present on the same location is limited. An extended version, which includes multiple depots and the assignment of orders to the depots, is studied in [12]. In their paper, [Ceselli et al. \(2009\)](#) [7] consider several operational constraints that arise in a real life situation. In this application, the routes performed thanks to a heterogeneous fleet may start from different depots, and may be open paths and not closed paths. Time windows are associated with the depots and the customers, which may be served according to a split delivery policy. The authors consider also

social driver regulations and incompatibilities between products, depots, vehicles and customers. Last, outsourcing is made possible by using express courier services. [Parragh et al. \(2012\)](#) [28] and [Beaudry et al. \(2010\)](#) [3] study dial-a-ride problems enriched by several complicating operational constraints which arise in the hospital context. For instance, these constraints required for patient transportation are: precedence constraints, the desired pickup or delivery time restrictions for patients and hospitals, etc. Last, the paper by [Rancourt et al. \(2013\)](#) [31] has a score of 2 according to SCs, but it considers various driver safety rules imposed for long-haul trips in North America under different objectives. The authors consider many PPCs such as multiple time windows at customers and different subsets of heterogeneous vehicles, which served subsets of customers.

Cluster C3 includes papers dealing with variants of the VRP with the lowest scores for the PPCs and different scores for the SCs. 15 papers are in C3 : [1, 2, 4, 5, 6, 10, 14, 17, 18, 21, 22, 23, 24, 25, 30] as well as the 5 standard VRPs defined in [Toth and Vigo \(2002\)](#). Since they are in the same group as the standard VRPs, the problems addressed in these papers have a low level of richness a priori. Indeed, only 5 papers out of 15 are case studies and most of the diverse real-life constraints considered in the papers of C1 are not present. For instance, the articles by [Mendoza et al. \(2009, 2011, 2010\)](#) [23, 24, 25] are the only ones in which incompatibility constraints are taken into account. Split deliveries is an option in 2 papers only, [Magalhães and Sousa \(2006\)](#) [22] and [Bolduc et al. \(2006\)](#) [4]. [Magalhães and Sousa \(2006\)](#) [22] study a pharmaceutical case-study in a dynamic multi-period environment using a heterogeneous fleet. Each vehicle may cover several routes and is loaded according to a chronological order. In [4], the authors address an inventory routing problem on a multi-period horizon using a heterogeneous limited fleet and a multiple use of vehicles. [Fügenschuh \(2006\)](#) [14] consider a real world bus scheduling problem taking into account school starting times and public bus schedule. The assignment of several trips to buses is the unique additional SC imposed. In their work, [Prodhon and Prins \(2008\)](#) [30] tackle a VRP variant closed to a standard one. The paper by [Crainic et al. \(2009\)](#) [10] addresses a multi-depot periodic VRP including time constraints and using a homogeneous unlimited fleet. Last, in their works, [Mendoza et al. \(2009, 2011, 2010\)](#) [23, 24, 25] describe routing problems for compartmentalized vehicles in a stochastic environment which are close to the classic VRPs. [Hvattum et al. \(2006\)](#) [18] consider an extended variant of the VRP to take into account some aspects of a real case faced by a major distribution company in Norway. The problem is modeled as a periodic dynamic and stochas-

tic problem with temporal constraints for customers and the depot. This paper is slightly different from those included in C3.

Last, [Armentano et al. \(2011\)](#) [1], [Boudia et al. \(2007\)](#) [6], [Boudia and Prins \(2009\)](#) [5], and [Bard and Nananukul \(2009\)](#) [2] integrate two types of strategic decision by coordinating the production, inventory and routing decisions over a multi-period horizon. The customer demands are satisfied either from inventory and/or thanks to the daily production of a single facility. The integration of two levels of decision illustrates the complexity of real-life supply chain management problems. It adds various constraints to the basic model to ensure the balance between demand, production, inventory and deliveries at the facility and at customers.

Cluster C2 includes the following papers: [13, 15, 16, 19, 20, 32, 34, 35, 36, 37, 39, 40, 41]. It lies between cluster C1, which gathers articles devoted to real cases and multi-constrained problems, and cluster C3, which contains papers addressing VRPs which seem not to be rich. In C2, we can identify variants of the VRP closed to those tackled in the papers of C1 or of C3 in terms of SCs and PPCs. Thus, [Reimann and Ulrich \(2006\)](#) [32] and [Ileri et al. \(2006\)](#) [19] describe problems which are not entirely different from the standard VRPs. The problem studied in [32] differs from the basic VRPB [93] by imposing time windows at customers and depots, by bounding the route length and by defining a lexicographic objective function. In [19], the problem addressed is a variant of the VRPPD [94] where multiple time windows are associated with customers, possible additional stops between the origin and the destination are considered, and the social rules for the drivers are satisfied. In their paper, [Derigs and Döhmer \(2008\)](#) [13] have addressed a pickup and delivery problem with time window using an unlimited homogeneous capacitated fleet. The work of [Wen et al. \(2009\)](#) [40] on the vehicle routing with cross-docking differs from the VRPPD by adding restrictions to synchronize inbound and outbound flows. However, the terminal management is not modeled since constraints related to the resource limitations are not included. For instance, the authors do not take into account the capacity of the cross-dock, the number and the availability of docks, and the scheduling of trucks processed at the dock doors, see e.g., ([Kreng and Chen, 2008](#), [Boysen and Fliedner, 2009](#), [Miao et al., 2009](#)). Last, [Kok et al. \(2010\)](#) [20] study a routing and crew scheduling problem with time window. In this work, the European social legislation on drivers' working hours is considered, but basic physical and the scenario characteristics are included.

Papers devoted to VRPs with 6 to 8 PPCs are also present in C2. Such problems could be considered as RVRPs. For instance, in their paper, [Schmid et al. \(2009\)](#)

[37] develop a hybrid solution approach for a rich application in the concrete industry. It consists in delivering concrete produced at different plants to construction sites at their preferred periods. Multiple visits using a heterogeneous capacitated fleet are allowed. Different incompatibility constraints between products, vehicles and customers must be satisfied as well as specific constraints related to the delivery process at construction sites. [Schilde et al. \(2009\)](#) [36] study the patient transportation problem faced by the Austrian Red Cross daily. The problem is modeled as a multi-objective DARP which aims to satisfy dynamic and stochastic requests in their specified time intervals using a fixed fleet. [Zäpfel and Bögl \(2008\)](#) [41] combine a multi-period vehicle routing problem with a crew scheduling problem to address a postal case-study. They consider pickup and delivery routes and time constraints at the customers and the depots. Driver outsourcing is also considered.

The analysis of the three clusters shows that the multi-constrained VRPs are considered in papers included mainly in C1 or C2. At the opposite, problems close to standard VRPs are addressed in papers belonging to clusters C3 and C2. The foregoing discussion leads us to define a frontier which discriminates the selected papers (see Figure 1.1). Below the frontier, there are standard variants of the VRP [90, 91, 92, 93, 94] and papers describing problems close to those variants. Above the frontier, papers are devoted to multi-constrained problems.

With respect to the two clustering criteria, we note that multi-constrained problems have at least 4 SCs and 6 PPCs in the papers included in C1. Rich combinations contain more PPCs than SCs. This is not a general rule, but it indicates combinations leading to rich vehicle routing studies. In addition, the previous discussion put into highlights that other combinations may lead to RVRPs. Those are studies above the frontier and can be characterized as follows: i) papers in which the strategic level includes several constraints related to the distribution system (5 SCs or more) , see, e.g., [18, 22]; ii) papers addressing variants in which the strategic and tactical levels remain standard but the operational characteristics of the route planning are increased (9 PPCs or more), see, e.g., [8, 29, 31, 38]. It is noteworthy that some VRPs based on real case studies could not be viewed as RVRPs as illustrated by papers [14, 19, 40]. The definition of RVRPs based on the taxonomy relies on the characteristics of the selected papers. In the literature, the RVRP is defined as a problem which simultaneously includes several types of challenging and complicating features. It is associated with the complexity of real-life routing problems. Now, we propose a more precise definition:

Definition

A RVRP extends the academic variants of the VRP in the different decision levels by considering at least four strategic and tactical aspects in the distribution system and including at least six different daily restrictions related to the physical characteristics. When a VRP is mainly defined through strategic and tactical aspects, at least five of them are present in a RVRP. When a VRP is mainly defined through physical characteristics, at least nine of them are present in a RVRP.

Clearly, the state of the art of RVRPs has changed since 2006. Some variants described as rich by their authors in 2006 may not be considered as such anymore whereas recent papers contain more complex aspects of reality.

1.4 Conclusions

In this chapter, a general taxonomy for RVRPs is proposed. Several papers have been classified and analyzed. A more elaborate definition of the RVRP than the existing ones results from this analysis. The taxonomy has been proven to be valid since some variants of the VRP are classified. Nevertheless, this taxonomy may be limited and not applicable for some exotic VRPs. In some countries, the distribution process includes primarily specific operational constraints. It may end up, rarely, with an uncommon problem that contains mainly particular constraints (2.4). Whether such problem should be classified according to the taxonomy, it could be not considered as a RVRP according to our taxonomy. Moreover, the RVRP taxonomy developed should not remain unchanged over time. It should be updated as new industrial challenges arise and new attributes may be added. Last, as the RVRPs incorporate complex features of real-life routing problems, there should be unified approximate methods to provide good solutions for these problems. Such heuristics should be generic and able to solve several VRP variants studied in the literature. A first step in this direction has been made by ([Røpke and Pisinger, 2006b](#), [Subramanian et al., 2011](#), [Derigs and Vogel, 2014](#), [Vidal et al., 2013b, 2014](#)).

Chapter 2

Models for a rich vehicle routing problem with compartments

In this chapter, we formally introduce the multi-depot multi-compartment multi-commodity vehicle routing problem with time windows (MDMCMC_m-VRPTW). We propose a mixed integer linear programming formulation for this problem. To improve the lower bounds obtained from the linear relaxation and to derive easily feasible solution, we apply the Dantzig-Wolfe decomposition to the previous model. The model associated with the pricing subproblem is then formulated and discussed.

2.1 Introduction

The use of multi-compartment vehicles is essential for several industries such as the transportation of food or petroleum products. In Chapter 1 we provide an overview of MC-VRPs while in this section we will further deepen this review. A literature review on the VRP with compartments in the food and petrol industries was proposed by Derigs et al. (2011a). Few published papers were devoted to food transportation. Such cases concern often delivering dry, refrigerated and frozen commodities where compartments are equipped with adequate cold holding equipment to maintain temperatures for fresh delivery. The use of fleet with several compartments is common in fuel and oil distribution (Brown et al., 1987, Cornillier et al., 2008a, 2009, 2012, Fagerholt and Christiansen, 2000, Relvas et al., 2014, Van der Bruggen et al., 1995). Fuel delivery applications are often considered as routing problems ensuring the delivery of petroleum products to underground tanks located in gas stations (Brown et al., 1987, Van der Bruggen et al., 1995) or as inventory-routing problems (Popović et al., 2012, Coelho and Laporte, 2013a, Vidović et al., 2014). This problem is often referred to the petrol station replenishment problem and it has been widely addressed by Cornillier et al. (2008b,a, 2009, 2012). Such problems arise also in maritime transportation where a fleet of ships equipped with flexible cargoes is engaged to distribute liquefied natural gas (Stålhane et al., 2012, Halvorsen-Weare and Fagerholt, 2013) or of non-mixable cement products (Christiansen et al., 2011) from loading ports to customers around the world. For a recent survey on maritime inventory routing problems, see (Papageorgiou et al., 2014).

Transporting oil and fuel with multi-compartment vehicles is more challenging and interesting from a scientific point of view than transporting food, where different commodities can be pre-assigned to suitable compartments. In this case, the loading problem reduces to a simple capacity checking procedure (Derigs et al., 2011a, El Fallahi et al., 2008, Melechovsky, 2013, Muyldermans and Pang, 2010b). In contrast, in fuel transportation, a routing problem and a compartment assignment problem must be solved jointly. For more details on the loading aspect arising in vehicle routing problems with compartments, see section 3.4.2.2 in Chapter 3 and (Pirkwieser et al., 2012b).

An homogeneous fleet is often assumed in the MC-VRP literature since the fleet is often homogeneous in distribution companies, and the problem is already very hard to address, (Wang et al., 2014). When distributing different types of fuel, vehicles are often not equipped with debit meters. As a consequence, a given compartment must hold at most only one customer demand. This scenario has been

widely studied in the MC-VRP literature, see e.g., (Cornillier et al., 2008b,a, 2009, 2012, Popović et al., 2012, Vidović et al., 2014). Coelho and Laporte (2013a) introduce and define the generalized case where compartments can be equipped with debit meters and the load of a compartment can be split between different customers. They also distinguish between the cases where customers may or may not receive the visit of more than one vehicle per period. The authors highlight the difficulty of the problem with split compartments and multiple visits per period.

In this chapter, we provide a three-index vehicle flow formulation for the MDMCMCm-VRPTW. Cordeau et al. (2002a) point out that the network lower bounds of the equivalent model for the VRPTW obtained by relaxing the time windows and the capacity constraints is often of poor quality. Similarly, the linear programming lower bound provides a weak linear relaxation. A possible way to overcome this drawback is to apply appropriate decomposition procedures, such as the Dantzig-Wolfe decomposition, which typically provides a tight linear programming relaxation.

The remainder of this chapter is organized as follows. In Section 2.2 we introduce and describe formally the MDMCMCm-VRPTW. Then we present a three-index vehicle flow formulation for the problem, along with a set of valid inequalities. In Section 2.3 we give an overview of the Dantzig-Wolfe decomposition and apply it on the MDMCMCm-VRPTW. The problem reformulation decomposes the compact formulation into smaller subproblems which have their solutions combined in a master problem. We describe the relation between the Dantzig-Wolfe decomposition and the column generation approach and we review its principles. Section 2.4 is devoted to describe and model the subproblem. Section 2.5 provides some concluding remarks.

2.2 Mixed integer linear program for the MDMC MCm-VRPTW

2.2.1 Problem description

The basic node routing problem is the CVRP, a well-known *NP*-hard problem (Garey and Johnson, 1990). The MC-VRP is a generalization of the CVRP in which each vehicle has only one compartment and it is shown to be *NP*-hard in (El Fallahi et al., 2008). Being a generalization of the MC-VRP, the Multi-Compartments Vehicle Routing Problem with Time Windows (MC-VRPTW) is also *NP*-hard. The

MC-VRP is obtained when the earliest time and latest time for each customer are set to 0 and infinity respectively.

The MDMCMCm-VRPTW enriches the CVRP in many ways by considering several complicated constraints encountered in common practical situations. In this problem, the routes performed by an homogeneous and sufficiently large fleet may start from different depots. A customer may have different orders, each corresponding to a single product. A key feature is that some products are incompatible and must be kept separated during transportation. There are also incompatibility relations between products and compartments. Thus, these features lead to use multi-compartments vehicles. Time windows are associated with the depots and the customers, which may be served according to a specific split delivery policy. Feasible routes are limited in capacity and time. The MDMCMCm-VRPTW consists then in designing vehicle routes of least total cost including the routing costs, the vehicle fixed costs and the total waiting time. Each vehicle route starts and ends at its associated depot such that all the customers orders are satisfied, possibly through several routes, while respecting temporal and physical restrictions. Therefore, a problem solution specify (1) the assignment of customers to vehicles, (2) how many vehicles to use, (3) the assignment of orders to compartments, (4) the sequence of the customer visits per vehicle such that the capacity and incompatibility restrictions are satisfied, (5) and the starting and ending times of vehicle routes.

All the problem assumptions are highlighted in Figure 2.1 and marked with an asterisk. As we can see, the MDMCMCm-VRPTW is much more characterized by operational daily characteristics than by strategic ones with a score of 4 for the SC and 8 for the PPC. The problem belongs to cluster $C1$ (see Chapter 1, Section 1.3) and is positioned over the frontier to the left. Classifying the MDMCMCm-VRPTW on the basis of the taxonomy attributes confirms that addressing this problem is quite challenging. To the best of our knowledge, this rich VRP has not been formally addressed before.

The MDMCMCm-VRPTW is a generalization of the VRP with capacity as it is shown in Figure 2.2. This diagram shows the derived problems obtained by relaxing one or more constraints of the MDMCMCm-VRPTW. To simplify the illustration, we take into consideration only the most discriminating problem attributes which have a significant impact on the problem definition. These attributes, as depicted in Figure 2.2, are : multi-depot (MD), multi-compartment (MC), time windows (TW) and multi-commodity (MCm). The classical constraints, such as capacity constraints and maximum route duration restrictions are not considered. Note that

1 Scenario characteristics	2 Problem physical characteristics
1.1 Input data	2.1 Vehicles
1.1.1 Static *	2.1.1 Type
1.1.2 Dynamic	2.1.1.1 Homogeneous *
1.1.3 Deterministic *	2.1.1.2 Heterogeneous
1.1.4 Stochastic	2.1.2 Number
1.2 Decision management components	2.1.2.1 Fixed
1.2.1 Routing *	2.1.2.2 Unlimited *
1.2.2 Inventory and routing	2.1.3 Structure
1.2.3 Location and routing	2.1.3.1 Compartmentalized *
1.2.4 Routing and driver scheduling	2.1.3.2 Not compartmentalized
1.2.5 Production and distribution planning	2.1.4 Capacity constraints *
1.3 Number of depots	2.1.5 Loading Policy
1.3.1 Single	2.1.5.1 Chronological order
1.3.2 Multiple *	2.1.5.2 No policy *
1.4 Operation type	2.1.6 Drivers regulations
1.4.1 Pickup or delivery *	2.2 Time constraints
1.4.2 Pickup and delivery	2.2.1 Restriction on customer *
1.4.3 Backhauls	2.2.2 Restriction on road access *
1.4.4 Dial-a-ride	2.2.3 Restriction on depot *
1.5 Load splitting constraints	2.2.4 Service time *
1.5.1 Splitting allowed *	2.2.5 Waiting time *
1.5.2 Splitting not allowed	2.3 Time window structure
1.6 Planning period	2.3.1 Single time window *
1.6.1 Single period *	2.3.2 Multiple time windows
1.6.2 Multi-period	2.4 Incompatibility constraints *
1.7 Multiple use of vehicles	2.5 Specific constraints
1.7.1 Single trip *	2.6 Objective function
1.7.2 Multi-trip	2.6.1 Single objective *
	2.6.2 Multiple objectives

Figure 2.1: The MDMCMCm-VRPTW assumptions

the MCM-VRP may be considered as a simplified version of the SDVRP in which the splitting scenarios is fixed a priori.

Formally, the MDMCMCm-VRPTW can be defined on a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E}) = (\mathcal{V} \cup \mathcal{D}, \mathcal{E})$ where $\mathcal{V} = \{1, \dots, n\}$ is the set of n customers and $\mathcal{D} = \{n + 1, n + 2, \dots, n + m\}$ is the set of m depots. \mathcal{E} is the arc set where $\mathcal{E} = \{(i, j) : i, j \in \mathcal{V} \cup \mathcal{D}, i \neq j\}$. If $i, j \in \mathcal{D}$, then arc $(i, j) \notin \mathcal{E}$, i.e. inter-depots routes are prohibited. The arc $(i, j) \in \mathcal{E}$ is associated with the lowest cost path connecting vertices i and j . A travel cost α_{ij} and a travel time t_{ij} are associated with each arc $(i, j) \in \mathcal{E}$. We assume that the cost matrix is symmetric and costs satisfy the triangle inequality. With each customer $i \in \mathcal{V}$, are associated a hard time window $[e_i, l_i]$, within which the deliveries of i take place, and a service time s_i . In case of early arrival at customer i , the vehicle is allowed to wait until e_i which incurs a waiting cost. Time windows are also associated with depots $d \in \mathcal{D}$. They correspond to the opening hours. They are denoted by $[e_d, l_d]$ where e_d represent the earliest possible departure from the depot d and l_d the latest possible arrival at the

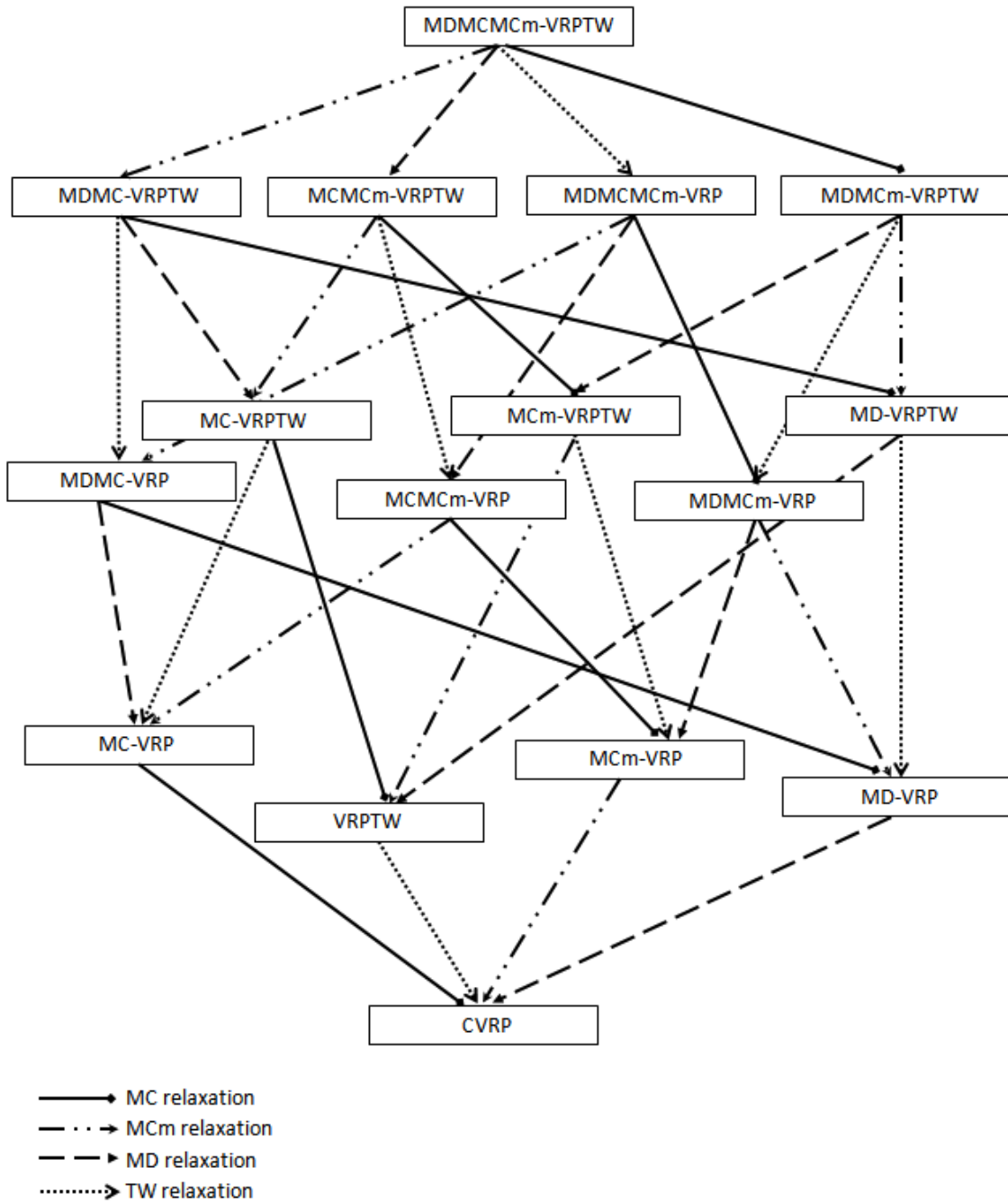


Figure 2.2: The related problems to the MDMCMCm-VRPTW and their interconnections

depot d . Service times for the depots are null. The total duration of any vehicle route must not exceed a preset maximal route duration T^{max} .

We consider a set \mathcal{K} of available homogeneous vehicles $\mathcal{K} = \{1, \dots, K\}$. We assume that K is large enough to guarantee the existence of a feasible solution. We denote by Q the carrying capacity of vehicle k and by β the fixed cost incurred. With each vehicle k is associated a set of compartments $\mathcal{W} = \{1, \dots, W\}$. Each compartment $w \in \mathcal{W}$ has a capacity Q^w and is equipped with a debit meter. Feasible vehicle routes are single-depot routes starting and ending at the same depot $d \in \mathcal{D}$. We note d_o^k and d_d^k the origin and destination depots for vehicle k .

There is a set $p \in \mathcal{P} = \{1, \dots, P\}$ of products. Each customer $i \in \mathcal{V}$, can place several orders, each referring to one single product p . We denote by $o_i^p \in \mathcal{O}$ the order placed by customer i for product p . With each order o_i^p we associate a demand q_i^p . A key feature of the MDMCMCm-VRPTW is that the number of orders may differ from one customer to another. We assume that there is at least one order for each product type p . The delivery of any product must not be split but a customer may receive several visits corresponding to different orders. We refer to this assumption as *partial customer satisfaction* which offers an intermediate level of splitting. The set of orders of one customer may be delivered by several routes starting from different depots. The set $\mathcal{IP} \subseteq \mathcal{P} \times \mathcal{P}$ denotes the incompatibility relation between products. $(p, q) \in \mathcal{IP}$ means that products p and q must not be carried together in the same compartment. The set $\mathcal{IPC} \subseteq \mathcal{K} \times \mathcal{W} \times \mathcal{P}$ defines incompatibilities between products and compartments, forbidding product p to be carried in compartment w . To sum up its description, we provide in Table 2.1 a summary of the MDMCMCm-VRPTW notations.

In Figure 2.3 we depict a MDMCMCm-VRPTW solution for a small example. Consider we have the following problem data:

- $\mathcal{V} = \{1, \dots, 6\}$
- $\mathcal{D} = \{1\}$
- $\mathcal{P} = \{p1, p2, p3, \}$
- $\mathcal{O} = \{1, \dots, 12\}$
- $\mathcal{K} = \{1, 2\}$
- $\mathcal{W} = \{w1, w2, w3\}$
- $\mathcal{IP} = \{(p2, p3)\}$

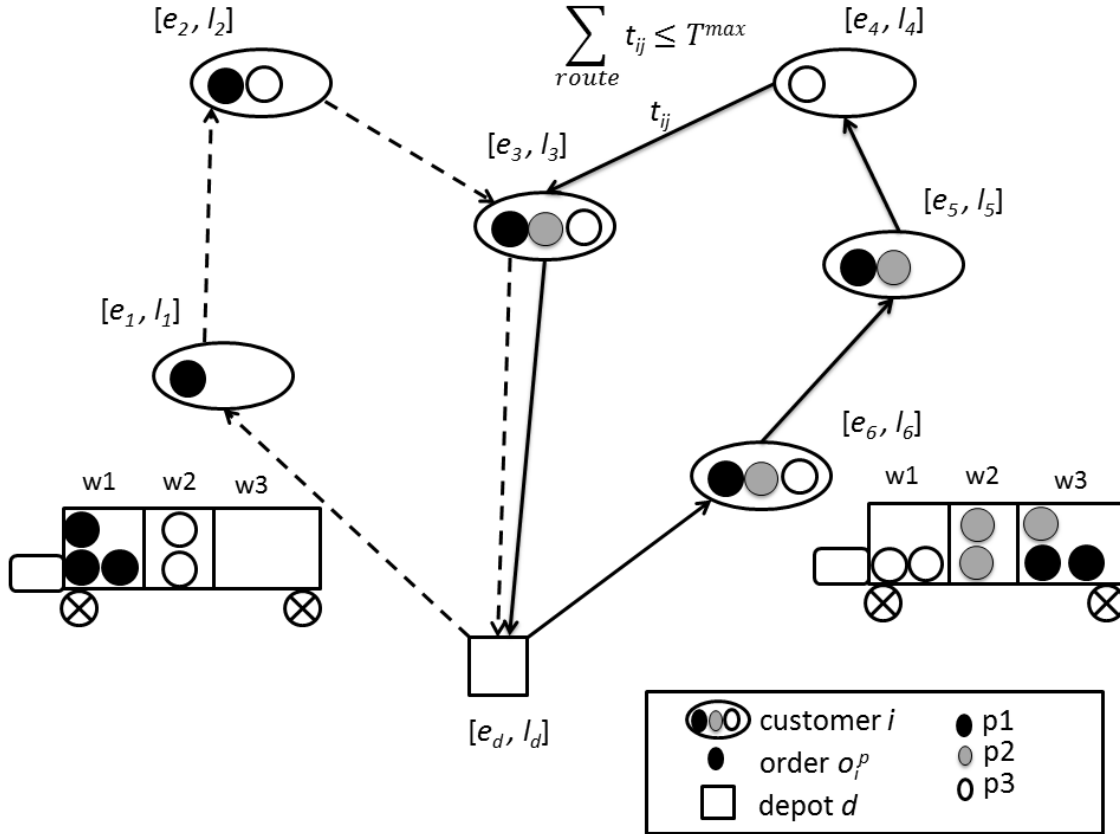


Figure 2.3: Example of solution for the MDMCMCm-VRPTW

- $IPC = \{(1, w2, p1), (2, w2, p1)\}$

A solution consists of two vehicles routes covering all the customer orders within their time windows, while respecting the compartments capacity, the incompatibility relations and the maximum route duration. Vehicle $k = 1$ performs the route $(Depot, 1, 2, 3, Depot)$ while the second vehicle $k = 2$ performs the route $(Depot, 6, 5, 4, 3, Depot)$.

2.2.2 Mathematical model

We propose a three-index vehicle flow formulation for the MDMCMCm-VRPTW. Three-index vehicle flow models have been extensively used to model multi-attributes VRPs due to their greater flexibility in incorporating several features. We define the following decision variables. Binary variables x_{ij}^k indicate if arc $(i, j) \in \mathcal{E}$ is traversed by vehicle k . Binary variables y_{jp}^{wk} indicate if product p of customer j is carried in compartment w of vehicle k , and variables b_i^k correspond to the time at which customer i is served thanks to vehicle k . We define variables y_{jp}^{wk} for all

Table 2.1: Notations used in the description of the MDMCMCm-VRPTW

\mathcal{D}	set of depots with cardinality m
\mathcal{IP}	set of incompatible products
\mathcal{IPC}	set of incompatible products and compartments
\mathcal{K}	set of vehicles with cardinality K
\mathcal{N}	set of vertices $\mathcal{V} \cup \mathcal{D}$ with cardinality $m + n$
\mathcal{O}	set of orders
\mathcal{P}	set of products with cardinality P
\mathcal{R}	set of routes
$\bar{\mathcal{R}}$	subset of routes $\subset \mathcal{R}$
\mathcal{V}	set of customers with cardinality n
$\bar{\mathcal{V}}$	set of customers with one associated depot
\mathcal{W}	set of compartments with cardinality W
α_{ij}	transportation cost associated with arc (i, j)
β	fixed cost for vehicle use
γ	cost of waiting time
π_i^p	profit associated with order σ_i^p
c^r	cost of route r
e_i	earliest arrival time at customer i
l_i	latest arrival time at customer i
σ_i^p	order of product p of customer i
q_i^p	quantity associated with order σ_i^p
Q	vehicle capacity
Q^w	capacity of compartment w
s_i	service time at customer i
t_{ij}	travel time associated with arc (i, j)
T^{max}	maximal route duration

combinations of customers and products because this simplifies the formulation. However, since not all customers place orders for all the products, we put $y_{jp}^{wk} = 0$ if customer j does not request product p , i.e., order o_j^p does not exist. We define the parameter γ as a normalization coefficient. The mathematical model for the MDMCMCm-VRPTW is as follows:

$$\min \sum_{(ij) \in \mathcal{E}} \sum_{k \in \mathcal{K}} \alpha_{ij} x_{ij}^k + \sum_{j \in \mathcal{V}} \sum_{k \in \mathcal{K}} \beta x_{d_o^k j}^k + \gamma \sum_{k \in \mathcal{K}} (b_{d_d^k}^k - b_{d_o^k}^k) \quad (2.1)$$

subject to

$$\sum_{i \in \mathcal{N}} x_{ij}^k \leq 1 \quad j \in \mathcal{N}, k \in \mathcal{K} \quad (2.2)$$

$$\sum_{i \in \mathcal{N}} x_{il}^k - \sum_{j \in \mathcal{N}} x_{lj}^k = 0 \quad l \in \mathcal{N}, k \in \mathcal{K} \quad (2.3)$$

$$b_j^k \geq b_i^k + s_i + t_{ij} - M_{ij}(1 - x_{ij}^k) \quad i, j \in \mathcal{N}, k \in \mathcal{K} \quad (2.4)$$

$$e_i \sum_{j \in \mathcal{N}} x_{ij}^k \leq b_i^k \leq l_i \sum_{j \in \mathcal{N}} x_{ij}^k \quad i \in \mathcal{N}, k \in \mathcal{K} \quad (2.5)$$

$$\sum_{(i,j) \in \mathcal{E}} t_{ij} x_{ij}^k \leq T^{max} \quad k \in \mathcal{K} \quad (2.6)$$

$$y_{jp}^{wk} \leq \sum_{i \in \mathcal{N}} x_{ij}^k \quad j \in \mathcal{V}, k \in \mathcal{K}, w \in \mathcal{W}, p \in \mathcal{P} \quad (2.7)$$

$$\sum_{k \in \mathcal{K}} \sum_{w \in \mathcal{W}} y_{jp}^{wk} = 1 \quad j \in \mathcal{V}, p \in \mathcal{P}, q_j^p \geq 0 \quad (2.8)$$

$$y_{jp}^{wk} + y_{iq}^{wk} \leq 1 \quad i, j \in \mathcal{V}, k \in \mathcal{K}, w \in \mathcal{W}, (p, q) \in \mathcal{IP} \quad (2.9)$$

$$y_{jp}^{wk} = 0 \quad j \in \mathcal{V}, k \in \mathcal{K}, w \in \mathcal{W}, (p, w^k) \in \mathcal{IPC} \quad (2.10)$$

$$\sum_{j \in \mathcal{V}} \sum_{p \in \mathcal{P}} q_j^p y_{jp}^{wk} \leq Q^w \quad k \in \mathcal{K}, w \in \mathcal{W} \quad (2.11)$$

$$x_{ij}^k \in \{0, 1\} \quad i, j \in \mathcal{N}, k \in \mathcal{K} \quad (2.12)$$

$$y_{jp}^{wk} \in \{0, 1\} \quad j \in \mathcal{V}, k \in \mathcal{K}, w \in \mathcal{W}, p \in \mathcal{P} \quad (2.13)$$

$$b_i^k \geq 0 \quad i \in \mathcal{N}, k \in \mathcal{K}. \quad (2.14)$$

Objective (2.1) minimizes the total transportation costs including routing costs, fixed vehicles costs and waiting time costs. We do not consider a multi-objective problem but we aggregate the objective functions hierarchically. The constraints (2.2)-(2.14) may be classified into three groups of constraints.

The first set of constraints (2.2)-(2.3) is devoted to the routing restrictions. Constraints (2.2) mean that a customer can be visited at most once by a given vehicle, but may be visited by several vehicles. This enables to have some split delivery policy for customers. For $j \in \mathcal{D}$, constraints (2.2) state that each vehicle returns at

most once to its associated depot. Constraints (2.3) specify flow conservation at both customers and depots. They ensure the continuity of each route, i.e., if a vehicle visits any customer, it has to leave it to reach another customer and/or depot. Constraints (2.2) and (2.3) both ensure that each vehicle leaves its associated depot and return to it at most once.

Constraints (2.4)-(2.6) maintain temporal feasibility of vehicles routes. Constraints (2.4)-(2.5) are time window constraints and eliminate subtours, (Cordeau et al., 2002a). Constraints (2.6) imply that vehicles routes have a limited duration.

The third set of constraints (2.7)-(2.11) describes deliveries in terms of capacity, loading and physical compatibility restrictions. Constraints (2.7) allow delivery of product p to customer j only if vehicle k visits customer j . Each product order is delivered by a single vehicle loaded in one compartment thanks to constraints (2.8). Constraints (2.9) guarantee that each compartment carries only compatible products. Constraints (2.10) forbid carrying products in incompatible compartments. Constraints (2.11) prevent compartment capacity violation. Finally, constraints (2.12)-(2.14) impose integrality and non-negativity conditions on the decision variables.

The formulation (2.1)-(2.14) is sufficient to model the MDMCMCm-VRPTW. Some additional constraints can however be added to strengthen this formulation. Such constraints are called valid inequalities. According to Nemhauser and Wolsey (1988), an inequality $d^T x \leq d_0$ is valid for a set $S \subseteq R^n$ if $d^T x \leq d_0$ for all $x \in S$. Given a formulation P , identifying valid inequalities and adding them to the formulation, can *cut* away regions of P that contain no feasible solutions, and thus leads to better model. The MDMCMCm-VRPTW formulation can be enriched by the introduction of the constraints (2.15)-(2.16) to enforce logical relations between routing and assignment variables.

$$\sum_{i \in \mathcal{N}} x_{ij}^k \leq \sum_{w \in \mathcal{W}} \sum_{p \in \mathcal{P}} y_{jp}^{wk} \quad j \in \mathcal{V}, k \in \mathcal{K} \quad (2.15)$$

$$\sum_{w \in \mathcal{W}} y_{jp}^{wk} \leq \sum_{i \in \mathcal{N}} x_{ij}^k \quad j \in \mathcal{V}, k \in \mathcal{K}, p \in \mathcal{P}. \quad (2.16)$$

Constraints (2.15)-(2.16) impose upper and lower bounds on the routing variables. Through constraints (2.15), we ensure that if customer j is visited by vehicle k , then at least one product p should be delivered to customer j . Constraints (2.16) ensure that if an order σ_j^p is loaded into a given vehicle k then the customer must be served

by this vehicle. Note that constraints (2.16) can be obtained from constraints (2.7) by applying a lifting procedure.

In some industrial contexts, compartments are not equipped with debit meters, then the content of a compartment can hold at most one customer order for reasons of weighting and billing. The compartment must be entirely emptied once delivery has started. In such cases, the MDMCMCm-VRPTW formulation must include restrictions on the number of orders per compartment while respecting the compartment capacity. This is ensured by introducing constraints (2.17).

$$\sum_{j \in \mathcal{V}} \sum_{p \in \mathcal{P}} y_{jp}^{wk} \leq 1 \quad k \in \mathcal{K}, w \in \mathcal{W}. \quad (2.17)$$

When compartments are not equipped with debit meters, the number of possible visited customers in each route is bounded by the number of the vehicle compartments which may simplify the problem. However, in some real-applications, using multi-compartments vehicles gives rise to other complicating constraints. This is the case when vehicles are either used over different periods or allowed to perform several routes over a given period and that cleaning or disinfection activities are required between successive routes. In such cases, one must determine for each vehicle used, the optimal sequencing of vehicle routes to minimize the cleaning costs incurred.

Generally speaking, the integrality gap obtained with such vehicle flow formulation, is not tight enough to be solved a priori using a straightforward branch-and-bound algorithm, except for small instances (Cordeau et al., 2002a). To overcome this drawback, several decomposition procedures were proposed in the VRP literature, exploiting the structure of the underlying problem. Moreover, three-index models are not suited when a special attention must be paid to the problem symmetry, as it is the case for the MDMCMCm-VRPTW. Symmetry can lead to multiple representations of the same route which may be a source of inefficiency in a branch-and-bound algorithm. Specifically, in the MDMCMCm-VRPTW, exchanging the orders assignment between any two compartments or two vehicles produces alternative solutions. Thus, the three-index vehicle flow formulation is expected to perform poorly when the loading problem is complex. To avoid such pitfalls, we propose to reformulate it as a set covering model having a better linear relaxation and a small number of constraints. This new model can be obtained through a Dantzig-Wolfe decomposition and is introduced in the next section.

2.3 Reformulation based on Dantzig-Wolfe decomposition

The Dantzig-Wolfe (DW) decomposition (Dantzig and Wolfe, 1960) is a technique to improve the solution of structured linear problems (LP). The general idea behind DW decomposition is to exploit the LP structure to reformulate the problem into another equivalent problem that is more tractable with regard to complexity, computational running time, or quality of lower bounds. This decomposition algorithm allows changing any LP with a huge number of constraints to a problem with a reduced number of constraints which typically has a tighter linear relaxation than the original formulation. It generates a new problem, the so called *master problem* (MP) with many more variables than the original formulation and a number of disjoint sets, each associated with a subsystem of constraints. The DW decomposition is very closely connected to column generation and they are often used interchangeably, (Villeneuve et al., 2005, Desrosiers and Lübbecke, 2005). Even if the MP has many variables (and associated columns), solving the linear relaxation of the MP does not require an explicit enumeration of all its variables, since the column generation algorithm consists in generating columns if needed. In this section, we illustrate the decomposition technique on a simple example. Then we apply it to the MDMCMCm-VRPTW vehicle flow formulation. Based on the resulting MP, we give an overview of the basic column generation algorithm.

2.3.1 Decomposition principle

Consider a minimization integer program of the form:

$$\min \sum_{k=1}^K C^k x^k \quad (2.18)$$

subject to

$$\sum_{k \in K} B^k x^k = f \quad (2.19)$$

$$D^k x^k \leq e^k \quad k \in \{1, \dots, K\} \quad (2.20)$$

$$x^k \in \mathbb{Z}_+^n \quad k \in \{1, \dots, K\}. \quad (2.21)$$

This program is referred as the original problem where K is the set of blocks, and B^k and D^k are the constraints matrices of these blocks. Notice that constraints (2.20) together with (2.21) have a *bloc angular* structure, meaning that the con-

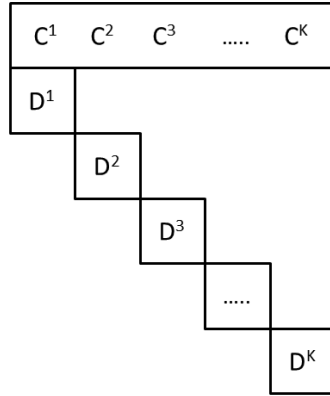


Figure 2.4: Block-angular matrices for K blocks

straint matrix is divided into blocks of non-zero variable coefficients. Each block corresponds to a subsystem of constraints solved separately for each $k \in \{1, \dots, K\}$ since the variables vector X can be partitioned into K disjoint subsets of variables $x = (x^1, x^2, \dots, x^K)$. In practice, these constraints are referred to *separable* or *easy* constraints as they usually form a more tractable model. On the other hand, constraints (2.19) link together variables from several disjoint sets. They are often referred as *joint*, *binding* or *difficult* constraints. These constraints are kept in a superior level and form the master problem. The matrix structure of the original formulation is depicted in Figure 2.4. The DW decomposition consists in rewriting the model (2.18)-(2.21) as another program that contains only equivalent constraints to binding constraints and a set of convexity constraints. The new model involves a very large number of variables defined based on separable constraints. Let $X^k = \text{conv}\{x^k \in \mathbb{Z}_+^n : D^k x^k \leq e^k\}$ be a feasible convex domain defining the K disjoint sets. Given the convex set X^k , Minkowski-Weyl theorem (cf. Nemhauser and Wolsey (1988)) states that X^k can be represented as a convex combination of its extreme points $\{x^{kr}\}_{r \in \mathcal{R}^k}$ and a linear combination of its extreme rays $\{\rho^{kp}\}_{p \in \Omega^k}$ such that $x^k \in X^k$ can be written as:

$$x^k = \sum_{r \in \mathcal{R}^k} x^{kr} \Lambda^{kr} + \sum_{p \in \Omega^k} \rho^{kp} \theta^{kp} \quad (2.22)$$

$$\sum_{r \in \mathcal{R}^k} \Lambda^{kr} = 1 \quad (2.23)$$

$$\Lambda^{kr} \geq 0 \quad r \in \mathcal{R}^k \quad (2.24)$$

$$\theta^{kp} \geq 0 \quad p \in \Omega^k \quad (2.25)$$

where \mathcal{R}^k is the index set of extreme points and Ω^k is the index set of extreme rays.

In this thesis, we assume that the set X^k is a large but *finite* set. Thus, it may be represented by considering only the extreme points $\{x^{kr}\}_{r \in \mathcal{R}^k}$. Specifically, we propose to consider the discretization approach of the DW decomposition set (compared to the convexification approach) since the goal behind the decomposition is to apply a column generation approach to obtain integer variables representing the extreme points of the X^k . Both approaches give the same linear relaxation of the MP if the decision variables are binary (see e.g., (Vanderbeck, 2000, Desrosiers and Lübbecke, 2011)). X^k can be replaced by:

$$x^k = \sum_{r \in \mathcal{R}^k} x^{kr} \Lambda^{kr} \quad (2.26)$$

$$\sum_{r \in \mathcal{R}^k} \Lambda^{kr} = 1 \quad (2.27)$$

$$\Lambda^{kr} \in \{0, 1\} \quad r \in \mathcal{R}^k \quad (2.28)$$

Substituting x^k in the original problem (2.18)-(2.21) leads to the following integer program (2.29)-(2.32), the MP. The new variables Λ^{kr} in the MP correspond to feasible solutions subject to the subsystem of constraints of each of the independent blocks.

$$\min \sum_{k=1}^K \sum_{r \in \mathcal{R}^k} C^k x^{kr} \Lambda^{kr} \quad (2.29)$$

subject to

$$\sum_{k=1}^K \sum_{r \in \mathcal{R}^k} B^k x^{kr} \Lambda^{kr} = f \quad (2.30)$$

$$\sum_{r \in \mathcal{R}^k} \Lambda^{kr} = 1 \quad k \in \{1, \dots, K\} \quad (2.31)$$

$$\Lambda^{kr} \in \{0, 1\} \quad r \in \mathcal{R}^k, k \in \{1, \dots, K\}. \quad (2.32)$$

The linear relaxation of the MP usually provides better bounds than the linear relaxation of the original formulation. This is due to the relaxation of all integrality requirements, i.e., the linear relaxation of the original formulation. For more details on the DW decomposition and its application on integer programs, we refer to (Vanderbeck, 2000, Vanderbeck and Savelsbergh, 2006, Vanderbeck and Wolsey, 2010, Frangioni and Gendron, 2013).

2.3.2 DW reformulation for the MDMCMCm-VRPTW

The MDMCMCm-VRPTW can be decomposed obviously into an assignment-type problem ensuring that all the customers orders are satisfied, i.e., the master problem, and several disjoint sets of constraints, one for each available vehicle. In the MDMCMCm-VRPTW model (2.1)-(2.14), constraints (2.8) are the binding constraints. The MP includes then the objective function (2.1), the covering of each customer order exactly once (2.8) and the binary requirements (2.13) on the assignment variables. The remaining constraints sets (2.2)-(2.7) and (2.9)-(2.14) ensure the feasibility of each vehicle route with respect to routing and loading restrictions. They define the feasible region of each set of routes associated with vehicle k . Let $A^k = \{(x^k, y^k, b^k) | (x^k, y^k, b^k) \text{ satisfies } (2.2)-(2.7); (2.9)-(2.14)\}$ be the set of routes satisfying the constraints associated with vehicle $k \in \mathcal{K}$. Then, there exists a finite set of feasible routes $\{a^{kr}\}_{r \in \mathcal{R}^k}$ with $a^{kr} = (x^{kr}, y^{kr}, b^{kr})$ and \mathcal{R}^k the index set such that $a^k \in A^k$ can be replaced by $a^k = (x^{kr}, y^{kr}, b^{kr})$.

$$y^k = \sum_{r \in \mathcal{R}^k} y^{kr} \Lambda^{kr} \quad (2.33)$$

$$x^k = \sum_{r \in \mathcal{R}^k} x^{kr} \Lambda^{kr} \quad (2.34)$$

$$b^k = \sum_{r \in \mathcal{R}^k} b^{kr} \Lambda^{kr} \quad (2.35)$$

$$\sum_{r \in \mathcal{R}^k} \Lambda^{kr} = 1 \quad (2.36)$$

$$\Lambda^{kr} \in \{0, 1\} \quad r \in \mathcal{R}^k \quad (2.37)$$

The definition of the MP as a set partitioning (SP) problem requires the following additional notation: we denote c^{kr} the total cost of route r for a given vehicle k such that $c^{kr} = \sum_{(i,j) \in \mathcal{E}} \alpha_{ij} x_{ij}^k + \sum_{j \in \mathcal{V}} \beta x_{d_o^k j}^k + \gamma (b_{d_d^k}^k - b_{d_o^k}^k)$ while a_{jp}^{kr} denotes a binary parameter equal to 1 if order o_j^p is covered by route r of vehicle k .

When all the vehicles are identical, as for the MDMCMCm-VRPTW, the DW decomposition can be carried even further. The corresponding set of feasible domains A^k can be decomposed into independent and identical blocks. The MP admits then a vehicle aggregation which allows to formulate the MP with fewer variables and constraints. An additional limit K is put on the number of routes of all vehicles used which results in $\sum_{r \in \mathcal{R}} \Lambda^r = K$. This makes the index k irrelevant and it can be removed from parameters c^{kr} and a_{jp}^{kr} . Let \mathcal{R} be the index set of all vehicles routes. After substitution and standard adjustment, the set partitioning formulation of the

MP can be formulated as follows:

$$\min \sum_{r \in \mathcal{R}} c^r \Lambda^r \quad (2.38)$$

subject to

$$\sum_{r \in \mathcal{R}} a_{jp}^r \Lambda^r = 1 \quad j \in \mathcal{V}, p \in \mathcal{P} \quad (2.39)$$

$$\sum_{r \in \mathcal{R}} \Lambda^r = K \quad (2.40)$$

$$\Lambda^r \in \{0, 1\} \quad r \in \mathcal{R}. \quad (2.41)$$

The objective (2.38) is to minimize the sum of fixed and variable transportation costs. Constraints (2.39) state that each customer order is satisfied exactly once. The second set of constraints (2.40) limits the number of routes. Constraints (2.40) hold for the MDMCMCm-VRPTW because *empty* vehicle routes are permitted. However, since we assume that we dispose of a sufficiently large fleet, constraints (2.40) can be dropped from the model.

Set covering formulation A closely related formulation to the SP model is the set covering (SC) formulation. Like it is commonly done in reformulations for routing problems, we choose to replace partitioning constraints by covering constraints. This is due to the fact that doing so, dual variables associated are positive instead of having no sign restrictions. Consequently, the dual solution space is reduced which improves the convergence of the solution method. Furthermore, when the cost matrix satisfies the triangle inequality, each customer is visited exactly once in an optimal solution. Note also that any feasible solution for the SP formulation is feasible for the SC formulation and any feasible solution for the SC may be transformed into a feasible solution for the SP with a smaller cost. Indeed, if a customer order is covered more than once in a SC solution, this customer order can be removed from the most "expensive" route, in a post-processing phase (see Section 5.2.3 in Chapter 5), reducing then the overall solution cost.

Consider the linear relaxation of the MP (2.38)-(2.41), the set covering model for the MDMCMCm-VRPTW is expressed as follows:

$$\min \sum_{r \in \mathcal{R}} c^r \Lambda^r \quad (2.42)$$

subject to

$$\sum_{r \in \mathcal{R}} a_{jp}^r \Lambda^r \geq 1 \quad j \in \mathcal{V}, p \in \mathcal{P} \quad (\pi_{jp} \geq 0) \quad (2.43)$$

$$\Lambda^r \geq 0 \quad r \in \mathcal{R}. \quad (2.44)$$

In the above SC formulation, the number of routes $\{a^r\}_{r \in \mathcal{R}}$ to be considered is very large. Generating explicitly all MP variables (columns) and using a straightforward branch-and-bound algorithm is impractical, but we can proceed implicitly by column generation. The column generation algorithm is a method able to solve the MP obtained through the DW decomposition and it is pioneered by (Dantzig and Wolfe, 1960, Gilmore and Gomory, 1961). In our implementation, the column generation scheme starts by initializing the MP with a small set of routes, denoted by $\bar{\mathcal{R}} \subset \mathcal{R}$. We define $\text{MP}(\bar{\mathcal{R}})$ the restriction of the MP to a subset of columns, referred as the Restricted Master Problem (RMP). The column generation approach consists in solving iteratively the linear relaxation of the RMP with an augmented subset of routes in each iteration. Following each optimization of the RMP, dual solution is transferred to a set of constraints defined by the feasible region $\text{conv}\{A\}$. The objective is to identify the most promising feasible solutions of the independent disjoint sets, often called pricing subproblems, such that the objective of the RMP can be improved. Solving a subproblem consists in identifying new columns with negative reduced cost (for a minimization problem). The new generated columns will be added to the RMP. This iterative process continues as long as there are routes with negative reduced costs in the set \mathcal{R} which have not been added yet to the RMP. The RMP is solved by the primal simplex algorithm.

The RMP can be expressed as follows:

$$\min \sum_{r \in \bar{\mathcal{R}}} c^r \Lambda^r \quad (2.45)$$

subject to

$$\sum_{r \in \bar{\mathcal{R}}} a_{jp}^r \Lambda^r \geq 1 \quad j \in \mathcal{V}, p \in \mathcal{P} \quad (2.46)$$

$$\Lambda^r \geq 0 \quad r \in \bar{\mathcal{R}}. \quad (2.47)$$

Let π_{jp} be the dual variables associated with constraints (2.46). We denote by $\tilde{c}^r = c^r - \sum_{j \in \mathcal{V}} \sum_{p \in \mathcal{P}} a_{jp}^r \pi_{jp}$ the reduced cost of a route $a^r \in A$. In a column generation scheme, given a dual solution of the RMP, the pricing subproblem identifies

the route a^r with the minimum reduced cost:

$$\tilde{a}^r = \operatorname{argmin}\{\hat{c}^r\}_{a^r \in A} = \operatorname{argmin}\{c^r - \sum_{j \in \mathcal{V}} \sum_{p \in \mathcal{P}} a_{jp}^r \pi_{jp}\} \quad (2.48)$$

Determining the route $\tilde{a}^r \in A$ which satisfies (2.48) is an optimization subproblem which is described in Section 2.4.

2.4 Pricing subproblem formulation

Based on the DW reformulation, the pricing subproblem includes implicitly two optimization problems that have to be solved jointly: a routing problem and a loading problem. The formulation relies on variables x_{ij}^k , y_{jp}^{kw} and b_i^k defined in Section 2.2. We omit the index k when we write the decision variables since the pricing subproblem is solved for each vehicle k . To clarify the problem structure, we define the binary variable z_i which is equal to 1 if customer i is visited and 0 otherwise. The routing variables z_i and x_{ij} identify which customers to visit and in what order. Once a customer i is selected, the assignment variables y_{ip}^w specify the orders to deliver. We recall that π_{ip} is the dual variable associated with the delivery of order p for customer i . π_{ip} can be interpreted as the "profit" obtained through the corresponding delivery. Let $\bar{\mathcal{V}} = \{d, 1, \dots, n\}$ be the vertex set including the n customers and the vehicle associated depot d . The pricing subproblem can be written as follows:

$$\max \sum_{j \in \bar{\mathcal{V}}} \sum_{p \in \mathcal{P}} \sum_{w \in \mathcal{W}} \Pi_{jp} y_{jp}^w - \sum_{(i,j) \in \bar{\mathcal{E}}} \alpha_{ij} x_{ij} - \sum_{j \in \bar{\mathcal{V}}} \beta x_{d_o j} - \gamma (b_{d_d} - b_{d_o}) \quad (2.49)$$

subject to

$$\sum_{j \in \bar{\mathcal{V}}} x_{ij} = z_i \quad i \in \bar{\mathcal{V}} \quad (2.50)$$

$$\sum_{i \in \bar{\mathcal{V}}} x_{ij} = z_j \quad j \in \bar{\mathcal{V}} \quad (2.51)$$

$$\sum_{i \in \mathcal{S}, j \in \mathcal{S}, (i,j) \in \bar{\mathcal{E}}} x_{ij} \leq \sum_{l \in \mathcal{S} \setminus \{m\}} z_l \quad m \in \mathcal{S}, \mathcal{S} \subset \bar{\mathcal{V}} \setminus \{d\}, 3 \leq |\mathcal{S}| \leq n - 3 \quad (2.52)$$

$$b_j \geq b_i + s_i + t_{ij} - M_{ij}(1 - x_{ij}) \quad i, j \in \bar{\mathcal{V}} \quad (2.53)$$

$$e_i z_i \leq b_i \leq l_i z_i \quad i \in \bar{\mathcal{V}} \quad (2.54)$$

$$\sum_{(i,j) \in \bar{\mathcal{E}}} t_{ij} x_{ij} \leq T^{max} \quad (2.55)$$

$$\sum_{w \in \mathcal{W}} y_{ip}^w \leq z_i \quad i \in \bar{\mathcal{V}}, p \in \mathcal{P} \quad (2.56)$$

$$z_i \leq \sum_{p \in \mathcal{P}} \sum_{w \in \mathcal{W}} y_{ip}^w \quad i \in \bar{\mathcal{V}} \quad (2.57)$$

$$y_{ip}^w + y_{jq}^w \leq 1 \quad i, j \in \bar{\mathcal{V}}, w \in \mathcal{W}, (p, q) \in \mathcal{IP} \quad (2.58)$$

$$y_{ip}^w = 0 \quad i \in \bar{\mathcal{V}}, w \in \mathcal{W}, (p, w) \in \mathcal{IPC} \quad (2.59)$$

$$\sum_{i \in \bar{\mathcal{V}}} \sum_{p \in \mathcal{P}} y_{ip}^w q_{ip} \leq Q^w \quad w \in \mathcal{W} \quad (2.60)$$

$$x_{ij}, y_{ip}^w, z_i \in \{0, 1\} \quad i, j \in \bar{\mathcal{V}}, w \in \mathcal{W}, p \in \mathcal{P} \quad (2.61)$$

$$b_i \geq 0 \quad i \in \bar{\mathcal{V}}. \quad (2.62)$$

A feasible vehicle route aims to maximize the collecting profit while minimizing the incurred costs. Constraints (2.50)-(2.52) are routing constraints. They ensure the flow conservation and eliminate subtours. Time windows and maximum route duration are handled in a classical way by constraints (2.53)-(2.55). Constraints (2.52) are redundant in the presence of the constraints (2.53) but they strengthen the model (Cordeau et al., 2002a). Constraints (2.56)-(2.60) handle incompatibility relations and ensure that vehicle compartment capacities are respected as described in Section 2.2.

The pricing subproblem can be viewed as a bi-objective routing problem with two opposite criteria, one consists in visiting more customers to maximize the total profit collected and the other aiming to reduce the route costs by identifying a subset of served customers. Feillet et al. (2005) distinguish three generic single vehicle routing problems with profits gathered under the name of Traveling Salesman Problem with Profits (TSP with profits). Depending on the objective function, three classes of TSP with profits can be distinguished: (i) when both criteria are combined linearly in the objective function, the problem is so-called the Profitable Tour Problem (PTP) (Dell'Amico et al., 1995); (ii) when the profit is maximized and the travel costs have to be less than a preset value, the problem is referred to as the Orienteering Problem (OP) (Tsiligirides, 1984) or the selective traveling salesman problem; (iii) when the objective is to minimize the travel costs and the profit collected must exceed a preset lower bound g_{min} , the problem is called the Prize Collecting TSP (PCTSP) (Balas, 2007).

Many papers and book chapters are devoted to TSP with profits, since many industrial applications are modeled as such problems (e.g., (Laporte and Martello, 1990, Gendreau et al., 1998a, Fischetti et al., 2007, Balas, 2007)). Some surveys have appeared over the last three decades and attempt to review the variants and the ap-

plications of these problems and to differentiate them from other routing problems (see e.g., (Feillet et al., 2005, Vansteenwegen et al., 2011, Archetti et al., 2013)).

The pricing subproblem can be seen as a rich profitable tour problem with time windows and maximum route duration, referred to Rich Profitable Tour Problem (RPTP). The RPTP is an extension of the classical PTP in many ways. It can be considered as a time constrained capacitated profitable tour problem with multiple products and incompatibility constraints. Dealing with several complicated constraints encountered in common practical situations makes this problem more challenging than the classical PTP.

2.5 Conclusions

In this chapter we introduced the rich MDMCMCm-VRPTW modeled as a mixed-integer mathematical program. This three-index vehicle flow formulation has a weak linear relaxation. Therefore, a DW decomposition has been applied to improve the original formulation bounding. The problem reformulation leads to a master problem, modeled as a set covering formulation, and a subset of independent pricing subproblems. The set covering formulation cannot be solved with a straightforward branch-and-bound method since the number of possible routes grows exponentially with the number of customers orders. However, a column generation based algorithm may solve such large-scale problem. As a next step, we are interested in developing an efficient matheuristic to solve the pricing subproblem to be embedded later in a column generation scheme. The purpose of the matheuristic is to solve medium and large size instances of the RPTP, as it will be described in Chapter 4.

Chapter 3

Matheuristic design for a multi-constrained TSP with profits

In this chapter we describe the components of the proposed matheuristic for a multi-constrained TSP with profits. First, we review the papers addressing the flexibility and reusability concerns of VRP solution methods. Then, we describe the unified matheuristic proposed to solve the MDMCMCm-VRPTW for one vehicle. We motivate the choice for a variable neighborhood search (VNS) based approach. It combines several removal and insertion routing heuristics as well as efficient constraint checking procedures. The loading problem due to the use of a multi-compartment vehicle is addressed carefully. Two loading neighborhoods based on the solution of mathematical programs are proposed to intensify the search. They interact with the heuristic routing neighborhoods in the line of matheuristics.

Parts of this work at different stages were presented at the *Eight Triennial Symposium on Transportation Analysis* (TRISTAN VIII) (Lahyani et al., 2013a), at the *5th International Conference on Modeling, Simulation and Applied Optimization* (ICMSAO) (Lahyani et al., 2013c), as well as at ROADEF 2013 and ROADEF 2014 conferences (Lahyani et al., 2013b, 2014).

3.1 Introduction

In the VRP literature, some studies address variants of the Team Orienteering Problem with multi-constraints (multi-vehicle Orienteering Problem) (Garcia et al., 2010, Tricoire et al., 2010, Souffriau et al., 2013). However, no previous work has been devoted to solve multi-constrained TSPs with profits. There are few studies dealing with basic extended variants of TSPs with profits, e.g., the capacitated PTP (CPTP) (Archetti et al., 2009, Jepsen, 2011), or the OPTW (Righini and Salani, 2006, Tricoire et al., 2010, Labadie et al., 2011, 2012).

In this chapter, we propose a unified solution approach for a large class of TSPs with profits, ranging from pure academic problems to multi-constrained variants. We also investigate a loading subproblem arising when the vehicle has multi-compartment. To the best of our knowledge, there is only one work focusing on the loading subproblem arising in the MC-VRP (Pirkwieser et al., 2012a, Pirkwieser, 2012). The authors propose first-fit, best-fit and best-fit decreasing heuristics and a constraint programming algorithm to solve the problem of assigning products to compartments considering incompatibility constraints. Previous works either solve the loading problem implicitly (Derigs et al., 2011a) or consider a simpler scenario with two compartments and two products, each compartment being dedicated to one product (El Fallahi et al., 2008, Muyltermans and Pang, 2010a, Wang et al., 2014, Reed et al., 2014). In such cases, the loading problem reduces to checking capacity restrictions. Here, we follow the definition given by Pirkwieser et al. (2012a) and we call the loading subproblem the Compartment Assignment Problem (CAP).

The contributions in this chapter are threefold. First, we develop a unified matheuristic combining routing and loading neighborhoods. Second, we propose several insertion and removal heuristics. Last, we propose loading neighborhoods based on the solution of mathematical programs for the CAP.

This chapter is organized as follows. Section 3.2 reviews a selection of papers addressing flexibility and reusability concerns in VRP solution methods. Section 3.3 presents the main features of the matheuristic. Section 3.4 presents in detail the constructive heuristic, the routing and the loading neighborhoods as well as the route feasibility check procedures.

3.2 Describing available unified methods for VRPs

Unified methods are rather seldom in the VRP field despite of their academic and practical interest. Only a little attention has been paid to this challenging research area in the last decade. In (Vidal et al., 2013a, 2011), the authors provide a unified view and a transversal analysis on timing problems and their corresponding algorithms. Based on the knowledge developed through these surveys, the authors "*invented then a kind of Swiss knife*" for the routing problems, referred to as Unified Hybrid Genetic Search (UHGS). The UHGS distinguishes three classes of attributes impacting on three aspects of the VRP: the assignment of customers and routes to resources (ASSIGN), the sequence choices (SEQ), and the evaluation of fixed sequences (EVAL), see (Vidal et al., 2013a) for more details. The proposed method addresses efficiently up to 26 extensions of VRPs, such as the VRPTW and its variants, the MPVRP, the OVRP, the PVRP, the VRPB, the VRPPD, (Vidal et al., 2014) but it does not address a variety of constraints simultaneously. The most constrained VRP addressed by the UHGS is the vehicle routing and truck driver scheduling problem (VRTDSP) with different driver regulation rules. The authors consider that the VRTDSP is a RVRP. According to the definition proposed in Chapter 1, this problem has a score of 2 for the SCs and 7 for the PPCs and cannot be considered as rich.

Derigs and Vogel (2014) propose a unified matheuristic framework, based on combining 5 different neighborhood search heuristics, to solve 5 academic routing problems namely: the MCVRP, the PVRP, the SDVRP, the VRPTW and the truck and trailer routing problem (TTRP). Their work is based on the thesis of Ulrich Vogel (Vogel, 2012), in which the author starts by describing the architecture of this VRP solver for the case of standard VRP. Then he proposes an *adaptation layer* which allows him to modify the solution algorithm and the solution representation from one problem to another. The author provides some examples of adaptations to address complex and rich VRPs such as the DVRP, the MDVRP, the VRP with multiple use of vehicles, the VRPPD with time windows, the fleet size and mix VRP, etc. The author claims that these problems are rich but experiments have not been conducted explicitly in his thesis.

Subramanian (2012) and Pisinger and Røpke (2007) propose general heuristics and hybrid algorithms to solve a set of VRP variants. They adapt the proposed metaheuristic to tackle multiple variants of the VRP. Each variant can be formulated as a special case of the underlying problem. Obviously, such approaches depend

strongly on the generality of the first problem addressed and are not suitable when other constraints are present. Irnich (2008) provides a modeling framework covering a variety of standard and nonstandard VRP constraints. He proposed a giant-tour representation enabling a unified view of moves in local search procedures. The author claims that these search procedures can be used to solve RVRPs efficiently including time windows, pickup-and-delivery operations, heterogeneous fleet, compartment constraints, etc. However, solutions are presented only for the VRPTW and the multi-depot VRPTW as the author considers these problems as "prototypes" for RVRPs. This brief survey puts into highlights that there is a lack of unified and flexible approaches which are designed explicitly to address RVRPs and which can be easily enriched accordingly whenever further specific requirements arise.

In Figure 3.1, we present the design of the proposed unified framework. Figure 3.1 sketches the transformation of rich and basic VRPs instances to be solved by the unified matheuristic. For example, when an instance of the CVRP has to be solved, it is transformed into an equivalent MDMCMCm-VRPTW instance. The latter is solved by the proposed unified matheuristic without any customization. The solution obtained is then interpreted as a CVRP instance. Still, designing flexible solution methods for multi-constrained VRPs requires to rely on existing solution methods that have been applied to VRPs successfully as it will be shown in Section 3.3.

3.3 Matheuristic approach

Cordeau et al. (2002b) explain that good VRP heuristics require four attributes: *accuracy*, *speed*, *flexibility* and *simplicity*. Designing a unified method for RVRPs represents a considerable research challenge which makes difficult to meet these four criteria simultaneously. The underlying matheuristic proposed in this chapter will rather focus on flexibility and simplicity. Most of the VRP heuristics proposed to solve one variant concentrate rather on accuracy and speed. To maintain a good compromise between flexibility and simplicity, a special attention must be paid to the main components of the problem and of the solution.

The multi-attribute PTP is a *NP*-hard combinatorial problem since the OP is *NP*-hard (Golden et al., 1987). Moreover, Gendreau et al. (1998b) outline some reasons explaining the difficulty of designing high quality heuristics for the OP. Part of the trouble lies in the fact that profits associated with customers and the distances between them are independent and lead to define conflicting objectives. It is usually difficult to select the customers that are part of the solution and considering time

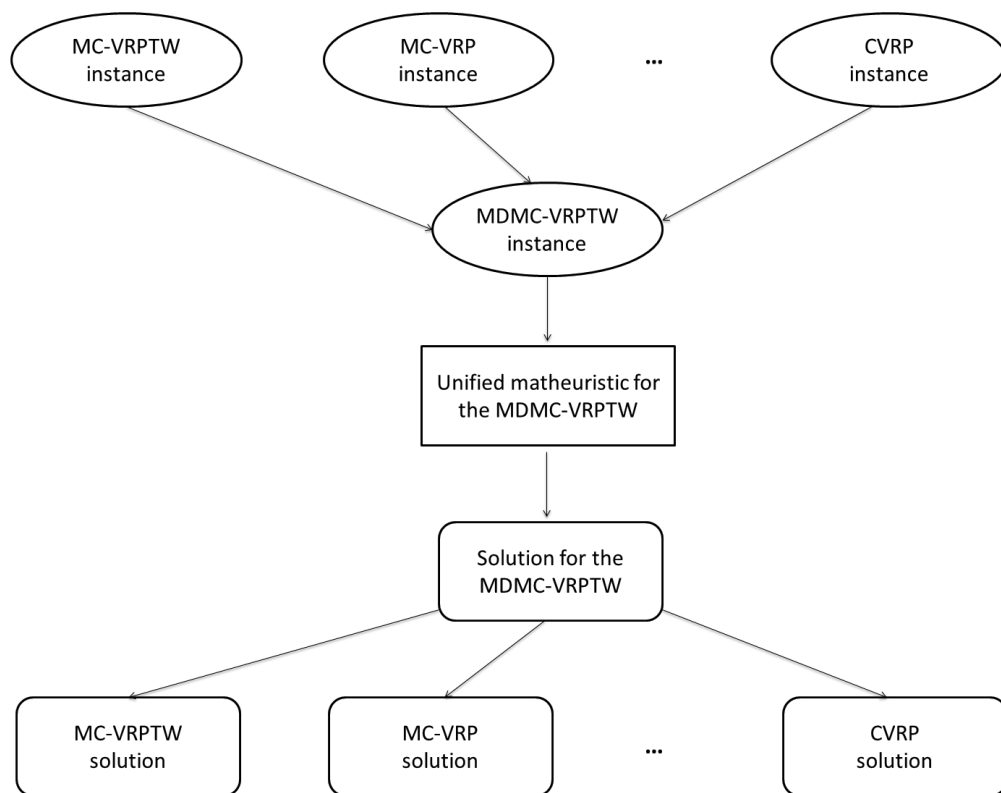


Figure 3.1: Transformation process for solving rich and basic VRPs with a unified matheuristic, adapted from [Røpke and Pisinger \(2006a\)](#)

window further complicates the solution process. Therefore, the design of a simple method providing high quality solution is quite challenging.

In this chapter, we propose a matheuristic based on a Variable Neighborhood Search (VNS) algorithm which includes exact procedures for the examination of some neighborhoods. We denote this method VNS*. Let us motivate our choice to select the VNS (Mladenovic and Hansen, 1997) to solve the RPTP. The VNS is an approach able to take into account a wide variety of constraints, which proves to be very efficient for solving complex VRPs (Wen et al., 2011, Schilde et al., 2011). This success may be explained either by its ability (i) to escape from local optima compared with population based metaheuristics; (ii) to exploit unexplored parts of the solution space by applying different problem specific neighborhoods. The basic steps of the proposed matheuristic are provided in Algorithm 1.

The proposed matheuristic performs two main steps. The first step corresponds to the constructive stage (lines 5-8 in Algorithm 1) and consists in identifying a good initial solution s . This solution is obtained thanks to an adapted Nearest Neighbor Algorithm (NNA) which tries to insert each order in the current circuit while satisfying the temporal, capacity and compatibility constraint (see Section 3.4.1). The second step, described in lines 8-25, is an attempt to diversify and improve the initial solution by applying VNS*. The key idea of such matheuristic is to exploit a systematically neighborhood structures during the search. Given an initial solution s , VNS* procedure alternates the perturbation phase and the improvement phase (see Sections 3.3.1 and 3.3.2) for a preset number $iter^{max}$ of iterations.

Figure 3.2 sketches the two main steps of the VNS*. It also illustrates an example of the solution representation for the remainder of this chapter. Let \mathcal{L} be the set including all customers i covered by route s and \mathcal{M} be the set including all removed customers i from s . The set \mathcal{M} is initially empty. We denote by $\mathcal{U} \subseteq \mathcal{O}$ the set including all the orders o_i^p placed by the customers $i \in \mathcal{V}$ and not covered by the current route s . For an insertion move, the current route may not include all orders o_i^p associated with customer $i \in \mathcal{V}$, whereas, a deletion move removes from the route all orders o_i^p placed by customer i . Since the orders placed by a customer $i \in \mathcal{U}$ are known and not delivered, we use interchangeably the expression *inserting customers* or *inserting orders* from the pool \mathcal{U} in the remainder of this chapter.

3.3.1 Perturbation phase

The perturbation phase, also known as the shaking phase, aims to guarantee a proper compromise between the diversification of the search and maintaining the promising parts of the incumbent solution, without spending too much computa-

Algorithm 1: Variable Neighborhood Search* (VNS*)

```
input : ( $s$ )/* (obtained after applying the constructive NNA) */
output: ( $s^*$ )
1 Let  $s^+$  be an intermediate solution;
2  $s^* \leftarrow s$ ;
3  $iter \leftarrow 1$ ;
4 while  $iter \leq iter^{max}$  do
5   if ( $iter \neq 1$ ) then
6     |  $s \leftarrow$  new built solution; /* (obtained by the multi-start NNA) */
7   end
8    $s^+ \leftarrow s$ ;
9   REDO:
10  for ( $i \leftarrow 1$  to 4) /* (Select a removal heuristic  $H_i^-$ ) */ do
11    | for ( $j \leftarrow 1$  to 6) /* (Select an insertion heuristic  $H_j^+$ ) */ do
12      | Apply perturbation phase on  $s$ ;
13      | Apply local search heuristic on  $s$ ;
14      | Apply 2_opt local search heuristic on  $s$ ;
15      | if ( $s$  is better than  $s^+$ ) then
16        | |  $s^+ \leftarrow s$ ;
17        | | goto REDO ;
18      | end
19    | end
20  end
21  if ( $s^+$  is better than  $s^*$ ) then
22    |  $s^* \leftarrow s^+$  ;
23  end
24  Apply loading based improvement on  $s^*$ ; /* (The Linear Multiple
    Knapsack Problem with Conflicts) */
25   $iter ++$  ;
26 end
27 Waiting_Time_Optimization ( $s^*$ );
28 return  $s^*$ ;
```

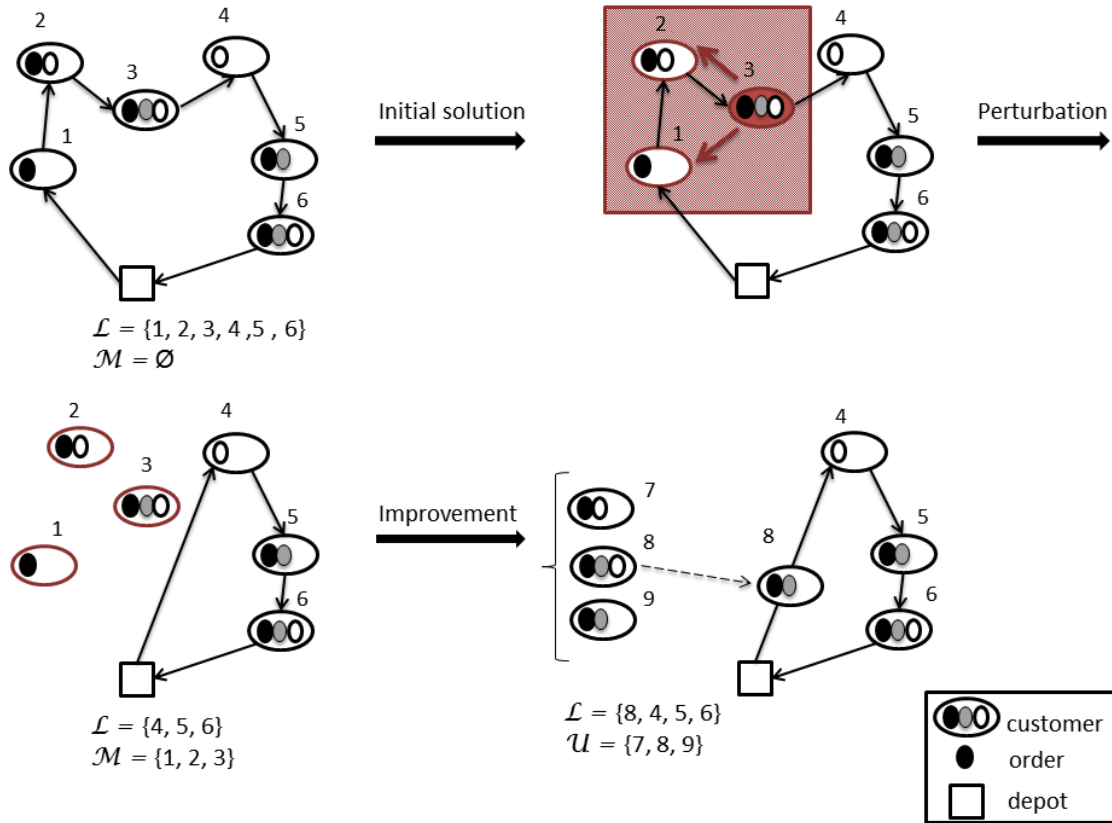


Figure 3.2: Representing the VNS* general behavior

tional time. For this reason, we propose to implement the perturbation phase as an adapted Large Neighborhood Search (LNS) heuristic (Shaw, 1997, 1998). The perturbation phase of the VNS* consists in partially destroying the current solution without trying to reinsert removed customers again. Rather than using one large neighborhood as in a standard LNS, we consider a set of four removal heuristics H_i^- , $i \in \{1, \dots, 4\}$: the similarity removal heuristic, the random removal heuristic, the worst profit removal heuristic and the spatio-temporal removal heuristic (see Section 3.4.2.1 for more details). Each removal neighborhood eliminates a number of kd customers from the current solution s according to a predefined criteria. The perturbation phase is followed by local search procedures aiming to gradually improve the shaken solution.

3.3.2 Improvement phase

The local search approach iteratively improves the solution obtained from the perturbation phase by moving from neighbors of the current solution to local optima.

Two important issues must be addressed when local search heuristics are designed: (i) the quality of the solutions obtained (ii) the complexity of the local search heuristic. An efficient local search heuristic should reach a good balance. The Variable Neighborhood Descent (VND) (Hansen and Mladenovic, 2003) satisfies both criteria and has been applied successfully to many optimization problems. It enables to reach global optima more probably than with a single neighborhood structure. In this thesis, we propose a VND which considers alternatively routing based neighborhoods and loading based neighborhoods. The proposed VND includes three different local search procedures described in what follows.

Local search heuristic : The local search heuristic, denoted for short by (LS), is a modified extension of the Ruin and Recreate paradigm introduced by Schrimph et al. (2000). It consists in removing randomly one or more customers from the current solution and inserting unvisited customers' orders from pool \mathcal{U} , unlike the Ruin and Recreate procedure which tries to re-insert the removed customers in better positions. The LS is a first improvement heuristic, i.e., the search procedure stops as soon as a solution improving the current objective value is found. We propose six sequential insertion heuristics, denoted by H_i^+ , $i \in \{1, \dots, 6\}$. Since the objective in the RPTP combines the total profit, the total travel costs and the total waiting time costs, the insertion criterion proposed either ensure a good trade-off between two or three of these terms or focus on a single selection measure. Given an insertion heuristic, each order is considered sequentially to select the best feasible insertion. The selected order is then cost-effectively inserted in the solution. Each local search step is parameterized according to the number of customers to remove, kd , the maximum number of customers to insert, kc , and the position in the route from which customer(s) may be removed.

2-opt Local Search : This heuristic attempts to reoptimize the solution obtained by the LS heuristic by decreasing the total travel time. First, it removes two arcs from a given route and reconnects the route by inserting two other arcs. When the time window constraints are considered, the orientation of the path may be reversed. The evaluation of the solution feasibility with respect to time window constraints is ensured by an effective time feasibility algorithm described in Section 3.4.3.2. If the new feasible solution is better than the current best solution in terms of total travel time, the procedure is reiterated. The algorithm stops in a local optima when no 2-opt exchange is possible. When an improved solution is identified, the heuristic

tries to insert new customers from the pool of unvisited customers \mathcal{U} given an insertion heuristic H_i^+ . The acceptance criterion consists in accepting all the solutions improving the objective value. Note that feasible 2-opt exchanges are seldom found since time window constraints are considered. Nevertheless, experimental results demonstrate that it is worthy to keep this improvement technique since it improves the current solution.

Loading Based Improvement : Delivering a set of products using a multi-compartment vehicle while satisfying incompatibilities constraints between products and between products and compartments results in an *NP*-hard packing problem, referred to as the CAP. This single vehicle packing problem is a key feature of the RPTP. We propose two exact loading neighborhoods based on the solution of two mathematical programs: the Quadratic Multiple Knapsack Problem with Conflicts and the Linear Multiple Knapsack Problem with Conflicts. These neighborhoods, proposed in Section 3.4.2.2, aim to reoptimize the loading plan of the current solution. The first neighborhood explores the search space by swapping products between compartments. The second one ensures more intensification by exchanging some loaded orders with more attractive ones from the pool \mathcal{U} .

Note that, given an empty solution s , solving the CAP requires an assignment heuristic able to insert orders in a feasible way. We propose a fast and effective solution procedure providing feasible assignments of orders while satisfying the loading constraints. Section 3.4.3.1 is devoted to describe this heuristic.

3.4 Main features of the matheuristic approach

3.4.1 Multi-start constructive heuristic

Perttunen (1994) proves that using initial solutions generated by constructive heuristics outperforms randomly generated initial solutions for TSPs. In this section, we propose a simple and fast NNA able to construct a feasible solution s by inserting orders from \mathcal{U} . Starting from an empty route, the NNA tries to insert all orders o_i^p of customer i , $p \in \mathcal{P} = \{1, \dots, P\}$ in the last position of the route as long as the temporal, capacity and compatibility restrictions are satisfied. The insertion is repeated until no feasible insertion exists. We consider the insertion of order o_j^p associated with customer j (and profit g_j^p), after an order o_i^p associated with customer i in the current route s . An insertion is evaluated with the coefficient \hat{c}_{ij}^p , expressed in equation (3.3). Let g_{crt} , d_{crt} , t_{crt} denote respectively the current total profit, the

current total distance and the current total time of the route. d_{ij} corresponds to the distance between customers i and j . Let $shift_{t_{ij}}^{NNA}$ and $shift_{d_{ij}}^{NNA}$ be the time increase and the distance increase if order σ_j^p is to be inserted at the end of the route s , expressed by equations (3.1) and (3.2). $w_j = \max\{0, e_j - a_j\}$ is the waiting time in j and a_j corresponds to the vehicle arrival time at customer j .

$$shift_{d_{ij}}^{NNA} = d_{ij} + d_{j0} - d_{i0} \quad (3.1)$$

$$shift_{t_{ij}}^{NNA} = t_{ij} + w_j + t_{j0} - t_{i0} \quad (3.2)$$

$$\hat{c}_{ij}^p = \frac{(d_{crt} + shift_{d_{ij}}^{NNA} + t_{crt} + shift_{t_{ij}}^{NNA})}{(g_{crt} + g_j^p)} \quad (3.3)$$

The NNA repeatedly selects as new vertex the vertex having the lowest coefficient \hat{c}_{ij}^p . This criterion promotes vertices with high profits and generating low time and distance increases. Such criterion has been experimentally proved to be more efficient than other criteria focusing either on time, distance or profit. The complexity of this heuristic is $O(n)$ since only the last position of the current route is considered when an insertion move is evaluated.

The metaheuristics based on local optimization, usually need some diversification to escape from local optimality. To achieve diversification, we propose a multi-start NNA starting from a new solution once a region of the solution space has been explored. The NNA is applied $iter^{max}$ times. The first solution is obtained applying the NNA as described above. For the remaining iterations, the score matrix g_i^p associated with order σ_i^p and the distance matrix between all the customers $i, j \in \bar{V}$ are randomized. The randomization is controlled by a parameter σ which varies between $[1/2, 3/2]$.

3.4.2 Neighborhoods

The proposed matheuristic may be seen as an iterative sequence of *Ruin and Recreate* steps since it destroys a part of the solution in the shaking phase and repairs it by means of the local search procedures. The search may be guided either for diversification, i.e., examining new regions of the solution space, or for intensification, i.e., focusing on promising regions. The neighborhoods developed for the RPTP solution may be classified in two categories:

- Routing neighborhoods: They modify the sequence route and the customers visited in a given solution. They include six insertion heuristics and four

removal heuristics. Some of these heuristics are adapted from the VRP literature. They may require some parameters denoted by Greek letters which will be fixed in Section 4.2 in Chapter 4.

- Loading neighborhoods: We propose two loading neighborhoods with different objectives. The first neighborhood attempts to reoptimize the compartments loading while maintaining the route sequence unchanged, while the second one removes less attractive orders from the route and replace them by more profitable orders without inserting any new customers.

The variety of the proposed neighborhoods allows a good exploration of the solution space as it is discussed in Section 4.2 in Chapter 4.

3.4.2.1 Routing neighborhoods

3.4.2.1.a Removal heuristics

The first three removal heuristics are adapted from the literature while the last one is new. A pseudo-code outlining the generic removal approach is presented in Algorithm 2. It takes as input an initial feasible solution s and returns a partially destroyed solution s^* . It removes kd customers according to a predefined removal operator. Let \mathcal{L}_i be the i th customer of \mathcal{L} . It is worthwhile to mention again that removal heuristics eliminate orders associated with a customer from a route. The process is controlled by the parameter ψ and the randomization parameters ϕ .

Similarity removal heuristic : This removal heuristic was proposed by (Shaw, 1997, 1998) and implemented by (Røpke and Pisinger, 2006b, Pisinger and Røpke, 2007, Ribeiro and Laporte, 2012, Demir et al., 2012). The aim of the similarity removal heuristic is to remove a set of customers that are similar with regards to a predefined similarity measure. Removing similar customers promotes the insertion of more customers which may lead probably to better solutions. We define the criterion $Sim(i, j)$ between two customers i and j , $Sim(i, j) = \phi_1 d_{ij} + \phi_2 |b_i - b_j| + \phi_3 |shift_i^g - shift_j^g|$. The similarity criterion includes three terms. The first one represents the distance d_{ij} between i and j . Temporal similarity is expressed through the difference between the departure times in i and j . The third term measures the difference of attractiveness between i and j . $shift_i^g$ denotes the contribution of customer i to s in terms of profit, such that $shift_i^g = \sum_{p \in \mathcal{P}} g_i^p, \forall \sigma_i^p \notin \mathcal{U}$. ϕ_1, ϕ_2 and ϕ_3 are normalized weights.

The heuristic initially selects a customer i randomly and removes it from the solution s . For the subsequent $kd - 1$ iterations, the heuristic selects customers that are

Algorithm 2: Generic structure of removal heuristics

input : $(s, kd, \mathcal{L}, \psi \geq 1)$
output: (s^*)

- 1 Initialize $\mathcal{M}, \mathcal{M} \leftarrow \emptyset$;
- 2 Eliminate randomly a vertex i from s ;
- 3 $\mathcal{M} = \{i\}$;
- 4 **while** $(|\mathcal{M}| \leq kd)$ **do**
- 5 Select randomly a customer i from \mathcal{M} ;
- 6 Update \mathcal{L} ;
- 7 Compute the removal ratio;
- 8 Sort all the customers $j \in \mathcal{L}$ according to the removal ratio;
- 9 Generate a random number $\phi \in [0, 1]$;
- 10 $j \leftarrow \lfloor \phi^\psi |\mathcal{L}| \rfloor$;
- 11 Eliminate randomly \mathcal{L}_j from s ;
- 12 $\mathcal{M} = \mathcal{M} \cup \{\mathcal{L}_j\}$;
- 13 **end**
- 14 Remove from s customers in \mathcal{M} , let s^* be the resulting solution;
- 15 **return** s^* ;

similar to the removed vertex according to $Sim(i, j)$. A lower value of $Sim(i, j)$ indicates that i and j are similar. In line 8 of Algorithm 2 the vertices in \mathcal{L} are sorted in an increasing order according to $Sim(i, j)$. Some randomness is introduced while selecting the customers thanks to the parameter ϕ . Given a value of ϕ , a low value of ψ ($\psi = 1$) corresponds to complete determinism and implies removing vertex j with a low value of $Sim(i, j)$ while the probability of choosing a vertex j much less similar to i increases with the increasing value of ψ . The time complexity of this heuristic in the worst cases is $O(n^2)$. Figure 3.3 summarizes the principal steps of this heuristic where $kd = 4$. The similarity removal heuristic selects randomly from the solution s customer 3 and puts it in the pool \mathcal{M} . Then it computes the $Sim(i, j)$ value between customer 3 and all the remaining customers in s . Customers 2, 4 and 5 are the most similar ones to customer 3. Given the values of ϕ and ψ , the heuristic removes customer 2 from the solution. The heuristic reiterates until 4 customers (3, 2, 4, 5) are removed from the solution s .

Random removal heuristic : This neighborhood consists in randomly removing kd customers. It may be seen as a special case of the similarity removal heuristic with $\phi_1 = \phi_2 = \phi_3 = 0$. This heuristic may be implemented in $O(1)$.

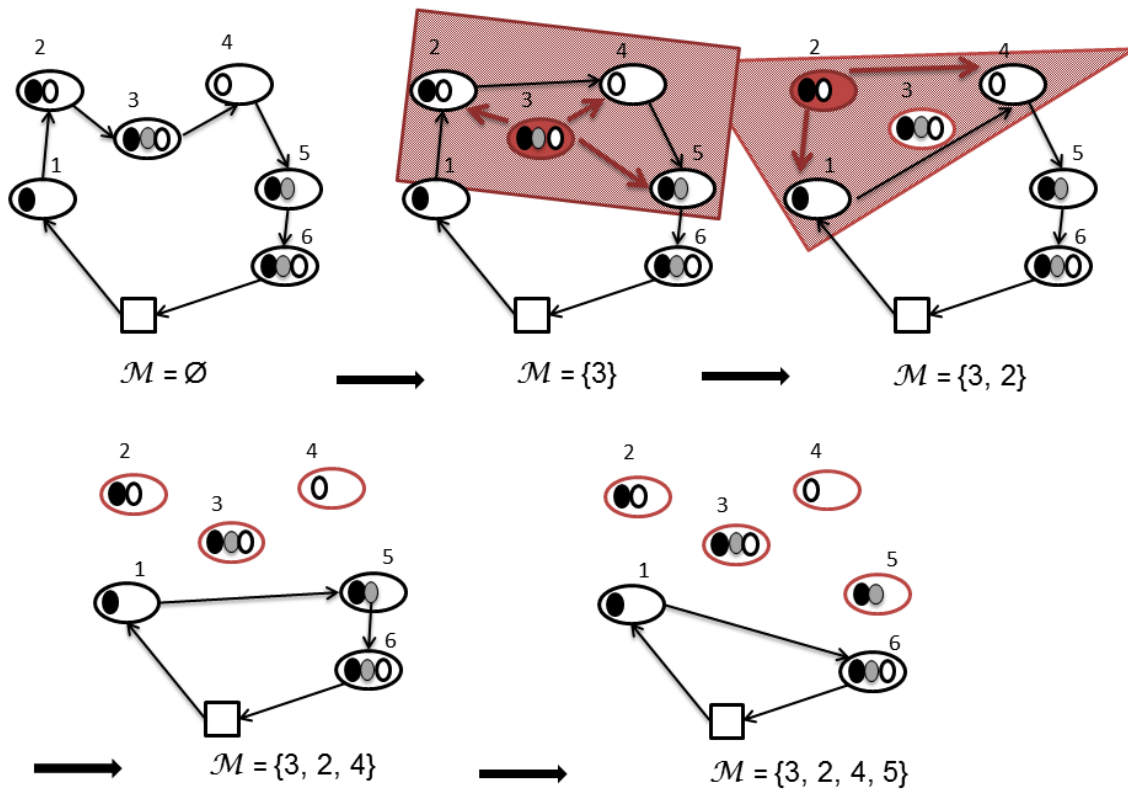


Figure 3.3: Similarity removal heuristic

Spatio-temporal removal heuristic : This heuristic is another adaptation of the removal heuristic proposed by (Shaw, 1997, 1998). It aims to remove customers similar in terms of distance and time. It differs from the similarity removal neighborhood by its removal criterion. The spatio-temporal ratio was proposed by Prescott-Gagnon et al. (2009) and is defined by equation (3.4).

$$ST_{ij} = \frac{1}{\left(\frac{d_{ij}}{d_i^{max}} + \frac{1}{T_{ij}^{ST} + T_{ji}^{ST}}\right)} \quad (3.4)$$

We set $d_i^{max} = \max_{j \in s} \{d_{ij}\}$, $i \in s$. If customer i is visited immediately before customer j , T_{ij}^{ST} measures the proximity of time windows between i and j and is equal to the expression $\max\{1, \min\{l_j, l_i + t_{ij}\} - \max\{a_j, a_i + t_{ij}\}\}$ (Gendreau et al., 1995). The main steps in this heuristic are those described in Algorithm 2. The customers served by the current route are sorted in decreasing order with respect to ST_{ij} value. The larger ST_{ij} is, the closer are the two customers i and j . The heuristic removes node $j = \operatorname{argmax}_{j \in s} \{ST_{ij}\}$. The removal move is also randomized and controlled by parameters ϕ and ψ . The spatio-temporal removal is implemented in $O(n^2)$ in the worst case.

Worst profit removal heuristic : The worst profit removal heuristic iteratively removes the less attractive customers according to the profit. The idea is to select the customers that do not contribute enough to the total solution profit. First, all the customers $i \in \mathcal{L}$ are sorted in increasing order according to the total profit of the served orders $shift_i^g$. Then, the heuristic selects a new customer with the lowest $shift_i^g$. Some randomization is ensured by parameters ϕ and ψ . This heuristic is implemented in $O(n)$.

3.4.2.1.b Insertion heuristics

The four insertion heuristics described below are used either to repair a partially destroyed solution or to improve a current solution. The first four heuristics are adapted from the OPTW literature, whereas the last two are new. As a general rule, the heuristic considers each order o_i^p from \mathcal{U} and tries to insert it in the best possible position according to a predefined insertion ratio. The whole process is repeated for kc iterations. After each insertion, the solution has to be updated. Obviously, the orders of one customer are served in sequence, i.e. during a single visit to the customer. In the solution process such sequences are enforced by null distances and null times between these orders. In addition, the insertion of orders of

customers partially favored would be ensured by the NNA and the second loading neighborhood. Therefore, insertion heuristics aim to consider the insertion of new customers. In each insertion heuristic, the feasibility of an insertion move is checked before it is implemented. Since the checking feasibility algorithms have different complexities, we order them. The most computationally consuming checking algorithm is the last invoked. The worst case time complexity of the six heuristics is in $O(n^2)$. The generic pseudo-code is shown in Algorithm 3.

Algorithm 3: Generic structure of insertion heuristics

```

input :  $(s, kc)$ 
output:  $(s^*)$ 
1 while  $(kc \geq 0)$  do
2   for  $(i \leftarrow 1$  to  $|\mathcal{U}|)$  do
3     if (the compatibility and capacity constraints are satisfied) then
4       for  $(j \leftarrow 1$  to  $|\mathcal{L}|)$  do
5         if  $(e_i + t_{ij} \leq l_j \text{ and } t_{crt} + t_{ij} \leq T^{max})$  then
6           Compute the insertion ratio;
7           Memorize and update the best position  $best\_pos$ ;
8            $i^* \leftarrow i$ ;
9         end
10      end
11    end
12  end
13  Insert the most attractive  $i^*$  in  $best\_pos$ ; /* (Best insertion) */
14  if (the temporal feasibility of  $s^*$  is maintained) then
15    | Update  $s^*$ ;
16  else
17    | Delete  $i^*$  from  $s^*$ ;
18  end
19   $kc - -$ ;
20 end
21 return  $s^*$ ;

```

Insertion heuristic (1) : This insertion heuristic is adapted from the constructive heuristic proposed by [Labadie et al. \(2011\)](#) for the single-product OPTW. It consists in considering over all orders in \mathcal{U} and in trying to insert new ones as long as possible while maintaining the feasibility of capacity, temporal and incompatibilities constraints. Given a current route s , the heuristic examines all the feasible positions of the order σ_i^p in s and inserts it, providing the best compromise between profit and time increasing. An insertion of candidate order σ_i^p in route s between

orders o_j^p and o_{j+1}^p results in a new route s^* . $shift_i^t$ corresponds to the route time increasing such that $shift_i^t = t_{ji} + w_i^s + t_{i,j+1} + w_{j+1}^{s^*} - t_{j,j+1} - w_{j+1}^s$ with $w_{j+1}^{s^*}$ corresponds to the waiting time incurred at customer $j+1$ in route s^* . In a first attempt, the defined ratio favors choosing customers having more reachable neighbors. Let E_j correspond to the set of reachable customers i from j such that $e_j + t_{ji} \leq l_i$ and $t_{crt} + t_{ji} < T^{max}$. The selected customer is $i^* = \operatorname{argmax}_{o_i^p \in U} \left\{ \frac{|E_i| * g_i^p}{shift_i^t} \right\}$. Preliminary results show that favoring customers having many temporal reachable neighbors may not lead to find a successor vertex that respect all constraints. This could be explained by the fact that the definition of a reachable neighbor should not rely only on temporal feasibility but also on physical feasibility. Then, we define the best insertion i^* such that $i^* = \operatorname{argmax}_{o_i^p \in U} \left\{ \frac{g_i^p}{shift_i^t} \right\}$.

Insertion heuristic (2) : This heuristic was proposed first by [Vansteenwegen et al. \(2009a\)](#) for the Team OPTW (TOPTW) and by [Garcia et al. \(2010\)](#) for the VRPTW. The ratio proposed by [Vansteenwegen et al. \(2009a\)](#) is equal to $\frac{(g_i^p)^2}{shift_i^t}$. Preliminary results show that this ratio is not of interest for the RPTP since the solution quality depends on other attributes. We should consider insertion principle considering the time, distance and capacity attributes. Let $shift_i^d$ denotes the distance increase if order o_i^p have to be inserted. The ratio gives the increase for each attribute on the current route with respect to the available quantity. The insertion ratio has to be computed as expressed by expression (3.5). The best insertion corresponds to the order with the highest ratio.

$$\frac{(g_i^p)^2}{\frac{shift_i^t}{availableTime} + \frac{q_i^p}{availableCapacity} + \frac{shift_i^d}{availableDistance}} \quad (3.5)$$

Insertion heuristic (3) : This insertion heuristic is adapted from the two-phase heuristic proposed by [Atahran \(2012\)](#) in his thesis for the DARP with time window. This heuristic consists in sequentially adding orders to a given route considering both the spatial and temporal closeness of customers, while keeping feasibility of the route. Assume that vertex i is defined by its geographical coordinates x_i and y_i and its associated time window $[e_i, l_i]$. We define a third temporal dimension $z_i = \frac{1}{2}(e_i + l_i)$. Each vertex i is represented by a three-dimensional vector $[x_i, y_i, z_i]$. Then, we transform the coordinates x_i , y_i and z_i to put them on a comparable scale. We consider the centered and scaled values \bar{x}_i , \bar{y}_i , \bar{z}_i and \tilde{x}_i , \tilde{y}_i , \tilde{z}_i respectively. Considering the coordinate x , we define $\bar{x}_i = \frac{1}{n} \sum_{i=1}^n x_i$ and $\sigma_{x_i} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x}_i)^2}$. Then, each x_i is replaced by its centered reduced value $\tilde{x}_i = \frac{x_i - \bar{x}_i}{\sigma_{x_i}}$. We proceed in a similar way for

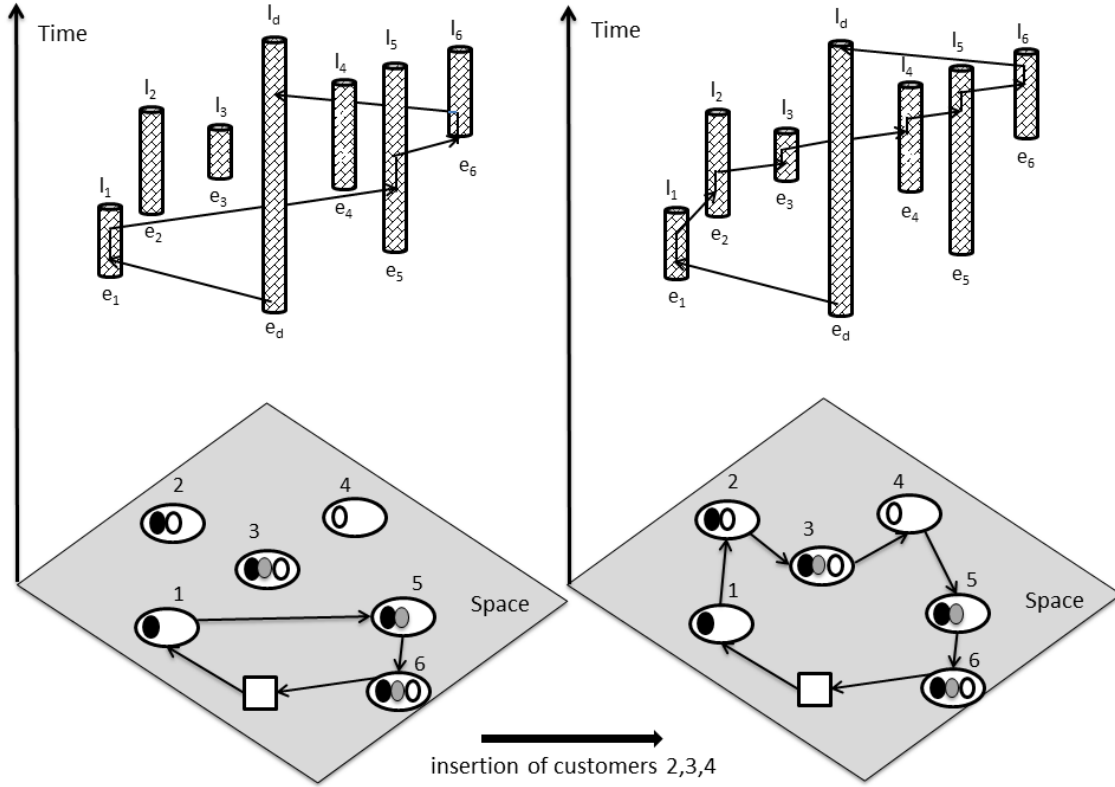


Figure 3.4: Spatio-temporal insertion heuristic

coordinates y_i and z_i . To measure the distance between two customers i and j , we use the euclidean metric defined by $d(i, j) = \sqrt{(\tilde{x}_i - \tilde{x}_j)^2 + (\tilde{y}_i - \tilde{y}_j)^2 + (\tilde{z}_i - \tilde{z}_j)^2}$. Figure 3.4 illustrates an insertion with a spatio-temporal dimension where cylinders correspond to time windows. Customers 2, 3 and 4 are close geographically and can be served in the same time slot.

Insertion heuristic (4) : This heuristic is a modified version of the best insertion heuristic proposed by Solomon (1987). It proceeds in two steps. First, for each order σ_i^p from \mathcal{U} , the heuristic computes the best feasible insertion, which is the one that involves the lowest distance increase. Then, the heuristic selects the best feasible insertion such that the extra time is minimized.

Insertion heuristic (5) : This heuristic inserts orders with the best compromise between the total insertion cost and the profit . The proposed ratio considers the difference between the three hierarchical objectives. For $i \in \mathcal{U}$ and $p \in \mathcal{P}$, the best order insertion $i^* = \operatorname{argmin}_{\sigma_i^p \in \mathcal{U}} \left\{ \frac{\text{shift}_i^t + \text{shift}_i^d}{g_i^p} \right\}$.

Insertion heuristic (6) : This heuristic focuses on the most important attribute in the RPTP objective function which is the profit. It selects orders with the highest profits while satisfying the route feasibility and attempts to insert them in s in the first feasible positions.

3.4.2.2 Loading neighborhoods

Quadratic Multiple Knapsack Problem with Conflicts : This neighborhood is an exact product rearrangement method. It consists in optimizing the loading plan of the vehicle. It aims to minimize the number of compartments used, while maintaining the route sequence unchanged, (see Figure 3.5). We model this problem as a Quadratic Multiple Knapsack Problem with incompatibility constraints between products and products and products and compartments. This problem is *NP*-hard, (Golumbic, 2004). Let $\mathcal{O}^s \subseteq \mathcal{O}$ corresponds to the set of orders covered by the route s . To simplify the formulation, we refer to order o_i^p served by route s by o . With each order o , is associated a customer $i \in \mathcal{L}$, a quantity q_o and a product $p_o \in \mathcal{P}$. Binary variables x_{ow} indicate whether or not order o is loaded in compartment w . The problem considered is:

$$\max \sum_{w \in \mathcal{W}} (Q^w - \sum_{o \in \mathcal{O}^s} x_{ow} q_o)^2 \quad (3.6)$$

subject to

$$\sum_{o \in \mathcal{O}^s} x_{ow} q_o \leq Q^w \quad w \in \mathcal{W} \quad (3.7)$$

$$\sum_{w \in \mathcal{W}} x_{ow} = 1 \quad o \in \mathcal{O}^s \quad (3.8)$$

$$x_{ow} + x_{kw} \leq 1 \quad o \in \mathcal{O}^s, k \in \mathcal{O}^s, w \in \mathcal{W}, p \in \mathcal{P}, q \in \mathcal{P}, \\ (p_o, q_k) \in \mathcal{IP} \quad (3.9)$$

$$x_{ow} = 0 \quad o \in \mathcal{O}^s, w \in \mathcal{W}, (p_o, w) \in \mathcal{IPC} \quad (3.10)$$

$$x_{ow} \in \{0, 1\} \quad o \in \mathcal{O}^s, w \in \mathcal{W}. \quad (3.11)$$

Defining a suitable objective function in any neighborhood search is a key factor. In this case, the maximization of the compartments loading or the maximization of the residual compartments capacity is useless it provides the same objective value (see example in Figure 3.5). The proposed neighborhood aims to minimize the number of compartments used. The objective function should focus then on filling efficiently used compartments to keep other compartments unused. The proposed objective function consists in maximizing the sum of squares of the residual compartments

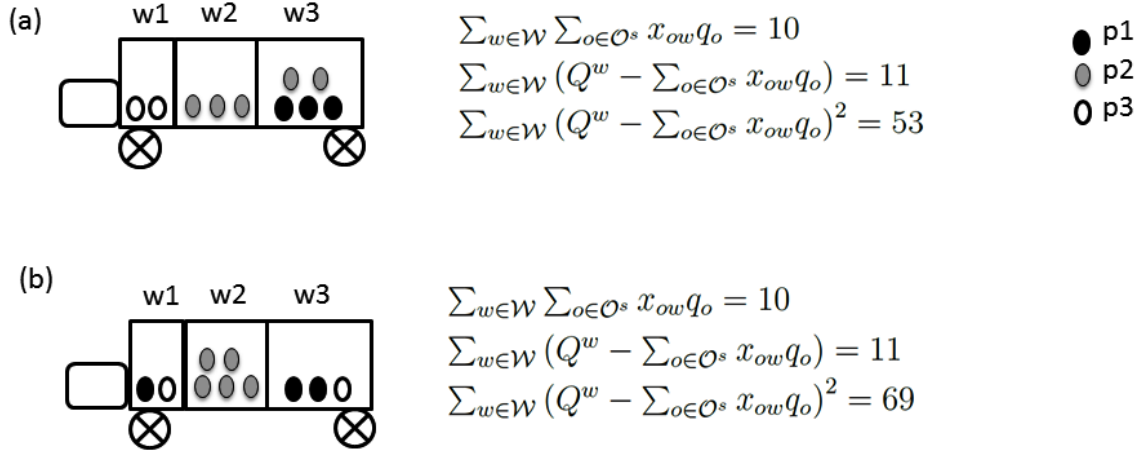


Figure 3.5: Example of a solution based on the Quadratic Multiple Knapsack Problem with Conflicts

Suppose $\mathcal{P} = \{p1, p2, p3\}$, $\mathcal{O}^s = \{1, \dots, 10\}$, $q_o = 1 \forall o \in \mathcal{O}^s$, $\mathcal{W} = \{w1, w2, w3\}$, $Q^{w1} = 3$, $Q^{w2} = 7$, $Q^{w3} = 11$, $\mathcal{IP} = \{(p2, p3)\}$ and $\mathcal{IPC} = \{(1, w2, p1)\}$. Figure (3.5a) corresponds to the current loading of a given solution s and Figure (3.5b) corresponds to a solution based on solving the Quadratic Knapsack Problem with Conflicts.

capacity, as it is expressed by (3.6). Constraints (3.7) ensure that compartment capacities are respected. Constraints (3.8) impose that each order $o \in \mathcal{O}^s$ is placed in exactly one compartment $w \in \mathcal{W}$. Constraints (3.9) - (3.10) express the incompatibility relations between products and between products and compartments. This model is solved thanks to the commercial solver IBM CPLEX 12.5.

Multiple Knapsack Problem with Conflicts : In order to maximize the solution profit, this neighborhood aims to insert orders, placed by customers served in the current solution s but not covered by the route s , and to remove less attractive orders from the current solution s . It is considered as a products reallocation method and modeled as a Multiple Knapsack Problem with incompatibility constraints between products and products and compartments. This problem is known also as the disjunctively constrained Knapsack Problem (Pferschy and Schauer, 2009) and is *NP*-hard. Let $\bar{\mathcal{O}}$ be the set of orders \bar{o}_i^p , placed by customers $i \in \mathcal{L}$. For sake of simplicity, we refer to \bar{o}_i^p by \bar{o} with which is associated with a product $p_{\bar{o}} \in \mathcal{P}$, a customer $i \in \mathcal{L}$, a quantity $q_{\bar{o}}$ and a profit $g_{\bar{o}}$. The decision variable $y_{\bar{o}w}$ is equal to one if order \bar{o} is loaded in compartment w and zero otherwise. Now we consider the following integer linear program:

$$\max \sum_{w \in \mathcal{W}} \sum_{\bar{o} \in \bar{\mathcal{O}}} y_{\bar{o}w} g_{\bar{o}} \quad (3.12)$$

subject to

$$\sum_{\bar{o} \in \bar{\mathcal{O}}} q_{\bar{o}} y_{\bar{o}w} \leq Q^w \quad w \in \mathcal{W} \quad (3.13)$$

$$\sum_{w \in \mathcal{W}} y_{\bar{o}w} \leq 1 \quad \bar{o} \in \bar{\mathcal{O}} \quad (3.14)$$

$$y_{\bar{o}w} + y_{kw} \leq 1 \quad \bar{o} \in \bar{\mathcal{O}}, k \in \bar{\mathcal{O}}, w \in \mathcal{W}, p \in \mathcal{P}, q \in \mathcal{P}, (p_{\bar{o}}, q_k) \in \mathcal{IP} \quad (3.15)$$

$$y_{\bar{o}w} = 0 \quad \bar{o} \in \bar{\mathcal{O}}, w \in \mathcal{W}, (p_{\bar{o}}, w) \in \mathcal{IPC} \quad (3.16)$$

$$y_{\bar{o}w} \in \{0, 1\} \quad \bar{o} \in \bar{\mathcal{O}}, w \in \mathcal{W}. \quad (3.17)$$

The objective expressed by constraints (3.12) maximizes the collected profit by placing the maximum orders \bar{o} in compartment w . Constraints (3.13) - (3.17) may be interpreted similarly to constraints (3.7) - (3.11). The model is solved using the commercial integer programming solver IBM CPLEX 12.5 and the performance on small and large instances is discussed in Section 4.2.3 in Chapter 4.

3.4.3 Route feasibility check

To speed up the solution evaluation and the cost computation, it is crucial to implement feasibility check algorithms efficiently. This section is devoted to two feasibility checks algorithms and one route-optimization heuristic. The feasibility checking routine proceeds as follows:

- The capacity and incompatibility constraints are checked first through the *Get_Compartment* heuristic, which is called in each insertion move. This heuristic provides a feasible order assignment to compartments, if it exists.
- The *TW_Feasibility_Check* is an exact algorithm checking the feasibility of the route with regards to time windows. It is called after each removal heuristic or insertion heuristic.
- The purpose of the *Waiting_Time_Optimization* exact algorithm is to minimize the waiting time in a given solution. This algorithm is called at the end of the search, it postpones the departure time from the depot as much as possible.

3.4.3.1 Loading feasibility check

A fast and effective assignment heuristic is required to solve the CAP as it must be solved frequently. The *Get_Compartment* heuristic first assesses the feasibility of

a potential order insertion with respect to capacity and incompatibility constraints. Then, it assigns the order to a compartment considering compartments according to their increasing residual capacity. The *Get_Compartment* heuristic is able to identify a feasible assignment for a new order, given an initial solution. The principal steps of this heuristic are summarized in pseudo-algorithm 4.

Algorithm 4: Get_Compartment heuristic

input : $(s, o \in \mathcal{U})$
output: (compartment index)

- 1 Let *Sort* be the table including the compartments index sorted according to the decreasing residual capacity in s ;
- 2 **for** $(i \leftarrow 1$ **to** $|\mathcal{W}|)$ **do**
- 3 **if** $((p_o, \text{Sort}[i]) \notin \mathcal{IPC})$ and $(q_o < \text{residual capacity}[\text{Sort}[w]])$ and $(Q[\text{Sort}[i]] \neq 0)$ and $((p_o, \text{all the orders loaded in } \text{Sort}[i]) \notin \mathcal{IP})$ **then**
- 4 **return** $\text{Sort}[i]$;
- 5 **end**
- 6 **end**
- 7 **return** -1 ; /* (No feasible assignment exists) */

3.4.3.2 Temporal feasibility check and optimization

The fundamental algorithms for handling time windows in routing problems were proposed by [Savelsbergh \(1992\)](#). An improved version of these algorithms was introduced by [Cornillier et al. \(2009\)](#). The *TW_Feasibility_Check* and *Waiting_Time_Optimization* algorithms are both based on the algorithms proposed by [Cornillier et al. \(2009\)](#). In what follows, we present the key formulas of these algorithms and we refer to [Cornillier et al. \(2009\)](#) for detailed proofs.

Given a route s , we define for each customer $i \in s$ a normalized time window $[e_i^n, l_i^n]$ if waiting times are not allowed. They indicate to which time the service can be preponed or postponed while still allowing timely arrivals at other customers. We set $e_i^n = (e_i - \sum_{j=0}^{i-1} t_{j,j+1} - \sum_{j=0}^{i-1} s_j)$ and $l_i^n = (l_i - \sum_{j=0}^{i-1} t_{j,j+1} - \sum_{j=0}^{i-1} s_j)$. In this case, the route is feasible if and only if the intersection of all normalized time windows is not empty. However, when waiting times are allowed, as it is the case for the RPTP, a route s is time-window feasible if and only if for each customer $i \in s$, $\max_{0 \leq j < i} \{e_j^n\} \leq l_i^n$. The *TW_Feasibility_Check* algorithm is applied whenever a feasibility evaluation is required. Its time complexity in the worst case is $O(n)$. To minimize the computation times, we indicate the position in the route from which the feasibility evaluation procedure should start. Once a route s is feasible, one may calculate its minimal

waiting time w^s such that $w^s = \max\{0, \max_{i \in \bar{V}}\{e_i^n\} - \max_{i \in \bar{V}}\{l_i^n\}\}$. Adding the minimal waiting time to the sum of travel and service times we obtain the minimal route duration. Then, we can compute the normalized departure and arrival times associated to the depot, $[e_d^n, l_d^n]$, such that the total waiting time of s is minimized. We set $e_d^n = \max_{i \in \bar{V}}\{e_i^n\} - w_s$ and $l_d^n = \min_{i \in \bar{V}}\{l_i^n\}$.

3.5 Conclusions

In this chapter, we described a unified matheuristic based on routing and loading heuristics, denoted as VNS*. An effective multi start-construction heuristic has been proposed to build the initial solution. To diversify and intensify the search, we suggested removal and insertion heuristics as well as different local search procedures. We tried to focus on the loading aspect of the problem which was barely considered in the MC-VRP literature. We introduced an heuristic to ensure and check the assignment of products to compartments and we proposed two loading neighborhoods based on the solution of mathematical programs. We incorporated these neighborhoods in the matheuristic approach to optimize the loading plan for the current solution.

Chapter 4

From methods to implementation and results

After describing the main matheuristic components, we present the data structures and the object classes used to model the RVRP solutions in Section 4.1. In Section 4.2, the performance of the VNS* is assessed on a large set of instances for the OP and the OPTW including up to 288 customers. The proposed matheuristic is very competitive compared with the state-of-the-art methods. To evaluate its performance, we generate a new testbed for the MDMCMCm-VRPTW, in Section 4.2.3.1, including a large variety of general attributes. Extensive computational experiments on the new testbed confirm the efficiency of the matheuristic. A sensitivity analysis follows in Section 4.2.3.2 to highlight which feature of the matheuristic contributes most to the solution quality. Conclusions are drawn in Section 4.3.

Preliminary results of this work were presented at the *Eight Triennial Symposium on Transportation Analysis* (TRISTAN VIII) (Lahyani et al., 2013a), at the *5th International Conference on Modeling, Simulation and Applied Optimization* (ICMSAO) (Lahyani et al., 2013c), as well as at ROADEF 2013 and ROADEF 2014 conferences (Lahyani et al., 2013b, 2014). An article based on Chapters 3 and 4 was submitted for publication: R. Lahyani, M. Khemakehm, F. Semet. A unified matheuristic for solving multi-attribute traveling salesman problems with profits. *EURO Journal on Computational Optimization*, submitted for publication.

4.1 Design of data structure classes

4.1.1 Challenges

A VRP solution is composed by a set of routes, describing the sequences of served customers. In a RVRP, more decisions have to be made simultaneously. In the MDMCMCm-VRPTW, a complex loading problem must be solved jointly to the routing and sequencing problem which adds up to the problem difficulty. Part of this difficulty comes from (i) the presence of compatibility constraints while loading orders into compartments (ii) the presence of orders rather than customers (iii) the different number and types of products delivered to each customer. Determining which orders to deliver to a customer with a given vehicle represents an additional decision to make. Delivering orders rather than visiting customers leads to a considerable increase of the problem size when many different products are considered: for P product types the number of orders is up to $|\mathcal{V}| \times P$. Therefore, addressing a VRP with multi-compartments, multiple products requires a special attention when the solution method and the data structures are designed.

The solution description of the MDMCMCm-VRPTW or more generally a solution for a real-life problem has to take into account several attributes associated with constraints, decisions and objectives. In the MDMCMCm-VRPTW, orders can be interpreted as dummy customers and the set of orders associated with a customer share the same attributes (e.g., coordinates, time windows, service time, etc). In addition, we have to keep the orders sequence for a route as well as for the compartment loading plan during the search process. The solution data structure is then much more difficult to design and is unlikely to be modeled by a single class holding all the routing and loading data. Therefore, we introduce the following classes: the *Order Class* and the *Compartment Class* in addition to the *Problem Class* and the *Solution Class*.

Since the routing and loading heuristics aim to apply moves to a solution and the checking procedures evaluate the solution feasibility, all details of each data structure should be accessed and modified. Two main challenges are encountered while designing the four data structures. On the one hand, we should maintain a moderate level of granularity while describing the data elements because storing and updating irrelevant data may slow down the optimization process. On the other hand, we must pay a special attention to the selection of data access techniques. To this end, we choose to implement all data classes in the programming language *C* and follow an object-oriented like design. Object-oriented programming represents a powerful tool for developing unified solution methods. Each object corresponds to

an instance of a class with its associated attributes and allow, at the same time, an easy access to other object through functions and pointers. The concept of pointers is one of the key concepts of language *C*. It enables to create dynamic data structures and to access data stored in the computer memory easily. Tables 4.1-4.3 list the elements of the main classes designed for the MDMCMCm-VRPTW. In these tables, we highlight the relationship between classes indicating with a (*) the pointer to other classes and with with a (**) the array of pointers to other classes.

4.1.2 Data structures classes

The *Problem Class* holds the data about the problem which have been extracted from a text file. This data remains unchanged during the solution process. This class includes the values defining the sets described in Section 2.2 in Chapter 2 in addition to the distance and time matrices. The distance matrix includes Euclidean distances computed from (x, y) coordinates of each vertex. By default, the distances are rounded down to the second decimal. However for a fair comparison with results obtained with other VRPs methods, distances may be rounded down to the first decimal.

The *Solution Class* represents a vehicle route starting from and returning to the depot. Each route is represented by a *doubly linked list* linking the orders covered by this route. From a theoretical point of view, linked lists allow to store similar data in memory efficiently. The solution class enables access to the other three classes through a specific pointer to each one. The solution class includes also relevant data that are used to check the feasibility and to evaluate the solution. These are the solution cost value, the total distance, the total wait, the total profit, the number of customers visited and the number of orders delivered.

Every non-depot node in the list corresponds to a given order which gives rise to the *Order Class*. This class specifies all attributes required to perform an operation on a given order. These attributes include the identification number in \mathcal{O} , the customer involved, the associated product, the compartment in which it is loaded, if one exists, and the vehicle arrival and departure times at the customer placing this order. Each order is linked to its predecessor and its successor on the route and the compartment. These links are referred to respectively by R_pred , R_succ , C_pred and C_succ . Choosing doubly-linked lists while designing the MDMCMCm-VRPTW data structures has many advantages. It allows to examine the sequence of orders on the route in both directions, from the first order, RF_ord , to the last order, RL_ord and vice-versa. The presence of predecessors and successors, even if null, associated with each order allows to keep track of the order positions during the

Table 4.1: Details of the order class

Order Class	
<i>Element</i>	<i>Description</i>
<i>Num</i>	Order index in \mathcal{O}
<i>Cust</i>	Index of the associated customer
<i>Prod</i>	Index of the associated product
<i>Comp</i>	Index of the associated compartment, -1 by default
<i>A_time</i>	Vehicle arrival time at <i>Cust</i> , -1 by default
<i>D_time</i>	Vehicle departure time from <i>Cust</i> , -1 by default
<i>*R_pred</i>	Pointer to the order predecessor on the route, NULL by default
<i>*R_succ</i>	Pointer to the order successor on the route, NULL by default
<i>*C_pred</i>	Pointer to the order predecessor in the compartment, NULL by default
<i>*C_succ</i>	Pointer to the order successor in the compartment, NULL by default

routing and the loading process. It enables also to quickly and efficiently detect unassigned orders to a route. Furthermore, an order can be inserted into (or removed from) a route or a compartment without searching the complete route or the whole content of a compartment to find the address of the previous order.

The information related to the compartment containing a given order is part of the solution representation. The *Compartments Class* aims to maintain data about the current vehicle load consistently in order to speed up the check feasibility procedure. It involves the current free capacity of the vehicle compartments and an array of compartments index sorted decreasingly with respect to the available capacity. This class involves pointers to the first and the last order currently loaded in the compartment. Thus, we can easily browse all orders loaded in this compartment.

Figure 4.1 illustrates the solution representation. Consider the route performed by vehicle $k = 2$ in the example described in Section 2.2.1 in Chapter 2 (see Figure 2.3). The vehicle route is $(Depot, 6, 5, 4, 3, Depot)$ including 7 orders. Figure 4.1 highlights how the use of doubly linked lists allows access to the associated order successors and predecessors easily. In particular, Figure 4.1 sketches the links between orders hold in compartment $w3$. Consider customer 6 which is the first customer visited on the route and has three orders delivered (1, 2, 3). Although order 2 associated with customer 6 is the second order on the route, is not necessarily the second order in compartment $w3$. Note that efficient cloning procedures are developed to create copies of a given class when it is required during the search process.

Table 4.2: Details of the compartment class

Compartment Class	
<i>Element</i>	<i>Description</i>
<i>Free_Q</i>	Array of size W , contains the current free capacities of compartments
<i>Sort</i>	Array of size W , contains the index of compartments sorted decreasingly with respect to <i>Free_Q</i> content
<i>**CF_ord</i>	Array of size W , contains pointers to the first order hold in each compartment
<i>**CL_ord</i>	Array of size W , contains pointers to the last order hold in each compartment

Table 4.3: Details of the solution class

Solution Class	
<i>Element</i>	<i>Description</i>
<i>*Prob</i>	Pointer to the problem class
<i>**Ord_list</i>	Array of size $ \mathcal{O} $, contains pointers to the orders covered by this route
<i>*Compartment</i>	Pointer to the compartment class
<i>*RF_ord</i>	Pointer to the first order in the route, NULL by default
<i>*RL_ord</i>	Pointer to the last order in the route, NULL by default
<i>#Cust</i>	Total number of the route customers
<i>#Ord</i>	Total number of the route orders
<i>Obj</i>	Route cost
<i>T_distance</i>	Total route distance
<i>T_time</i>	Total route time
<i>T_wait</i>	Total route wait
<i>T_profit</i>	Total route profit

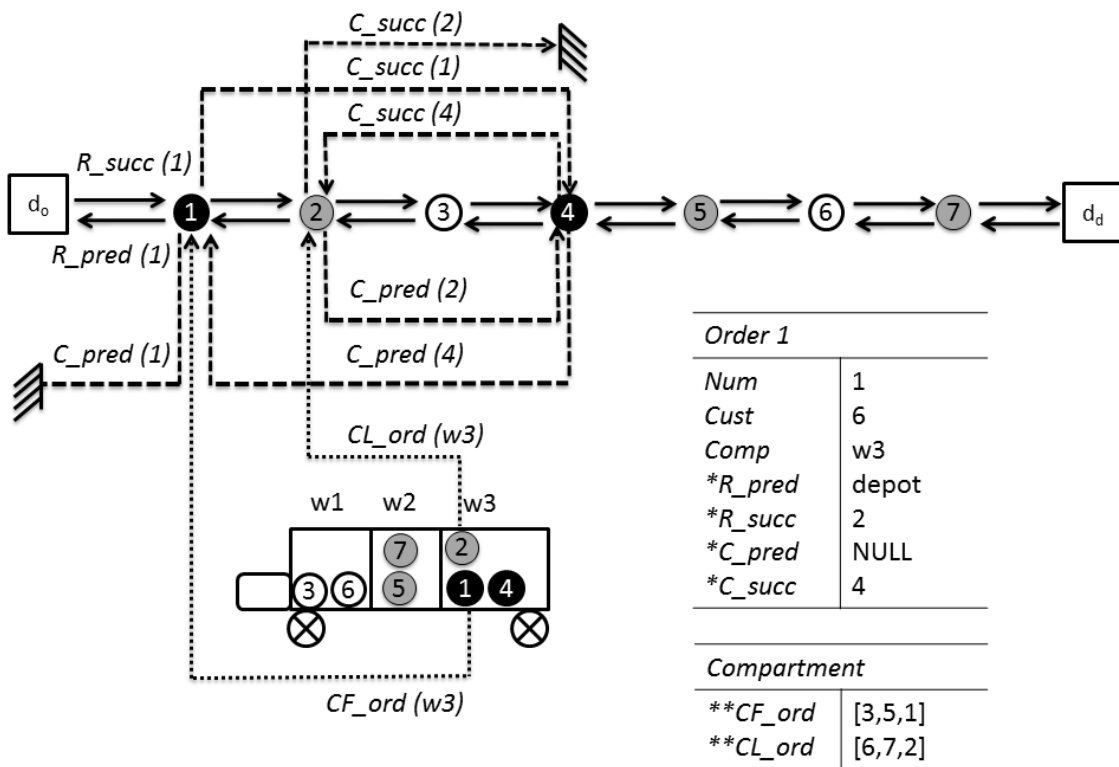


Figure 4.1: Solution representation using the order class and the compartment class

4.2 Computational experiments

To assess the efficiency of the proposed matheuristic, we report experimental results on problems related to the RPTP, sharing common features with it. To preserve the efficiency of the proposed matheuristic compared with state-of-the-art methods, we select 4 insertion heuristics among 6 in the improvement phase of the VNS* for all the tests carried out in this Chapter. The invoked insertion heuristics are heuristics (1), (2), (5) and (6). This section presents experimental results and is divided in three main parts. In Section 4.2.1, we report results obtained by applying the proposed matheuristic on OP instances. In Section 4.2.2, computational experiments are conducted on the OPTW, a more difficult RPTP. Since the loading neighborhoods are not relevant for the OP and OPTW instances, we generate new instances under three real-life scenarios to better evaluate the VNS*. Extensive computational experiments and a sensitivity analysis for the RPTP demonstrate the contribution of the main components of the matheuristic to the solution quality.

The matheuristic has been coded in language *C* and executed on an Intel Quad Core with 2.66 GHz and 4 GB Ram. Results are summarized in Tables 4.4-4.8 and 4.10-4.12. In these tables, data are grouped by instance class and average statistics over the 5 runs are reported for each instance class. Columns headed *Class* identify the instance class, columns headed *Gap%* report the percentage gap to the best known value solution and columns headed *Time(s)* give the computational time in seconds for each instance class. Detailed computational experiments are reported in Appendix A.

Preliminary experiments have been carried out on sample instances to determine the best parameter setting with respect to speed and efficiency. The maximum VNS* iterations, $iter^{max}$, is fixed to 20. The maximum number of deleted customers in the shaking phase is $kd = 2$. The local search procedures try to insert new customers while $kc \leq 2 * kd$. For the routing heuristics, we used the same parameter setting determined in (Røpke and Pisinger, 2006b), $(\phi_1, \phi_2, \phi_3) = (9, 3, 2)$ and (Prescott-Gagnon et al., 2009), $\psi = 35$. Preliminary experiments also reveal that the performance of the VNS* is improved when $iter^{max}$ (re-starting in every iteration from a new solution) or kd (exploring new solutions neighbors) are increased. Since this improvement is at the expense of computational times, we do not consider such settings. We keep the same parameter values for all the benchmarks to demonstrate that the performance of the proposed matheuristic is not subject to any customization, as in Vidal et al. (2014).

4.2.1 OP instances

[Tsiligrirides \(1984\)](#) propose an exact algorithm for 3 sets of instances for the OP. The sets (1_p21, 1_p32 and 1_p33) include 18, 11 and 20 instances respectively. The number of customers ranges from 21 to 33. A second testbed with larger instances was generated by [Chao et al. \(1996\)](#). The authors proposed five steps heuristic named CGW which was evaluated on 2 sets of instances (1_p64 and 1_p66) with 14 and 26 instances including 64 and 66 customers respectively. We test the VNS* over these 5 instances sets. We compare the results with the optimum values published by [Tsiligrirides \(1984\)](#), the best known (BK) solutions values published by [Chao et al. \(1996\)](#) and the following 4 state-of-the-art methods:

- A 3-run trained Ant Colony Optimization with 20 iterations (ACO_{20}^t) proposed by [Souffriau et al. \(2008\)](#),
- A 1-run deterministic Guided Local Search (GLS) proposed by [Vansteenwegen et al. \(2009b\)](#),
- A 10-run Pareto Ant Colony Optimization (P-ACO) and 10-run Pareto Variable Neighborhood Search (P-VNS) proposed by [Schilde et al. \(2009\)](#),
- A 10-run GRASP with path relinking (GRASP) proposed by [Campos et al. \(2013\)](#).

Tables 4.4 and 4.5 summarize the computational results on [Tsiligrirides \(1984\)](#) and [Chao et al. \(1996\)](#) instances sets respectively. Columns headed *Gap%* in Table 4.4 and Table 4.5 give the average percentage gaps to the optimal value identified by [Tsiligrirides \(1984\)](#) for the sets 1_p21, 1_p32 and 1_p33 and to the BK value for the sets 1_p64 and 1_p66 respectively. Note that for the three [Tsiligrirides \(1984\)](#) instances sets, both methods proposed by [Schilde et al. \(2009\)](#) find the optimal values but computational times were not reported. Table 4.4 shows that the proposed matheuristic VNS* outperforms all the published methods by providing the optimal values of less than one second, 0,67 seconds in average. GLS is the fast method on average on the three instances classes. It takes 0.44 seconds but provides worst results, 1.74% of average gap. In Table 4.5, the VNS* finds the best gap value for the class 1_p64 in 4.64 seconds while the CGW finds solutions with the same gap value, with more computational effort equal to 177 seconds. Moreover, the VNS* achieves only slightly worse average gap equal to 0.08% compared to 0.06% obtained by P_ACO. Nevertheless, we cannot conclude that the VNS* is the most competitive among the state-of-the-art methods since the running times

for P_ACO , P_VNS and $GRASP$ are missing. For [Chao et al. \(1996\)](#) instances, [Schilde et al. \(2009\)](#) and [Campos et al. \(2013\)](#) report only the running time elapsed to find the BK value which is useless for this comparison.

According to [Dongarra \(2013\)](#), computers used for state-of-the-art methods and the VNS* have similar performance (similar enough to compare absolute run times). Our computer is similar to the machine used to run the CGW heuristic and two time faster than the machine used to run the GLS heuristic.

Table 4.4: Comparison of VNS* to the state-of-the-art methods on [Tsiligirides \(1984\)](#) data sets

<i>Class</i>	<i>Gap%</i>			<i>Time(s)</i>		
	<i>GLS</i>	<i>ACO</i> ₂₀ ^t	<i>VNS</i> *	<i>GLS</i>	<i>ACO</i> ₂₀ ^t	<i>VNS</i> *
1_p21	1.34	4.78	0.00	0.25	-	0.33
1_p32	3.28	1.80	0.00	0.52	-	0.70
1_p33	0.60	1.79	0.00	0.55	-	0.98

Table 4.5: Comparison of VNS* to the state-of-the-art methods on [Chao et al. \(1996\)](#) data sets

<i>Class</i>	<i>Gap%</i>						<i>Time(s)</i>					
	<i>CGW</i>	<i>GLS</i>	<i>P_ACO</i>	<i>P_VNS</i>	<i>GRASP</i>	<i>VNS</i> *	<i>CGW</i>	<i>GLS</i>	<i>P_ACO</i>	<i>P_VNS</i>	<i>GRASP</i>	<i>VNS</i> *
1_p64	0.07	1.09	0.13	0.17	0.14	0.07	177.04	2.18	1.97	2.56	0.08	4.64
1_p66	0.43	1.31	0.00	0.13	0.00	0.10	158.65	2.26	0.22	1.18	0.07	4.94

4.2.2 OPTW instances

The OPTW is a simplified version of the RPTP where the vehicle has only one compartment w and there is only one product demand p associated with each customer i . The OPTW has received a lot of attention in the literature and a large set of instances has been proposed. The instances were obtained from the data sets proposed by [Solomon \(1987\)](#) for the VRPTW and the data sets of [Cordeau et al. \(1997\)](#) generated for the periodic MDVRPTW. Based on the Solomon's instances, [Righini and Salani \(2006\)](#) generated 58 instances for the OPTW by considering 50 and 100 customers (c-50, r-50, rc-50, c-100, r-100, rc-100). They derived a second set of 10 instances based on Cordeau's data sets (pr01-pr10). [Righini and Salani \(2006, 2009\)](#) solved the derived sets of the OPTW to optimality thanks to a dynamic programming approach. [Montemanni and Gambardella \(2009\)](#) proposed an Ant Colony

System (ACS) and added 2 data sets by considering 27 Solomon's instances with 100 customers (c2-100, r2-100, rc2-100) and other 10 Cordeau's instances (pr11-pr20). The Solomon's instances referred to by (c2-100, r2-100, rc2-100) as well as the Cordeau's instances are characterized by wide time windows and are known to be hard to solve (Righini and Salani, 2006, Labadie et al., 2011). Most approximate algorithms failed to obtain optimal solutions in a reasonable time for these instances. In total, 105 instances for the OPTW divided into 11 classes are available. The number of customers in Cordeau's data sets ranges from 48 to 288 while the Solomon's instances contain 50 or 100 customers. The demand of each customer in each instance represents the associated score. The maximum duration T^{max} of a route equals the closing time of the starting point. These test instances can be downloaded from <http://www.mech.kuleuven.be/en/cib/op>.

The performance of the proposed matheuristic is now compared to the OPTW published results, in particular, the Iterative Local Search (ILS) (Vansteenwegen et al., 2009a), the ACS (Montemanni and Gambardella, 2009), the VNS (Tricoire et al., 2010), the Greedy Randomized Adaptive Search Procedure (GRASP) (Labadie et al., 2011), the Granular VNS (GVNS) (Labadie et al., 2012) and the Slow Simulated Annealing (SSA) (Lin and Yu, 2012). Vansteenwegen et al. (2009a) proposed a fast ILS based on a multi-start strategy to solve the OPTW and the TOPTW instances. Tricoire et al. (2010) designed a VNS to deal with instances of the multi-period orienteering problem with multiple time windows, the OPTW and the TOPTW. A GRASP hybridized with an evolutionary local search algorithm was proposed in Labadie et al. (2011). Recently, Labadie et al. (2012) develop an effective GVNS. To fairly compare the performance of the GVNS to the ILS, the authors report the detailed results of a fast version of the GVNS which terminates once the solution value of the ILS is retrieved. Lin and Yu (2012) propose a fast and a slow version of an algorithm based on Simulated Annealing ((FSA) and (SSA)). The SSA outperforms the FSA due to the different stopping criterion. SSA stops when the best solution visited is not improved for a preset number of iterations. The SSA improves the BK solutions for 4 instances: rc2-104, pr11, pr17 and pr18. In the remainder of this chapter, we compute all average percentage gaps for ACS, ILS, VNS, GRASP and GVNS taking into consideration these new BK values.

In this chapter, the Euclidean distance for all the instances are rounded down to the second decimal as in (Righini and Salani, 2009, Montemanni and Gambardella, 2009, Labadie et al., 2011, 2012). It is worth mentioning that the remaining state-of-the-art methods compute distances rounded down to the second digit for Cordeau's instances and to the first digit for Solomon's instances.

For a fair comparison, we compare the computers on which computational results were conducted (Dongarra, 2013). The performance of our computer is equivalent to the computers used by Vansteenwegen et al. (2009a) and Lin and Yu (2012). For the other state-of-the-art methods, our processor is approximately two times faster. It should be noted that ILS and SSA are deterministic algorithms and were run only once. ACS, GRASP and GVNS were executed 5 times while the results for VNS were obtained with 10 runs. The comparison to previous methods is shown in Tables 4.6-4.8.

A large variance of computational time can be observed for the results of Solomon’s instances. For example, for the instance class r*2-100, the CPU ranges from 1.66 seconds to 3600 seconds. To obtain a meaningful comparison, we propose to divide methods into slow algorithms: ACS and VNS, (see Table 4.6) and fast algorithms: ILS, GRASP, GVNS and SSA (see Table 4.7). In Table 4.6, we report results when we limit the CPU time to 120 and 300 seconds. In Table 4.7, we report the results of VNS* when the stopping criteria, $iter^{max}$, is fixed to 20 and the results when the computation time is limited to 20 seconds. Results for Cordeau’s instances are summarized in Table 4.8. We report the computational results of VNS* for 120 and 300 seconds of computation time.

In the detailed experimentations results, we state that VNS* is able to obtain the BK solution at least once for most of the Solomon’s instances. In Tables 4.6, 4.7 and 4.8 the efficiency of VNS* increases with running time. Looking at the results after 120 seconds of computation time, VNS* provides a better solution value than the ACS on 8 out of 9 classes. After 300 seconds, VNS* and VNS provide similar results for the different instances classes. On average, Tricoire et al. (2010) obtain a gap to BK values equal to 0.33% in 318 seconds while we obtain 0,38% in 300 seconds.

When the stopping criteria is the fixed number of iterations, VNS* leads to near-optimal solutions with a little computational effort. The average gap to the BK solutions ranges from 0% to 2.77% over all the instances classes within an average computation time equal to 7.86 seconds. VNS* outperforms then the GVNS in terms of solution quality and provides similar average gap than GRASP (0.88% vs. 0.73% respectively). For the instances classes with 50 and 100 customers, the SSA provides slightly lower average gaps than VNS* at the expense of larger computational times. Computation times indicate that ILS is certainly the fastest method but it provides the highest average gap equal to 1.93%. This is not surprising because ILS is a deterministic algorithm designed to reach good quality solution very quickly.

Considering the first seven instances classes, VNS* 20 seconds achieves better results compared to VNS* with $iter^{max}$ as stopping criterion. For the instances classes (r2-100 and rc2-100) known to be hard, VNS* is able to provide good quality solution with an average gap equal to 3.05% at most. We can conclude then that the VNS* is competitive as the state-of-the-art methods on the Solomon's instances.

On the Cordeau's data sets, VNS* is not as efficient as on the Solomon's instances. After 120 seconds and 300 seconds of computation time, the average gaps to BK on Cordeau's instances (pr01-pr10) and (pr11-pr20) are 4% and 3.06% respectively better than the values obtained by ILS (6.91%) and ACS (6.02%) with an average computational times of 1.86 seconds and 3600 seconds respectively. Results in Table 4.8 indicate that the SSA provides the best solution values on average and finds new BK solutions for three of the Cordeau's instances: pr11, pr17 and pr18. For the state-of-the-art methods, the average gap from BK solutions ranges from 0.97% to 10.84%. Due to their characteristics (large TW, large time duration per route), Cordeau's instances require much more diversification than intensification. Therefore, to assess the efficiency of the proposed matheuristic for more difficult instances, we propose a slightly modified version of the VNS* by increasing the number of deleted customers kd to $\frac{\#Cust}{3}$ in each iteration and by reducing $iter^{max}$ to 5. This method, VNS*C, explores the solution space more widely. The average gap decreases significantly. After 300 seconds of running time, VNS*C is as efficient as SSA: 0.99% and 2.96% versus 0.97% and 3.25% in 112.21 seconds and 162.40 seconds respectively. In addition, it is worth mentioning that VNS* improves the BK solution for instance pr11 while VNS*C improves the BK solution for instance pr13. Note that in VNS*, the maximum number kd of deleted customers is set to 2. We have to preserve the speed of the proposed matheuristic and it is shown experimentally that it turns to be time consuming to explore the solution space extensively as VNS*C does on Solomon's instances.

Altogether, the performance of the unified matheuristic is very encouraging. The results have shown that the proposed algorithm can compete with the current state-of-the-art methods. It is able to produce optimal and near-optimal solutions for slightly longer running times. This is not surprising since the code was not modified to solve these problems. Therefore, some procedures are invoked even if they are of no interest in this case. This represents the cost to pay when the method implemented is able to solve a wider range of multi-constrained problems.

Table 4.6: Average results of the slow methods on Solomon’s data sets

<i>Class</i>	<i>Gap%</i>				<i>Time(s)</i> VNS
	<i>ACS</i>	<i>VNS</i>	<i>VNS*</i>	<i>VNS*</i>	
	<i>(3600s)</i>		<i>(120s)</i>	<i>(300s)</i>	
c-50	0.00	0.00	0.00	0.00	53.63
r-50	0.00	0.00	0.33	0.33	24.18
rc-50	0.00	0.19	0.00	0.00	31.93
c-100	0.00	0.11	0.00	0.00	98.39
r-100	0.24	0.05	0.06	0.05	89.10
rc-100	0.00	0.04	0.00	0.00	65.21
c2-100	0.58	0.21	0.42	0.33	560.17
r2-100	3.16	1.05	2.10	1.53	1065.82
rc2-100	2.04	1.35	1.61	1.23	869.41

Table 4.7: Average results of the fast methods on Solomon’s data sets

<i>Class</i>	<i>Gap%</i>					<i>Time(s)</i>					
	<i>ILS</i>	<i>GRASP</i>	<i>GVNS</i>	<i>SSA</i>	<i>VNS*</i> <i>(20 iter)</i>	<i>VNS*</i> <i>(20s)</i>	<i>ILS</i>	<i>GRASP</i>	<i>GVNS</i>	<i>SSA</i>	<i>VNS*</i> <i>(20 iter)</i>
c-50	0.33	0.00	-	-	0.00	0.00	0.27	7.01	-	-	0.88
r-50	0.63	0.47	-	-	0.33	0.33	0.20	0.93	-	-	0.80
rc-50	2.21	1.11	-	-	0.10	0.00	0.18	0.91	-	-	0.65
c-100	1.11	0.00	1.22	0.00	0.49	0.00	0.33	22.59	166.46	21.07	1.88
r-100	1.90	0.22	2.68	0.11	0.25	0.09	0.19	3.51	29.43	23.34	2.75
rc-100	2.92	0.40	3.51	0.00	0.19	0.00	0.23	1.99	9.80	22.19	2.03
c2-100	2.28	0.61	1.11	0.13	1.16	0.85	1.71	32.18	192.40	37.49	11.28
r2-100	2.89	1.61	3.37	1.29	2.77	3.05	1.66	11.18	33.82	45.83	29.46
rc2-100	3.43	2.20	3.96	0.96	2.65	2.60	1.63	8.21	16.01	50.25	21.05

Table 4.8: Average results of the state-of-the-art methods on Cordeau’s data sets

<i>Class</i>	<i>Gap%</i>		<i>VNS</i>	<i>GRASP</i>	<i>GVNS</i>	<i>SSA</i>	<i>VNS*C</i> <i>(120 s)</i>	<i>VNS*C</i> <i>(300s)</i>	<i>VNS*</i> <i>(120 s)</i>	<i>VNS*</i> <i>(300s)</i>
	<i>ILS</i>	<i>ACS</i> <i>(3600s)</i>								
pr01-10	4.72	1.20	1.08	1.44	1.61	0.97	1.20	0.99	2.41	1.77
	1.75		822.07	5.03	12.37	112.21				
pr11-20	9.11	10.84	2.92	2.92	3.81	3.25	3.45	2.96	5.60	4.36
	1.98		1045.93	7.90	24.22	162.40				

4.2.3 Experimental results for the RPTP

4.2.3.1 New testbed

Since no data sets are available for the RPTP, we generate a new testbed to evaluate the VNS*. The proposed testbed is based on the Solomon's data sets with 50 and 100 customers and on the extended instances including 200 customers proposed for the VRPTW by [Gehring and Homberger \(1999\)](#). We consider a total of 172 original instances classified into 18 classes. We introduce three types of products $p \in \mathcal{P} = \{1, 2, 3\}$ with the vector profit $(10, 15, 20)$. We split the original customer demand into three amounts randomly. A customer i may not place an order for a given product p , i.e., $q_i^p = 0$. The gain g_i^p associated with each order o_i^p is obtained by multiplying the quantity q_i^p by the associated product profit.

Three compartments $w \in \mathcal{W} = \{1, 2, 3\}$ are considered for each vehicle. The compartments capacities (Q^1, Q^2, Q^3) are obtained by dividing the original capacity Q into three parts according to the following distribution $(0.2 * Q, 0.3 * Q, 0.5 * Q)$. \mathcal{W} and \mathcal{P} are the same for all 172 instances. As in the original data sets, instances belonging to the same class have the same customers locations but have different time windows and quantity (eventually, null) of products for each customer. Another relevant design parameter is the incompatibility relations. To study different loading problems as they arise in real-life scenarios, we propose to generate three types of incompatibilities as it is shown in [Table 4.9](#), each type contains 172 instances divided into 18 classes.

The first type of instances (*type A*) is the simplest one. In this data set, there is no incompatibility constraints, i.e., each product may be loaded in any compartment and with any other product. Therefore, a solution for an instance of *type A* is feasible for the corresponding OPTW instances. *Type B* instances correspond to the distribution of liquid products to customers or animal feeds to farms or to the waste collection. In such cases, there are no product-compartment incompatibilities and all the products are incompatible pairwise. The third instances type, *type C*, corresponds to the general one. We choose to maintain a moderate level of incompatibility restrictions. Two products must be kept segregated during transportation and each product may be loaded in two compartments out of three.

Table 4.9: Incompatibility configurations in generated data sets

<i>Sets</i>	<i>Type A</i>	<i>Type B</i>	<i>Type C</i>
$\mathcal{IP} \subseteq \mathcal{P} \times \mathcal{P}$	$\mathcal{IP} = \{\emptyset\}$	$\mathcal{IP} = \{(1,2), (1,3), (2,3)\}$	$\mathcal{IP} = \{(1,3)\}$
$\mathcal{IPC} \subseteq \mathcal{K} \times \mathcal{W} \times \mathcal{P}$	$\mathcal{IPC} = \{\emptyset\}$	$\mathcal{IPC} = \{\emptyset\}$	$\mathcal{IPC} = \{(1,1,1), (1,2,2), (1,3,3)\}$

The instances names are as follows: the first two letters give the type of customers distribution followed by an asterisk to differentiate the instances designed for the VRPTW with compartments from the original ones designed for the VRPTW. The first digit gives the type of time window (1: narrow, 2: large), the second three-digits indicate the number of customers.

Table 4.10 reports the performance of VNS* on the three data sets. The first two columns report the number of customers and the number of orders included in the solution respectively. Columns headed *Obj* report the average objective value obtained with the standard setting of parameters. Table 4.10 shows that solutions for instances of *type A*, provide higher average objective than solutions for instances of *type B* and *type C*, although the average number of covered customers is almost the same between the three instances types. This could be expected since instances of class *A* are less constrained. Indeed, the number of visited customers in each route is limited by the temporal restrictions or/and capacity for the three data sets while the absence of incompatibility relations enables to freely assign products to compartments and increasing the objective value.

For instances of *type B*, identifying very good solutions requires large computational times almost equal to the double of average running time required by solving instances of *type A* and *type C*. This is likely due to the fact that the assignment of each product to any compartment induces some symmetry which can not be broken easily. Finally, VNS* seems sensitive not only to incompatibility constraints but also to the type of the time windows associated with customers, since few customers are inserted for instances with narrow time windows. These observations seem to be consistent with previous research (Righini and Salani, 2006, Labadie et al., 2011) as we pointed out in Section 4.2.2.

Now, we assess the efficiency of the loading neighborhoods. We compare two new versions of VNS* as follows.

- VNS* = The Linear Multiple Knapsack Problem with Conflicts is solved at the end of each VNS iteration.

- VNS*I= The Quadratic Multiple Knapsack Problem with Conflicts is solved at the end of each VNS iteration. Next an insertion heuristic attempting to insert new orders is applied. Last, we solve the Linear Multiple Knapsack Problem with Conflicts.
- VNS*II= VNS* without any loading neighborhoods.

To avoid misleading results, the remaining tests are performed on instances of *type C*. Results for each instance class are reported in Table 4.11.

Obviously, VNS*II provides good but not convincing results. The solutions provided serve almost the same number of customers compared with the solutions provided by applying the loading neighborhoods. However, VNS*II provides lower average objective values by 10% than those provided by VNS* and VNS*I. As we expected, optimizing the loading plan in the MC-VRP has a significant impact on the solution quality.

Extensive computational experiments with VNS*I, put into highlight that improving the quality solution is unlikely within a reasonable amount of computation time. When we solve the Quadratic Multiple Knapsack Problem with Conflicts,

Table 4.10: Average results of VNS* on RPTP data sets

<i>Class</i>	<i>Type A</i>				<i>Type B</i>				<i>Type C</i>			
	<i># Cust</i>	<i># Ord</i>	<i>Obj</i>	<i>Time(s)</i>	<i># Cust</i>	<i># Ord</i>	<i>Obj</i>	<i>Time(s)</i>	<i># Cust</i>	<i># Ord</i>	<i>Obj</i>	<i>Time(s)</i>
c*1-50	11.33	19.89	3478.47	2.10	11.11	18.44	3360.23	5.37	11.56	21.56	3236.88	2.43
c*2-50	33.38	82.88	9900.73	13.03	33.38	77.13	8734.90	83.08	33.00	81.38	9264.00	17.52
r*1-50	8.58	20.67	2824.96	1.32	8.58	20.17	2623.35	2.62	8.75	21.67	2785.72	1.41
r*2-50	38.91	92.18	8326.97	12.67	38.64	91.64	8207.63	40.49	38.00	90.36	8063.30	15.00
rc*1-50	8.63	18.63	3128.93	1.79	9.25	18.13	2871.65	2.95	9.00	19.63	2975.40	1.51
rc*2-50	35.75	89.88	10752.40	11.27	35.50	87.75	10326.02	94.09	34.75	88.00	10216.72	14.94
c*1-100	11.11	16.56	3553.52	3.97	11.11	16.00	3478.68	6.05	11.22	20.89	3279.04	3.68
c*2-100	31.63	66.63	11447.27	28.25	31.63	71.50	10118.81	107.14	32.00	72.25	10381.67	31.76
r*1-100	11.42	23.58	3318.47	4.02	10.83	19.92	3177.76	7.41	11.67	24.08	3160.05	4.17
r*2-100	49.82	122.55	13371.22	61.29	49.45	114.73	11981.13	316.85	49.64	121.55	12311.07	71.99
rc*1-100	10.38	20.13	3343.15	4.79	10.13	17.75	3176.81	6.02	10.50	21.13	3207.39	3.46
rc*2-100	42.63	102.50	13199.75	46.43	43.00	94.75	11715.23	222.65	42.63	103.88	11962.38	52.16
c*1-200	11.60	18.40	3555.19	7.18	11.20	17.00	3449.21	10.18	11.80	21.00	3273.55	7.06
c*2-200	33.50	66.40	11799.60	64.79	33.60	73.70	10423.75	165.82	33.70	70.10	10347.67	68.44
r*1-200	20.40	23.10	3621.71	16.71	16.60	21.90	3291.21	20.60	16.00	25.60	3209.68	14.37
r*2-200	87.10	128.90	16938.42	371.27	84.80	134.00	14568.86	267.57	80.40	135.00	16034.64	397.53
rc*1-200	20.50	22.90	3642.42	14.85	18.60	24.60	3316.47	30.64	17.40	26.30	3260.44	14.04
rc*2-200	81.10	128.40	16815.84	272.69	80.90	135.00	13516.57	232.84	79.00	134.00	16012.48	332.15
Average	30.43	59.12	7945.50	52.13	29.91	58.56	7129.90	90.13	29.50	61.02	7387.89	58.53

Table 4.11: Average results of VNS* variants on RPTP data sets

<i>Class</i>	<i>VNS*</i>				<i>VNS*I</i>				<i>VNS*II</i>			
	<i># Cust</i>	<i># Ord</i>	<i>Obj</i>	<i>Time(s)</i>	<i># Cust</i>	<i># Ord</i>	<i>Obj</i>	<i>Time(s)</i>	<i># Cust</i>	<i># Ord</i>	<i>Obj</i>	<i>Time(s)</i>
c*1-50	11.56	21.56	3236.88	2.43	11.33	23.89	3210.17	3.91	11.56	22.44	3170.84	1.55
c*2-50	33.00	81.38	9264.00	17.52	33.00	81.13	9186.04	522.33	33.00	73.63	8369.57	12.59
r*1-50	8.75	21.67	2785.72	1.41	8.75	21.75	2759.27	4.26	8.75	21.42	2762.88	0.98
r*2-50	38.00	90.36	8063.30	15.00	38.18	90.55	8066.45	458.08	37.91	84.55	7679.49	12.33
rc*1-50	9.00	19.63	2975.40	1.51	8.75	20.75	2965.76	3.29	9.00	20.38	2948.56	1.00
rc*2-50	34.75	88.00	10216.72	14.94	34.75	87.50	10194.07	481.74	34.88	80.88	9466.77	11.37
c*1-100	11.22	20.89	3279.04	3.68	11.44	21.56	3266.81	5.31	11.11	20.33	3217.79	2.97
c*2-100	32.00	72.25	10381.67	31.76	32.13	73.00	10185.00	521.78	32.38	68.63	9497.83	27.78
r*1-100	11.67	24.08	3160.05	4.17	11.75	25.00	3126.62	12.69	11.58	24.33	3101.71	3.41
r*2-100	49.64	121.55	12311.07	71.99	49.64	120.91	12286.95	580.93	49.27	107.18	11018.73	61.56
rc*1-100	10.50	21.13	3207.39	3.46	10.63	21.75	3173.95	5.76	10.50	20.88	3150.69	2.78
rc*2-100	42.63	103.88	11962.38	52.16	43.00	103.88	11932.72	558.44	42.75	94.38	10983.27	45.84
c*1-200	11.80	21.00	3273.55	7.06	11.90	21.90	3247.12	8.53	11.80	22.80	3205.34	6.35
c*2-200	33.70	70.10	10347.67	68.44	33.70	68.90	10242.72	534.96	34.60	68.80	9641.03	63.50
r*1-200	16.00	25.60	3209.68	14.37	19.00	29.70	3166.47	83.40	20.50	27.40	3036.41	13.44
r*2-200	80.40	135.00	16034.64	397.53	83.40	139.50	15923.66	907.89	86.30	126.10	13147.38	347.32
rc*1-200	17.40	26.30	3260.44	14.04	17.70	29.00	3201.31	111.79	19.10	26.50	3133.65	12.96
rc*2-200	79.00	134.00	16012.48	332.15	82.30	140.00	15912.79	846.41	84.40	129.90	13070.97	283.31
Average	29.50	61.02	7387.89	58.53	30.07	62.26	7335.99	313.97	30.52	57.81	6700.16	50.61

From methods to implementation and results

we stop CPLEX when an integer feasible solution has been proved to be within 0.05% of optimality or when running time limit set to 25 seconds is reached. Given the results presented in Table 4.11, VNS*I is two to thirty times slower than VNS* without a significant improvement on the solution quality. VNS* provides slightly better results than those provided by VNS*I compared to the results obtained without loading neighborhoods. Despite this, we kept the same stopping criteria when solving the quadratic program. Indeed, deriving better results at the expense of longer run times is not our goal. To conclude, VNS* is the best configuration providing high quality solutions without a large computational effort. The efficiency of the VNS* depends on the type of the time windows and the instances size. This finding suggests that the larger computational times required by VNS* are always due to the solution of instances (c*2-200, r*2-200 and rc*2-200). Detailed results are provided in Appendices A.5-A.9.

4.2.3.2 Sensitivity analysis

The proposed matheuristic embeds different components that contribute to the performance of VNS*. In order to better analyze the contribution of the main components, we conduct some additional experiments reported in this section. In these experiments, the performance of each setting is assessed by reporting the average deviation from the best solution found over the three classes of instances (c*1-100, r*1-100 and rc*1-100). The results provided by VNS* are used as reference solutions. The parameters setting is unchanged.

First, we study the impact of the multi-start constructive heuristic by running VNS* with the same initial solution for the 20 iterations, i.e., lines (5-7) are removed from Algorithm 1 in Chapter 3. We obtain an average percentage gap of 1.04%, 6.33% and 4.42% for classes: c*1-100, r*1-100 and rc*1-100 respectively. These results confirm that starting from a new solution at each iteration plays an important role in the effective exploration of the solution space.

One critical component in VNS* is the *Get_Compartment* heuristic. To prove its efficiency, we solve the Quadratic Multiple Knapsack Problem with Conflicts for the final solution s^* obtained by VNS*. We denote the new solution \bar{s} . We compare the residual capacity of solutions s^* and \bar{s} . No improvement has been obtained on the sample instances. Therefore, the loading feasibility check heuristic provides a very effective (most likely the optimal) assignment of products to compartments. It represents a key component in the VNS* framework. Furthermore, we test the VNS* without the *Waiting_time_optimization* heuristic. The average results obtained are slightly worse than those obtained when the

Waiting_time_optimization heuristic is applied. Hence, optimizing the total route duration contributes to the quality of the obtained solution.

Table 4.12 summarizes the behavior of each removal and insertion heuristic, invoked in the current version of VNS*, when used in a LNS scheme combining one insertion heuristic and one removal heuristic. For these tests, we compare the quality of the solution obtained by VNS* without the *Waiting_time_optimization* heuristic. The first four configurations describe the impact of the removal heuristics when the insertion heuristic is fixed. The results show that there is no significant difference between the 4 removal heuristic performance. The aim of a removal heuristic is to diversify the solution, which seems to be ensured by the 4 heuristics in a similar way. However, the combination of the 4 removal heuristics leads to improve the exploration of the solution space.

For the following experiments (conf.5, conf.6 and conf.7), we select arbitrarily the similarity removal heuristic to perturb the solution. In these experiments, all insertion heuristics perform well expect insertion heuristic (1). The insertion heuristic (5) performs better with an average gap equal to 1.32% followed by the insertion heuristics (6) and (2). The insertion heuristic (1) gives the worst average gap 12.20%. Among the 8 settings described in Table 4.12, conf.6 turns out to be the most efficient combination on the testbed. Nevertheless, it is worth while that applying conf.6, coupling the insertion heuristic (5) and the similarity removal heuristic, on the OP and the OPTW fails to provide good quality solutions while the VNS* performs particularly well. These observations support the idea that combining several removal and insertion heuristics may have a positive impact on the solution quality and that designing the matheuristic with different neighborhoods is definitely a good option.

Table 4.12: Effectiveness of the insertion and removal heuristics

<i>Conf.</i>	Similarity	<i>Removal heuristics</i>		Spatio-temporal	<i>Insertion heuristics</i>				<i>Gap%</i> c*1-100 r*1-100 rc*1-100
		Worst	Random		Insert (1)	Insert (2)	Insert (5)	Insert (6)	
1	•					•			3.84 3.78 3.70
2		•				•			3.19 4.81 3.65
3			•			•			3.66 4.29 3.95
4				•		•			3.78 5.12 3.61
5	•				•				5.33 12.85 18.41
6	•						•		0.75 1.88 1.32
7	•							•	3.05 3.08 3.57
VNS*	•	•	•	•	•	•	•	•	0.00

4.3 Conclusions

In this chapter, we presented the data structure developed of our unified framework, representing the problem instance and the solution. Then, extensive experimental results show that VNS* competes with the best known state-of-the-art methods proposed for the OP and the OPTW without any customization. To better evaluate the matheuristic performance on rich instances, we generated three data sets. Each data set includes 172 instances and describes three real-life scenarios. As expected, VNS* performs specially well on the less constrained instances. In the presence of incompatibility restrictions, VNS* produces high quality solutions on average with a larger computational effort. The sensitivity analysis reveals that the exact loading neighborhoods contribute to the VNS* performance significantly. They lead to improve the solution quality with no significant time increase. Furthermore, starting from a different solution at each iteration and combining removal and insertion neighborhoods contribute to the quality of solutions.

As presented in Chapter 2, the RPTP solution corresponds to the solution of the pricing subproblem arising in the column generation scheme designed for solving the MDMCMCm-VRPTW. In the next chapter we investigate this research avenue.

Chapter 5

Column generation heuristic

This chapter is devoted to describe the column generation heuristic, which hybridizes the VNS* matheuristic and the column generation approach to solve a variety of VRPs. In the following sections, we first describe the components of the column generation heuristic. Specifically we present the heuristic for generating the initial set of columns. Next, we summarize the accelerating strategies performed to solve the NP -hard pricing subproblem. Last, we model the post-processing procedure invoked to re-optimize the final solutions. The performance of the column generation heuristic is assessed on a set of 14 CVRP instances including up to 199 customers and the test bed for the MCMCm-VRPTW generated in Chapter 4. The computational results reported show that the proposed column generation heuristic performs moderately well in terms of solution quality and computational efficiency, so far. In this chapter, we describe a preliminary version of column generation heuristic and several potential strategies will be explored in depth in future research for further improvements.

5.1 Introduction

As pointed out in Chapter 2, the DW decomposition leads to a model with a potentially exponential number of new variables and to a set of pricing problems. We showed in Chapter 2 that it can be modeled as a rich profitable tour problem. We refer to [Lübbecke and Desrosiers \(2005\)](#) as well as the book [Desaulniers et al. \(2005\)](#) for comprehensive surveys on column generation.

Since the subproblem is NP -hard, it seems to be very difficult to solve it efficiently. Therefore, we choose to solve it heuristically. The VNS* matheuristic can handle all the complex issues of the MDMCMCm-VRPTW efficiently. Figure 5.1 sketches the proposed column generation heuristic. Initialization involves adding

to the RMP a number of columns to obtain a feasible solution. The initial set of columns is obtained by applying an effective constructive heuristic described in Section 5.2.1. An iteration of a column generation consists in solving the RMP in order to (i) identify the current optimal objective function value $\sum_{r \in \bar{\mathcal{R}}} c^r$ and dual variables π_j^p associated with active constraints and to (ii) find a new column, Λ^r , if there is still one, with a negative reduced cost \hat{c}^r . The RMP is gradually enriched by new columns by iteratively solving pricing subproblems, which corresponds to solving the MDMCMCm-VRPTW with one vehicle. At each iteration, the VNS* matheuristic is invoked to rapidly generate columns with positive costs since the subproblem is formulated as a maximization problem. If there is no improving columns or the maximum column generation iterations, $iter_{CG}^{Max}$, is reached, we solve the resulting RMP with integer requirements. Otherwise, the column found is added to the RMP which is solved by the primal simplex method.

5.2 Column generation heuristic components

5.2.1 Initial solution heuristic

Starting from a promising initial solution is a key feature in column generation. This helps the column generation process get a fast increase in the quality of dual variables. We use a fast heuristic based on the VNS* matheuristic described in Chapter 3 which produces a set of feasible routes. It is noteworthy that generating columns provided by a few iterations of the VNS* or by the constructive heuristic NNA as well as the trivial depot-customer-depot routes yields to poor quality initial solution. The basic steps of the proposed matheuristic are provided in Algorithm 5.

The heuristic is performed L times. In each iteration ℓ , a solution \mathcal{S}_ℓ is build including a set of routes which satisfy all the customers orders. We propose a method to implicitly cluster the non-satisfied customer orders located in the same region. At each iteration, we build a route depending on a seed customer denoted h . For the first route built customer h corresponds to the ℓ *th* farthest customer from the depot d_o . For the remaining routes, the seed customer h corresponds to the farthest customer from the depot with at least one non-satisfied order. For each new route, we need to recalculate the profit g_i^p associated with each order o_i^p with respect to the distance, $dis_{d_o h}$, separating the seed customer h and the depot according to equations 5.1. The better choices are then to visit customers as close as possible to the seed customer h with the highest profit g_i^p . Through preliminary experimentations, we notice that instances with a large number of orders require more diversification

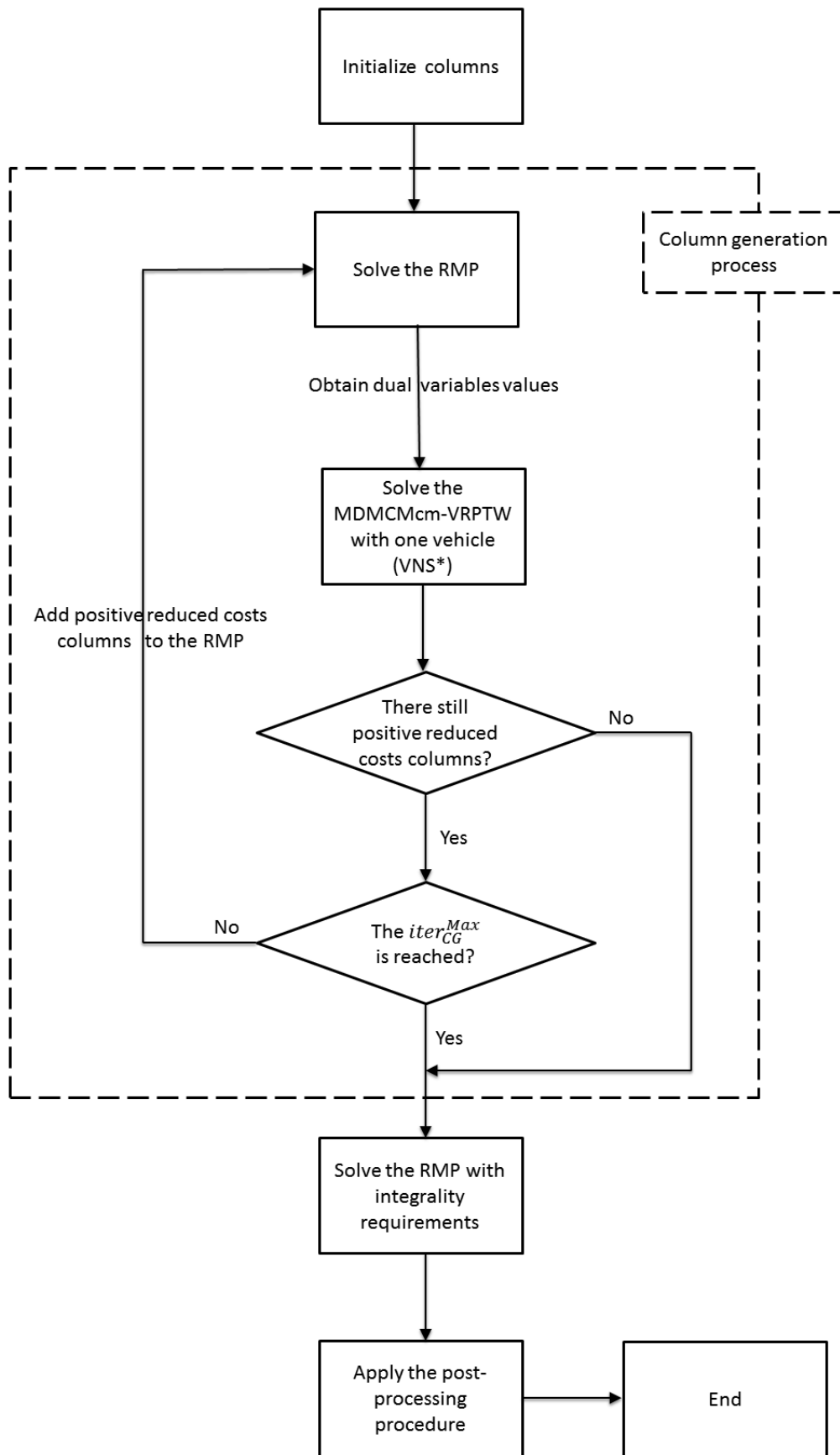


Figure 5.1: Column generation based heuristic

than small ones. We set L to $n/2$ for instances with up to 75 customers and to n for instances with more than 75 customers. This heuristic produces a set a meaningful routes satisfying all the customers orders. The obtained feasible solutions routes are then included in the initial set of columns for the column generation heuristic. With these initial columns, column generation is usually faster and can produce higher quality heuristic solutions.

$$g_i^p = \left(\frac{dis_{dol}}{dis_{doi} + dis_{il}} \right)^3 * 100 \quad (5.1)$$

Algorithm 5: Heuristic to construct an initial set of columns

```

input : (Problem Class )
output: ( $\{\mathcal{S}_\ell \forall \ell \in L : \mathcal{S}_\ell = \{s_\ell^1, s_\ell^2, \dots, s_\ell^j\}\}$  )
1 for ( $\ell \leftarrow 1$  to  $L$ ) do
2    $exist \leftarrow 1$ ;
3    $j \leftarrow 1$ ;
4   Initialize  $\mathcal{S}_\ell, \mathcal{S}_\ell \leftarrow \emptyset$ ;
5   while ( $exist == 1$ ) do
6      $exist \leftarrow 0$ ;
7     if ( $j == 1$ ) then
8        $h \leftarrow$  the  $\ell$  th farthest customer from the depot;
9     else
10       $h \leftarrow$  the farthest customer from the depot;
11    end
12    Compute the profits associated with undelivered orders;
13    Let  $s_\ell^j$  be an empty solution;
14    Apply the NNA heuristic on  $s_\ell^j$ ;
15    Apply the VNS* with  $iter^{max} = 5$  on  $s_\ell^j$ ;
16    Apply the Waiting_Time_Optimization procedure on  $s_\ell^j$ ;
17     $\mathcal{S}_\ell = \mathcal{S}_\ell \cup s_\ell^j$ ;
18    for (all the orders in the Problem Class) do
19      if (there still at least one order not satisfied) then
20         $exist \leftarrow 1$ ;
21        break;
22      end
23    end
24     $j ++$  ;
25  end
26  return  $\mathcal{S}_\ell$ ;
27 end

```

5.2.2 Accelerating strategies for solving the pricing problem

Generating an attractive and reasonable number of columns is definitely the bottleneck for column generation, paired with the time to solve the RMP with a large number of columns to integrality. Some accelerating strategies and dominance rules are used to speed up the generation process.

- Regarding the column generation process, we apply two dominance rules while solving the pricing subproblem. First, we retain only distinct columns having a different reduced cost. Second, in each call for the VNS* matheuristic, we remove dominated routes from the pool of columns added to the RMP. A route s^* dominates a route \bar{s} if \bar{s} covers the same subset of customers visited by s^* and incurs a higher travel cost.
- A well known strategy for accelerating column generation is to return many positive reduced costs columns at each iteration when we solve the pricing subproblem. Even though only one column needs to be returned, returning several ones when they are available have been proved to be beneficial experimentally (Larsen, 1999, Desaulniers et al., 2005).
- The column generation approach does not automatically guarantee integer solutions. The solutions obtained will indeed be fractional. We impose decisions on the route variables Λ^r to reduce the search space before the linear optimization is started. This can be achieved by fixing some variables to 1 when no improvement of larger than 0.05% in the objective value of the RMP has been realized in the last $iter_{CG}^{Fix}$ column generation iterations ($iter_{CG}^{Fix}$ was set to 3 for our experiments). It may be of interest to fix variables, since due to the reduced size of the resulting search tree, potentially more improvements could be possible in limited time. Nevertheless, note that so far, we consider the strategy that keeps all columns in the RMP once they were added because, for some instances columns with fractional value near to 0 are part of the optimal solution.
- Since we aim to design a heuristic approach which is competitive to other state-of-the-art methods, we choose to not solve the linear RMP relaxation to optimality. We put a limit on the number of column generation iterations $iter_{CG}^{Max}$ which equals to 100 for our experiments. In addition, we set an early termination criterion by means of a gap limit equals to 3%. For many of the instances considered, this limit does not deter to obtain good solutions while

it could in turn lead to worse solutions for larger instances.

5.2.3 Post-processing procedure

In Chapter 2, the MP is modeled as a set covering problem where constraints 2.43 enable a customer order to be covered more than once in a solution. We overcome this problem by removing redundant orders (i.e., delivered more than once in the final solution) and apply a post-processing procedure on these routes. This procedure completes the resulting column generation heuristic and allows to find consistently solutions that improve the distance costs. This repair process is performed only once for the final routes and is based on the solution of an assignment problem. Each customer is treated as an independent vertex regardless of the number and type of orders covered by this route. We compute c_i^r the distance incurred by visiting customer i by route r . c_i^r takes a big positive value if customer $i \notin r$ in order to forbid relocating customers between routes. We define $\hat{\mathcal{R}}$ the set of final routes obtained from solving the integer MP and \mathcal{P}_i the set of orders placed by customer i . We introduce the following decision variables. Binary variables x_i^{pr} indicate if order o_i^p is covered by route r . Binary variables y_i^r indicate if customer i is visited by route r . The mathematical model for the post-processing phase is as follows:

$$\min \sum_{r \in \hat{\mathcal{R}}} \sum_{i \in \mathcal{V}} c_i^r y_i^r \quad (5.2)$$

subject to

$$\sum_{r \in \hat{\mathcal{R}}} x_i^{pr} = 1 \quad i \in \mathcal{V}, p \in \mathcal{P}_i \quad (5.3)$$

$$\sum_{p \in \mathcal{P}_i} x_i^{pr} \geq y_i^r \quad i \in \mathcal{V}, r \in \hat{\mathcal{R}} \quad (5.4)$$

$$\sum_{p \in \mathcal{P}_i} x_i^{pr} \leq |\mathcal{P}_i| y_i^r \quad i \in \mathcal{V}, r \in \hat{\mathcal{R}} \quad (5.5)$$

$$x_i^{pr} = 0 \quad i \in \mathcal{V}, p \in \mathcal{P}_i, o_i^p \notin r. \quad (5.6)$$

The objective function (5.2) minimizes the sum of the distance costs. Constraints (5.3) ensure that an order o_i^p is covered exactly once. Constraints (5.4) guarantee that if a customer i is visited by a route r then at least one order o_i^p should be delivered by this route. Constraints (5.5) put an upper bound on the maximum number

of delivered orders for a given customer i by a route r . Constraints (5.6) put to 0 variables x_i^{pr} if order o_i^p does not appear on route r .

5.3 Computational experiments

Due to the large variety of constraints which have been considered in the MDMCMCm-VRPTW, a large number of derived problems exists and it is not possible to perform tests on all of them. Hence, we choose to solve the MCMCm-VRPTW, positioned at the top of Figure 2.2 and the CVRP positioned at the bottom of this figure. We first present the results on instances for the CVRP and then on the testbed generated in Chapter 4. We ran the experiments on a computer with 2.50 GHz and 4 GB Ram. The algorithms have been implemented in language C and we use Cplex 12.5 to solve mathematical programs. The number of deleted customer kd has been adjusted through preliminary tests and it is set to 4 for the experimentations conducted in this chapter. We allow the VNS* invoked to build the initial solution to run over 5 iterations. All the remaining parameters are maintained unchanged.

5.3.1 CVRP instances

The first set of instances is the well-known CVRP instances introduced by Christofides et al. (1979) which can be downloaded from <http://www.bernabe.dorronsoro.es/vrp/>. The 14 problem instances (vrpnc) contain 50 to 199 customers. Instances vrpnc6– vrpnc10, vrpnc13 and vrpnc14 include route length restrictions. Results are summarized in Table 5.1. To assess the efficiency of the initial constructive heuristic and the post-processing procedure, we report three costs obtained in three different phases of the column generation heuristic. $Cost_1$ corresponds to the initial objective value of the RMP. $Cost_2$ and $Cost_3$ correspond respectively to the objective value of the RMP before and after the post-processing procedure. The BK column reports the BK solutions for the CVRP as listed in Laporte et al. (2000). The Euclidean distance for the CVRP instances are real distances.

Although the framework proposed in this thesis is designed to solve RVRPs, it provides reasonable results on the CVRP, especially for small and medium-sized instances. The average gap obtained is equal to 5.41% for the instances vrpnc1–vrpnc3 and vrpnc6–vrpnc8 including up to 100 customers. Unfortunately on larger instances, the column generation heuristic behavior has to be improved. The cardinality of \mathcal{R} becomes prohibitive and we still have to develop some efficient rules to

manage such a large pool.

Preliminary experimentations show that incorporating the 6 insertion heuristics described in Chapter 2 in the VNS* improve the overall average results on the CVRP instances. For example, the average gap obtained for instance *vrpnc3* is reduced from 5% to 2.5%. However, this configuration turns out to be very time consuming, so it will not be considered.

The column generation heuristic takes longer running times than the best known classical VRP heuristics. It takes less than 5 minutes on average to solve instances with 50, 75 and 100 customers. This is not surprising since some procedures are invoked even if they are of no interest for the solution of CVRP instances.

The costs of the initial set of columns computed with the heuristic described in Section 5.2.1 are close to those resulting from the column generation heuristic. *Cost_3* outperforms *Cost_1* with less than 10% on average. Indeed, since the heuristic starts with a set of feasible and good quality columns, it is called with tight dual variables values which avoids generating many columns with a large computational effort. The success of post-processing procedure is apparent from most of the instances since the total costs is reduced after its use.

Table 5.1: Performance of the column generation heuristic on the CVRP instances

<i>Instance</i>	<i>n</i>	<i>BK</i>	<i>Column Generation heuristic</i>				
			<i>Cost_1</i>	<i>Cost_2</i>	<i>Cost_3</i>	<i>Gap%</i>	<i>Time(s)</i>
<i>vrpnc1</i>	50	524.61	577.02	524.93	524.93	0.06	65.31
<i>vrpnc2</i>	75	835.32	973.91	891.54	890.78	6.64	103.41
<i>vrpnc3</i>	100	826.14	1031.78	867.70	867.64	5.02	477.76
<i>vrpnc4</i>	150	1031.07	1370.59	1099.27	1096.57	6.35	1338.27
<i>vrpnc5</i>	199	1311.35	1705.94	1375.32	1371.7	4.60	3157.9
<i>vrpnc6</i>	50	555.43	606.40	583.33	583.33	5.02	85.87
<i>vrpnc7</i>	75	909.68	1072.41	964.88	964.88	6.07	133.17
<i>vrpnc8</i>	100	865.94	958.74	953.12	949.33	9.63	750.21
<i>vrpnc9</i>	150	1162.89	1333.36	1293.37	1293.37	11.22	2596.44
<i>vrpnc10</i>	199	1404.75	1549.13	1519.27	1516.63	7.96	4817.52
<i>vrpnc11</i>	120	1042.11	1458.36	1293.62	1293.62	24.13	1686.33
<i>vrpnc12</i>	100	819.56	1080.63	952.31	927.97	13.23	289.83
<i>vrpnc13</i>	120	1545.93	1864.93	1864.93	1864.93	20.63	635.64
<i>vrpnc14</i>	100	866.37	1017.77	1017.77	1017.77	17.48	684.41

5.3.2 MCMCm-VRPTW instances

We conduct experiments on the test bed generated in Chapter 4, specifically on instances of *type C* with 50 and 100 customers. For the 100-customers instances, we consider only the sets r*1-100, c*1-100, rc*1-100 with tight time windows. For Solomon’s instances, the average width of the time windows and the number of customers with constraining time windows vary from one instance to another within the same class. In the test bed generated for the MCMCm-VRPTW, the difficulty of instances is correlated to the number of orders rather than to the number of customers. As shown in Table 5.2, the average number of orders vary from one instance to another within the same class. For example, instances with 50 customers contain between 50 and 150 customer orders since $P = 3$. Detailed computational experiments are reported in Appendix A.

Table 5.2: Classes description

<i>Class</i>	<i>n</i>	<i>#Orders</i>
c*1-50	50	123.00
c*2-50	50	126.13
r*1-50	50	112.08
r*2-50	50	112.27
rc*-50	50	124.25
rc*2-50	50	127.25
c*1-100	100	251.00
r*1-100	100	219.67
rc*1-100	100	235.75

Table 5.3 presents the results of the column generation heuristic on 6 data sets of the MCMCm-VRPTW including 112 different instances. Only $Cost_1$, $Cost_3$, the running time in seconds and the average number of used vehicles are reported for the sake of brevity. Column headed $|\mathcal{K}|_{VRPTW}$ indicates the average number of used vehicles in the BK or the optimal solution of Solomon’s instances designed for the VRPTW. We succeed to solve the 112 instances. Table 5.3 shows that the average solution time is almost 25 minutes for instances with 100 customers and less than 4 minutes for instances of type 1 including 50 customers. However, it is noteworthy that for solved instances, the heuristic stopping criterion is the fixed number of column generation iterations. This clearly shows that the column generation heuristic could have found better solutions if we have allowed more time to

Table 5.3: Results of the column generation heuristic on the MCMCm-VRPTW instances

textitClass	<i>Cost_1</i>	<i>Cost_3</i>	$ \mathcal{K} $	$ \mathcal{K} _{VRPTW}$	<i>Costs gap%</i>	<i>Time(s)</i>
c*1-50	556.61	503.79	6.33	4.8	10.54	341.16
c*2-50	817.69	757.32	4.75	2.75	8.90	1462.84
r*1-50	919.44	826.62	8.08	8.41	11.88	190.77
r*2-50	978.17	923.08	3.82	4.18	6.84	2037.73
rc*1-50	883.15	875.84	7.88	7	0.75	182.80
rc*2-50	1136.11	1083.69	4.38	4.25	5.60	3725.05
c*1-100	1137.48	1071.66	12.56	10	6.10	1641.92
r*1-100	1410.48	1297.07	13.33	13.5	8.91	1492.93
rc-1-100	1573.27	1509.35	13.75	13.12	4.17	1383.09

solve the instances. Nevertheless, we had to make a trade-off between solution quality and computational time.

We tried to solve the MCMCm-VRPTW instances which have 100 customers and larger time windows (c*2-100, r*2-100, rc*2-100) with the same parameter setting. We succeed to solve many of these instances despite of the straightforward implementation of the current version of column generation heuristic. However, for some instances, the heuristic runs out of memory because it cannot deal with the very large number of generated columns when the RMP is solved with integer requirements.

Considering the number of vehicles, we notice that the column generation heuristic produces solutions with almost the same or a slightly larger fleet size than the one used in the optimal solution of the VRPTW despite of the physical incompatibilities considered in the MCMCm-VRPTW. This highlights the performance of the proposed heuristic especially as the primary objective of the VRPTW consists in the minimization of the number of vehicles used.

The proposed column generation heuristic depends on some control parameters. Most of them are used to limit computational times in one or another way. Parameter setting should be done on different instances sets with different characteristics. A sensitivity analysis on the most sensible parameters should be conducted to adapt these parameters to each instances set.

5.4 Conclusions

We described a column generation heuristic for the MDMCMC_m-VRPTW. The most important contributions of this chapter are the design of this heuristic and the introduction of the post-processing algorithm to re-optimize the resulting routes based on the solution of a mathematical program. Although a basic implementation of these ideas only allows the solution of medium sized instances, by managing more efficiently the pool of generated columns we believe that we will be able to solve instances with up to 200 customers. Future research will focus also on developing more efficient accelerating strategies.

We tried to assess the performance of the column generation heuristic on real data sets arising in the olive oil collection process in Tunisia. This real-life case study is the topic of Chapter 6. The real instances are defined over a set of periods and use a limited size fleet. Cleaning activities may be required between successive routes. One can run the column generation heuristic on each period. The solutions obtained may be modeled as minimum-cost flow with the cost associated with each arc is equal to the cost incurred by the cleaning activity if it is required, 0 otherwise. However, we are not able to overcome the difficulty of limiting the fleet size in the column generation process. Therefore, we propose an exact branch-and-cut algorithm to solve these real data sets.

Chapter 6

Real case study: The collection of olive oil in Tunisia

In this chapter, we introduce, model and solve to optimality a rich multi-product, multi-period and multi-compartment vehicle routing problem with a required compartment cleaning activity. This real-life application arises in the olive oil collection process in Tunisia, where regional collection offices dispose of a fleet of vehicles to collect one or several grades of olive oils from a set of producers. For each grade, the quantity offered by a producer changes dynamically over the planning horizon. We provide a mathematical formulation for the problem, along with a set of known and new valid inequalities. We propose an exact branch-and-cut algorithm in order to solve the problem. We evaluate the performance of the algorithm on real data sets under different transportation scenarios to demonstrate to our industrial partner the advantages of using multi-compartment vehicles.

An article based on this chapter was submitted for publication: R. Lahyani, L.C. Coelho, M. Khemakhem, G. Laporte, F. Semet. A multi-compartment vehicle routing problem arising in the collection of olive oil in Tunisia. *OMEGA*, submitted for publication.

6.1 Introduction

We introduce, model and solve a real-world application of a multi-product, multi-period and multi-compartment vehicle routing problem (MPPC-VRP) arising in the collection of olive oil in Tunisia. In 2012, that country was the fourth largest exporter of olive oil worldwide, with an export production of 163,000 tons. This amount was expected to increase in 2013 according to the General Directorate for Research at the Ministry of Agriculture. For climatic and geographical reasons, olive groves are rather widespread in the central part of the country, as shown in Figure 6.1. Collecting olive oil is particularly important during the four-month production season. It mobilizes considerable human and material resources, and timeliness is crucial in this operation. Producers work non-stop 24 hours a day in order not to damage the harvest. On any given day, olive oil collection is carried out over six periods lasting almost 24 hours in total. This activity is performed by a fleet of capacitated heterogeneous vehicles, often with compartments of equal or different sizes, all equipped with a debit meter, enabling the decision maker to have full knowledge of the load contained in each compartment at all times. The oil must be collected before the producer runs out of storage space. A good forecast is available for the production rate of each product by each producer.

Olive oil comes in three different grades known as *extra*, *virgin*, and *lampante*. The top two grades with superior tastes are extra and virgin, which are suitable for consumption, whereas lampante oil is mostly destined for industrial uses. The transportation is regulated by law in order to protect the natural flavors of the oils. In particular, the different grades must be kept separate during transportation, hence the need to have multi-compartment vehicles. It is forbidden to load superior grades and lampante oil consecutively in the same compartment, unless it has been cleaned before the changeover. The cleaning activity generates a cost and takes time.

Routing problems with cleaning activity have not been widely studied from a scientific perspective, but similar constraints appear in other contexts, although several features of our problem are different. Thus, [Oppen et al. \(2010\)](#) consider the problem of transporting different types of live animals from farms to slaughterhouses by means of multi-compartment vehicles. They add time between consecutive tours to allow for unloading and disinfection of the vehicles. [Hvattum et al. \(2009\)](#) deal with a tank allocation problem arising in the shipping of bulk oil and chemical products by tanker ships. They consider that a cleaning activity is required if two incompatible products are assigned to the same compartment within less than three trips.



🏠: Regional offices, (1-9, A-G): Producers

Figure 6.1: Map of Tunisia pinpointing producers and regional offices locations (Source: Google Maps, accessed March 2014)

Transporting oil and fuel with multi-compartment vehicles is more challenging and interesting from a scientific point of view than transporting food, as explained in Chapter 2. In this chapter, we introduce other complicating constraints reflecting the complexities of the olive oil transport. The mathematical model developed for this application considers two vehicle types which differ in the number and sizes of the compartments. In the olive oil industry, a producer typically offers several products in different quantities in each period. The grades and quantities offered depend on the producer. All offers must be picked up in each period. These assumptions, together with the presence of the cleaning activity, increase the difficulty of the problem and make it very difficult to be solved by exact algorithms for most instances of realistic sizes.

From a scientific perspective, we introduce, model, and solve exactly a difficult and rich variation of the well-known vehicle routing problem. Part of the complexity of the problem comes from the requirement to clean the compartments, depending

on the assignment and scheduling decisions. From a practical standpoint, we provide a tool that can assist managerial decision making at the tactical and operational levels. In particular, we are able to compare several transportation alternatives for the service provider and to evaluate the potential routing savings yielded by replacing single compartment vehicles with multi-compartment ones.

The remainder of this chapter is organized as follows. In Section 6.2 we provide a precise description of the problem and we introduce a mathematical model complemented with known and new valid inequalities. The branch-and-cut algorithm is described in Section 6.3. We present computational results on real and artificially generated instances in Section 6.4, followed by conclusions in Section 6.5.

6.2 Mathematical description of the problem

We first introduce some notation, followed by a mathematical model and valid inequalities.

6.2.1 Notation

The MPPC-VRP is defined on an undirected graph $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{\mathcal{E}})$, where $\bar{\mathcal{V}}$ is the vertex set and $\bar{\mathcal{E}}$ is the edge set. The set $\bar{\mathcal{V}} = \{0, \dots, n\}$ contains the locations of the depot 0 and of the producers $\mathcal{V} = \{1, \dots, n\}$. The set $\bar{\mathcal{E}} = \{(i, j) : i \in \bar{\mathcal{V}}, j \in \mathcal{V}, i < j\}$ defines the edges of the problem. A routing cost α_{ij} is known for each edge $(i, j) \in \bar{\mathcal{E}}$. The problem is defined over a finite planning horizon $\mathcal{T} = \{1, \dots, T\}$, and a set $\mathcal{K} = \{1, \dots, K\}$ of heterogeneous vehicles is available at the depot. Each vehicle k contains a set \mathcal{W}^k of compartments equipped with a debit meter, and each compartment $w \in \mathcal{W}^k$ has a capacity Q^{wk} . The use of vehicle k incurs a fixed cost β^k . Identical products collected from different producers can be loaded into the same compartment provided there is sufficient capacity. Also, each producer may be visited by more than one vehicle in any given period.

A set $\mathcal{P} = \{1, \dots, P\}$ of products are offered. For ease of notation, we assume that products are ordered according to their grade, and if the product of the lowest grade, i.e., P , is loaded into a compartment of a vehicle, it contaminates the compartment and a cleaning procedure is required to load any product $p < P$ ($p \in \mathcal{P}$) in subsequent periods. If a cleaning procedure is undertaken, it generates a cost δ . Each producer $i \in \mathcal{V}$ offers a quantity q_i^{pt} of product of type p in period t with which is associated a transportation request. Not all producers offer all types of products, and the available quantities change from one period to the next.

6.2.2 Mathematical model

We now provide an integer linear programming formulation for the MPPC-VRP, followed by known and new valid inequalities used to strengthen the mathematical model. The decision variables are defined as follows. The routing variables x_{ij}^{kt} are equal to the number of times edge (i, j) is used by vehicle k in period t ; the visiting variables z_i^{kt} are binary and equal to one if and only if vertex i is visited by vehicle k in period t ; the assignment variables y_{ip}^{wkt} are equal to one if and only if product p of producer i is loaded into compartment w of vehicle k in period t ; likewise, the variables w_p^{wkt} are equal to one only if product p is loaded in compartment w of vehicle k in period t . Finally, the variables u^{wkt} are binary and indicate whether or not compartment w of vehicle k is cleaned at the beginning of period t .

Recall that not all producers offer all products at each period. Nevertheless, we define variables y_{ip}^{wkt} for all combinations of producers and products, because this simplifies the formulation. If a producer i does not offer product p in period t , we then set $y_{ip}^{wkt} = 0$ for all compartments w and vehicles k . Finally, we need to define the set \mathcal{T}_s^t containing the periods elapsed between s and t , i.e., $\mathcal{T}_s^t = \{s + 1, \dots, t - 1\}$.

The problem can then be modeled as follows:

$$\text{minimize } \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} \sum_{(i,j) \in \bar{\mathcal{E}}} \alpha_{ij} x_{ij}^{kt} + \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{V}} \beta^k z_0^{kt} + \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{K}} \sum_{w \in \mathcal{W}^k} \delta u^{wkt} \quad (6.1)$$

subject to

$$\sum_{i \in \mathcal{V}} q_i^{pt} y_{ip}^{wkt} \leq Q^{wk} \quad k \in \mathcal{K}, w \in \mathcal{W}^k, p \in \mathcal{P}, t \in \mathcal{T} \quad (6.2)$$

$$y_{ip}^{wkt} \leq z_i^{kt} \quad i \in \mathcal{V}, k \in \mathcal{K}, w \in \mathcal{W}^k, p \in \mathcal{P}, t \in \mathcal{T} \quad (6.3)$$

$$y_{ip}^{wkt} \leq w_p^{wkt} \quad i \in \mathcal{V}, k \in \mathcal{K}, w \in \mathcal{W}^k, p \in \mathcal{P}, t \in \mathcal{T} \quad (6.4)$$

$$w_p^{wkt} \leq \sum_{i \in \mathcal{V}} y_{ip}^{wkt} \quad k \in \mathcal{K}, w \in \mathcal{W}^k, p \in \mathcal{P}, t \in \mathcal{T} \quad (6.5)$$

$$\sum_{p \in \mathcal{P}} w_p^{wkt} \leq 1 \quad k \in \mathcal{K}, w \in \mathcal{W}^k, t \in \mathcal{T} \quad (6.6)$$

$$\sum_{j \in \bar{\mathcal{V}}, i < j} x_{ij}^{kt} + \sum_{j \in \bar{\mathcal{V}}, j < i} x_{ji}^{kt} = 2z_i^{kt} \quad i \in \bar{\mathcal{V}}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.7)$$

$$\sum_{i \in \mathcal{Z}} \sum_{j \in \mathcal{Z}, i < j} x_{ij}^{kt} \leq \sum_{i \in \mathcal{Z}} z_i^{kt} - z_z^{kt} \quad \mathcal{Z} \subseteq \mathcal{V}, z \in \mathcal{Z}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.8)$$

$$\sum_{k \in \mathcal{K}} \sum_{w \in \mathcal{W}^k} y_{ip}^{wkt} = 1 \quad i \in \mathcal{V}, p \in \mathcal{P}, t \in \mathcal{T} \quad (6.9)$$

$$w_P^{wks} + w_p^{wkt} \leq 1 + u^{wkt} + \sum_{t' \in \mathcal{T}_s^t} u^{wkt'} \quad k \in \mathcal{K}, w \in \mathcal{W}^k, p \in \mathcal{P} \setminus \{P\},$$

$$s, t \in \mathcal{T} \quad (6.10)$$

$$x_{0j}^{kt} \in \{0, 1, 2\} \quad j \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.11)$$

$$x_{ij}^{kt} \in \{0, 1\} \quad i, j \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.12)$$

$$y_{jp}^{wkt}, z_i^{kt}, w_p^{wkt}, u^{wkt} \in \{0, 1\} \quad i \in \bar{\mathcal{V}}, j \in \mathcal{V}, k \in \mathcal{K}, w \in \mathcal{W}^k,$$

$$p \in \mathcal{P}, t \in \mathcal{T}. \quad (6.13)$$

The objective function (6.1) minimizes the sum of the routing cost, the vehicle fixed cost and the compartment cleaning cost. Constraints (6.2) ensure that the capacity of each compartment is not violated. Constraints (6.3)–(6.5) link the variables y_{jp}^{wkt} , w_p^{wkt} and z_i^{kt} . Specifically, constraints (6.3) ensure that a product from a producer is loaded into a given compartment of a vehicle at a given period only if the producer is served by the vehicle. Constraints (6.4) and (6.5) guarantee that a compartment is allowed to carry a product in a given period only if the vehicle visits a producer offering that product in the same period. Constraints (6.6) ensure that each compartment carries at most one type of product at a time. Constraints (6.7) and (6.8) are degree and subtour elimination constraints, respectively. Constraints (6.9) ensure that all the quantities being offered are collected. Constraints (6.10) mean that a cleaning operation is performed if necessary. The term $\sum_{t' \in \mathcal{T}_s^t} u^{wkt'}$ keeps track of each compartment cleaning operations for the interval between periods s and t . Constraints (6.11)–(6.13) define the integer and binary nature of the variables.

6.2.3 Valid inequalities

The formulation defined by (6.1)–(6.13) is sufficient to model the MPPC-VRP. We can, however, strengthen it through the inclusion of valid inequalities in the form of symmetry breaking constraints and additional cuts imposing upper bounds on the integer variables. The first one is related to the period in which a cleaning operation takes place. For example, suppose a contaminating product is transported by a given compartment in period $t = 1$, and the next use of this compartment is to carry a higher grade product in period $t = 5$. Then, at least four optimal solutions exist, by cleaning the compartment in period 2, 3, 4, or 5. In order to avoid such

symmetries, we impose constraints (6.14) which postpone the cleaning operation as much as possible:

$$u^{wkt} \leq \sum_{p \in \mathcal{P}} w_p^{wkt} \quad k \in \mathcal{K}, w \in \mathcal{W}^k, t \in \mathcal{T}. \quad (6.14)$$

We also integrate the vehicle and the compartment symmetry breaking constraints for the first period, when no cleaning operation is necessary. These constraints are inspired from those proposed in (Coelho and Laporte, 2013a). Constraints (6.15) and (6.16) are valid if the considered vehicles and compartments are identical. We define the set $\mathcal{K}' \subset \mathcal{K}$ containing only the homogeneous vehicles of \mathcal{K} . Note that in the olive oil industry as in petrol distribution, multi-compartment vehicles with different compartments capacities are seldom used in order to reduce the imbalance of loaded vehicles on the road (Derigs et al., 2011a). These symmetry breaking constraints are

$$\sum_{i \in \bar{\mathcal{V}}} z_i^{k1} \leq \sum_{i \in \bar{\mathcal{V}}} z_i^{k-1,1} \quad k \in \mathcal{K}' \setminus \{1\} \quad (6.15)$$

and

$$w_h^{wk1} \leq \sum_{p \in \mathcal{P}} w_p^{w-1,k,1} \quad k \in \mathcal{K}, w \in \mathcal{W}^k \setminus \{1\}, h \in \mathcal{P}. \quad (6.16)$$

Constraints (6.15) rank identical vehicles according to the index of the producers served. In particular, they ensure that among vehicles of the same type, vehicle k cannot serve any customer if vehicle $k - 1$ has not already been used in the first period. Constraints (6.16) state that if deliveries are performed using compartment w , then compartment $w - 1$ is already used. These rules cannot be generalized to the remaining periods because they may impact the cleaning operation and ultimately affect the solution cost.

We also make use of a known set of useful cuts to enforce logical relationships between routing, visiting and assignments variables. For more details on logical inequalities for routing problems, see Coelho and Laporte (2013a) and Gendreau et al. (1998c). These cuts are as follows:

$$x_{0i}^{kt} \leq 2z_i^{kt} \quad i \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.17)$$

$$x_{ij}^{kt} \leq z_i^{kt} \quad i, j \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.18)$$

$$z_i^{kt} \leq z_0^{kt} \quad i \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.19)$$

$$z_i^{kt} \leq \sum_{p \in \mathcal{P}} \sum_{w \in \mathcal{W}^k} y_{ip}^{wkt} \quad i \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.20)$$

$$z_i^{kt} \leq \sum_{p \in \mathcal{P}} \sum_{w \in \mathcal{W}^k} w_p^{wkt} \quad i \in \mathcal{V}, k \in \mathcal{K}, t \in \mathcal{T} \quad (6.21)$$

$$\sum_{i \in \mathcal{V}} z_i^{kt} \leq \sum_{i \in \mathcal{V}} \sum_{p \in \mathcal{P}} \sum_{w \in \mathcal{W}^k} y_{ip}^{wkt} \quad k \in \mathcal{K}, t \in \mathcal{T}. \quad (6.22)$$

Constraints (6.17), (6.18) and (6.19) are referred to as routing cuts, as a way to ensure that if a producer i is visited, i.e., (6.17) holds, or a producer j is the successor of a producer i , i.e., (6.18) holds on the route of vehicle k in period t , then the producer i to be visited, i.e., $z_i^{kt} = 1$. Constraints (6.18) can also be considered as subtour elimination constraints (6.8) when $|\mathcal{Z}| = 2$. Inequalities (6.19) guarantee that if vehicle k visits producer i in period t , then the depot must be included in the route of vehicle k in period t . Through constraints (6.20) and (6.21), we ensure that if a producer i is visited in period t by vehicle k , then at least one product p should be loaded in some compartment of vehicle k in that period. Constraints (6.22) strengthen the relationships between the collection routes, products and compartments. Specifically, a collection route using vehicle k in period t exists to ensure the pick-up of some products from a producer i and load them in some compartment of that vehicle. We also note that constraints (6.22) are the sum over the locations of (6.20). Even if these constraints are redundant in this context, they are known to help the mathematical programming solvers to derive new cuts and improve the overall algorithmic performance (Coelho and Laporte, 2014, Gendron and Crainic, 1994, Jena et al., 2012, 2013).

6.3 Branch-and-cut algorithm

We have implemented a branch-and-cut algorithm capable of solving the formulation just presented. All variables of the formulation are explicitly handled by the algorithm. Since the number of constraints (6.17)–(6.22) is polynomial, they are

added a priori in the model. In the sequel, we will show how each subset of inequalities impacts its solution. At the opposite, we cannot generate all subtour elimination constraints (6.8) a priori since their number is exponential. These are dynamically generated as cuts as they are found to be violated. The formulation is then solved by branch-and-cut as follows. At a generic node of the search tree, a linear program with relaxed integrality constraints is solved, a search for violated constraints is performed, and appropriate valid inequalities are added to eliminate subtours, and the current subproblem is then reoptimized. This process is reiterated until a feasible or dominated solution has been reached, or until no more cuts can be added. At this point, branching on a fractional variable occurs. We provide in Algorithm 6 a sketch of the branch-and-cut scheme.

6.4 Computational experiments

In this section we describe the computational experiments carried out in order to assess the performance of our model and algorithm. We provide in Section 6.4.1 details of the real instances we have obtained from our Tunisian partner, and the instances generated with a different fleet composition. In Section 6.4.2 we describe the results of computational experiments performed to evaluate the effectiveness of the cuts and valid inequalities, and we compare our solutions with those corresponding to the current situation.

We have coded the algorithm in C++ using IBM CPLEX Studio 12.5.1 as the MIP solver. All computations were executed on a grid of Intel XeonTM processors running at 2.66 GHz with up to 96 GB of RAM installed per node, with the Scientific Linux 6.1 operating system. All instances and detailed results are available from <http://www.leandro-coelho.com>.

6.4.1 Instance generation

We have created a set of five instances based on real data gathered from industrial partners in the regions of Sfax and Kairouan in Tunisia. In terms of size, we handle instances with one depot and up to 45 transportation requests loaded on three or four vehicles. The product quantities being offered are either obtained from our partner, or estimated when these could not be made publicly available for confidentiality reasons. There is no restriction on the number of producers that can be visited on a route. The quantities of available extra, virgin and lampante oils represent 56%, 30% and 14% of the total available production, respectively. The routable

Algorithm 6: Branch-and-cut algorithm

```

1 At the root node of the search tree, generate and insert all valid inequalities
  into the program.
2  $z^* \leftarrow \infty$ .
3 Termination check:
4 if there are no more nodes to evaluate then
5 |   Stop with the incumbent and optimal solution of cost  $z^*$ .
6 else
7 |   Select one node from the branch-and-bound tree.
8 end
9 Subproblem solution: solve the LP relaxation of the node and let  $z$  be its cost.
10 if the current solution is feasible then
11 |   if  $z \geq z^*$  then
12 | |   Go to the termination check.
13 |   else
14 | |    $z^* \leftarrow z$ .
15 | |   Update the incumbent solution.
16 | |   Prune the nodes with a lower bound larger than or equal to  $z^*$ .
17 | |   Go to the termination check.
18 |   end
19 end
20 Cut generation:
21 if the solution of the current LP relaxation violates any cuts then
22 |   Identify connected components as in Padberg and Rinaldi \(1991\).
23 |   Determine whether the component containing the producer is weakly
  |   connected as in Gendreau et al. \(1997\).
24 |   Add violated subtour elimination constraints (6.8).
25 |   Go to the subproblem solution.
26 end
27 Branching: branch on one of the fractional variables.
28 Go to the termination check.

```

network dataset is constructed using real travel distances. Only driving distances provided by the regional office of Sfax take traffic congestion and the road state into account. The geographical location of the producers of this case study are depicted in Figures 6.2 and 6.3 and Table 6.1.

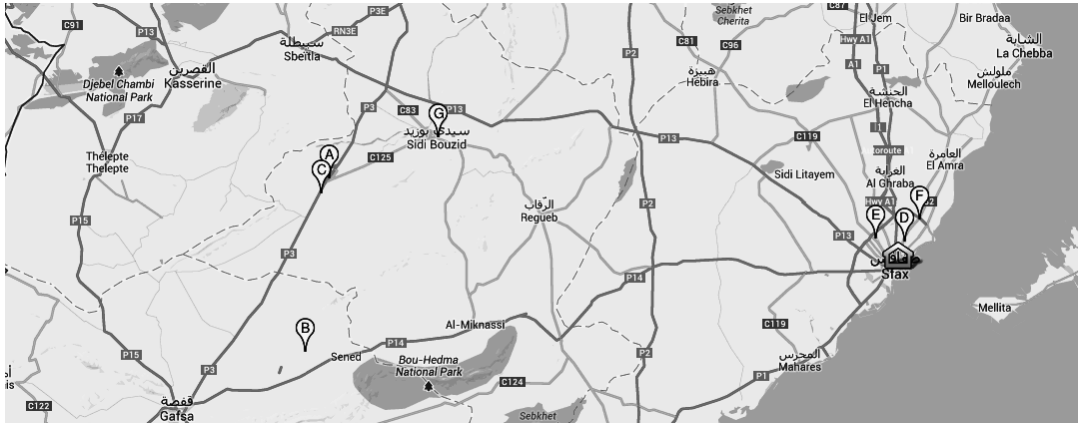


Figure 6.2: Geographical locations of the producers around the region of Sfax (Source: Google Maps, accessed March 2014)

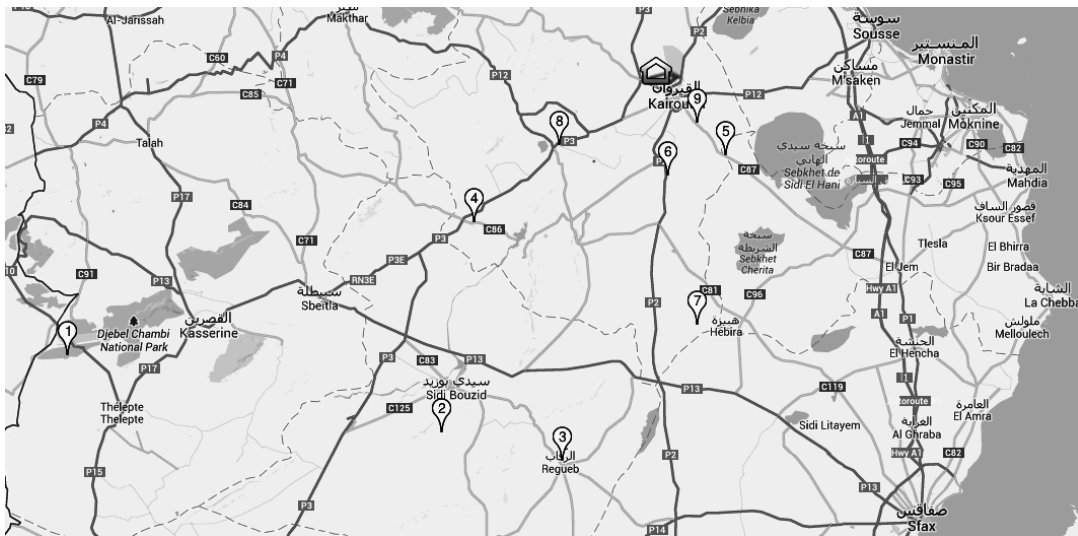


Figure 6.3: Geographical locations of the producers around the region of Kairouan (Source: Google Maps, accessed March 2014)

The industrial partners of the regions Sfax and Kairouan do not currently dispose of the same fleet composition. The first office controls a heterogeneous limited size fleet of single-compartment vehicles having a capacity of 10 tons, and double-compartment vehicles in which each compartment has a capacity of five tons. These two configurations will be denoted by type I and type II, respectively. The regional

Table 6.1: Producers locations in Figures 6.2 and 6.3

<i>Producer</i>	<i>Location</i>	<i>Producer</i>	<i>Location</i>
1	Fériana, Kasserine	A	Bir El Hfey, Sidi Bouzid
2	Jelma, Sidi Bouzid	B	Sened, Gafsa
3	Regueb, Sidi Bouzid	C	Sidi Ali Ben Aoun, Sidi Bouzid
4	Hajeb El Ayoun, Kairouan	D	North of Sfax
5	El Khazaziya, Kairouan	E	Road Gremda, Sfax
6	Boussari, Kairouan	F	Road Mahdia, Sfax
7	Cherarda, Kairouan	G	Sidi Bouzid
8	El Houareb, Kairouan		
9	South of Kairouan		

office of Kairouan, which presently uses single-compartment vehicles only, will revise its short-term procurement policy once we will have identified the cost savings achieved by using multi-compartment vehicles. Some drivers are outsourced, which enables the service providers to perform tours and to exploit the available fleet over six periods spread out during day-time and in the evening. The data exploitation results in five original instances, with an asterisk prepended to their names. To cover the different scenarios under a different fleet composition, we generate 10 instances identical to the original ones, but with a different set of vehicles. The names of the test instances highlight the factors that may affect the results. These factors include the first four letters of the regional office they refer to, the total number of requests, and the number of vehicles of type I and type II, respectively. Since the original instances may contain some products with a supply of more than five tons, some restrictions had to be made while generating the fleet composition. Each collection route requires at least one single-compartment vehicle for the Kairouan region, and two single-compartment vehicles for the Sfax region. Table 6.2 summarizes the characteristics of these instances. Regarding the objective function, we have used the following parameters after a tuning phase and discussions with our industrial partner:

- $\alpha_{ij} = \text{€}1$ per driving kilometer between vertices i and j ;
- $\beta^k = \text{€}15$ per vehicle k used per period t regardless of the vehicle type;
- $\delta = \text{€}10$ per cleaning activity.

6.4.2 Computational results

We have run the proposed algorithm over the data sets shown on Table 6.2. Table 6.3 summarizes the performance of the algorithm. We assess the performance

Table 6.2: Configurations of the real-world instances

<i>Instance</i>	<i># Producers</i>	<i># Requests</i>	<i>Fleet composition</i>	
			<i>Type I</i>	<i>Type II</i>
*Kair_27_3_0	6	27	3	0
Kair_27_1_2	6	27	1	2
Kair_27_2_1	6	27	2	1
*Kair_33_3_0	5	33	3	0
Kair_33_1_2	5	33	1	2
Kair_33_2_1	5	33	2	1
*Kair_34_3_0	6	34	3	0
Kair_34_1_2	6	34	1	2
Kair_34_2_1	6	34	2	1
*Kair_45_3_0	7	45	3	0
Kair_45_1_2	7	45	1	2
Kair_45_2_1	7	45	2	1
*Sfax_39_2_2	7	39	2	2
Sfax_39_3_1	7	39	3	1
Sfax_39_4_0	7	39	4	0

of the routing and assignment cuts by comparing the solutions obtained for different configurations. The implementation with constraints (6.1)–(6.16) is used as a reference point. Specifically, for each instance test we present the number of nodes explored in the branch-and-cut tree, the ratio of the lower bound at the root node between the configuration with cuts and the basic configuration (6.1)–(6.16), and finally the running time in seconds.

The algorithm proves optimality over all the 15 instances within very short computing times. For most of the instances of the region of Kairouan with three vehicles and up to seven producers, the algorithm takes less than one second to reach optimality. However, it requires more computational effort for the instances of Sfax, especially when the proposed cuts are disabled. This may be explained by the fact that increasing the number of vehicles generates much more symmetry.

A deeper analysis of the tested configurations shows that the introduction of valid inequalities significantly improves the performance of the algorithm. The average running time is reduced from 50.58 to 2.77 seconds and the average number of explored nodes goes down from 101,184 to 384 when the full model is implemented. The short computational time results from the fact that the model provides a high quality lower bound at the root node of the search tree. On average, the lower bound of the model with all cuts is almost equal to 1.5 times the initial lower bound value of the basic formulation (6.1)–(6.16). The full set of proposed cuts is essential to achieve the best algorithmic performance. In particular, the assignment cuts (6.20)–(6.22) have a more positive impact on these instances than the routing cuts (6.17)–(6.19). Without the assignment cuts, the model requires more iterations and

Table 6.3: Summary of computational results of the test instances with respect to several configurations

Model	Instance	# Nodes	Lower bound increase	Time(s)	Instance	# Nodes	Lower bound increase	Time(s)	
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	*Kair_27_3_0	1250	1	4.32	Kair_34_2_1	219	1	3.36
	(6.1)-(6.19)		98	1.28	1.19		114	1.25	2.45
	(6.20)-(6.22)		125	1.45	1.16		88	1.29	1.89
	(6.1)-(6.22)		1	1.54	0.30		32	1.40	1.46
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	Kair_27_1_2	3242	1	19.92	*Kair_45_3_0	801	1	7.11
	(6.1)-(6.19)		212	1.28	3.49		94	1.20	2.9
	(6.20)-(6.22)		237	1.13	2.51		17	1.18	1.32
	(6.1)-(6.22)		119	1.54	2.07		1	1.39	0.54
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	Kair_27_2_1	50	1	1.98	Kair_45_1_2	230	1	5.52
	(6.1)-(6.19)		14	1.25	0.73		256	1.21	5.61
	(6.20)-(6.22)		7	1.29	0.82		78	1.08	3.3
	(6.1)-(6.22)		1	1.39	0.26		93	1.26	2.89
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	*Kair_33_3_0	85	0.89	0	Kair_45_2_1	88	1	3.86
	(6.1)-(6.19)		22	1.57	0.68		39	1.19	2.18
	(6.20)-(6.22)		4	1.36	0.22		76	1.12	2.63
	(6.1)-(6.22)		1	1.80	0.16		5	1.31	1.20
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	Kair_33_1_2	34	1	2.2	*Sfax_39_2_2	18008	1	98.51
	(6.1)-(6.19)		112	1.40	0.64		6056	1.37	31.61
	(6.20)-(6.22)		74	1.13	1.73		5314	1.10	26.5
	(6.1)-(6.22)		28	1.43	0.57		1090	1.85	7.06
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	Kair_33_2_1	105	1	2.2	Sfax_39_3_1	19090	1	99.01
	(6.1)-(6.19)		76	1.51	0.96		9373	1.29	48.64
	(6.20)-(6.22)		28	1.25	1.31		732	1.04	7.34
	(6.1)-(6.22)		21	1.68	0.70		1801	1.77	10.66
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	*Kair_34_3_0	678	1	3.11	Sfax_39_4_0	108657	1	504.11
	(6.1)-(6.19)		54	1.28	1.68		16831	1.31	76.38
	(6.20)-(6.22)		108	1.43	1.75		2943	1.04	14.92
	(6.1)-(6.22)		6	1.53	0.60		2504	1.79	12.15
(6.1)-(6.16), (6.20)-(6.22)	(6.1)-(6.16)	Kair_34_1_2	216	1	2.56				
	(6.1)-(6.19)		82	1.28	1.24				
	(6.20)-(6.22)		72	1.13	1.47				
	(6.1)-(6.22)		60	1.29	0.87				

explores more nodes to identify the best solution. On average, the algorithm explores respectively 660 and 2,253 nodes when using separately assignment cuts and routing cuts. Under these two configurations, we have obtained almost the same improvement in the lower bound value (1.31 against 1.20) by imposing the assignment cuts.

When this study was undertaken, our industrial partner was trying to find ways to minimize the total associated logistic costs, i.e., the fixed and variable costs related to transport and cleaning related costs. In particular, it was paying close attention to the the fleet composition component and its impact on the overall costs. Using our methodology, we can provide alternative solutions to the managers by generating 10 instances while the overall capacity remains unchanged and the number of vehicles of types I and II varies. Table 6.4 summarizes these results. We note that substantial savings are achieved if both types of vehicles are used. We observe that a combination of vehicles yields better quality solutions for all Kairouan related instances, with improvements ranging from 1.9% to 21.7%, and averaging 11.7%. For the instances related to the office of Sfax, the best solutions are obtained with the current fleet composition, i.e., by using two vehicles of each type. One possible ex-

Table 6.4: Relative savings compared to existing situation

<i>Instance</i>	<i>Optimized solution</i>			<i>Manual solution</i>		
	<i># Km</i>	<i># Vehicles</i>	<i># Cleanings</i>	<i># Km</i>	<i># Vehicles</i>	<i># Cleanings</i>
*Kair_27_3_0	2361	17	1	2435	17	0
Kair_27_1_2	1848	15	0	-	-	-
Kair_27_2_1	1848	15	1	-	-	-
*Kair_33_3_0	2173	16	0	2339	16	0
Kair_33_1_2	1990	16	0	-	-	-
Kair_33_2_1	2104	16	0	-	-	-
*Kair_34_3_0	2737	17	1	2829	17	1
Kair_34_1_2	2322	16	1	-	-	-
Kair_34_2_1	2424	17	1	-	-	-
*Kair_45_3_0	4477	18	0	4721	18	0
Kair_45_1_2	4008	17	3	-	-	-
Kair_45_2_1	4388	17	0	-	-	-
*Sfax_39_2_2	2755	21	0	2866	21	0
Sfax_39_3_1	2787	21	0	-	-	-
Sfax_39_4_0	2811	21	0	-	-	-

planation is that compartments equipped with debit meters enable the collection of small quantities of the same product and the segregation of non-mixable products.

Finally, we compare our results to the solution currently applied by our industrial partner. Table 6.4 indicates that the proposed method provides an improvement over the current solution designed manually by the dispatcher since it reduces the overall costs for all the data sets. Our solutions minimize the distance traveled with an improvement of up to 7% and optimize the products assignment process to avoid unnecessary cleaning costs. However, the same number of vehicles is needed to cover all the producers' locations. We have run further test with an hierarchical objective function, which first minimizes the required number of vehicles, and then the routing and cleaning costs. These tests reveal that the current number of vehicles used by our industrial partner is in fact optimal. For the managers, optimizing both the collection routes and the fleet composition is important but difficult to achieve through manual methods. They were asked to evaluate the solutions obtained and declared themselves very satisfied with the results.

6.5 Conclusions

We tackled a real-world and rich multi-compartment vehicle routing problem arising in the olive oil collection industry. This practical application concerns the pick-up of one or more commodities from a set of geographically scattered producers in the center of Tunisia. We presented a mathematical model including known and new valid inequalities, as well as a branch-and-cut algorithm for its solution. We tested

the algorithm on several sets of realistic instances. Our experimental results show the effectiveness of the proposed algorithm. Extensive tests enabled us to generate solutions that can help support decision making at the tactical level, i.e., purchasing new vehicles, and at the operational level, i.e., redesigning vehicles loading and collection routes.

Conclusions

This thesis addressed the emergent research area reflecting the challenges of real-life VRPs, known as Rich VRPs. We provided first a comprehensive and relevant survey of the RVRP literature as well as a taxonomy. Then we proposed an elaborate definition of RVRPs. The proposed taxonomy served as an illustration of the relevance of flexible solution methods to incorporate real-life requirements arising in various industries. Few approaches explicitly address the solution of RVRPs in general. This motivates the development of a unified metaheuristic framework to solve a RVRP and some of its variants.

Specifically, we solved the MDMCMC_m-VRPTW. First, we introduced a three-index vehicle flow formulation for this problem and we applied the DW decomposition on this formulation. This decomposition gives rise to a more tractable integer problem. We proposed a unified column generation heuristic cooperating with a VNS to solve the pricing problem. The VNS combines several removal and insertion routing heuristics as well as computationally efficient constraint checking procedures in order to solve the MDMCMC_m-VRPTW with a single vehicle. A special attention was paid to the loading problem resulting from the use of multi-compartment vehicles. Two loading neighborhoods based on the solution of mathematical programs are proposed to intensify the search. They interact with the heuristic routing neighborhoods in the line of matheuristics. On a set of 191 instances of less complex routing problems, the unified matheuristic turns to be competitive. A sensitivity analysis, performed on more complex generated instances reveals the importance of some algorithmic features and of loading neighborhoods for reaching high quality solutions. The VNS based matheuristic is embedded in a column generation heuristic to solve the MDMCMC_m-VRPTW. We proposed a set of techniques to improve the solution quality as well as an exact post-processing method to optimize the assignment of customers to vehicle routes. While developing the unified matheuristic, we focused on the definition of a generic data structure which offers flexibility to extend or to specialize the proposed matheuristic easily. Last, we tackled the real case-study that motivated this work. We introduced, modeled and solved to optimality a rich

multi-product, multi-period and multi-compartment vehicle routing problem with a required compartment cleaning activity. This real-life application arises in the olive oil collection process in Tunisia. We proposed an exact branch-and-cut algorithm to solve the problem. We evaluated the performance of the algorithm on real data sets under different transportation scenarios to demonstrate to our industrial partner the advantages of using multi-compartment vehicles.

For some benchmarks, the proposed matheuristic outperforms state-of-the-art methods with respect to both solution quality and computational time. On other problems, it is dominated by specialized algorithms but still produce good results. This explains the average behavior of the proposed method on some benchmarks. This drawback is offset by the increase of flexibility when the proposed matheuristic is extended/specilized to deal with real-world requirements in solving VRPs.

Many avenues could be explored in future research. A number of other VRP extensions are relevant in the context of the MC-VRPs. We believe that the most interesting variant is probably a simple version of the MDMCMCm-VRPTW without considering the temporal restrictions. Modeling this problem would require a smaller number of variables, parameters and constraints, probably making it significantly more tractable. This problem can be considered as a special case of the SDVRP with multi-compartment vehicles where some splitting scenarios are fixed a priori. Simplifying the problem could be a promising incentive to enrich the loading neighborhood search methods, in line with works on VRPs with loading constraints. The loading neighborhoods available in this context are clearly very numerous. For example, a possible neighborhood would consist in isolating a certain proportion of the orders that are considered the most incompatible, or to rearrange the loading plan to maximize the number of compartment filled to 100%.

There are many potential starting points for further improvement of the proposed column generation heuristic. We may improve the overall performance of the framework by designing innovative strategies to add variables and by defining better initialization procedures. Larger instances remain difficult to solve. This clearly calls for increasing the number of generated columns and/or avoiding degeneracy caused by the large dual space. The aspect of preventing dual variables from taking extreme values, is theoretically interesting. Satbilization methods could also be investigated further to enhance the column generation convergence.

Through this thesis, we attempted to answer the primary research question: "*unified matheuristic: birth or burial?*". Evidently, this thesis represents a step in this research avenue and works on the subject can be pursued by considering more VRP variants. In addition, I started this trip with the aim of providing additional guide-

lines for hybrid solution approaches for RVRPs. Combining matheuristics, which in some way exploit the mathematical model of a problem, is very promising and may produce effective solution approaches. I look forward to discover these future researches development, which I hope not only to observe but in some way to participate in, too.

References

- Y. Adulyasak, J. F. Cordeau, and R. Jans. Optimization-based adaptive large neighborhood search for the production routing problem. *Transportation Science*, 48: 20–45, 2012. [17](#)
- Y. Adulyasak, J. F. Cordeau, and R. Jans. The production routing problem: A review of formulations and solution algorithms. Technical report, 2013-49, CIRRELT, Montreal, QC, Canada, 2013. [17](#)
- F. Alonso, M. J. Alvarez, and J. E. Beasley. A tabu search algorithm for the periodic vehicle routing problem with multiple vehicle trips and accessibility restrictions. *Journal of the Operational Research Society*, 59:963–976, 2008. [20](#)
- H. Andersson, A. Hoff, M. Christiansen, G. Hasle, and A. Løkketangen. Industrial aspects and literature survey: Combined inventory management and routing. *Computers & Operations Research*, 37(9):1515–1536, 2010. [17](#)
- C. Archetti and M. G. Speranza. An overview on the split delivery vehicle routing problem. In *Operations Research Proceedings 2006*, pages 123–127. Springer, 2007. [19](#)
- C. Archetti and M. G. Speranza. Vehicle routing problems with split deliveries. *International transactions in operational research*, 19(1-2):3–22, 2012. [3](#), [19](#)
- C. Archetti, D. Feillet, A. Hertz, and M. G. Speranza. The capacitated team orienteering and profitable tour problems. *Journal of the Operational Research Society*, 60(2):831–842, 2009. [60](#)
- C. Archetti, M. G. Speranza, and D. Vigo. Vehicle routing problems. Technical report, Department of Economics and Management University of Brescia, Italy, 2013. [57](#)
- V. A. Armentano, A. L. Shiguemoto, and A. Løkketangen. Tabu search with path relinking for an integrated production–distribution problem. *Computers & Operations Research*, 38(8):1199–1209, 2011. [16](#), [17](#), [33](#)

- A. Atahran. *Étude et résolution d'un problème de Transport à la Demande multi-critère*. PhD thesis, Ph. D. thesis, Université François Rabelais de Tours, France, 2012. [75](#)
- N. Azi, M. Gendreau, and J. Y. Potvin. An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. *European Journal of Operational Research*, 202(3):756–763, 2010. [20](#)
- N. Azi, M. Gendreau, and J. Y. Potvin. A dynamic vehicle routing problem with multiple delivery routes. *Annals of Operations Research*, 199(1):103–112, 2012. [20](#)
- A. Balakrishnan, J. E. Ward, and R. T. Wong. Integrated facility location and vehicle routing models: Recent work and future prospects. *American Journal of Mathematical and Management Sciences*, 7(1-2):35–61, 1987. [18](#)
- E. Balas. *The prize collecting traveling salesman problem and its applications*. In: Gutin G, Punnen AP (eds) *The traveling salesman problem and its variations*, pages 663–695. Kluwer Academic Publisher, 2007. [56](#)
- R. Baldacci, M. Battarra, and D. Vigo. Routing a heterogeneous fleet of vehicles. In *The vehicle routing problem: latest advances and new challenges*, pages 3–27. Springer, 2008. [20](#), [23](#)
- R. Baldacci, E. Bartolini, A. Mingozzi, and A. Valletta. An exact algorithm for the period routing problem. *Operations research*, 59(1):228–241, 2011. [20](#)
- J. F. Bard and N. Nananukul. The integrated production-inventory-distribution-routing problem. *Journal of Scheduling*, 12(3):257–280, 2009. [17](#), [33](#)
- A. Beaudry, G. Laporte, T. Melo, and S. Nickel. Dynamic transportation of patients in hospitals. *Operations Research Spectrum*, 32:77–107, 2010. [22](#), [23](#), [32](#)
- W. J. Bell, L. M. Dalberto, M. L. Fisher, A. J. Greenfield, R. Jaikumar, P. Kedia, R. G. Mack, and P. J. Prutzman. Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces*, 13(6):4–23, 1983. [17](#)
- E. J. Beltrami and L. D. Bodin. Networks and vehicle routing for municipal waste collection. *Networks*, 4(1):65–94, 1974. [19](#)

- G. Berbeglia, J. F. Cordeau, I. Gribkovskaia, and G. Laporte. Static pickup and delivery problems: a classification scheme and survey. *Trabajos de Investigación Operativa*, 15:1–31, 2007. [19](#)
- G. Berbeglia, J. F. Cordeau, and G. Laporte. Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8–15, 2010. [16](#)
- O. Berman, P. Jaillet, and D. Simchi-Levi. Location-routing problems with uncertainty. In *Facility Location: A Survey of Applications and Methods*, pages 427–452. In Drezner, Z. (Eds.), New York: Springer, 1995. [18](#)
- L. Bertazzi, M. Savelsbergh, and M. G. Speranza. Inventory routing. In *The vehicle routing problem: latest advances and new challenges*, pages 49–72. Springer, 2008. [17](#)
- L. Bodin. A taxonomic structure for vehicle routing and scheduling problems. *Computers & Urban Society*, 1(1):11–29, 1975. [12](#)
- L. Bodin and B. Golden. Classification in vehicle routing and scheduling. *Networks*, 11:97–108, 1981. [12](#)
- M. C. Bolduc, J. Renaud, and B. Montreuil. Synchronized routing of seasonal products through a production distribution network. *Central European Journal of Operations Research*, 14:209–228, 2006. [32](#)
- M. Boudia and C. Prins. A memetic algorithm with dynamic population management for an integrated production–distribution problem. *European Journal of Operational Research*, 195(3):703–715, 2009. [17](#), [33](#)
- M. Boudia, M. A. O. Louly, and C. Prins. A reactive grasp and path relinking for a combined production-distribution problem. *Computers & Operations Research*, 34(11):3402–3419, 2007. [17](#), [33](#)
- N. Boysen. Truck scheduling at zero-inventory cross docking terminals. *Computers & Operations Research*, 37:32–41, 2010. [25](#)
- N. Boysen and M. Flidner. Cross dock scheduling: classification, literature review and research agenda. *Omega*, 38:413–422, 2009. [33](#)
- J. Brandão. A tabu search algorithm for the open vehicle routing problem. *European Journal of Operational Research*, 157(3):552–564, 2004. [25](#)

- O. Bräysy, W. Dullaert, G. Hasle, D. Mester, and M. Gendreau. An effective multirestart deterministic annealing metaheuristic for the fleet size and mix vehicle-routing problem with time windows. *Transportation Science*, 42(3): 371–386, 2008. [20](#)
- G. G. Brown and G. W. Graves. Real-time dispatch of petroleum tank trucks. *Management Science*, 27(1):19–32, 1981. [21](#)
- G. G. Brown, C. J. Ellis, G. W. Graves, and D. Ronen. Real-time, wide area dispatch of mobile tank trucks. *Interfaces*, 17:107–120, 1987. [38](#)
- O. Bräysy and M. Gendreau. Vehicle routing problem with time windows, part i: Route construction and local search algorithms. *Transportation Science*, 39(1): 104–118, 2005a. [22](#)
- O. Bräysy and M. Gendreau. Vehicle routing problem with time windows, part ii: Metaheuristics. *Transportation Science*, 39(1):119–139, 2005b. [22](#)
- A. Campbell, L. Clarke, A. Kleywegt, and M. Savelsbergh. The inventory routing problem. In *Fleet management and logistics*, pages 95–113. Springer, 1998. [17](#)
- A. M. Campbell and J. H. Wilson. Forty years of periodic vehicle routing. *Networks*, 63(1):2–15, 2014. [3](#), [20](#)
- V. Campos, R. Marti, J. Sánchez-Oro, and A. Duarte. Grasp with path relinking for the orienteering problem. *Journal of the Operational Research Society*, 2013. ISSN 0160-5682. URL <http://dx.doi.org/10.1057/jors.2013.156>. [90](#), [91](#)
- M. Caramia and F. Guerriero. A milk collection problem with incompatibility constraints. *Interfaces*, 40(2):130–143, 2010. [25](#)
- F. Carrabs, R. Cerulli, and J. F. Cordeau. An additive branch-and-bound algorithm for the pickup and delivery traveling salesman problem with lifo or fifo loading. *INFOR: Information Systems and Operational Research*, 45(4):223–238, 2007. [21](#)
- D. Cattaruzza, N. Absi, D. Feillet, and T. Vidal. A memetic algorithm for the multi trip vehicle routing problem. *European Journal of Operational Research*, 236(3): 833–848, 2014. [20](#)
- A. Ceselli, G. Righini, and M. Salani. A column generation algorithm for a vehicle routing problem with economies of scale and additional constraints. *Transportation Science*, 43(1):56–69, 2009. [23](#), [25](#), [31](#)

- E. D. Chajakis and M. Guignard. Scheduling deliveries in vehicles with multiple compartments. *Journal of Global Optimization*, 26(1):43–78, 2003. [21](#)
- P. Chandra and M. L. Fisher. Coordination of production and distribution planning. *European Journal of Operational Research*, 72(3):503–517, 1994. [17](#)
- I. M. Chao, B. L. Golden, and E. A. Wasil. A fast and effective heuristic for the orienteering problem. *European Journal of Operational Research*, 88:475–489, 1996. [xiii](#), [90](#), [91](#), [173](#)
- Z. L. Chen. Integrated production and distribution operations. In *Handbook of quantitative supply chain analysis*, pages 711–745. Springer, 2004. [17](#)
- M. Christiansen, K. Fagerholt, T. Flatberg, Ø. Haugen, O. Kloster, and E. H. Lund. Maritime inventory routing with multiple products: A case study from the cement industry. *European Journal of Operational Research*, 208(1):86–94, 2011. [38](#)
- N. Christofides, A. Mingozzi, and P. Toth. The vehicle routing problem. In *Combinatorial Optimization*. John Wiley, Chichester, 1979. [111](#)
- L. C. Coelho. Flexibility and consistency in inventory-routing. *4OR*, 11(3):297–298, 2013. [17](#)
- L. C. Coelho and G. Laporte. Classification, models and exact algorithms for multi-compartment delivery problems. Technical Report CIRRELT-2013-39, Montreal, Canada, 2013a. [8](#), [38](#), [39](#), [123](#)
- L. C. Coelho and G. Laporte. A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem. *International Journal of Production Research*, 51(23-24):7156–7169, 2013b. [17](#)
- L. C. Coelho and G. Laporte. The exact solution of several classes of inventory-routing problems. *Computers & Operations Research*, 40(2):558–565, 2013c. [17](#)
- L. C. Coelho and G. Laporte. Improved solutions for inventory-routing problems through valid inequalities and input ordering. *International Journal of Production Economics*, forthcoming, 2014. doi: 10.1016/j.ijpe.2013.11.019. [17](#), [124](#)
- L. C. Coelho, J. F. Cordeau, and G. Laporte. The inventory-routing problem with transshipment. *Computers & Operations Research*, 39(11):2537–2548, 2012a. [17](#)

- L. C. Coelho, J. F. Cordeau, and G. Laporte. Consistency in multi-vehicle inventory-routing. *Transportation Research Part C: Emerging Technologies*, 24:270–287, 2012b. [17](#)
- L. C. Coelho, J. F. Cordeau, and G. Laporte. Thirty years of inventory routing. *Transportation Science*, 48(1):1–19, 2013. [17](#)
- J. F. Cordeau and G. Laporte. The dial-a-ride problem: models and algorithms. *Annals of Operations Research*, 153:29–46, 2007. [16](#), [19](#)
- J. F. Cordeau, M. Gendreau, and G. Laporte. A tabu search heuristic for periodic and multidepot problems. *Networks*, 30:105–119, 1997. [20](#), [91](#), [173](#)
- J. F. Cordeau, G. Desaulniers, J. Desrosiers, M. M. Solomon, and F. Soumis. *VRP with time windows*. In: *Toth P, Vigo D (eds) The Vehicle Routing Problem*, pages 157–193. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, 2002a. [39](#), [47](#), [48](#), [56](#)
- J. F. Cordeau, M. Gendreau, G. Laporte, J. Y. Potvin, and F. Semet. A guide to vehicle routing heuristics. *Journal of the Operational Research Society*, 53(5): 512–522, 2002b. [4](#), [62](#)
- J. F. Cordeau, G. Laporte, M. W. F. Savelsbergh, and D. Vigo. *Vehicle routing*. In: *Barnhart C, Laporte G, (eds) Handbooks in Operations Research and Management Science, Transportation*, pages 367–428. North Holland, 1 edition, Amsterdam: Elsevier, 2007. [12](#), [17](#)
- F. Cornillier, F. F. Boctor, G. Laporte, and J. Renaud. An exact algorithm for the petrol station replenishment problem. *Journal of the Operational Research Society*, 59(5):607–615, 2008a. [21](#), [38](#), [39](#)
- F. Cornillier, F. F. Boctor, G. Laporte, and J. Renaud. A heuristic for the multi-period petrol station replenishment problem. *European Journal of Operational Research*, 191(2):295–305, 2008b. [21](#), [24](#), [38](#), [39](#)
- F. Cornillier, G. Laporte, F. F. Boctor, and J. Renaud. The petrol station replenishment problem with time windows. *Computers & Operations Research*, 36(3): 919–935, 2009. [21](#), [24](#), [38](#), [39](#), [80](#)
- F. Cornillier, F. F. Boctor, and J. Renaud. Heuristics for the multi-depot petrol station replenishment problem with time windows. *European Journal of Operational Research*, 220(2):361–369, 2012. [21](#), [38](#), [39](#)

- T. G. Crainic, G. C. Crisan, M. Gendreau, N. Lahrichi, and W. Rei. Multi-thread integrative cooperative optimization for rich combinatorial problems. In *Parallel & Distributed Processing, 2009. IPDPS 2009. IEEE International Symposium on*, pages 1–8. IEEE, 2009. [32](#)
- L. Cruz Reyes, J. J. González Barbosa, D. Romero Vargas, H. J. Fraire Huacuja, N. Rangel Valdez, J. A. Herrera Ortiz, A. B. Arrañaga Cruz, and J. F. Delgado Orta. A distributed methaheuristic for solving real world scheduling routing loading problem. *Lecture Notes in Computer Science*, 4742:68–77, 2007. [21](#), [25](#), [31](#)
- L. Cruz Reyes, J. F. Delgado Orta, J. J. González Barbosa, J. T. Jimenez, H. J. Fraire Huacuja, and A. B. Arrañaga Cruz. An ant colony system algorithm to solve routing problems applied to the delivery of bottled products. In *Foundations of Intelligent Systems*, pages 329–338. Springer, 2008. [31](#)
- G. Dantzig, R. Fulkerson, and S. Johnson. Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America*, 2(4):393–410, 1954. [17](#)
- G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, 6(1):80–91, 1959. [12](#)
- G. B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations research*, 8(1):101–111, 1960. [49](#), [54](#)
- M. Dell’Amico, F. Maffioli, and P. Varbrand. On prize-collecting tours and the asymmetric travelling salesman problem. *International Transactions in Operational Research*, 2(3):297–308, 1995. [56](#)
- E. Demir, T. Bektas, and G. Laporte. An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, 223(2):346–359, 2012. [70](#)
- E. Demir, T. Bektas, and G. Laporte. A review of recent research on green road freight transportation. *European Journal of Operational Research*, 2014. doi: <http://dx.doi.org/10.1016/j.ejor.2013.12.033>. [24](#)
- U. Derigs and T. Döhmer. Indirect search for the vehicle routing problem with pickup and delivery and time windows. *OR Spectrum*, 30(1):149–165, 2008. [33](#)

- U. Derigs and U. Vogel. Experience with a framework for developing heuristics for solving rich vehicle routing problems. *Journal of Heuristics*, 20(1):75–106, 2014. [4](#), [13](#), [35](#), [61](#)
- U. Derigs, J. Gottlieb, J. Kalkoff, M. Piesche, F. Rothlauf, and U. Vogel. Vehicle routing with compartments: applications, modelling and heuristics. *OR Spectrum*, 33(4):885–914, 2011a. [21](#), [38](#), [60](#), [123](#)
- U. Derigs, R. Kurowsky, and U. Vogel. Solving a real-world vehicle routing problem with multiple use of tractors and trailers and EU-regulations for drivers arising in air cargo road feeder services. *European Journal of Operational Research*, 213(1):309–319, 2011b. [22](#)
- U. Derigs, M. Pullmann, and U. Vogel. Truck and trailer routing problems, heuristics and computational experience. *Computers & Operations Research*, 40(2):536–546, 2013. [25](#)
- G. Desaulniers, J. Desrosiers, A. Erdmann, M. M. Solomon, and F. Soumis. *VRP with pickup and delivery*. In: *Toth P, Vigo D (eds) The vehicle routing problem*, pages 225–242. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, 2002. [19](#)
- G. Desaulniers, J. Desrosiers, and M. M Solomon. *Column generation*, volume 5. Springer, 2005. [105](#), [109](#)
- M. Desrochers, J. K. Lenstra, and M. W. P. Savelsbergh. A classification scheme for vehicle routing and scheduling problems. *European Journal of Operational Research*, 46:322–332, 1990. [12](#), [23](#)
- J. Desrosiers and M. E. Lübbecke. A primer in column generation. In *Column generation*, pages 1–32. Springer, 2005. [49](#)
- J. Desrosiers and M. E. Lübbecke. Branch-price-and-cut algorithms. *Wiley encyclopedia of operations research and management science*, 2011. [51](#)
- K. F. Doerner, G. Fuellerer, R. F. Hartl, M. Gronalt, and M. Iori. Metaheuristics for the vehicle routing problem with loading constraints. *Networks*, 49(4):294–307, 2007. [21](#)
- K.F. Doerner and V. Schmid. Survey: matheuristics for rich vehicle routing problems. In *Hybrid Metaheuristics*, pages 206–221. Springer, 2010. [12](#)

- A. Dohn, E. Kolind, and J. Clausen. The manpower allocation problem with time windows and job-teaming constraints: A branch-and-price approach. *Computers & Operations Research*, 36(4):1145–1157, 2009. [23](#)
- A. Dohn, M. S. Rasmussen, and J. Larsen. The vehicle routing problem with time windows and temporal dependencies. *Networks*, 58(4):273–289, 2011. [23](#)
- J. J. Dongarra. Performance of various computers using standard linear equations software. Technical report, Electrical Engineering and Computer Science Department, University of Tennessee and Computer Science and Mathematics Division, University of Manchester, 2013. [91](#), [93](#)
- M. Drexl. Rich vehicle routing in theory and practice. *Logistics Research*, 5(1-2):47–63, 2012a. [12](#), [13](#)
- M. Drexl. Synchronization in vehicle routing—a survey of vrps with multiple synchronization constraints. *Transportation Science*, 46(3):297–316, 2012b. [23](#)
- M. Dror and P. Trudeau. Savings by split delivery routing. *Transportation Science*, 23(2):141–145, 1989. [19](#)
- B. Eksioglu, Vural A. V., and A. Reisman. The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, 57:1472–1483, 2009. [12](#), [13](#)
- A. El Fallahi, C. Prins, and R. Wolfler Calvo. A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem. *Computers & Operations Research*, 35:1725–1741, 2008. [38](#), [39](#), [60](#)
- S. Erdoğan and E. Miller-Hooks. A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):100–114, 2012. [24](#)
- K. Fagerholt and M. Christiansen. A combined ship scheduling and allocation problem. *Journal of the Operational Research Society*, 51(7):834–842, 2000. [38](#)
- D. Feillet, P. Dejax, and M. Gendreau. Traveling salesman problems with profits. *Transportation Science*, 39(2):188–205, 2005. [17](#), [56](#), [57](#)
- M. Fischetti, J. J. Salazar-Gonzalez, and P. Toth. *The generalized traveling salesman and orienteering problems*. In: Gutin G, Punnen AP (eds) *The traveling salesman problem and its variations*, pages 609–662. Kluwer Academic Publisher, 2007. [56](#)

- M. L. Fisher, A. Greenfield, R. Jaikumar, and P. Kedia. Real-time scheduling of a bulk delivery fleet: Practical application of lagrangean relaxation. Technical report, The Wharton School, University of Pennsylvania, 1982. [17](#)
- T. Flatberg, G. Hasle, O. Kloster, E. J. Nilssen, and A. Rüise. Dynamic and stochastic aspects in vehicle routing—a literature survey. *STF90 A05413 SINTEF Applied Mathematics*, page 16, 2005. [16](#)
- B. Fleischmann. The vehicle routing problem with multiple use of vehicles. Technical report, Fachbereich Wirtschaftswissenschaften Universität Hamburg, 1990. [20](#)
- P. Francis and K. Smilowitz. Modeling techniques for periodic vehicle routing problems. *Transportation Research Part B: Methodological*, 40(10):872–884, 2006. [20](#)
- P. M. Francis, K. R. Smilowitz, and M. Tzur. The period vehicle routing problem and its extensions. In *The vehicle routing problem: latest advances and new challenges*, pages 73–102. Springer, 2008. [20](#)
- A. Frangioni and B. Gendron. A stabilized structured dantzig–wolfe decomposition method. *Mathematical Programming*, 140(1):45–76, 2013. [51](#)
- R. Freling, D. Huisman, and A. Wagelmans. Models and algorithms for integration of vehicle and crew scheduling. *Journal of Scheduling*, 6(1):63–85, 2003. [18](#)
- Z. Fu, R. Eglese, and L. Y. O. Li. A new tabu search heuristic for the open vehicle routing problem. *Journal of the Operational Research Society*, 56(3):267–274, 2005. [25](#)
- G. Fuellerer, K. F. Doerner, R. F. Hartl, and M. Iori. Ant colony optimization for the two-dimensional loading vehicle routing problem. *Computers & Operations Research*, 36(3):655–673, 2009. [21](#)
- G. Fuellerer, K. F. Doerner, R. F. Hartl, and M. Iori. Metaheuristics for vehicle routing problems with three-dimensional loading constraints. *European Journal of Operational Research*, 201(3):751–759, 2010. [21](#)
- A. Fügenschuh. The vehicle routing problem with coupled time windows. *Central European Journal of Operations Research*, 14(2):157–176, 2006. [23](#), [32](#)
- F. Fumero and C. Vercellis. Synchronized development of production, inventory, and distribution schedules. *Transportation science*, 33(3):330–340, 1999. [17](#)

- A. Garcia, P. Vansteenwegen, W. Souffriau, O. Arbelaitz, and M. Linaza. Solving multi constrained team orienteering problems to generate tourist routes. Technical report, Centre for Industrial Management / Traffic & Infrastructure, 2010. [60](#), [75](#)
- M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990. ISBN 0716710455. [39](#)
- H. Gehring and J. Homberger. A parallel hybrid evolutionary metaheuristic for the vehicle routing problem with time windows. In *Proceedings of the Evolutionary Algorithms in Engineering and Computer Science*, pages 57–64, Jyväskylä, Finland, 1999. University of Jyväskylä. [96](#)
- M. Gendreau and C. D. Tarantilis. Solving large-scale vehicle routing problems with time windows: The state-of-the-art. Technical report, 2010-04, CIRRELT, Montreal, QC, Canada, 2010. [22](#)
- M. Gendreau, A. Hertz, G. Laporte, and M. Stan. A generalized insertion heuristic for the traveling salesman problem with time windows. *Operations Research*, 46(3):330–335, 1995. [73](#)
- M. Gendreau, G. Laporte, and R. Séguin. Stochastic vehicle routing. *European Journal of Operational Research*, 88(1):3–12, 1996. [16](#)
- M. Gendreau, G. Laporte, and F. Semet. The covering tour problem. *Operations Research*, 45(4):568–576, 1997. [126](#)
- M. Gendreau, G. Laporte, and F. Semet. A branch and cut algorithm for the undirected selective travelling salesman problem. *Networks*, 32(4):263–273, 1998a. [56](#)
- M. Gendreau, G. Laporte, and F. Semet. A tabu search heuristic for the undirected selective travelling salesman problem. *European Journal of Operational Research*, 106(2-3):539–545, 1998b. [62](#)
- M. Gendreau, G. Laporte, and F. Semet. A tabu search heuristic for the undirected selective travelling salesman problem. *European Journal of Operational Research*, 106(2-3):539–545, 1998c. [123](#)
- B. Gendron and T. G. Crainic. Relaxations for multicommodity capacitated network design problems. Technical Report CRT-965, Centre de Recherche sur les Transports, Montreal, 1994. [124](#)

- J. C. Gerdessen. Vehicle routing problem with trailers. *European Journal of Operational Research*, 93(1):135–147, 1996. [25](#)
- P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting-stock problem. *Operations research*, 9(6):849–859, 1961. [54](#)
- F. Glover, G. Jones, D. Karney, D. Klingman, and J. Mote. An integrated production, distribution, and inventory planning system. *Interfaces*, 9(5):21–35, 1979. [17](#)
- A. Goel. Vehicle scheduling and routing with drivers’ working hours. *Transportation Science*, 43(1):17–26, 2009. [18](#), [22](#)
- A. Goel. A column generation heuristic for the general vehicle routing problem. In *Learning and Intelligent Optimization*, pages 1–9. Springer, 2010. [23](#)
- A. Goel and V. Gruhn. A general vehicle routing problem. *European Journal of Operational Research*, 191(3):650–660, 2008. [23](#)
- A. Goel and T. Vidal. Hours of service regulations in road freight transport: an optimization-based international assessment. *Transportation Science*, 2013. doi: 10. [22](#)
- M. Goetschalckx and C. Jacobs-Blecha. The vehicle routing problem with backhauls. *European Journal of Operational Research*, 42(1):39–51, 1989. [19](#)
- B. Golden, L. Levy, and R. Vohra. The orienteering problem. *Naval Research Logistics*, 34(3):307–318, 1987. [62](#)
- B. Golden, S. Raghavan, and E. Wasil. *The vehicle routing problem: Latest advances and new challenges*. Springer, New York, 2008. [12](#)
- M. C. Golumbic. *Algorithmic graph theory and perfect graphs*. Annals of Discrete Mathematics, Rutgers University, Piscataway, NJ, USA, 2004. [77](#)
- D. Gulczynski, B. Golden, and E. Wasil. The period vehicle routing problem: New heuristics and real-world variants. *Transportation Research Part E: Logistics and Transportation Review*, 47(5):648–668, 2011. [20](#)
- K. Haase, G. Desaulniers, and J. Desrosiers. Simultaneous vehicle and crew scheduling in urban mass transit systems. *Transportation Science*, 3(35):286–303, 2001. [18](#)

- E. Halvorsen-Weare and K. Fagerholt. Routing and scheduling in a liquefied natural gas shipping problem with inventory and berth constraints. *Annals of Operations Research*, 203(1):167–186, 2013. [38](#)
- P. Hansen and N. Mladenovic. *Variable neighborhood search*. In: *Glover F, Kochenberger GA (eds) Handbook of Metaheuristics*, pages 145–184. Kluwer Academic Publisher, 2003. [67](#)
- R. F. Hartl, G. Hasle, and G. K. Janssens. Special issue on rich vehicle routing problems. *Central European Journal of Operations Research*, 14(2):103–104, 2006. [12](#)
- G. Hasle and O. Kloster. *Industrial vehicle routing problems*. In Hasle, G., Lie, K. A., Quak, E. (Eds.), *Geometric modelling, numerical simulation and optimization: Applied mathematics, SINTEF*. Berlin, Germany: Springer, 2007. [16](#), [22](#)
- G. Hasle, A. Løkketangen, and S. Martello. Rich models in discrete optimization: Formulation and resolution. *European Journal of Operational Research*, 175(3):1752–1753, 2006. [12](#)
- V. Hemmelmayr, K. F. Doerner, R. F. Hartl, and S. Rath. A heuristic solution method for node routing based solid waste collection problems. *Journal of Heuristics*, 19(2):129–156, 2013. [18](#)
- V. C. Hemmelmayr, K. F. Doerner, and R. F. Hartl. A variable neighborhood search heuristic for periodic routing problems. *European Journal of Operational Research*, 195(3):791–802, 2009. [20](#)
- A. Hoff and A. Løkketangen. A tabu search approach for milk collection in western norway using trucks and trailers. *Proc. Sixth Triennial Sympos. Transportation Anal.(TRISTAN VI), Phuket Island, Thailand*, 2007. [25](#)
- A. Hoff, H. Andersson, M. Christiansen, G. Hasle, and A. Løkketangen. Industrial aspects and literature survey: Fleet composition and routing. *Computers & Operations Research*, 37(12):2041–2061, 2010. [20](#)
- L. M. Hvattum, A. Løkketangen, and G. Laporte. Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic. *Transportation Science*, 40(4):421–438, 2006. [32](#)
- L. M. Hvattum, K. Fagerholt, and V. A. Armentano. Tank allocation problems in maritime bulk shipping. *Computers & Operations Research*, 36(11):3051–3060, 2009. [118](#)

- Y. Ileri, M. Bazaraa, T. Gifford, G. Nemhauser, J. Sokol, and E. Wikum. An optimization approach for planning daily drayage operations. *Central European Journal of Operations Research*, 14(2):141–156, 2006. 24, 33
- M. Iori and S. Martello. Routing problems with loading constraints. *Top*, 18(1):4–27, 2010. 21
- Stefan Irnich. A unified modeling and solution framework for vehicle routing and local search-based metaheuristics. *INFORMS Journal on Computing*, 20(2):270–287, 2008. 62
- S. D. Jena, J. F. Cordeau, and B. Gendron. Modeling and solving a logging camp location problem. *Annals of Operations Research*, 2012. 124
- S. D. Jena, J. F. Cordeau, and B. Gendron. Dynamic facility location with generalized modular capacities. Technical report, CIRRELT-2013-18, Montreal, 2013. 124
- M. K. Jepsen. *Branch-and-cut and branch-and-cut-and-price algorithms for solving vehicle routing problems*. Dissertation, Technical University of Denmark, 2011. URL http://orbit.dtu.dk/fedora/objects/orbit:89357/datastreams/file_6317942/content. 60
- N. Jozefowicz, F. Semet, and E. G. Talbi. Multi-objective vehicle routing problems. *European Journal of Operational Research*, 189(2):293–309, 2008. 26
- B. Kallehauge. Formulations and exact algorithms for the vehicle routing problem with time windows. *Computers & Operations Research*, 35(7):2307–2330, 2008. 22
- B. Kallehauge, J. Larsen, O. B. G. Madsen, and M. M. Solomon. *Vehicle routing problem with time windows*. Springer, 2005. 22
- P. Kaminsky and E. Simchi-Levi. *Designing and managing the supply chain: concepts, strategies, and case studies*. McGraw Hill Professional, 2003. 16
- D. Kirby. Is your fleet the right size? *Journal of the Operational Research Society*, 10(4):252–252, 1959. 20
- D. Kirchler and R. Wolfler Calvo. A granular tabu search algorithm for the dial-a-ride problem. *Transportation Research Part B: Methodological*, 56:120–135, 2013. 19

- K. Knight and J. Hofer. Vehicle scheduling with timed and connected calls: A case study. *Operational Research Quarterly*, 19(3):299–310, 1968. 22
- A. Kok, C. Meyer, H. Kopfer, and J. Schutten. A dynamic programming heuristic for the vehicle routing problem with time windows and european community social legislation. *Transportation Science*, 44:442–454, 2010. 22, 33
- A. A. Kovacs, S. N. Parragh, K. F. Doerner, and R. F. Hartl. Adaptive large neighborhood search for service technician routing and scheduling problems. *Journal of scheduling*, 15(5):579–600, 2012. 24
- V. B. Kreng and F. T. Chen. The benefits of a cross-docking delivery strategy: A supply chain collaboration approach. *Production Planning and Control*, 19:229–241, 2008. 33
- N. Labadie, J. Melechovsk, and R. W. Calvo. Hybridized evolutionary local search algorithm for the team orienteering problem with time windows. *Journal of Heuristics*, 17(6):729–753, 2011. 60, 74, 92, 97
- N. Labadie, R. Mansini, J. Melechovsk, and R. W. Calvo. The team orienteering problem with time windows: An lp-based granular variable neighborhood search. *European Journal of Operational Research*, 220(1):15–27, 2012. 60, 92
- R. Lahyani, M. Khemakhem, F. Semet, and H. Chabchoub. Taxonomy for rich vehicle routing problems. In *Proceedings of the 1st International Conference on Logistics Operations Management, IEEE International Conference*, pages 75–78, Le Havre, France, 2012. 11
- R. Lahyani, M. Khemakhem, and F. Semet. Solving rich profitable tour problems. The Eight Triennial Symposium on Transportation Analysis (TRISTAN VIII), San Pedro de Atacama, Chile, 2013a. http://www.isci.cl/tristan/data/Routing/TRISTAN8_paper_158.pdf/. 59, 83
- R. Lahyani, M. Khemakhem, and F. Semet. A unified heuristic for solving rich variants of orienteering problems. ROADEF 2013, Troyes, France, 2013b. 59, 83
- R. Lahyani, M. Khemakhem, and F. Semet. Heuristics for rich profitable tour problems. In *5th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO), Tunisia*, pages 1–3. IEEE, 2013c. 59, 83
- R. Lahyani, M. Khemakhem, and F. Semet. On unified solution approach for solving multi-constraint traveling salesman problems with profits. ROADEF 2014, Bordeaux, France, 2014. 59, 83

- G. Laporte. Location-routing problems. In *Vehicle Routing: Methods and Studies*, pages 163–198. In Golden, B. L., Assad, A. A. (Eds.), North-Holland, Amsterdam, 1988. [18](#)
- G. Laporte. A survey of algorithms for location-routing problems. *Investigación Operativa*, 1(2):93–118, 1989. [18](#)
- G. Laporte. Fifty years of vehicle routing. *Transportation Science*, 43:408–416, 2009. [3](#), [12](#)
- G. Laporte and S. Martello. The selective travelling salesman problem. *Discrete Applied Mathematics*, 26(2-3):193–207, 1990. [56](#)
- G. Laporte and I. H. Osman. Routing problems: A bibliography. *Annals of Operations Research*, 61:227–262, 1995. [12](#)
- G. Laporte, Y. Nobert, and D. Arpin. *Optimal solutions to capacitated multidepot vehicle routing problems*. École des hautes études commerciales, 1984. [18](#)
- G. Laporte, Y. Nobert, and S. Taillefer. Solving a family of multi-depot vehicle routing and location-routing problems. *Transportation science*, 22(3):161–172, 1988. [18](#)
- G. Laporte, M. Gendreau, J. Y. Potvin, and F. Semet. Classical and modern heuristics for the vehicle routing problem. *International transactions in operational research*, 7(4-5):285–300, 2000. [111](#)
- A. Larsen, O. B. G. Madsen, and M. M. Solomon. Recent developments in dynamic vehicle routing systems. In *The Vehicle Routing Problem: Latest Advances and New Challenges*, pages 199–218. Springer, 2008. [16](#)
- J. Larsen. *Parallelization of the vehicle routing problem with time windows*. PhD thesis, Technical University of Denmark, 1999. [109](#)
- Y. H. Lee, J. W. Jung, and K. M. Lee. Vehicle routing scheduling for cross-docking in the supply chain. *Computers & Industrial Engineering*, 51:247–256, 2006. [25](#)
- A. N. Letchford, J. Lygaard, and R. W. Eglese. A branch-and-cut algorithm for the capacitated open vehicle routing problem. *Journal of the Operational Research Society*, 58(12):1642–1651, 2007. [25](#)
- Y. Li, A. Lim, and B. Rodrigues. Manpower allocation with time windows and job-teaming constraints. *Naval Research Logistics (NRL)*, 52(4):302–311, 2005. [23](#)

- C. J. Liao, Y. Lin, and S. C. Shih. Vehicle routing with cross-docking in the supply chain. *Expert Systems with Applications*, 37:6868–6873, 2010. [25](#)
- A. Lim, B. Rodrigues, and L. Song. Manpower allocation with time windows. *Journal of the Operational Research Society*, 55(11):1178–1186, 2004. [23](#)
- C. Lin, K. L. Choy, G. T. S. Ho, S. H. Chung, and H. Y. Lam. Survey of green vehicle routing problem: past and future trends. *Expert Systems with Applications*, 41(4):1118–1138, 2014. [24](#)
- S. W. Lin and V. F. Yu. A simulated annealing heuristic for the team orienteering problem with time windows. *European Journal of Operational Research*, 217(1):94–107, 2012. [92](#), [93](#)
- F. Louveaux and G. Laporte. Stochastic vehicle routing problems. In *Encyclopedia of Optimization*, pages 3829–3832. Springer, 2009. [16](#)
- M. E. Lübbecke and J. Desrosiers. Selected topics in column generation. *Operations Research*, 53(6):1007–1023, 2005. [105](#)
- R. Macedo, C. Alves, J. M. Valério De Carvalho, F. Clautiaux, and S. Hanafi. Solving the vehicle routing problem with time windows and multiple routes exactly using a pseudo-polynomial model. *European Journal Of Operational Research*, 214(3):536–545, 2011. [20](#)
- O. B. G. Madsen. *Optimal scheduling of trucks - A routing problem with tight due times for delivery*, pages 126–136. Strobel H, Genser R, Etschmaier M (eds) Optimization applied to transportation systems. International Institute for Applied System Analysis, Laxenburgh, Austria, 1976. [22](#)
- J. M. Magalhães and J. P. Sousa. Dynamic vrp in pharmaceutical distribution: a case study. *Central European Journal of Operations Research*, 14:177–192, 2006. [32](#)
- A. McKinnon, M. Browne, and A. Whiteing. *Green Logistics: Improving the environmental sustainability of logistics*. Kogan Page Publishers, 2012. [24](#)
- J. Melechovsky. A variable neighborhood search for the selective multi-compartment vehicle routing problem with time windows. *Lecture Notes in Management Science*, 5:159–166, 2013. [38](#)

- J. E. Mendoza, B. Castanier, C. Guéret, A. L. Medaglia, and Velasco. N. A simulation-based moea for the multi-compartment vehicle routing problem with stochastic demands. In *8th Metaheuristics International Conference (MIC), Hamburg, Germany*, 2009. [32](#)
- J. E. Mendoza, B. Castanier, C. Guéret, A. L. Medaglia, and N. Velasco. A memetic algorithm for the multi-compartment vehicle routing problem with stochastic demands. *Computers & Operations Research*, 37:1886–1898, 2010. [32](#)
- J. E. Mendoza, B. Castanier, C. Guéret, A. L. Medaglia, and N. Velasco. Constructive heuristics for the multicompartment vehicle routing problem with stochastic demands. *Transportation Science*, 45(3):346–363, 2011. [32](#)
- Z. Miao, A. Lim, and H. Ma. Truck dock assignment problem with operational time constraint within crossdocks. *European Journal of Operational Research*, 192:105–115, 2009. [33](#)
- J. Michallet, C. Prins, L. Amodeo, F. Yalaoui, and G. Vitry. Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services. *Computers & Operations Research*, 41:196–207, 2014. [20](#)
- H. Min, V. Jayaraman, and R. Srivastava. Combined location-routing problems: A synthesis and future research directions. *European Journal of Operational Research*, 108(1):1–15, 1998. [18](#)
- N. Mladenovic and P. Hansen. Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100, 1997. [4](#), [64](#), [168](#)
- R. Montemanni and L. M. Gambardella. An ant colony system for team orienteering problem with time windows. *Foundations of Computing and Decision Sciences*, 34(4):287–306, 2009. [91](#), [92](#)
- I. Moon, J. H. Lee, and J. Seong. Vehicle routing problem with time windows considering overtime and outsourcing vehicles. *Expert Systems with Applications*, 39(18):13202–13213, 2012. [24](#)
- I. Muter, J. F. Cordeau, and G. Laporte. A branch-and-price algorithm for the multidepot vehicle routing problem with interdepot routes. *Transportation Science*, 2014. doi: <http://dx.doi.org/10.1287/trsc.2013.0489>. [18](#)

- L. Muyltermans and G. Pang. On the benefits of co-collection: experiments with a multi-compartment vehicle routing algorithm. *European Journal of Operational Research*, 206(1):93–103, 2010a. [60](#)
- L. Muyltermans and G. Pang. On the benefits of co-collection: Experiments with a multi-compartment vehicle routing algorithm. *European Journal of Operational Research*, 206(1):93–103, 2010b. [21](#), [38](#)
- G. Nagy and S. Salhi. Location-routing: Issues, models and methods. *European Journal of Operational Research*, 177(2):649–672, 2007. [18](#)
- G. L. Nemhauser and L. A. Wolsey. *Integer and combinatorial optimization*. Wiley-Interscience, New York, NY, USA, 1988. ISBN 0-471-82819-X. [47](#), [50](#)
- A. Olivera and O. Viera. Adaptive memory programming for the vehicle routing problem with multiple trips. *Computers & Operations Research*, 34:28–47, 2007. [20](#)
- J. Oppen and A. Løkketangen. A tabu search approach for the livestock collection problem. *Computers & operations research*, 35(10):3213–3229, 2008. [20](#), [21](#), [23](#), [26](#), [30](#)
- J. Oppen, A. Løkketangen, and J. Desrosiers. Solving a rich vehicle routing and inventory problem using column generation. *Computers & Operations Research*, 37(7):1308–1317, 2010. [20](#), [23](#), [26](#), [30](#), [118](#)
- M. W. Padberg and G. Rinaldi. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review*, 33(1):60–100, 1991. [126](#)
- D. J. Papageorgiou, G. L. Nemhauser, J. Sokol, M. K. Cheon, and A. B. Keha. MIRPLib—a library of maritime inventory routing problem instances: survey, core model, and benchmark results. *European Journal of Operational Research*, forthcoming, 2014. doi: 10.1016/j.ejor.2013.12.013. [38](#)
- J. Paquette, J. F. Cordeau, and G. Laporte. Quality of service in dial-a-ride operations. *Computers and Industrial Engineering*, 56:1721–1734, 2009. [19](#)
- J. Paquette, J. F. Cordeau, G. Laporte, and M. Pascoal. Combining multicriteria analysis and tabu search for dial-a-ride problems. *Transportation Research Part B: Methodological*, 52:1–16, 2013. [19](#)

- S. N. Parragh. Introducing heterogeneous users and vehicles into models and algorithms for the dial-a-ride problem. *Transportation Research Part C: Emerging Technologies*, 19(5):912–930, 2011. 26
- S. N. Parragh, K. F. Doerner, and R. F. Hartl. A survey on pickup and delivery models : Part i: Transportation between customers and depot. *Journal für Betriebswirtschaft*, 58:21–51, 2008a. 19
- S. N. Parragh, K. F. Doerner, and R. F. Hartl. A survey on pickup and delivery models: Part ii: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58:81–117, 2008b. 19
- S. N. Parragh, J. F. Cordeau, K. F. Doerner, and R. F. Hartl. Models and algorithms for the heterogeneous dial-a-ride problem with driver-related constraints. *OR spectrum*, 34(3):593–633, 2012. 22, 23, 26, 32
- P. Pellegrini. Application of two nearest neighbor approaches to a rich vehicle routing problem. Technical report, Technical Report TR/IRIDIA/2005-15, IRIDIA, Université Libre de Bruxelles, Belgium, 2005. 13, 26
- P. Pellegrini, D. Favaretto, and E. Moretti. Multiple ant colony optimization for a rich vehicle routing problem: a case study. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 627–634. Springer, 2007. 23, 26
- J. Perttunen. On the significance of the initial solution in traveling salesman heuristics. *Journal of the Operational Research Society*, 45(10):1131–1140, 1994. 68
- R. J. Petch and S. Salhi. A multi-phase constructive heuristic for the vehicle routing problems with multiple trips. *Discrete Applied Mathematics*, 133:69–92, 2003. 20
- U. Pferschy and J. Schauer. The knapsack problem with conflict graphs. *Journal of Graph Algorithms and Applications*, 13(2):233–249, 2009. 78
- V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia. A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1):1–11, 2013. 3, 16
- D. S. Pirkwieser. *Hybrid metaheuristics and matheuristics for problems in bioinformatics and transportation*. PhD thesis, Vienna University of Technology, 2012. 60

- S. Pirkwieser, G. R. Raidl, and J. Gottlieb. Improved packing and routing of vehicles with compartments. *Lecture Notes in Computer Science*, Springer(4):392–399, 2012a. [60](#)
- S. Pirkwieser, G. R. Raidl, and J. Gottlieb. Improved packing and routing of vehicles with compartments. *Lecture Notes in Computer Sciences*, 6927:392–399, 2012b. [38](#)
- D. Pisinger and S. Røpke. A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8):2403–2435, 2007. [61](#), [70](#)
- D. Popović, M. Vidović, and G. Radivojević. Variable neighborhood search heuristic for the inventory routing problem in fuel delivery. *Expert Systems with Applications*, 39(18):13390–13398, 2012. [38](#), [39](#)
- W. B. Powell, J. A. Shapiro, and H. P. Simao. A representational paradigm for dynamic resource transformation problems. *Annals of Operations Research*, 104(1):231–279, 2001. [16](#)
- W. B. Powell, B. Bouzaiene-Ayari, and H. P. Simao. Dynamic models for freight transportation. *Handbooks in Operations Research and Management Science: Transportation*, 14:285–365, 2003. [16](#)
- É. Prescott-Gagnon, G. Desaulniers, and L. M. Rousseau. A branch-and-price-based large neighborhood search algorithm for the vehicle routing problem with time windows. *Networks*, 54(4):190–204, 2009. [73](#), [89](#)
- É. Prescott-Gagnon, G. Desaulniers, M. Drexler, and L. M. Rousseau. European driver rules in vehicle routing with time windows. *Transportation Science*, 44(4):455–473, 2010. [22](#)
- C. Prodhon and C. Prins. A memetic algorithm with population management (maipm) for the periodic location-routing problem. *Lecture Notes in Computer Science*, 5296:43–57, 2008. [32](#)
- H. N. Psaraftis. Dynamic vehicle routing problems. *Vehicle routing: Methods and studies*, 16:223–248, 1988. [16](#)
- H. N. Psaraftis. Dynamic vehicle routing: Status and prospects. *Annals of Operations Research*, 61(1):143–164, 1995. [16](#)

- J. Puchinger and G. R. Raidl. Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. In *Artificial intelligence and knowledge engineering applications: a bioinspired approach*, pages 41–53. Springer, 2005. [4](#)
- H. Pullen and M. Webb. A computer application to a transport scheduling problem. *The Computer Journal*, 10(1):10–13, 1967. [22](#)
- A. Rahimi-Vahed, T. G. Crainic, M. Gendreau, and W. Rei. A path relinking algorithm for a multi-depot periodic vehicle routing problem. *Journal of Heuristics*, 19(3):497–524, 2013. [18](#)
- M. E. Rancourt, J. F. Cordeau, and G. Laporte. Long-haul vehicle routing and scheduling with working hour rules. *Transportation Science*, 47(1):81–107, 2013. [22](#), [32](#)
- R. L. Rardin. *Optimization in operations research*, volume 166. Prentice Hall New Jersey, 1998. [ix](#), [2](#)
- M. Reed, A. Yiannakou, and R. Evering. An ant colony algorithm for the multi-compartment vehicle routing problem. *Applied Soft Computing*, 15:169–176, 2014. [21](#), [60](#)
- M. Reimann and H. Ulrich. Comparing backhauling strategies in vehicle routing using ant colony optimization. *Central European Journal of Operations Research*, 14(2):105–123, 2006. [33](#)
- A. Reisman. *Management science knowledge: it's creation, generalization and consolidation*. Quorum Books Publishing Company, Westport, CT, 1992. [13](#)
- S. Relvas, S. N. B. Magatão, A. P. F. D. Barbosa-Póvoa, and F. Neves Jr. Integrated scheduling and inventory management of an oil products distribution system. *Omega*, 41(6):955–968, 2014. [38](#)
- L. C. Reyes, Orta J. F. D., Barbosa J. J. G., J. T. Jimenez, H. J. F. Huacuja, and B. A. A. Cruz. An ant colony system algorithm to solve routing problems applied to the delivery of bottled products. In *Foundations of Intelligent Systems*, pages 329–338. Springer, 2008. [13](#)
- G. Ribeiro and G. Laporte. An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem. *Computers & Operations Research*, 39(3):728–735, 2012. [70](#)

- J. Rieck and J. Zimmermann. A new mixed integer linear model for a rich vehicle routing problem with docking constraints. *Annals of Operations Research*, 181(1): 337–358, 2010. [13](#), [23](#), [25](#), [30](#)
- G. Righini and M. Salani. Dynamic programming for the orienteering problem with time windows. Technical report, Dipartimento di Tecnologie dell’Informazione, Universita degli Studi Milano, Italy, 2006. [60](#), [91](#), [92](#), [97](#)
- G. Righini and M. Salani. Decremental state space relaxation strategies and initialization heuristics for solving the orienteering problem with time windows with dynamic programming. *Computers & Operations Research*, 36(4):1191–1203, 2009. [91](#), [92](#)
- U. Ritzinger and J. Puchinger. Hybrid metaheuristics for dynamic and stochastic vehicle routing. In *Hybrid Metaheuristics*, pages 77–95. Springer, 2013. [3](#), [16](#)
- S. Røpke and D. Pisinger. A unified heuristic for a large class of vehicle routing problems with backhauls. *European Journal of Operational Research*, 171(3):750–775, 2006a. [ix](#), [63](#)
- S. Røpke and D. Pisinger. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4): 455–472, 2006b. [4](#), [13](#), [35](#), [70](#), [89](#)
- R. Russell, W. C. Chiang, and D. Zepeda. Integrating multi-product production and distribution in newspaper logistics. *Computers & Operations Research*, 35 (5):1576–1588, 2008. [25](#)
- S. Salhi and R. J. Petch. A genetic algorithm based heuristic for the vehicle routing problem with multiple trips. *Journal of Mathematical Modeling and Algorithms*, 6:591–613, 2007. [20](#)
- S. Salhi, S. R. Thangiah, and F. Rahman. A genetic clustering method for the multi-depot vehicle routing problem. In *Artificial Neural Nets and Genetic Algorithms*, pages 234–237. Springer, 1998. [18](#)
- F. A. Santos, G. R. Mateus, and A. S. Cunha. A branch-and-price algorithm for a vehicle routing problem with cross-docking. *Electronic Notes in Discrete Mathematics*, 37:249–254, 2011a. [25](#)
- F. A. Santos, G. R. Mateus, and A. S. Cunha. A novel column generation algorithm for the vehicle routing problem with cross-docking. *Lecture Notes in Computer Science*, 6701:412–425, 2011b. [25](#)

- A. M. Sarmiento and R. Nagi. A review of integrated analysis of production–distribution systems. *IIE transactions*, 31(11):1061–1074, 1999. 17
- M. W. P. Savelsbergh. The vehicle routing problem with time windows: Minimizing route duration. *ORSA Journal on Computing*, 4(2):146–154, 1992. 80
- A. Sbihi and R. W. Eglese. The relationship between vehicle routing and scheduling and green logistics: a literature survey. Technical report, Lancaster University: The Department of Management Science, 2007a. 24
- A. Sbihi and R. W. Eglese. Combinatorial optimization and green logistics. *4OR*, 5(2):99–116, 2007b. 24
- M. Schilde, K. F. Doerner, F. H. Richard, and G. Kiechle. Metaheuristics for the bi-objective orienteering problem. *Swarm Intelligenc*, 3(3):179–201, 2009. 34, 90, 91
- M. Schilde, K. F. Doerner, F. H. Richard, and G. Kiechle. Metaheuristics for the dynamic stochastic dial-a-ride problem with expected return transports. *Computers & Operations Research*, 38(12):1719–1730, 2011. 64
- V. Schmid, K. F. Doerner, R. F. Hartl, M. W. P. Savelsbergh, and W. Stoecher. A hybrid solution approach for ready-mixed concrete delivery. *Transportation Science*, 43(1):70–85, 2009. 33
- V. Schmid, K. F. Doerner, and G. Laporte. Rich routing problems arising in supply chain management. *European Journal of Operational Research*, 224(3):435–448, 2013. 17
- L. Schrage. Formulation and structure of more complex/realistic routing and scheduling problems. *Networks*, 11(2):229–232, 1981. 25
- G. Schrimph, J. Schneider, H. Stamm-Wilbrandt, and G. Dueck. Record breaking optimization results using the ruin and recreate principle. *Journal of Computational Physics*, 159(2):139–171, 2000. 6, 67
- F. Semet and E. Taillard. Solving real-life vehicle routing problems efficiently using tabu search. *Annals of Operations research*, 41(4):469–488, 1993. 20, 25
- P. Shaw. A new local search algorithm providing high quality solutions to vehicle routing problems. Technical report, Department of Computer Science, University of Strathclyde, Scotland, 1997. 66, 70, 73

- P. Shaw. Using constraint programming and local search methods to solve vehicle routing problem. In *Proceedings CP-98 (Fourth International Conference on Principles and Practice of Constraint Programming)*, 1998. [66](#), [70](#), [73](#)
- M. M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265, 1987. [76](#), [91](#), [173](#)
- K. Sörensen, M. Sevaux, and P. Schittekat. Multiple neighbourhood search in commercial vrp packages: Evolving towards self-adaptive methods. In *Adaptive and Multilevel Metaheuristics*, pages 239–253. Springer, 2008. [12](#)
- W. Souffriau, P. Vansteenwegen, G. V. Berghe, and D. V. Oudheusden. Automated parameterisation of a metaheuristic for the orienteering problem. In *Adaptive and Multilevel Metaheuristics*, pages 255–269. Springer, 2008. [90](#)
- W. Souffriau, P. Vansteenwegen, G. V. Berghe, and D. V. Oudheusden. The multi-constraint team orienteering problem with multiple time windows. *Transportation Science*, 47(1):53–63, 2013. [60](#)
- M. Stålhane, J. G. Rakke, C. R. Moe, H. Andersson, M. Christiansen, and K. Fagerholt. A construction and improvement heuristic for a liquefied natural gas inventory routing problem. *Computers & Industrial Engineering*, 62(1):245–255, 2012. [38](#)
- A. Stenger, D. Vigo, S. Enz, and M. Schwind. An adaptive variable neighborhood search algorithm for a vehicle routing problem arising in small package shipping. *Transportation Science*, 47(1):64–80, 2013. [24](#)
- A. Subramanian. *Heuristic, exact and hybrid approaches for vehicle routing problems*. PhD thesis, Ph. D. thesis, Universidade Federal Fluminense, Niterói, Brazil, 2012. [61](#)
- A. Subramanian, E. Uchoa, and L. S. Ochi. A hybrid algorithm for general vehicle routing problems. In *Proceedings of Route 2011*, 2011. [4](#), [13](#), [35](#)
- P. Surekha and S. Sumathi. Solution to multi-depot vehicle routing problem using genetic algorithms. *World Applied Programming*, 1(3):118–131, 2011. [18](#)
- C. D. Tarantilis and C. T. Kiranoudis. Distribution of fresh meat. *Journal of Food Engineering*, 51(1):85–91, 2002. [25](#)

- C. D. Tarantilis, C. T. Kiranoudis, and V. S. Vassiliadis. A list based threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem. *Journal of the Operational Research Society*, 54(1):65–71, 2003. [20](#)
- C. D. Tarantilis, C. T. Kiranoudis, and V. S. Vassiliadis. A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem. *European Journal of Operational Research*, 152(1):148–158, 2004. [20](#)
- F. A. Tillman. The multiple terminal delivery problem with probabilistic demands. *Transportation Science*, 3(3):192–204, 1969. [18](#)
- P. Toth and D. Vigo. *The Vehicle routing problem*. SIAM Monographs on Discrete Mathematics and Application, Philadelphia, PA: SIAM, 2002. [12](#), [19](#), [30](#), [32](#)
- F. Tricoire, M. Romauch, K. F. Doerner, and R. F. Hartl. Heuristics for the multi-period orienteering problem with multiple time windows. *Computers & Operations Research*, 37(2):351–367, 2010. [60](#), [92](#), [93](#)
- F. Tricoire, K. F. Doerner, R. F. Hartl, and M. Iori. Heuristic and exact algorithms for the multi-pile vehicle routing problem. *OR spectrum*, 33(4):931–959, 2011. [21](#)
- T. Tsiligirides. Heuristic methods applied to orienteering. *Journal of the Operational Research Society*, 35(9):797–809, 1984. [xiii](#), [56](#), [90](#), [91](#), [173](#)
- B. Vahdani, R. Tavakkoli-Moghaddam, M. Zandieh, and J. Razmi. Vehicle routing scheduling using an enhanced robust hybrid optimization approach. *Journal of Intelligent Manufacturing Systems*, 23(3):759–774, 2012. [25](#)
- J. Van Belle, P. Valckenaers, and D. Cattrysse. Cross-docking: State of the art. *Omega*, 40(6):827–846, 2012. [25](#)
- L. Van der Bruggen, R. Gruson, and M. Salomon. Reconsidering the distribution structure of gasoline products for a large oil company. *European Journal of Operational Research*, 81(3):460–473, 1995. [38](#)
- F. Vanderbeck. On dantzig-wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm. *Operations Research*, 48(1):111–128, 2000. [51](#)
- F. Vanderbeck and M. W. P. Savelsbergh. A generic view of dantzig-wolfe decomposition in mixed integer programming. *Operations Research Letters*, 34(3):296–306, 2006. [51](#)

- F. Vanderbeck and L. A. Wolsey. Reformulation and decomposition of integer programs. In *50 Years of Integer Programming 1958-2008*, pages 431–502. Springer, 2010. [51](#)
- P. Vansteenwegen, W. Souffriau, G. V. Berghe, and D. V. Oudheusden. Iterated local search for the team orienteering problem with time windows. *Computers & Operations Research*, 36(12):3281–3290, 2009a. [75](#), [92](#), [93](#)
- P. Vansteenwegen, W. Souffriau, G.V. Berghe, and D. V. Oudheusden. A guided local search metaheuristic for the team orienteering problem. *European Journal of Operational Research*, 196(1):118–127, 2009b. [90](#)
- P. Vansteenwegen, W. Souffriau, and D. V. Oudheusden. The orienteering problem: a survey. *European Journal of Operational Research*, 209(1):1–10, 2011. [57](#)
- C. J. Vidal and M. Goetschalckx. Strategic production-distribution models: A critical review with emphasis on global supply chain models. *European Journal of Operational Research*, 98(1):1–18, 1997. [17](#)
- T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. *A unifying view on timing problems and algorithms*. CIRRELT, 2011. [61](#)
- T. Vidal, T. G. Crainic, M. Gendreau, N. Lahrichi, and W. Rei. A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. *Operations Research*, 60(3):611–624, 2012. [18](#), [20](#)
- T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. Heuristics for multi-attribute vehicle routing problems: a survey and synthesis. *European Journal of Operational Research*, 231(1):1–21, 2013a. [61](#)
- T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers & Operations Research*, 40(1):475–489, 2013b. [4](#), [13](#), [35](#)
- T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research*, 234(3):658–673, 2014. [4](#), [13](#), [35](#), [61](#), [89](#)
- D. S. Vidit. Time dependent truck routing and driver scheduling problem with hours of service regulations. Technical report, Master’s Thesis. USA, Boston: Northeastern University Libraries, 2008. [22](#)

- M. Vidović, D. Popović, and B. Ratković. Mixed integer and heuristics model for the inventory routing problem in fuel delivery. *International Journal of Production Economics*, 147(Part C):593–604, 2014. [38](#), [39](#)
- J. G. Villegas, C. Prins, C. Prodhon, A. L. Medaglia, and N. Velasco. A grasp with evolutionary path relinking for the truck and trailer routing problem. *Computers & Operations Research*, 38(9):1319–1334, 2011. [25](#)
- J. G. Villegas, C. Prins, C. Prodhon, A. L. Medaglia, and N. Velasco. A matheuristic for the truck and trailer routing problem. *European Journal of Operational Research*, 230(2):231–244, 2013. [25](#)
- D. Villeneuve, J. Desrosiers, M. E. Lübbecke, and F. Soumis. On compact formulations for integer programs solved by column generation. *Annals of Operations Research*, 139(1):375–388, 2005. [49](#)
- U. Vogel. *A flexible metaheuristic framework for solving rich vehicle routing problems*, volume 17. Shaker Verlag, 2012. [61](#)
- Q. Wang, Q. Ji, and C.H. Chiu. Optimal routing for heterogeneous fixed fleets of multi-compartment vehicles. *Mathematical Problems in Engineering*, forthcoming, 2014. doi: 10.1155/2014/847630. [38](#), [60](#)
- M. Wen, J. Larsen, J. Clausen, J. F. Cordeau, and G. Laporte. Vehicle routing with cross-docking. *Journal of the Operational Research Society*, 60:1708–1718, 2009. [25](#), [33](#)
- M. Wen, J. F. Cordeau, G. Laporte, and J. Larsen. The dynamic multi-period vehicle routing problem. *Computers & Operations Research*, 37(9):1615–1623, 2010. [20](#)
- M. Wen, E. Krappner, J. Larsen, and T. Stidsen. A multi-level variable neighborhood heuristic for a practical vehicle routing and driver scheduling problem. *Networks*, 58(4):311–322, 2011. [18](#), [22](#), [64](#)
- Y. Xiao, Q. Zhao, I. Kaku, and Y. Xu. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers & Operations Research*, 39(7):1419–1431, 2012. [24](#)
- H. Xu, Z. L. Chen, S. Rajagopal, and S. Arunapuram. Solving a practical pick-up and delivery problem. *Transportation Science*, 37(3):347–364, 2003. [18](#)

G. Zäpfel and M. Bögl. Multi-period vehicle routing and crew scheduling with outsourcing options. *International Journal of Production Economics*, 113:980–996, 2008. [18](#), [22](#), [24](#), [34](#)

Résumé

Une mathheuristique unifiée pour résoudre des problèmes de tournées de véhicules riches

Les problèmes considérés dans cette thèse se retrouvent dans le domaine du transport routier de marchandises relatifs à de nombreux contextes industriels réels. Ces problèmes correspondent souvent à des combinaisons complexes des décisions d'affectation, de chargement, de planification et de routage. Toutes ces décisions sont inter-corrélées afin d'optimiser les ressources utilisées, tout en satisfaisant les exigences des clients. Différents types de décision et de nombreuses caractéristiques doivent être considérés dans ces problèmes de transport qui sont généralement considérés comme *riches*. Un problème de transport riche peut être considéré comme un problème multi-attribut reflétant la complexité des systèmes de transport réels et combinant divers défis soulevés quotidiennement dans les industries. A travers la littérature consacrée pour les Problèmes de Tournées de Véhicules (VRPs), on a pu constaté un manque d'études sur ce domaine de recherche émergent, souvent connu sous le nom de Rich VRP (RVRPs), en dépit de plusieurs travaux dédiés à différentes variantes du VRP. La classe des RVRPs nécessite donc une classification taxonomique permettant d'identifier les problèmes qui peuvent être considérés comme riches parmi les nombreuses variantes du VRP. Le but de la première partie de la thèse est de fournir une taxonomie générique s'intéressant à la littérature des RVRPs.

Les méthodes de résolution proposées précédemment pour les VRPs peuvent souvent être adaptées à d'autres problèmes semblables, tandis que les problèmes inspirés de la réalité soulèvent de nouveaux défis méthodologiques en raison de leur variété. Cette variété peut affecter à la fois le modèle du problème et la méthode de résolution proposée. En outre, les RVRPs combinent souvent des décisions et des objectifs contradictoires. Résoudre ces problèmes de manière itérative en considérant un sous-ensemble de caractéristiques et/ou de décisions peut conduire à des décisions irréalisables. Par conséquent, proposer une nouvelle approche de so-

lution spécifique à chaque problème concret émergent ne représente plus un choix raisonnable ni pour les chercheurs, ni pour les praticiens. En effet, les logiciels commerciaux résolvant les systèmes de transport doivent être en mesure de faire face à une variété de problèmes réels et de contourner les multiples aspects combinatoires qui peuvent survenir. Ces observations motivent le développement d'une méthode unifiée capable de résoudre une large classe de VRP basiques et riches.

Plusieurs méthodes de résolution ont été proposées pour le VRP: des méthodes exactes qui fournissent la solution optimale d'un modèle, et les méthodes approchées qui produisent des solutions de bonne qualité dans un délai raisonnable, au détriment de l'optimalité. Comme la plupart des problèmes abordés sont NP-difficiles, les méthodes exactes sont rarement applicables dans des contextes réels. Ces contextes nécessitent souvent une solution rapidement pour résoudre un VRP multi-attribut. Une alternative aux méthodes exactes serait donc de combiner des algorithmes exactes et les métaheuristiques. Cette approche émergente, appelée matheuristiques ou métaheuristiques à base de modèles, semble être une voie très prometteuse vers la solution de problèmes riches d'optimisation combinatoire. Les matheuristiques permettent de profiter de la synergie entre les deux approches et conduisent souvent à des performances nettement plus élevées par rapport à la qualité de la solution et le temps de calcul. L'objectif méthodologique de cette thèse est de proposer un cadre de résolution unifiée pour les problèmes de tournées de véhicules multi-contraints (VRP). Ces travaux de recherche sont motivés par les défis complexes que posent les VRPs dans de nombreuses applications réelles formant la classe des problèmes de tournées de véhicules riches (RVRPs). On propose une heuristique de génération de colonne basée sur une matheuristique. La méthode proposée peut être utilisée pour résoudre un ensemble de VRPs de base et des VRPs riches. La matheuristique proposée repose sur la méthode de recherche à voisinage variable (VNS), ([Mladenovic and Hansen, 1997](#)).

Plus précisément, dans la première partie de la thèse, on présente une revue de la littérature complète des RVRPs ainsi qu'une taxonomie. Ensuite, on propose une définition élaborée des RVRPs. On sélectionne 41 articles, pour les diverses variantes des RVRPs qu'ils traitent, et on les classe en fonction de la taxonomie proposée. Une fois le classement est effectué, une analyse typologique, basée sur deux critères discriminatoires, est réalisée. Cette analyse débouche sur la définition des RVRPs. La taxonomie ainsi développée souligne le manque de méthodes unifiées capables de traiter de nombreuses variantes des VRPs de manière efficace. Dans ce cadre découle le but de la seconde partie de cette thèse qui consiste à proposer une méthode de génération de colonnes basée sur une matheuristique pour

résoudre les RVRPs. La méthode proposée est capable de traiter un large éventail de contraintes. Elle est ensuite utilisée pour résoudre une grande variété de VRPs.

Le RVRP étudié dans cette thèse concerne les industries où les marchandises sont non homogènes et doivent être séparés pendant leur transport. Cette caractéristique motive l'utilisation des véhicules multi-compartiments. Dans ce problème, un client peut avoir différentes commandes, chacune correspondant à un produit. La possibilité de livrer les commandes par des véhicules différents introduit des politiques de division spécifiques et augmente la difficulté du problème. Plus précisément, on s'intéresse à la résolution d'un problème de tournées de véhicules multi-dépôt multi-compartiment multi-produits avec fenêtres de temps (MDMCMCm-VRPTW).

On propose de concevoir et d'appliquer une méthode de génération de colonnes basée sur une matheuristique unifiée qui contribue à la solution d'un RVRP et certaines de ses variantes. Tout d'abord, on propose une modélisation mathématique du problème sous forme d'un programme linéaire en nombres entiers et mixtes. Puis, on applique la décomposition de Dantzig-Wolfe (DW) sur cette formulation. Cette décomposition donne naissance à un problème en nombre entier plus facile à traiter du point de vue combinatoire et un ensemble de sous-problèmes, un pour chaque véhicule. On propose une matheuristique de recherche à voisinage variable (VNS) pour résoudre le sous-problème. Cette méthode combine plusieurs heuristiques de routage de type destruction et insertion ainsi que des procédures efficaces de contrôle de réalisabilité des contraintes, afin de résoudre le MDMCMCm-VRPTW pour un seul véhicule. Le problème de chargement, venant de la présence de plusieurs compartiments, est examiné attentivement. Deux voisinages de chargement, basés sur la résolution de programmes mathématiques, sont proposés et résolus à l'optimalité afin d'intensifier la recherche. Ces voisinages interagissent avec les voisinages heuristiques de routage, comme cela est fait de manière courante dans les matheuristiques.

La description de la solution du MDMCMCm-VRPTW ou plus généralement une solution pour un problème réel doit tenir compte de plusieurs attributs relatifs aux contraintes, décisions et objectifs. Dans le MDMCMCm-VRPTW, les commandes peuvent être interprétées comme des clients fictifs et l'ensemble des commandes associées à un même client ont les mêmes attributs (par exemple, les coordonnées, les fenêtres de temps, le temps de service, etc). En outre, le processus de recherche doit garder en mémoire la séquence des commandes livrées sur une route ainsi que leur séquençage dans le compartiment. La structure des données d'une solution pour le MDMCMCm-VRPTW est alors beaucoup plus difficile à concevoir et ne peut pas être modélisé par une seule classe qui maintient toutes les données de routage et de chargement. Par conséquent, on introduit les classes suivantes: la

classe Commande, la classe Compartiment, la classe Problème et la classe Solution. La conception de structures de données est générique afin de faciliter la ré-utilisation du code et son extension facilement pour d'autres variantes du VRP. Ceci représente une autre contribution méthodologique de la thèse.

Des études expérimentales approfondies sont conduites sur un ensemble de 191 instances moins complexes pour tester la matheuristique VNS. Ces instances sont relatives aux variantes Orienteering Problem et Orienteering Problem with Time Windows. Les expérimentations valident la compétitivité de la matheuristique unifiée par rapport aux meilleurs résultats trouvés dans la littérature. Une analyse de sensibilité, effectuée sur des instances plus complexes générées pour le MDMCMCm-VRPTW, révèle l'importance de certains choix algorithmiques et des voisinages de chargement pour parvenir à des solutions de très bonne qualité.

La matheuristique basée sur la méthode de VNS est intégrée dans l'heuristique de génération de colonnes pour résoudre le MDMCMCm-VRPTW. On propose un ensemble de techniques visant à améliorer la qualité de la solution ainsi qu'une méthode exacte de post-traitement capable d'optimiser l'affectation des clients aux tournées de véhicules. Tout en développant la matheuristique unifiée, on se concentre sur la définition de structures de données génériques qui offre une grande flexibilité lorsqu'il s'agit d'étendre ou de spécialiser la matheuristique proposée.

Finalement, on s'intéresse à l'application réelle qui a motivé cette thèse. On présente le problème de tournées de véhicules riche multi-produit, multi-période multi-compartiment avec une activité de nettoyage des compartiments. Cette application réelle survient dans le processus de collecte de l'huile d'olive en Tunisie. On propose un algorithme exact de type branch-and-cut pour résoudre ce problème. On évalue la performance de l'algorithme proposé sur des données réelles selon différents scénarios de transport dans l'objectif de démontrer à notre partenaire industriel, les avantages de l'utilisation des véhicules multi-compartiments.

Abstract

Unified matheuristic for solving rich vehicle routing problems

The purpose of this thesis is to develop a solution framework for Rich Vehicle Routing Problems (RVRPs). We first provide a comprehensive survey of the RVRP literature as well as a taxonomy. Selected papers addressing various variants are classified according to the proposed taxonomy. A cluster analysis based on two discriminating criteria is performed and leads to define RVRPs. In this thesis we are interested in solving a multi-depot multi-compartment multi-commodity vehicle routing problem with time windows (MDMCMCm-VRPTW). We propose a unified column generation heuristic cooperating with a variable neighborhood search (VNS) matheuristic. The VNS combines several removal and insertion routing heuristics as well as computationally efficient constraint checking. Two loading neighborhoods based on the solution of mathematical programs are proposed to intensify the search. On a set of 191 instances of less complex routing problems, the unified matheuristic turns to be competitive. A sensitivity analysis, performed on more complex generated instances reveals the importance of some algorithmic features and of loading neighborhoods for reaching high quality solutions. The VNS based matheuristic is embedded in a column generation heuristic to solve the MDMCMCm-VRPTW. We propose an exact post-processing method to optimize the assignment of customers to vehicle routes. Last, we introduce, model and solve to optimality a RVRP arising in the olive oil collection process in Tunisia. We propose an exact branch-and-cut algorithm to solve the problem. We evaluate the performance of the algorithm on real data sets under different transportation scenarios.

Keywords: Rich Vehicle Routing Problems, Loading Problems, Taxonomy, Matheuristic, Variable Neighborhood Search, Column Generation Method.

Résumé

Une matheuristique unifiée pour résoudre des problèmes de tournées de véhicules riches

L'objectif de cette thèse est de développer un cadre méthodologique pour les problèmes de tournées de véhicules riches (RVRPs). Nous présentons d'abord une taxonomie et une définition élaborée des RVRPs basée sur une analyse typologique réalisée en fonction de deux critères discriminatoires. Dans cette thèse, nous nous intéressons à la résolution du problème de tournées de véhicules multi-dépôt multi-compartment multi-produits avec fenêtres de temps (MDMCMCm-VRPTW). Nous proposons une heuristique de génération de colonnes unifiée qui inclut une matheuristique de type VNS. La matheuristique combine plusieurs heuristiques de routage de type destruction et insertion ainsi que des procédures efficaces de contrôle de réalisabilité des contraintes afin de résoudre le MDMCMCm-VRPTW pour un seul véhicule. Deux voisinages de chargement, basés sur la résolution de programmes mathématiques sont proposées. Des études expérimentales approfondies sont conduites sur un ensemble de 191 instances pour des VRPs moins complexes. Les expérimentations valident la compétitivité de la matheuristique unifiée. Une analyse de sensibilité révèle l'importance de certains choix algorithmiques et des voisinages de chargement pour parvenir à des solutions de très bonne qualité. La matheuristique basée sur la méthode de VNS est intégrée dans l'heuristique de génération de colonnes pour résoudre le MDMCMCm-VRPTW. Nous proposons une méthode exacte de post-traitement capable d'optimiser l'affectation des clients aux tournées de véhicules. Enfin, nous résolvons un RVRP qui survient dans le processus de collecte de l'huile d'olive en Tunisie à l'aide d'un algorithme exact de type branch-and-cut.

Mots-Clefs: Problèmes de Tournées de Véhicules Riches, Problèmes de Chargement, Taxonomie, Matheuristique, Méthode de Recherche à Voisinages Variables, Méthode de Génération de Colonnes.

Electronic appendix A

In this appendix, we provide files with results for the full computational experiments carried out in this thesis.

- **Table A.1** provides full results of VNS* on [Tsiligirides \(1984\)](#) instances.
- **Table A.2** provides full results of VNS* on [Chao et al. \(1996\)](#) instances.
- **Table A.3** provides full results of VNS* on [Solomon \(1987\)](#) instances.
- **Table A.4** provides full results of VNS* on [Cordeau et al. \(1997\)](#) instances.
- **Table A.5** provides full results of VNS* on instances of *type A*.
- **Table A.6** provides full results of VNS* on instances of *type B*.
- **Table A.7** provides full results of VNS* on instances of *type C*.
- **Table A.8** provides full results of VNS*I on instances of *type C*.
- **Table A.9** provides full results of VNS*II on instances of *type C*.
- **Table A.10** provides full results of the column generation heuristic on the MCMCm-VRPTW instances of *type C*.

Columns indicate the following:

- *# Cust*: Total number of customers in the best solution.
- *# Ord*: Total number of orders in the best solution.
- *BK*: Best known solution.
- *Cost₁*: Initial objective value of the RMP.
- *Cost₃*: Objective value of the RMP after the post-processing procedure.
- *Costs Gap%*: Gap between *Cost₁* and *Cost₃*.
- *Gap%*: Average percentage gap to the BK value or optimal solution.
- *Instance*: Instance name.
- $|\mathcal{K}|$: Number of used vehicles.
- *n*: Number of customers in the instance.
- *Obj*: Show the minimum, maximum and average solution values over 5 runs.
- *Opt*: Optimal value.
- *T_{distance}*: Total distance of the best solution.
- *T_{profit}*: Total profit of the best solution.
- *T_{profit_{Avg}}*: Average total profit over 5 runs.
- *T_{wait}*: Total wait of the best solution.
- *Time(s)*: Average running time in seconds over 5 runs.
- *T^{max}*: Maximum route duration.

Table A.1 Results of VNS* on Tsiligrides (1984) instances

Set p_21

<i>Instance</i>	<i>T^{max}</i>	<i>Opt</i>	<i>T_profit_{Avg}</i>	<i>Gap%</i>	<i>Time(s)</i>
1	15	120	120	0.00	0.23
2	20	200	200	0.00	0.26
3	23	210	210	0.00	0.33
4	25	230	230	0.00	0.28
5	27	230	230	0.00	0.32
6	30	265	265	0.00	0.31
7	32	300	300	0.00	0.38
8	35	320	320	0.00	0.38
9	38	360	360	0.00	0.38
10	40	395	395	0.00	0.44
11	45	450	450	0.00	0.31
Average				0.00	0.33

Set p_32

<i>Instance</i>	<i>T^{max}</i>	<i>Opt</i>	<i>T_profit_{Avg}</i>	<i>Gap%</i>	<i>Time(s)</i>
1	5	10	10	0.00	0.04
2	10	15	15	0.00	0.06
3	15	45	45	0.00	0.16
4	20	65	65	0.00	0.32
5	25	90	90	0.00	0.45
6	30	110	110	0.00	0.53
7	35	135	135	0.00	0.66
8	40	155	155	0.00	0.80
9	46	175	175	0.00	0.93
10	50	190	190	0.00	0.96
11	55	205	205	0.00	0.94
12	60	225	225	0.00	0.97
13	65	240	240	0.00	0.97
14	70	260	260	0.00	0.97
15	73	265	265	0.00	0.98
16	75	270	270	0.00	0.95
17	80	280	280	0.00	0.96
18	85	285	285	0.00	0.98
Average				0.00	0.70

Set p_33

<i>Instance</i>	<i>T^{max}</i>	<i>Opt</i>	<i>T_profit_{Avg}</i>	<i>Gap%</i>	<i>Time(s)</i>
1	15	170	170	0.00	0.4
2	20	200	200	0.00	0.5
3	25	260	260	0.00	0.54
4	30	320	320	0.00	0.66
5	35	390	390	0.00	0.74
6	40	430	430	0.00	0.96
7	45	470	470	0.00	1.06
8	50	520	520	0.00	1.04
9	55	550	550	0.00	1.22

10	60	580	580	0.00	1.14
11	65	610	610	0.00	1.21
12	70	640	640	0.00	1.17
13	75	670	670	0.00	1.18
14	80	710	710	0.00	1.14
15	85	740	740	0.00	1.11
16	90	770	770	0.00	1.12
17	95	790	790	0.00	1.12
18	100	800	800	0.00	1.1
19	105	800	800	0.00	1.07
20	110	800	800	0.00	1.06
<hr/>					
Average				0.00	0.98
<hr/>					

Table A.2 Results of VNS* on Chao et al. (1996) instances

Set p_64

<i>Instance</i>	T^{max}	<i>BK</i>	T_{profit} <small><i>Avg</i></small>	<i>Gap%</i>	<i>Time(s)</i>
1	15	96	96	0.00	0.63
2	20	294	294	0.00	1.58
3	25	390	390	0.00	2.44
4	30	474	474	0.00	2.89
5	35	576	574.8	0.20	3.26
6	40	714	714	0.00	4.83
7	45	816	814.8	0.14	5.37
8	50	900	900	0.00	5.97
9	55	984	981.6	0.24	5.88
10	60	1062	1062	0.00	6.64
11	65	1116	1116	0.00	6.60
12	70	1188	1188	0.00	6.64
13	75	1236	1236	0.00	6.48
14	80	1284	1279.2	0.37	5.73
Average				0.07	4.64

Set p_66

<i>Instance</i>	T^{max}	<i>BK</i>	T_{profit} <small><i>Avg</i></small>	<i>Gap%</i>	<i>Time(s)</i>
1	5	10	10	0.00	0.12
2	10	40	40	0.00	0.29
3	15	120	120	0.00	0.62
4	20	205	205	0.00	0.97
5	25	290	286	1.37	1.29
6	30	400	400	0.00	1.84
7	35	465	465	0.00	2.18
8	40	575	575	0.00	3.02
9	45	650	650	0.00	3.56
10	50	730	730	0.00	4.06
11	55	825	825	0.00	4.91
12	60	915	915	0.00	5.57
13	65	980	980	0.00	5.76
14	70	1070	1070	0.00	6.62
15	75	1140	1140	0.00	6.85
16	80	1215	1215	0.00	7.77
17	85	1270	1270	0.00	7.58
18	90	1340	1340	0.00	7.97
19	95	1395	1395	0.00	7.90
20	100	1465	1455	0.68	7.94
21	105	1520	1520	0.00	7.95
22	110	1560	1558	0.12	7.61
23	115	1595	1594	0.06	7.14
24	120	1635	1629	0.36	6.70
25	125	1670	1670	0.00	6.19
26	130	1680	1680	0.00	5.90
Average				0.10	4.94

Table A.3 Results of VNS* on Solomon (1987) instances

<i>Instance</i>	<i>VNS* (20 iter)</i>		<i>VNS* (20 s)</i>	<i>VNS* (120 s)</i>	<i>VNS* (300s)</i>
	<i>Gap%</i>	<i>Time(s)</i>	<i>Gap%</i>	<i>Gap%</i>	<i>Gap%</i>
c101_50	0.00	0.66	0.00	0.00	0.00
c102_50	0.00	0.87	0.00	0.00	0.00
c103_50	0.00	1.00	0.00	0.00	0.00
c104_50	0.00	1.13	0.00	0.00	0.00
c105_50	0.00	0.80	0.00	0.00	0.00
c106_50	0.00	0.69	0.00	0.00	0.00
c107_50	0.00	0.84	0.00	0.00	0.00
c108_50	0.00	0.85	0.00	0.00	0.00
c109_50	0.00	1.04	0.00	0.00	0.00
r101_50	0.00	0.39	0.00	0.00	0.00
r102_50	1.52	0.82	1.52	1.52	1.52
r103_50	0.00	0.92	0.00	0.00	0.00
r104_50	0.00	0.97	0.00	0.00	0.00
r105_50	0.00	0.49	0.00	0.00	0.00
r106_50	2.40	0.89	2.40	2.40	2.40
r107_50	0.00	0.91	0.00	0.00	0.00
r108_50	0.00	0.98	0.00	0.00	0.00
r109_50	0.00	0.67	0.00	0.00	0.00
r110_50	0.00	0.79	0.00	0.00	0.00
r111_50	0.00	0.88	0.00	0.00	0.00
r112_50	0.00	0.90	0.00	0.00	0.00
rc101_50	0.00	0.56	0.00	0.00	0.00
rc102_50	0.00	0.61	0.00	0.00	0.00
rc103_50	0.00	0.67	0.00	0.00	0.00
rc104_50	0.00	0.80	0.00	0.00	0.00
rc105_50	0.00	0.57	0.00	0.00	0.00
rc106_50	0.00	0.57	0.00	0.00	0.00
rc107_50	0.83	0.65	0.00	0.00	0.00
rc108_50	0.00	0.76	0.00	0.00	0.00
c101_100	0.00	1.64	0.00	0.00	0.00
c102_100	0.00	1.88	0.00	0.00	0.00
c103_100	0.00	2.10	0.00	0.00	0.00
c104_100	0.95	2.29	0.00	0.00	0.00
c105_100	1.18	1.65	0.00	0.00	0.00
c106_100	1.76	1.70	0.00	0.00	0.00
c107_100	0.54	1.71	0.00	0.00	0.00
c108_100	0.00	1.89	0.00	0.00	0.00
c109_100	0.00	2.10	0.00	0.00	0.00
r101_100	0.00	1.37	0.00	0.00	0.00
r102_100	0.00	2.65	0.00	0.00	0.00
r103_100	0.00	2.88	0.00	0.00	0.00
r104_100	0.00	3.06	0.00	0.00	0.00
r105_100	0.00	2.28	0.00	0.00	0.00
r106_100	0.00	2.96	0.00	0.00	0.00
r107_100	0.74	3.29	0.67	0.67	0.67
r108_100	0.52	3.50	0.00	0.00	0.00

r109_100	0.00	2.70	0.00	0.00	0.00
r110_100	0.92	2.69	0.42	0.14	0.00
r111_100	0.74	3.04	0.00	0.00	0.00
r112_100	0.13	2.56	0.00	0.00	0.00
rc101_100	0.00	1.62	0.00	0.00	0.00
rc102_100	0.00	1.84	0.00	0.00	0.00
rc103_100	0.68	2.06	0.00	0.00	0.00
rc104_100	0.00	2.46	0.00	0.00	0.00
rc105_100	0.00	1.88	0.00	0.00	0.00
rc106_100	0.00	1.94	0.00	0.00	0.00
rc107_100	0.87	2.14	0.00	0.00	0.00
rc108_100	0.00	2.29	0.00	0.00	0.00
c201_100	0.46	8.62	0.00	0.00	0.00
c202_100	1.08	11.27	0.86	0.00	0.00
c203_100	1.25	12.57	0.42	0.00	0.00
c204_100	1.63	14.73	1.22	1.02	1.02
c205_100	1.1	9.52	0.88	0,00	0.00
c206_100	1.51	10.67	1.72	1.08	0.86
c207_100	1.08	11.56	0.86	0.86	0.43
c208_100	1.68	10.93	1.68	1.05	1.05
r201_100	1.73	15.73	1.53	1.13	0.60
r202_100	4.63	25.86	4.22	3.68	2.97
r203_100	2.31	33.42	4.11	2.10	1.68
r204_100	2.01	40.26	2.74	0.87	0.61
r205_100	3.17	22.45	2.31	1.24	0.82
r206_100	3.87	31.18	4.26	2.97	2.10
r207_100	2.48	34.77	2.91	1.79	1.77
r208_100	2.00	39.84	2.36	1.89	1.40
r209_100	2.40	22.91	2.44	1.92	1.31
r210_100	4.05	28.58	4.5	3.51	2.33
r211_100	1.82	29.08	2.16	2.05	1.28
rc201_100	0.96	13.32	0.70	0,00	0.00
rc202_100	4.10	20.10	4.66	2.99	1.84
rc203_100	2.59	26.64	2.59	1.91	1.73
rc204_100	2.26	30.42	2.60	1.30	1.02
rc205_100	1.72	17.06	1.37	0.77	0.61
rc206_100	2.99	15.71	2.41	1.72	1.23
rc207_100	4.17	21.09	4.23	2.79	2.08
rc208_100	2.43	24.07	2.22	1.39	1.29

Table A.4 Results of VNS* and VNS*C on Cordeau et al. (1997) instances

<i>Instance</i>	<i>n</i>	<i>VNS* (120 s)</i> <i>Gap%</i>	<i>VNS* (300s)</i> <i>Gap%</i>	<i>VNS*C (120 s)</i> <i>Gap%</i>	<i>VNS* C (300s)</i> <i>Gap%</i>
pr01	49	0.00	0.00	0.00	0.00
pr02	97	2.48	2.48	2.67	2.48
pr03	145	0.15	0.15	0.20	0.10
pr04	193	2.94	2.21	1.15	0.53
pr05	241	4.61	2.45	3.29	2.72
pr06	289	8.61	6.64	1.49	1.25
pr07	73	0.00	0.00	0.00	0.00
pr08	145	0.00	0.00	0.00	0.00
pr09	217	2.92	1.66	1.01	0.65
pr10	289	2.39	2.12	2.19	2.19
pr11	49	0.11	-0.11	0.57	0.40
pr12	97	0.95	0.90	0.54	0.77
pr13	145	0.74	0.09	0.39	-0.35
pr14	193	4.83	4.06	5.08	4.76
pr15	241	7.53	6.16	3.09	3.80
pr16	289	9.35	9.76	8.37	8.25
pr17	73	0.00	0.00	0.00	0.00
pr18	145	10.24	3.90	0.22	0.00
pr19	217	15.23	11.74	10.71	5.44
pr20	289	6.99	7.08	5.58	6.57

Table A.5 Results of VNS* on instances of Type A

<i>Instance</i>	<i># Cust</i>	<i># Ord</i>	<i>T_profit</i>	<i>T_distance</i>	<i>T_wait</i>	<i>Obj</i>			<i>Time(s)</i>
						<i>min</i>	<i>max</i>	<i>avg</i>	
c*101-50	11	20	3640.00	179.71	17.04	3339.01	3443.25	3409.42	1.95
c*102-50	11	24	3675.00	175.78	0.00	3452.11	3499.22	3475.97	1.96
c*103-50	12	16	3690.00	144.61	0.00	3411.54	3545.39	3472.17	2.64
c*104-50	12	18	3850.00	153.30	0.00	3596.12	3696.70	3658.70	2.42
c*105-50	11	23	3510.00	217.97	0.00	3193.50	3292.03	3256.88	2.00
c*106-50	12	22	3680.00	130.06	1.87	3492.05	3548.07	3517.83	1.95
c*107-50	11	17	3720.00	217.99	0.00	3473.93	3502.01	3485.40	1.93
c*108-50	11	23	3495.00	149.10	0.00	3309.92	3345.90	3332.68	2.16
c*109-50	11	16	3890.00	171.45	0.00	3667.26	3718.55	3697.17	1.90
c*201-50	32	84	10150.00	353.14	0.00	9796.86	9796.86	9796.86	12.65
c*202-50	34	84	9755.00	297.51	0.00	9424.61	9457.49	9445.76	9.53
c*203-50	34	83	10395.00	321.20	0.00	9933.09	10073.80	9967.46	11.99
c*204-50	34	86	10320.00	256.40	0.00	9955.95	10063.60	10008.26	18.84
c*205-50	32	79	10185.00	405.22	0.00	9748.87	9779.78	9772.55	14.15
c*206-50	33	82	10625.00	376.64	0.00	10241.24	10248.36	10245.51	12.84
c*207-50	34	79	10440.00	323.91	0.00	10011.48	10116.09	10035.89	14.37
c*208-50	34	86	10250.00	301.44	0.00	9899.52	9948.56	9933.53	9.89
r*101-50	5	13	2010.00	112.62	36.44	1860.94	1860.94	1860.94	0.61
r*102-50	8	21	3020.00	132.88	3.80	2873.68	2883.32	2875.61	1.28
r*103-50	9	22	3280.00	138.57	0.00	3113.45	3141.43	3124.64	1.32
r*104-50	10	25	3210.00	128.26	0.00	3051.78	3081.74	3075.75	1.76
r*105-50	6	12	2170.00	151.97	0.00	2018.03	2018.03	2018.03	0.64
r*106-50	8	20	3270.00	143.81	0.00	3032.49	3126.19	3069.97	1.24
r*107-50	10	26	3180.00	129.09	0.00	2987.95	3050.91	3016.45	1.55
r*108-50	10	17	3380.00	128.46	0.00	3188.81	3251.54	3212.65	2.18
r*109-50	8	21	2775.00	129.94	0.00	2645.06	2645.06	2645.06	1.10
r*110-50	9	23	3035.00	126.45	0.00	2908.55	2908.55	2908.55	1.16
r*111-50	10	23	3140.00	127.59	0.00	2996.99	3012.41	3006.24	1.31
r*112-50	10	25	3210.00	124.35	0.00	3085.65	3085.65	3085.65	1.66
r*201-50	26	62	6865.00	643.30	27.45	6124.12	6194.25	6159.65	7.73
r*202-50	35	82	8075.00	606.73	0.49	7222.26	7467.78	7344.99	9.87
r*203-50	40	95	9210.00	560.36	0.00	8516.38	8649.64	8568.84	11.71
r*204-50	44	103	9920.00	547.57	0.00	9269.03	9372.43	9337.64	12.59
r*205-50	35	80	8045.00	618.38	0.00	7300.12	7426.62	7358.35	12.06
r*206-50	38	90	9230.00	569.39	0.00	8451.32	8660.61	8538.67	13.75
r*207-50	41	98	9585.00	583.47	0.00	8916.20	9001.53	8949.59	12.79
r*208-50	48	112	10470.00	512.72	0.00	9811.24	9957.28	9875.43	12.36
r*209-50	38	90	8915.00	605.06	0.00	8246.64	8309.94	8288.82	14.58
r*210-50	41	102	8735.00	569.05	0.00	8056.59	8165.95	8113.07	15.08
r*211-50	42	100	9680.00	567.79	0.00	9001.62	9112.21	9061.65	16.85
rc*101-50	9	19	2800.00	121.47	0.00	2678.53	2678.53	2678.53	1.05
rc*102-50	7	15	3100.00	162.97	0.00	2933.90	2937.03	2935.78	1.03
rc*103-50	9	25	3205.00	148.83	0.00	3041.17	3056.17	3050.17	1.61
rc*104-50	10	20	3570.00	136.69	0.00	3355.14	3433.31	3407.27	1.74
rc*105-50	9	21	3450.00	144.35	0.00	3305.65	3305.65	3305.65	1.30
rc*106-50	9	16	3120.00	131.87	0.00	2986.68	2988.13	2987.26	4.52

rc*107-50	6	12	3350.00	175.68	0.00	3128.58	3174.32	3140.27	1.40
rc*108-50	10	21	3670.00	138.13	0.00	3518.49	3531.87	3526.52	1.65
rc*201-50	28	68	8900.00	642.64	0.00	8199.26	8257.36	8240.78	10.47
rc*202-50	32	75	11195.00	626.86	3.00	10429.40	10565.14	10496.56	7.26
rc*203-50	38	98	11710.00	557.20	0.00	11145.31	11152.80	11150.17	9.76
rc*204-50	45	120	13655.00	504.74	0.00	12968.75	13150.26	13091.23	15.19
rc*205-50	32	81	9990.00	619.55	0.82	9324.49	9369.63	9350.99	9.73
rc*206-50	33	85	10725.00	626.85	0.00	9829.83	10098.15	9971.97	9.46
rc*207-50	37	88	11715.00	585.10	0.00	10868.00	11129.90	11035.98	11.08
rc*208-50	41	104	13285.00	547.55	0.00	12467.19	12737.45	12681.52	17.24
c*101-100	10	16	3580.00	149.40	0.00	3381.35	3430.60	3403.29	3.34
c*102-100	11	18	3715.00	180.34	0.00	3512.33	3534.66	3525.25	4.40
c*103-100	12	17	3865.00	151.55	0.00	3616.78	3713.45	3678.97	5.44
c*104-100	12	16	3880.00	124.24	0.00	3698.61	3755.76	3739.26	4.58
c*105-100	11	14	3900.00	190.22	0.00	3579.52	3709.78	3649.24	3.20
c*106-100	11	20	3630.00	180.05	0.00	3331.20	3449.95	3383.85	3.04
c*107-100	11	12	3895.00	234.40	0.00	3480.82	3660.60	3591.81	3.70
c*108-100	11	21	3685.00	142.49	0.00	3367.49	3542.51	3456.84	3.79
c*109-100	11	15	3770.00	166.11	0.00	3519.03	3603.89	3553.13	4.21
c*201-100	32	76	11690.00	484.62	0.00	11091.85	11205.38	11141.80	21.15
c*202-100	31	66	12220.00	556.61	3.91	11438.65	11659.48	11525.89	27.87
c*203-100	32	63	12300.00	489.47	0.00	11575.62	11810.53	11698.78	31.69
c*204-100	32	68	12200.00	455.44	0.00	11673.62	11744.56	11702.81	36.38
c*205-100	30	62	11440.00	593.51	0.00	10675.56	10846.49	10761.30	25.06
c*206-100	32	63	12145.00	497.88	0.00	11532.21	11647.12	11604.90	28.14
c*207-100	32	67	12350.00	422.41	0.00	11663.28	11927.59	11835.41	27.96
c*208-100	32	68	11855.00	480.72	0.00	11264.62	11374.28	11307.26	27.77
r*101-100	9	24	2810.00	130.65	0.00	2679.35	2679.35	2679.35	2.02
r*102-100	12	21	3685.00	100.37	0.00	3547.63	3584.63	3576.73	3.84
r*103-100	12	26	3480.00	107.24	0.00	3308.21	3372.76	3342.80	4.39
r*104-100	12	24	3430.00	105.11	0.00	3197.60	3324.89	3260.69	5.15
r*105-100	10	24	3405.00	127.34	0.00	3256.88	3277.66	3269.35	2.89
r*106-100	11	30	3330.00	117.61	0.00	3178.70	3212.39	3198.91	3.89
r*107-100	13	28	3605.00	95.80	0.00	3387.89	3509.20	3436.94	4.84
r*108-100	13	24	3515.00	91.05	0.00	3312.69	3423.95	3384.51	4.74
r*109-100	10	19	3485.00	127.58	0.00	3254.22	3357.42	3298.29	3.58
r*110-100	10	21	3635.00	112.66	0.00	3432.85	3522.34	3470.73	4.33
r*111-100	13	21	3560.00	96.93	0.00	3345.64	3463.07	3400.39	4.70
r*112-100	12	21	3650.00	107.65	0.00	3472.69	3542.35	3502.94	3.92
r*201-100	37	93	11075.00	604.43	15.40	10288.65	10455.17	10365.40	33.35
r*202-100	47	115	12865.00	522.87	0.00	12156.18	12342.13	12251.89	47.79
r*203-100	54	131	14290.00	450.11	0.00	13492.39	13839.89	13671.62	64.95
r*204-100	55	128	15105.00	449.27	0.00	14536.08	14655.73	14598.36	80.43
r*205-100	45	109	13200.00	533.26	0.00	12162.76	12666.74	12534.27	46.22
r*206-100	50	130	14835.00	488.87	0.00	13765.74	14346.13	14097.16	62.19
r*207-100	54	131	15520.00	458.70	0.00	14633.10	15061.30	14749.04	72.39
r*208-100	58	151	15550.00	416.46	0.00	15011.77	15133.54	15069.67	86.85
r*209-100	47	112	13145.00	524.94	0.00	12418.86	12620.06	12499.68	54.62
r*210-100	49	126	13995.00	506.69	0.00	13319.17	13488.31	13416.51	60.99
r*211-100	52	122	14540.00	473.40	0.00	13561.53	14066.60	13829.84	64.38

rc*101-100	9	19	3300.00	141.59	0.00	3053.71	3158.41	3108.48	2.48
rc*102-100	10	20	3495.00	133.48	0.00	3281.42	3361.52	3313.65	2.93
rc*103-100	10	18	3525.00	132.88	0.00	3353.16	3392.12	3383.22	3.09
rc*104-100	10	21	3665.00	137.79	0.00	3441.45	3527.21	3490.51	3.84
rc*105-100	12	24	3530.00	116.84	0.00	3302.28	3413.16	3340.25	3.05
rc*106-100	11	22	3410.00	127.75	0.00	3256.47	3282.25	3271.94	3.52
rc*107-100	11	20	3450.00	119.36	0.00	3248.39	3330.64	3288.15	3.74
rc*108-100	10	17	3730.00	128.49	0.00	3527.49	3601.51	3549.03	15.70
rc*201-100	34	79	11420.00	618.39	0.00	10502.89	10801.61	10685.25	27.60
rc*202-100	42	104	13125.00	503.80	0.00	12273.08	12621.20	12439.46	42.11
rc*203-100	49	117	14845.00	469.32	0.00	14093.94	14375.68	14246.73	60.61
rc*204-100	51	113	15965.00	448.06	0.00	15191.14	15516.94	15390.09	69.33
rc*205-100	39	101	12385.00	565.61	0.00	11519.90	11819.39	11660.60	35.83
rc*206-100	37	93	13100.00	548.87	0.00	12053.70	12551.13	12415.66	35.53
rc*207-100	43	104	14890.00	500.12	0.00	13483.59	14389.88	13966.47	44.09
rc*208-100	46	109	15425.00	498.99	0.00	14662.80	14926.01	14793.71	56.30
c*101-200	10	21	3560.00	169.84	0.00	3316.87	3390.16	3361.65	4.72
c*102-200	12	16	3920.00	254.64	0.00	3620.00	3665.36	3641.51	8.47
c*103-200	13	18	3905.00	163.01	0.00	3668.63	3741.99	3711.66	8.29
c*104-200	13	21	3715.00	163.68	0.00	3527.86	3551.32	3543.63	8.43
c*105-200	11	19	3765.00	181.46	0.00	3472.28	3583.54	3544.15	5.15
c*106-200	11	15	3760.00	213.56	0.00	3414.38	3546.44	3482.82	6.27
c*107-200	11	17	3770.00	151.85	0.00	3494.65	3618.15	3553.90	8.39
c*108-200	12	18	3775.00	175.41	0.00	3513.16	3599.59	3554.18	7.21
c*109-200	11	20	3720.00	208.16	0.00	3473.16	3511.84	3497.72	7.22
c*110-200	12	19	3860.00	133.80	0.00	3603.52	3726.20	3660.69	7.69
c*201-200	31	69	12370.00	745.62	0.00	11468.85	11624.38	11550.20	44.28
c*202-200	33	66	12200.00	584.45	0.00	11484.98	11615.55	11557.00	63.76
c*203-200	35	63	12885.00	433.81	0.00	12127.58	12451.19	12281.11	82.11
c*204-200	35	56	13090.00	439.22	0.00	12482.18	12650.78	12564.27	83.92
c*205-200	34	77	11945.00	525.78	0.00	11320.29	11419.22	11365.86	51.75
c*206-200	34	71	12270.00	533.76	0.00	11562.96	11736.24	11650.69	54.07
c*207-200	32	71	12110.00	704.60	0.00	11153.41	11405.40	11274.45	60.97
c*208-200	34	66	12360.00	500.03	0.00	11705.94	11859.97	11770.05	68.93
c*209-200	34	64	12625.00	480.35	0.00	12062.19	12144.65	12098.76	68.37
c*210-200	33	61	12570.00	622.72	0.00	11808.19	11947.28	11883.61	69.69
r*101-200	13	17	3830.00	336.71	59.13	3380.64	3434.16	3411.14	7.82
r*102-200	22	24	3960.00	334.53	14.11	3502.18	3611.36	3556.53	15.31
r*103-200	19	20	3970.00	234.53	0.00	3588.38	3735.47	3689.63	17.69
r*104-200	21	21	4000.00	250.60	0.00	3696.59	3749.40	3726.74	21.17
r*105-200	19	22	3935.00	322.97	0.00	3534.78	3612.03	3574.51	9.54
r*106-200	19	21	3955.00	273.77	4.52	3567.86	3676.71	3617.49	16.08
r*107-200	23	25	3990.00	260.01	0.00	3641.10	3729.99	3675.62	20.75
r*108-200	23	26	3960.00	223.91	0.00	3670.03	3736.09	3702.31	26.82
r*109-200	22	28	3895.00	289.02	9.59	3536.65	3596.39	3577.82	13.62
r*110-200	23	27	3980.00	251.67	0.00	3630.42	3728.33	3685.31	18.33
r*201-200	67	116	17460.00	1632.96	0.00	15378.38	15827.04	15614.70	164.99
r*202-200	81	131	17685.00	1188.02	6.48	16299.18	16490.50	16410.02	286.53
r*203-200	97	148	18590.00	1171.94	1.16	17127.85	17416.90	17285.46	382.60
r*204-200	104	121	19350.00	1107.33	11.96	18060.10	18230.71	18159.74	577.94

r*205-200	72	122	18000.00	1278.39	38.89	16231.53	16682.72	16460.50	234.84
r*206-200	78	119	18280.00	1129.36	0.00	16839.25	17150.64	16972.11	346.07
r*207-200	94	140	18355.00	1140.36	7.47	17059.58	17207.17	17146.65	533.42
r*208-200	112	138	18825.00	1266.79	0.00	17361.44	17558.21	17475.09	556.60
r*209-200	79	124	18150.00	1419.20	12.64	16505.69	16718.16	16639.26	264.52
r*210-200	87	130	18660.00	1241.55	0.00	17070.90	17418.45	17220.63	365.17
rc*101-200	18	18	4000.00	318.91	28.28	3503.33	3652.81	3586.51	8.86
rc*102-200	22	24	3965.00	266.96	0.00	3569.52	3698.04	3636.57	16.14
rc*103-200	24	26	3975.00	220.87	0.00	3645.32	3754.13	3714.40	24.63
rc*104-200	27	29	3980.00	190.84	0.00	3731.59	3789.16	3756.56	25.91
rc*105-200	18	23	3900.00	247.84	0.00	3584.59	3652.16	3616.10	9.61
rc*106-200	19	23	3915.00	300.97	9.87	3536.04	3604.16	3575.99	12.46
rc*107-200	15	18	3900.00	212.31	0.00	3609.15	3687.69	3653.20	10.39
rc*108-200	20	22	3910.00	240.76	0.00	3509.16	3669.24	3590.89	13.59
rc*109-200	21	25	3910.00	258.27	6.75	3564.43	3644.98	3598.41	12.84
rc*110-200	21	21	4000.00	255.01	0.00	3649.62	3744.99	3695.55	14.03
rc*201-200	64	105	17350.00	1250.72	113.82	15724.45	15985.46	15857.47	145.84
rc*202-200	71	119	17580.00	902.32	29.30	16349.03	16648.38	16508.92	224.64
rc*203-200	93	138	18480.00	1137.56	6.71	17150.01	17335.73	17252.31	365.81
rc*204-200	113	134	19345.00	879.77	0.00	18232.45	18465.23	18339.01	511.34
rc*205-200	72	126	17765.00	1085.38	245.62	16017.59	16434.00	16188.15	191.49
rc*206-200	77	126	18220.00	1503.87	45.44	15936.17	16670.69	16156.93	177.54
rc*207-200	79	133	17795.00	1031.67	28.10	16428.51	16735.23	16617.15	244.09
rc*208-200	82	135	18530.00	992.84	2.03	17062.49	17535.13	17278.99	239.84
rc*209-200	79	134	18115.00	1027.63	0.00	16597.62	17087.37	16901.65	286.99
rc*210-200	81	134	18205.00	1037.28	0.00	16858.37	17167.72	17057.77	339.33

Table A.6 Results of VNS* on instances of Type B

<i>Instance</i>	<i># Cust</i>	<i># Ord</i>	<i>T_profit</i>	<i>T_distance</i>	<i>T_wait</i>	<i>Obj</i>			<i>Time(s)</i>
						<i>min</i>	<i>max</i>	<i>avg</i>	
c*101-50	11	20	3640.00	243.77	0.00	3302.14	3396.23	3359.56	4.54
c*102-50	11	20	3640.00	191.10	0.00	3301.05	3448.90	3366.54	5.69
c*103-50	11	17	3660.00	179.49	0.00	3416.62	3480.51	3460.51	5.61
c*104-50	12	16	3760.00	150.45	0.00	3523.49	3609.55	3554.87	5.68
c*105-50	11	21	3335.00	198.26	0.00	2945.03	3136.74	3047.37	5.77
c*106-50	12	19	3615.00	138.39	0.00	3187.71	3476.61	3345.20	6.06
c*107-50	11	20	3780.00	244.28	0.00	3443.24	3535.72	3463.91	4.69
c*108-50	11	23	3185.00	141.15	0.00	2969.56	3043.85	3005.48	5.93
c*109-50	10	10	3920.00	210.30	0.00	3600.46	3709.70	3638.66	4.33
c*201-50	33	73	9140.00	238.51	0.00	8718.87	8901.49	8844.47	66.61
c*202-50	35	80	8905.00	236.39	0.00	8469.44	8668.61	8583.53	80.89
c*203-50	32	75	9310.00	338.38	0.00	8334.64	8971.62	8571.22	80.39
c*204-50	34	80	9160.00	318.89	0.00	8528.26	8841.11	8652.01	118.23
c*205-50	32	77	9070.00	371.06	0.00	8605.12	8698.94	8652.44	82.64
c*206-50	34	77	9415.00	281.20	0.00	8562.76	9133.80	8901.89	72.56
c*207-50	33	76	9775.00	375.05	0.00	9100.99	9399.95	9235.63	62.44
c*208-50	34	79	8880.00	320.56	0.00	8319.29	8559.44	8438.04	100.85
r*101-50	5	13	2010.00	112.62	36.44	1734.85	1860.94	1810.50	0.78
r*102-50	8	20	3000.00	146.72	0.00	2748.36	2853.28	2769.34	1.63
r*103-50	10	23	3125.00	128.59	0.00	2972.33	2996.41	2977.15	3.68
r*104-50	10	24	3030.00	125.98	0.00	2887.15	2904.02	2890.52	2.68
r*105-50	6	11	2170.00	152.24	0.00	1947.08	2017.76	1988.10	0.87
r*106-50	9	23	3090.00	127.59	0.00	2797.79	2962.41	2893.58	2.45
r*107-50	9	23	2860.00	135.30	0.00	2634.49	2724.70	2688.04	3.53
r*108-50	10	15	3090.00	127.82	0.00	2917.49	2962.18	2933.45	3.52
r*109-50	8	21	2765.00	122.49	0.00	2206.73	2642.51	2378.35	2.36
r*110-50	9	22	2975.00	126.45	0.00	2848.55	2848.55	2848.55	3.92
r*111-50	9	22	2950.00	131.71	0.00	2606.45	2818.29	2733.97	2.48
r*112-50	10	25	2920.00	126.91	0.00	2393.25	2793.09	2568.59	3.55
r*201-50	27	65	6840.00	649.09	1.71	6144.45	6189.20	6163.58	16.07
r*202-50	34	82	8065.00	597.38	0.00	7326.20	7467.62	7377.00	29.48
r*203-50	39	90	9035.00	585.98	0.00	8352.75	8449.02	8406.67	38.48
r*204-50	45	104	9935.00	541.60	0.00	8904.30	9393.40	9165.77	49.94
r*205-50	34	80	7985.00	578.39	0.00	7293.17	7406.61	7343.95	28.94
r*206-50	38	90	9025.00	559.21	0.00	8365.06	8465.79	8413.17	39.85
r*207-50	41	98	9405.00	568.85	0.00	8426.80	8836.15	8696.96	45.57
r*208-50	47	109	10295.00	510.28	0.00	9634.35	9784.72	9705.66	55.56
r*209-50	37	88	8710.00	609.10	0.00	7759.14	8100.90	7937.44	39.33
r*210-50	41	102	8725.00	582.04	0.00	8088.37	8142.96	8115.18	50.52
r*211-50	42	100	9585.00	572.77	0.00	8883.94	9012.23	8958.54	51.65
rc*101-50	9	16	2670.00	125.87	0.00	2528.09	2544.13	2540.92	1.67
rc*102-50	8	17	2950.00	147.54	0.00	2632.18	2802.46	2695.46	1.50
rc*103-50	9	19	3050.00	143.24	0.00	2621.55	2906.76	2778.67	4.05
rc*104-50	10	18	3340.00	136.69	0.00	3167.74	3203.31	3181.97	4.81
rc*105-50	9	17	3240.00	144.35	0.00	2626.12	3095.65	2883.84	3.13
rc*106-50	10	23	2880.00	109.60	0.00	2618.60	2770.40	2735.54	2.21

rc*107-50	9	17	3145.00	142.84	0.00	2702.62	3002.16	2892.81	2.25
rc*108-50	10	18	3540.00	138.13	0.00	3074.13	3401.87	3263.98	4.00
rc*201-50	28	68	8845.00	609.86	0.00	8203.03	8235.14	8228.03	20.66
rc*202-50	31	74	11070.00	622.99	14.13	10267.24	10432.88	10331.13	26.12
rc*203-50	38	92	11505.00	564.85	0.00	10751.18	10940.15	10893.77	84.04
rc*204-50	45	116	12380.00	480.84	0.00	11517.02	11899.16	11810.78	403.41
rc*205-50	32	81	9990.00	621.37	0.00	9055.38	9368.63	9216.41	26.97
rc*206-50	33	83	10565.00	626.85	0.00	9764.52	9938.15	9851.41	45.91
rc*207-50	36	86	11525.00	592.38	0.00	10341.23	10932.62	10699.46	37.59
rc*208-50	41	102	12405.00	541.43	0.00	11221.78	11863.57	11577.15	108.04
c*101-100	10	16	3600.00	198.29	0.00	3266.38	3401.71	3340.83	4.94
c*102-100	11	17	3700.00	200.99	0.00	3462.25	3499.01	3481.44	5.50
c*103-100	12	20	3785.00	150.81	0.00	3579.47	3634.19	3610.76	6.54
c*104-100	12	12	3940.00	140.69	0.00	3393.84	3799.31	3571.71	8.34
c*105-100	11	17	3765.00	156.01	0.00	3430.69	3608.99	3549.51	5.87
c*106-100	11	18	3610.00	179.85	0.00	3148.73	3430.15	3280.89	6.09
c*107-100	10	10	3860.00	247.47	0.00	3532.28	3612.53	3575.50	4.76
c*108-100	11	15	3640.00	141.36	0.00	3319.82	3498.64	3406.44	6.06
c*109-100	12	19	3660.00	133.25	0.00	3368.08	3526.75	3491.05	6.34
c*201-100	32	70	11300.00	490.70	0.00	8728.08	10809.30	9723.10	108.21
c*202-100	30	56	9830.00	468.39	4.91	8954.17	9356.70	9088.20	98.27
c*203-100	32	72	11550.00	379.42	0.00	9192.08	11170.58	10575.78	112.06
c*204-100	33	76	10850.00	353.87	0.00	9322.43	10496.13	9595.85	151.28
c*205-100	30	67	10970.00	607.83	0.00	9889.40	10362.17	10231.36	74.38
c*206-100	32	80	10490.00	478.64	0.00	9706.34	10011.36	9869.54	96.33
c*207-100	33	78	11530.00	354.17	0.00	11011.11	11175.83	11082.93	94.00
c*208-100	31	73	11510.00	503.23	0.00	10441.27	11006.77	10783.72	122.61
r*101-100	7	19	2640.00	140.00	9.82	2328.38	2490.18	2376.87	2.95
r*102-100	12	20	3660.00	106.29	0.00	3539.63	3553.71	3545.10	5.93
r*103-100	11	20	3450.00	117.64	0.00	3201.00	3332.36	3284.49	9.50
r*104-100	11	22	3500.00	108.23	0.00	3054.56	3391.77	3138.91	9.84
r*105-100	10	19	3300.00	127.34	0.00	3172.66	3172.66	3172.66	5.73
r*106-100	11	26	3200.00	118.77	0.00	2887.47	3081.23	3018.75	7.45
r*107-100	11	18	3460.00	108.88	0.00	3190.53	3351.12	3283.34	8.23
r*108-100	12	18	3525.00	108.78	0.00	2840.62	3416.22	3149.56	8.29
r*109-100	11	19	3375.00	107.74	0.00	3041.71	3267.26	3133.48	6.37
r*110-100	10	19	3580.00	126.82	0.00	3283.85	3453.18	3401.08	7.82
r*111-100	13	23	3480.00	96.71	0.00	3195.03	3383.29	3258.03	8.26
r*112-100	11	16	3580.00	116.70	0.00	3281.43	3463.30	3370.87	8.58
r*201-100	39	94	10875.00	574.98	24.72	10115.33	10275.30	10207.95	101.63
r*202-100	46	109	12060.00	485.97	1.71	11375.71	11572.32	11440.57	258.77
r*203-100	50	114	13010.00	497.86	0.00	11673.57	12512.14	12004.02	403.00
r*204-100	56	122	13350.00	422.84	0.00	12578.34	12927.16	12795.13	389.31
r*205-100	44	102	12645.00	511.07	14.81	11661.05	12119.12	11862.17	127.17
r*206-100	53	131	13385.00	462.72	0.00	11315.13	12922.28	11714.36	293.55
r*207-100	52	120	14320.00	474.28	0.00	13619.16	13845.72	13749.64	377.68
r*208-100	57	132	13340.00	427.41	0.00	12350.92	12912.59	12528.49	529.63
r*209-100	45	106	11970.00	499.54	0.00	11196.08	11470.46	11290.18	344.55
r*210-100	50	114	12190.00	477.31	0.00	11568.32	11712.69	11638.64	413.84
r*211-100	52	118	13460.00	457.62	0.00	12088.83	13002.38	12561.29	246.23

rc*101-100	9	20	3105.00	141.59	0.00	2963.41	2963.41	2963.41	8.47
rc*102-100	10	17	3435.00	133.58	0.00	3061.60	3301.42	3180.86	5.43
rc*103-100	9	16	3380.00	146.48	0.00	3158.60	3233.52	3189.47	5.94
rc*104-100	11	17	3660.00	128.62	0.00	3271.85	3531.38	3419.50	5.13
rc*105-100	10	17	3420.00	133.78	0.00	3229.95	3286.22	3253.42	4.85
rc*106-100	11	19	3250.00	127.42	0.00	3112.01	3122.58	3119.18	4.33
rc*107-100	11	19	3220.00	117.21	0.00	2759.11	3102.79	2972.81	6.34
rc*108-100	10	17	3600.00	131.16	0.00	3080.20	3468.84	3315.84	7.64
rc*201-100	34	79	11035.00	610.25	0.00	10221.01	10424.75	10290.33	51.94
rc*202-100	43	98	12420.00	503.89	0.00	11275.74	11916.11	11650.52	220.24
rc*203-100	50	105	13100.00	450.13	0.00	11354.65	12649.87	12207.02	328.43
rc*204-100	53	105	12900.00	429.23	0.00	12137.66	12470.77	12291.93	479.46
rc*205-100	40	98	11610.00	554.30	0.00	10749.62	11055.70	10858.11	109.45
rc*206-100	38	93	12320.00	551.69	0.00	11074.85	11768.31	11355.36	119.33
rc*207-100	40	94	13975.00	529.38	0.00	13098.92	13445.62	13272.40	132.89
rc*208-100	46	86	12460.00	486.89	0.00	11631.78	11973.11	11796.20	339.45
c*101-200	10	17	3640.00	262.21	0.00	3313.59	3377.79	3351.96	8.27
c*102-200	12	17	3780.00	242.10	0.00	3456.16	3537.90	3484.96	10.46
c*103-200	13	22	3720.00	140.59	0.00	3472.88	3579.41	3520.30	11.37
c*104-200	12	18	3700.00	138.68	0.00	3176.86	3561.32	3419.55	11.87
c*105-200	10	10	3820.00	265.46	0.00	3407.24	3554.54	3465.66	9.45
c*106-200	11	15	3660.00	194.86	0.00	3332.82	3465.14	3398.99	8.97
c*107-200	11	18	3680.00	155.18	0.00	3452.21	3524.82	3495.85	8.83
c*108-200	11	20	3640.00	174.09	0.00	3312.10	3465.91	3402.55	10.31
c*109-200	11	18	3680.00	169.13	0.00	3150.79	3510.87	3375.96	10.28
c*110-200	11	15	3800.00	162.65	0.00	3479.58	3637.35	3576.31	12.02
c*201-200	32	71	10385.00	584.25	0.00	9364.24	9800.75	9507.54	97.82
c*202-200	32	69	11550.00	576.07	0.00	10908.31	10973.93	10934.66	119.32
c*203-200	34	77	10835.00	366.01	0.00	10350.22	10468.99	10428.78	246.94
c*204-200	36	72	10850.00	319.35	0.00	10067.76	10530.65	10424.17	231.64
c*205-200	32	69	10850.00	507.30	0.00	9196.34	10342.70	10084.75	159.85
c*206-200	34	77	11530.00	522.45	0.00	10371.87	11007.55	10614.21	165.32
c*207-200	33	75	11550.00	580.79	0.00	10666.46	10969.21	10801.51	139.32
c*208-200	34	79	10835.00	415.04	0.00	9831.48	10419.96	10288.38	166.76
c*209-200	34	72	10770.00	519.03	0.00	9902.59	10250.97	10041.25	141.01
c*210-200	35	76	11550.00	419.22	0.00	11086.09	11130.78	11112.23	190.26
r*101-200	14	23	3700.00	254.40	138.98	3040.04	3306.62	3108.37	15.06
r*102-200	20	23	3800.00	280.23	59.94	3258.41	3459.83	3377.90	22.79
r*103-200	19	19	4000.00	252.96	0.00	3276.91	3747.04	3473.37	20.03
r*104-200	20	23	3800.00	219.88	0.00	3303.69	3580.12	3409.04	23.10
r*105-200	15	18	3800.00	315.41	24.53	3396.84	3460.06	3433.22	12.78
r*106-200	11	15	3800.00	225.21	4.27	3461.24	3570.52	3518.27	17.15
r*107-200	21	26	3800.00	280.19	0.00	3142.64	3519.81	3434.83	24.18
r*108-200	16	24	3285.00	141.51	0.00	2943.46	3143.49	3045.48	20.62
r*109-200	17	28	3480.00	282.46	34.83	2892.64	3162.71	3066.87	26.06
r*110-200	13	20	3500.00	189.70	0.00	2869.63	3310.30	3044.70	24.23
r*201-200	66	117	16455.00	1477.39	50.34	14823.15	14927.27	14861.33	128.92*
r*202-200	89	145	16455.00	1555.53	32.86	13199.09	14866.61	14344.79	246.89*
r*203-200	89	141	14455.00	886.64	0.00	12872.67	13568.36	13404.81	332.43*
r*204-200	102	138	15455.00	1020.05	0.00	13481.96	14434.95	13874.99	389.10*

r*205-200	79	137	15955.00	1310.27	0.00	13023.05	14644.73	14274.96	166.28*
r*206-200	92	144	16455.00	1356.71	0.41	14600.55	15097.88	14804.14	279.27*
r*207-200	96	136	15955.00	1080.35	0.00	14641.88	14874.65	14729.72	322.69*
r*208-200	86	134	15955.00	786.44	0.00	14658.05	15168.56	15001.35	330.36*
r*209-200	79	125	16455.00	1337.98	0.00	15030.90	15117.02	15084.19	195.59*
r*210-200	70	123	16455.00	989.76	49.98	15217.85	15415.26	15308.36	284.16*
rc*101-200	22	25	3800.00	320.46	1.78	3261.98	3477.76	3332.10	24.77*
rc*102-200	14	20	3500.00	194.44	38.15	3136.57	3267.41	3171.99	20.49
rc*103-200	21	24	3800.00	238.34	0.00	3453.30	3561.66	3509.66	27.16
rc*104-200	20	24	3800.00	151.33	0.00	3258.19	3648.67	3481.16	100.10
rc*105-200	19	26	3800.00	309.46	2.81	3365.14	3487.73	3421.89	19.63
rc*106-200	19	22	3800.00	277.59	47.87	2890.82	3474.54	3184.91	23.64
rc*107-200	15	28	3455.00	241.19	6.36	2856.50	3207.45	2945.45	21.46
rc*108-200	21	26	3700.00	271.29	8.78	3089.25	3419.93	3271.98	20.12
rc*109-200	16	26	3700.00	209.68	0.00	3319.77	3490.32	3408.99	23.86
rc*110-200	19	25	3800.00	244.60	16.06	3219.14	3539.34	3436.61	25.17
rc*201-200	67	121	13455.00	1514.17	137.39	11755.29	11803.44	11792.58	123.73*
rc*202-200	78	133	14455.00	1063.08	5.20	13069.98	13386.72	13240.15	204.13*
rc*203-200	85	140	13955.00	767.20	0.00	13046.55	13187.80	13126.17	365.37*
rc*204-200	89	149	16455.00	554.78	0.00	13754.47	15900.22	15430.38	386.07*
rc*205-200	76	116	15455.00	1450.27	32.43	12982.61	13972.30	13390.40	153.65*
rc*206-200	68	126	14350.00	1201.21	79.88	12919.10	13068.91	12988.25	156.91*
rc*207-200	83	145	15955.00	1212.29	0.00	12768.23	14742.71	13220.47	200.56*
rc*208-200	86	134	15955.00	1125.37	15.77	13106.57	14813.86	13799.22	226.86*
rc*209-200	86	139	15455.00	962.09	0.00	14094.95	14492.91	14251.55	224.38*
rc*210-200	91	147	15955.00	1180.22	0.00	13539.21	14774.78	13926.57	286.75*

time(s)*: aborted prematurely with out-of-memory status in the solution
based on the Linear Multiple Knapsack Problem with Conflicts

Table A.7 Results of VNS* on instances of Type C

<i>Instance</i>	<i># Cust</i>	<i># Ord</i>	<i>T_profit</i>	<i>T_distance</i>	<i>T_wait</i>	<i>Obj</i>			<i>Time(s)</i>
						<i>min</i>	<i>max</i>	<i>avg</i>	
c*101-50	11	16	3500.00	209.91	0.00	3218.23	3290.09	3264.89	1.77
c*102-50	12	24	3290.00	124.17	0.00	3121.45	3165.83	3148.14	2.35
c*103-50	11	20	3485.00	178.60	0.00	3234.75	3306.40	3262.10	2.75
c*104-50	12	22	3470.00	124.83	0.00	3246.08	3345.17	3306.19	2.88
c*105-50	11	23	3380.00	149.38	0.00	3099.24	3230.62	3149.18	2.92
c*106-50	11	26	3400.00	197.19	0.00	3132.17	3202.81	3162.30	2.70
c*107-50	12	23	3415.00	153.22	0.00	3186.52	3261.78	3236.95	2.09
c*108-50	12	20	3480.00	154.96	0.00	3095.07	3325.04	3255.26	2.22
c*109-50	12	20	3500.00	120.47	0.00	3312.26	3379.53	3346.90	2.18
c*201-50	31	78	9660.00	399.77	0.00	8629.39	9260.23	8992.34	14.42
c*202-50	34	86	9200.00	290.91	0.00	8652.23	8909.09	8743.54	16.03
c*203-50	33	78	9895.00	383.48	0.00	9309.18	9511.52	9428.26	18.45
c*204-50	34	84	9670.00	309.98	0.00	8988.26	9360.02	9175.78	25.38
c*205-50	31	75	9695.00	392.35	0.00	8950.98	9302.65	9169.78	16.59
c*206-50	33	83	10080.00	359.91	0.00	9181.12	9720.09	9339.54	16.17
c*207-50	34	83	10030.00	299.11	0.00	9424.38	9730.89	9655.06	17.13
c*208-50	34	84	10150.00	312.03	0.00	9494.75	9837.97	9607.70	15.97
r*101-50	5	13	2010.00	112.62	36.44	1860.94	1860.94	1860.94	0.63
r*102-50	9	25	2975.00	132.51	0.00	2767.04	2842.49	2821.65	1.25
r*103-50	10	23	3230.00	128.90	0.00	2987.70	3101.10	3035.86	1.63
r*104-50	10	27	3210.00	128.26	0.00	2940.97	3081.74	3002.80	1.89
r*105-50	6	12	2170.00	151.97	0.00	2018.03	2018.03	2018.03	0.74
r*106-50	8	20	3150.00	142.12	4.84	3003.04	3003.04	3003.04	1.39
r*107-50	10	26	3180.00	129.09	0.00	2968.48	3050.91	3007.83	1.78
r*108-50	10	19	3350.00	126.87	0.00	3142.60	3223.13	3188.71	1.73
r*109-50	8	21	2775.00	129.94	0.00	2645.06	2645.06	2645.06	1.17
r*110-50	9	23	3035.00	126.45	0.00	2908.55	2908.55	2908.55	1.40
r*111-50	10	25	3125.00	128.01	0.00	2889.06	2996.99	2937.61	1.57
r*112-50	10	26	3180.00	126.91	0.00	2877.35	3053.09	2998.52	1.76
r*201-50	25	61	6825.00	612.75	2.05	6071.66	6210.20	6110.82	8.58
r*202-50	34	80	7890.00	604.62	0.00	6966.36	7285.38	7124.53	11.66
r*203-50	38	90	8955.00	590.43	0.00	7930.62	8364.57	8121.59	13.62
r*204-50	45	103	9760.00	533.78	6.68	8587.64	9219.54	8858.14	16.32
r*205-50	33	78	8020.00	586.91	0.00	7168.29	7433.09	7292.39	13.85
r*206-50	41	94	9025.00	583.44	0.00	7992.29	8441.56	8139.85	14.61
r*207-50	41	99	9310.00	555.91	0.00	8451.49	8754.09	8547.52	16.04
r*208-50	45	108	10280.00	524.42	0.00	9725.53	9755.58	9741.94	15.66
r*209-50	37	89	8855.00	578.39	0.00	7778.49	8276.61	8065.92	17.29
r*210-50	38	96	8550.00	525.46	0.00	7590.61	8024.54	7859.55	17.37
r*211-50	41	96	9455.00	544.42	0.00	8660.31	8910.58	8834.09	19.96
rc*101-50	9	19	2800.00	121.47	0.00	2678.53	2678.53	2678.53	1.07
rc*102-50	7	15	3100.00	162.97	0.00	2802.46	2937.03	2884.38	1.12
rc*103-50	9	25	3205.00	148.83	0.00	2863.07	3056.17	2983.00	1.66
rc*104-50	10	21	3375.00	138.38	0.00	3162.23	3236.62	3196.20	1.89
rc*105-50	9	22	3135.00	140.71	0.00	2776.20	2994.29	2874.41	1.41
rc*106-50	9	17	3115.00	133.32	0.00	2918.60	2981.68	2952.15	1.40

rc*107-50	9	18	3260.00	143.39	0.00	3076.51	3116.61	3096.68	1.69
rc*108-50	10	20	3310.00	134.37	0.00	3101.33	3175.63	3137.86	1.86
rc*201-50	27	67	8835.00	633.89	0.00	8131.27	8201.11	8164.95	11.05
rc*202-50	32	77	10780.00	616.17	0.26	9618.32	10163.57	9928.97	9.82
rc*203-50	37	95	10855.00	550.46	0.00	10295.39	10304.54	10301.20	14.56
rc*204-50	45	121	13415.00	493.02	0.00	11750.78	12921.98	12284.15	22.66
rc*205-50	30	75	9795.00	613.30	12.78	8749.99	9168.92	8902.43	10.64
rc*206-50	32	81	10555.00	632.76	0.00	9078.09	9922.24	9550.89	11.61
rc*207-50	36	88	11205.00	578.75	0.00	10470.13	10626.25	10556.46	15.92
rc*208-50	39	100	12785.00	535.52	0.00	11572.53	12249.48	12044.71	23.27
c*101-100	11	18	3445.00	167.25	0.00	3075.60	3277.75	3167.30	3.09
c*102-100	11	19	3500.00	121.55	1.00	3260.97	3377.45	3316.96	3.95
c*103-100	12	26	3440.00	129.64	0.00	3229.84	3310.36	3286.59	4.09
c*104-100	12	22	3490.00	113.51	0.00	3298.29	3376.49	3341.74	4.59
c*105-100	11	22	3455.00	160.39	0.00	3263.42	3294.61	3282.49	2.99
c*106-100	11	23	3410.00	168.53	0.00	3159.77	3241.47	3203.27	3.21
c*107-100	10	15	3465.00	133.31	0.00	3236.40	3331.69	3268.68	3.39
c*108-100	11	21	3455.00	148.84	0.00	3187.52	3306.16	3274.18	4.04
c*109-100	12	22	3500.00	110.90	0.00	3343.36	3389.10	3370.18	3.80
c*201-100	32	79	11230.00	476.32	0.00	10347.71	10753.68	10590.39	22.60
c*202-100	33	74	11545.00	399.02	0.00	9813.81	11145.98	10254.90	27.49
c*203-100	31	70	10900.00	463.33	0.00	9894.12	10436.67	10163.18	33.73
c*204-100	33	67	11465.00	367.10	0.00	10413.02	11097.90	10630.04	42.84
c*205-100	31	75	10320.00	394.23	31.78	9544.31	9893.99	9745.87	31.54
c*206-100	32	73	10960.00	478.50	0.00	9808.09	10481.50	10207.42	29.93
c*207-100	32	68	11585.00	412.40	0.00	10547.75	11172.60	10909.71	32.53
c*208-100	32	72	11285.00	410.66	0.00	10337.76	10874.34	10551.81	33.45
r*101-100	9	24	2810.00	130.65	0.00	2679.35	2679.35	2679.35	2.15
r*102-100	11	20	3465.00	113.72	0.00	3305.64	3351.28	3318.95	3.82
r*103-100	12	25	3340.00	106.43	0.00	3123.19	3233.57	3175.77	4.59
r*104-100	12	22	3400.00	108.72	0.00	3183.00	3291.28	3212.27	4.92
r*105-100	9	20	3300.00	122.15	0.00	3121.24	3177.85	3155.44	3.31
r*106-100	12	27	3130.00	106.49	0.00	2680.93	3023.51	2907.85	4.28
r*107-100	12	22	3460.00	92.70	0.00	3254.72	3367.30	3300.98	4.16
r*108-100	13	26	3445.00	91.06	0.00	3160.22	3353.94	3226.01	5.30
r*109-100	12	25	3380.00	100.16	0.00	3125.66	3279.84	3215.76	3.88
r*110-100	13	29	3310.00	97.20	0.00	3128.80	3212.80	3178.82	4.67
r*111-100	13	24	3420.00	90.81	0.00	3238.52	3329.19	3288.00	4.34
r*112-100	12	25	3395.00	99.08	0.00	3217.55	3295.92	3261.39	4.60
r*201-100	38	96	10925.00	606.55	4.16	9683.40	10314.29	10137.39	39.13
r*202-100	49	117	12065.00	503.68	0.00	10683.22	11561.32	11209.11	61.67
r*203-100	55	134	13320.00	439.51	0.00	11940.55	12880.49	12349.65	75.38
r*204-100	57	139	14360.00	429.00	0.00	13542.38	13931.00	13817.59	94.14
r*205-100	47	111	12405.00	521.24	0.00	11350.69	11883.76	11535.33	56.40
r*206-100	50	125	13800.00	487.42	0.00	12215.79	13312.58	12733.12	72.79
r*207-100	52	130	14945.00	474.19	0.00	12968.63	14470.81	13976.22	85.97
r*208-100	56	142	14145.00	439.32	0.00	13133.09	13705.68	13462.81	101.99
r*209-100	46	108	12485.00	533.28	0.00	11507.73	11951.72	11787.17	64.21
r*210-100	48	121	12655.00	507.82	0.00	11215.50	12147.18	11759.07	66.70
r*211-100	48	114	13710.00	492.79	0.00	12042.97	13217.21	12654.31	73.50

rc*101-100	10	23	3120.00	110.08	0.00	3009.92	3009.92	3009.92	2.78
rc*102-100	9	18	3410.00	117.69	0.00	3192.96	3292.31	3243.51	3.01
rc*103-100	10	21	3390.00	133.88	0.00	3166.78	3256.12	3215.93	3.29
rc*104-100	11	19	3430.00	125.76	0.00	3250.79	3304.24	3274.66	4.36
rc*105-100	10	18	3465.00	130.05	0.00	3253.78	3334.95	3299.93	3.28
rc*106-100	11	24	3410.00	127.42	0.00	3155.68	3282.58	3213.22	3.40
rc*107-100	11	19	3465.00	128.28	0.00	3200.68	3336.72	3241.82	3.71
rc*108-100	12	27	3350.00	110.11	0.00	3037.41	3239.89	3160.11	3.87
rc*201-100	33	79	10900.00	575.47	0.00	9747.32	10324.53	10079.38	32.57
rc*202-100	41	104	12615.00	538.33	2.14	11307.92	12074.53	11701.10	50.27
rc*203-100	47	112	13585.00	489.14	0.00	12688.62	13095.86	12930.21	67.50
rc*204-100	53	128	14770.00	424.43	0.00	13137.47	14345.57	14014.64	77.28
rc*205-100	39	96	12000.00	568.83	0.00	10549.46	11431.17	11062.06	37.65
rc*206-100	39	94	12345.00	510.61	0.00	10790.59	11834.39	11347.27	42.00
rc*207-100	42	106	13065.00	536.62	0.00	10423.06	12528.38	11127.07	47.40
rc*208-100	47	112	14690.00	475.71	0.00	12540.63	14214.29	13437.30	62.62
c*101-200	11	17	3475.00	175.85	0.00	3114.51	3299.15	3197.08	5.00
c*102-200	13	22	3500.00	147.02	0.00	3279.50	3352.98	3311.04	7.85
c*103-200	13	24	3435.00	158.49	0.00	3219.75	3276.51	3250.96	8.59
c*104-200	13	25	3495.00	124.18	0.00	3347.62	3370.82	3356.91	8.87
c*105-200	11	24	3430.00	216.61	0.00	3151.11	3213.39	3187.17	5.37
c*106-200	10	16	3480.00	141.03	0.00	3176.39	3338.97	3262.93	5.79
c*107-200	12	22	3480.00	147.98	0.00	3254.02	3332.02	3297.61	6.47
c*108-200	12	22	3490.00	161.30	0.00	3214.67	3328.70	3257.96	6.59
c*109-200	12	19	3485.00	188.81	0.00	3226.14	3296.19	3271.46	7.28
c*110-200	11	19	3490.00	128.37	0.00	3309.09	3361.63	3342.37	8.74
c*201-200	33	81	10480.00	553.81	0.00	9244.01	9926.19	9563.48	46.97
c*202-200	32	69	11610.00	624.12	0.00	9615.53	10985.88	10163.61	63.32
c*203-200	35	66	11930.00	393.28	0.00	10902.44	11536.72	11114.64	85.93
c*204-200	35	59	11320.00	389.38	0.00	10314.20	10930.62	10653.13	83.63
c*205-200	33	76	11005.00	575.44	0.00	9940.51	10429.56	10227.35	54.25
c*206-200	33	65	10990.00	546.79	0.00	9977.51	10443.21	10238.01	71.22
c*207-200	35	72	10905.00	446.10	0.00	9769.37	10458.90	10076.81	66.40
c*208-200	35	75	11990.00	433.22	0.00	10271.07	11556.78	11010.38	66.40
c*209-200	32	71	10905.00	564.82	0.00	9880.93	10340.18	10149.04	70.56
c*210-200	34	67	10980.00	447.46	0.00	9891.45	10532.54	10280.25	75.76
r*101-200	15	26	3500.00	245.81	106.90	2985.73	3147.29	3098.33	8.44
r*102-200	18	29	3500.00	234.82	16.09	3197.40	3249.09	3236.62	12.43
r*103-200	12	22	3500.00	178.13	0.00	3246.74	3321.87	3283.78	14.89
r*104-200	16	22	3500.00	185.02	0.00	3139.00	3314.98	3260.14	18.79
r*105-200	13	20	3500.00	255.99	5.84	3100.17	3238.17	3163.79	9.23
r*106-200	14	23	3500.00	193.29	74.83	3164.69	3231.88	3209.06	16.01
r*107-200	16	26	3500.00	226.74	0.00	3194.28	3273.26	3232.11	17.46
r*108-200	18	24	3425.00	175.48	0.00	3172.45	3249.52	3223.37	17.28
r*109-200	16	29	3500.00	263.39	58.98	3134.90	3177.63	3162.10	12.40
r*110-200	22	35	3500.00	243.52	0.00	3192.94	3256.48	3227.51	16.74
r*201-200	68	119	16540.00	1430.28	103.12	14519.79	15006.60	14757.54	186.91
r*202-200	87	145	17445.00	1393.31	81.92	15635.90	15969.77	15856.43	321.13
r*203-200	82	140	17500.00	1035.16	35.94	16236.74	16428.90	16334.29	426.14
r*204-200	79	129	17500.00	828.00	0.00	16407.53	16672.00	16526.38	598.13

r*205-200	74	132	17350.00	1394.97	44.12	15158.64	15910.91	15569.38	234.21
r*206-200	80	135	17500.00	1103.82	55.08	15949.98	16341.10	16122.16	388.51
r*207-200	89	147	17465.00	948.95	22.94	16238.12	16493.11	16400.44	554.68
r*208-200	87	140	17500.00	825.97	6.14	16450.88	16667.89	16556.77	566.36
r*209-200	83	128	17415.00	1341.00	17.21	15988.30	16056.79	16034.51	315.48
r*210-200	75	135	17500.00	1207.05	66.56	16142.74	16226.39	16188.50	383.78
rc*101-200	20	25	3500.00	218.60	1.11	3121.93	3280.29	3220.00	9.30
rc*102-200	16	19	3500.00	202.08	25.67	3156.75	3272.25	3226.99	15.53
rc*103-200	20	29	3500.00	185.95	0.00	3229.28	3314.05	3285.88	20.90
rc*104-200	21	34	3500.00	143.21	0.00	3285.70	3356.79	3323.71	24.29
rc*105-200	15	24	3500.00	204.17	15.61	3246.78	3280.22	3259.35	9.28
rc*106-200	12	21	3500.00	222.14	0.00	3188.50	3277.86	3242.25	10.88
rc*107-200	18	32	3500.00	192.35	22.21	3194.86	3285.44	3246.66	10.78
rc*108-200	18	30	3465.00	190.71	0.00	3080.73	3274.29	3192.77	13.17
rc*109-200	18	21	3500.00	163.04	0.00	3303.13	3336.96	3319.80	12.08
rc*110-200	16	28	3500.00	186.82	0.00	3237.11	3313.18	3286.98	14.21
rc*201-200	66	104	16985.00	1403.54	91.37	14799.61	15490.09	15165.26	146.06
rc*202-200	77	130	17500.00	1105.15	136.94	15983.41	16257.91	16060.67	270.26
rc*203-200	86	141	17500.00	919.10	87.33	16305.10	16493.57	16422.52	477.60
rc*204-200	82	137	17500.00	659.93	62.43	16647.85	16777.64	16724.58	627.00
rc*205-200	70	138	17270.00	1181.81	154.26	15140.84	15933.93	15612.17	223.79
rc*206-200	70	123	17375.00	1535.75	94.09	15439.39	15745.16	15602.28	208.97
rc*207-200	86	139	17500.00	1250.38	17.69	16011.37	16231.93	16142.45	290.45
rc*208-200	86	144	17370.00	1284.35	22.00	15723.64	16063.65	15916.58	373.62
rc*209-200	80	135	17405.00	1071.96	3.57	16103.09	16329.47	16215.07	301.63
rc*210-200	87	149	17495.00	900.11	5.65	16130.65	16589.24	16263.23	402.11

Table A.8 Results of VNS*I on instances of Type C

<i>Instance</i>	<i># Cust</i>	<i># Ord</i>	<i>T_profit</i>	<i>T_distance</i>	<i>T_wait</i>	<i>Obj</i>			<i>Time(s)</i>
						<i>min</i>	<i>max</i>	<i>avg</i>	
c*101-50	12	26	3390.00	129.42	0.00	3200.09	3260.58	3221.58	2.92
c*102-50	12	29	3330.00	110.96	0.00	3095.83	3219.04	3138.43	4.17
c*103-50	11	20	3485.00	178.60	0.00	3241.52	3306.40	3267.80	3.79
c*104-50	12	22	3455.00	145.85	0.00	3257.45	3309.15	3286.32	3.98
c*105-50	11	23	3320.00	149.38	0.00	3079.24	3170.62	3111.21	5.01
c*106-50	11	26	3400.00	197.19	0.00	3035.96	3202.81	3134.42	4.17
c*107-50	11	26	3370.00	128.19	0.00	3210.34	3241.81	3224.98	3.24
c*108-50	11	21	3435.00	147.73	0.00	3096.21	3287.27	3176.63	4.65
c*109-50	11	22	3480.00	119.21	0.00	3308.72	3360.79	3330.12	3.28
c*201-50	31	77	9610.00	399.77	0.00	8629.39	9210.23	8982.34	519.73
c*202-50	34	83	9340.00	287.63	0.00	8312.23	9052.37	8581.86	518.80
c*203-50	33	81	9835.00	375.71	0.00	9025.19	9459.29	9224.26	522.71
c*204-50	34	85	9660.00	309.98	0.00	8833.72	9350.02	9100.85	529.93
c*205-50	31	75	9695.00	392.35	0.00	8950.98	9302.65	9153.78	522.22
c*206-50	33	80	9900.00	359.91	0.00	9181.12	9540.09	9279.54	521.71
c*207-50	34	83	10030.00	299.11	0.00	9349.04	9730.89	9623.99	522.62
c*208-50	34	85	10090.00	312.03	0.00	9284.75	9777.97	9541.70	520.90
r*101-50	5	13	2010.00	112.62	36.44	1860.94	1860.94	1860.94	1.42
r*102-50	9	25	2975.00	132.51	0.00	2747.04	2842.49	2817.65	3.85
r*103-50	10	24	3205.00	129.13	0.00	3012.08	3075.87	3040.67	4.31
r*104-50	10	28	3125.00	128.58	0.00	2912.59	2996.42	2962.35	8.50
r*105-50	6	12	2170.00	151.97	0.00	2018.03	2018.03	2018.03	1.59
r*106-50	8	20	3150.00	142.12	4.84	3003.04	3003.04	3003.04	2.96
r*107-50	10	26	3115.00	127.05	0.00	2950.91	2987.95	2967.83	8.38
r*108-50	10	19	3300.00	126.87	0.00	3128.62	3173.13	3143.76	2.76
r*109-50	8	21	2775.00	129.94	0.00	2645.06	2645.06	2645.06	3.81
r*110-50	9	23	2985.00	138.07	0.00	2768.55	2846.93	2785.22	4.37
r*111-50	10	24	3095.00	128.01	0.00	2862.38	2966.99	2913.61	3.73
r*112-50	10	26	3140.00	124.35	0.00	2867.82	3015.65	2953.12	5.48
r*201-50	25	61	6825.00	612.75	2.05	6071.66	6210.20	6110.82	46.88
r*202-50	34	80	7890.00	604.62	0.00	6966.36	7285.38	7124.53	488.56
r*203-50	38	90	8955.00	590.43	0.00	7930.62	8364.57	8121.59	518.44
r*204-50	45	103	9760.00	533.78	6.68	8587.64	9219.54	8880.40	522.92
r*205-50	33	78	8020.00	586.91	0.00	7168.29	7433.09	7292.39	326.76
r*206-50	41	94	9025.00	583.44	0.00	7992.29	8441.56	8139.85	519.10
r*207-50	42	100	9320.00	557.24	0.00	8451.49	8762.76	8549.25	523.05
r*208-50	46	109	10340.00	531.49	0.00	9725.53	9808.51	9752.53	523.08
r*209-50	37	89	8855.00	578.39	0.00	7778.49	8276.61	8065.92	521.34
r*210-50	38	96	8550.00	525.46	0.00	7590.61	8024.54	7859.55	522.52
r*211-50	41	96	9455.00	544.42	0.00	8660.31	8910.58	8834.09	526.19
rc*101-50	9	19	2800.00	121.47	0.00	2678.53	2678.53	2678.53	2.39
rc*102-50	7	15	3100.00	162.97	0.00	2802.46	2937.03	2876.38	2.45
rc*103-50	9	24	3175.00	148.83	0.00	2863.07	3026.17	2968.00	3.56
rc*104-50	8	21	3405.00	157.59	0.00	3162.82	3247.41	3200.55	3.09
rc*105-50	9	22	3095.00	140.71	0.00	2776.20	2954.29	2866.41	6.66
rc*106-50	9	20	3115.00	136.45	0.00	2920.92	2978.55	2948.12	2.39

rc*107-50	9	19	3235.00	139.82	0.00	3022.69	3095.18	3064.79	2.58
rc*108-50	10	26	3280.00	135.79	0.00	3099.57	3144.21	3123.32	3.22
rc*201-50	27	67	8835.00	633.89	0.00	8131.27	8201.11	8164.95	204.07
rc*202-50	32	77	10780.00	616.33	0.00	9618.32	10163.67	9948.97	514.69
rc*203-50	37	93	10715.00	550.46	0.00	10155.39	10164.54	10161.20	519.94
rc*204-50	45	121	13415.00	493.02	0.00	11750.78	12921.98	12284.15	533.02
rc*205-50	30	75	9795.00	613.30	12.78	8749.99	9168.92	8902.43	514.62
rc*206-50	32	80	10495.00	632.76	0.00	9078.09	9862.24	9538.89	515.93
rc*207-50	36	87	11145.00	584.28	0.00	10470.13	10560.72	10525.28	520.94
rc*208-50	39	100	12785.00	535.52	0.00	11512.53	12249.48	12026.71	530.68
c*101-100	11	18	3445.00	167.25	0.00	3088.64	3277.75	3163.49	6.26
c*102-100	12	23	3440.00	108.98	0.00	3240.97	3331.02	3301.22	6.79
c*103-100	12	26	3440.00	129.64	0.00	3282.72	3310.36	3294.69	5.42
c*104-100	12	18	3500.00	124.76	0.00	3295.08	3375.24	3332.32	5.47
c*105-100	12	26	3410.00	99.54	0.00	3223.41	3310.46	3268.53	4.11
c*106-100	11	24	3325.00	111.70	0.00	3187.66	3213.30	3204.05	5.28
c*107-100	11	19	3430.00	165.96	0.00	3197.04	3264.04	3234.56	4.38
c*108-100	10	19	3500.00	195.74	0.00	3238.58	3304.26	3274.59	5.01
c*109-100	12	21	3500.00	122.55	0.00	3251.79	3377.45	3327.86	5.03
c*201-100	32	78	11000.00	476.32	0.00	9947.71	10523.68	10275.39	527.21
c*202-100	33	76	11190.00	401.10	0.00	9813.81	10788.90	10116.86	532.45
c*203-100	30	70	10665.00	467.06	0.00	9894.12	10197.94	10051.18	538.49
c*204-100	33	70	11350.00	371.73	0.00	10186.35	10978.27	10480.11	540.51
c*205-100	33	73	10225.00	404.68	0.00	9508.21	9820.32	9634.47	431.58
c*206-100	32	75	10570.00	483.72	0.00	9798.09	10086.28	9931.41	533.85
c*207-100	31	68	11200.00	407.15	0.00	10315.82	10792.85	10600.50	535.54
c*208-100	33	74	11140.00	410.72	0.00	10067.76	10729.28	10390.09	534.64
r*101-100	9	24	2810.00	130.65	0.00	2679.35	2679.35	2679.35	7.06
r*102-100	10	22	3425.00	111.44	0.00	3274.13	3313.56	3294.95	4.86
r*103-100	13	33	3265.00	91.19	0.00	3083.46	3173.81	3125.38	51.54
r*104-100	12	21	3300.00	108.72	0.00	3045.25	3191.28	3137.84	12.58
r*105-100	10	24	3265.00	127.34	0.00	3111.24	3137.66	3127.98	5.64
r*106-100	11	27	3060.00	104.80	0.00	2660.93	2955.20	2831.81	12.32
r*107-100	13	25	3360.00	94.08	0.00	3242.00	3265.92	3252.16	7.62
r*108-100	13	26	3445.00	91.06	0.00	3155.26	3353.94	3206.01	9.17
r*109-100	12	24	3380.00	105.67	0.00	3125.66	3274.33	3187.65	6.62
r*110-100	13	31	3340.00	97.12	0.00	3102.17	3242.88	3160.15	10.03
r*111-100	12	19	3460.00	103.21	0.00	3205.25	3356.79	3276.39	8.69
r*112-100	13	24	3415.00	88.99	0.00	3175.35	3326.01	3239.72	16.16
r*201-100	38	95	10795.00	558.80	10.33	9683.40	10225.87	10052.48	544.11
r*202-100	49	117	12065.00	503.68	0.00	10453.22	11561.32	11163.11	570.41
r*203-100	55	134	13320.00	439.51	0.00	11840.55	12880.49	12329.65	585.72
r*204-100	57	139	14340.00	429.00	0.00	13542.38	13911.00	13805.59	600.87
r*205-100	46	106	12075.00	510.83	0.00	11258.97	11564.17	11428.30	561.74
r*206-100	50	125	13800.00	487.42	0.00	12215.79	13312.58	12733.12	583.93
r*207-100	52	129	14855.00	474.19	0.00	12968.63	14380.81	13958.22	598.39
r*208-100	56	142	14145.00	439.32	0.00	13083.09	13705.68	13452.81	616.95
r*209-100	46	108	12485.00	533.28	0.00	11227.73	11951.72	11731.17	571.16
r*210-100	48	121	12655.00	507.82	0.00	10955.50	12147.18	11653.07	575.18
r*211-100	49	114	13920.00	496.31	0.00	12424.47	13423.69	12848.96	581.82

rc*101-100	9	20	3075.00	144.03	0.00	2879.40	2930.97	2914.67	5.11
rc*102-100	10	20	3440.00	127.50	0.00	3182.96	3312.50	3231.96	3.99
rc*103-100	10	22	3380.00	133.88	0.00	3175.47	3246.12	3215.13	5.40
rc*104-100	11	19	3405.00	119.01	0.00	3142.27	3285.99	3207.97	6.87
rc*105-100	10	18	3465.00	130.05	0.00	3213.17	3334.95	3285.57	4.35
rc*106-100	11	24	3380.00	127.42	0.00	3125.68	3252.58	3176.84	4.48
rc*107-100	12	24	3395.00	116.60	0.00	3144.74	3278.40	3224.82	10.33
rc*108-100	12	27	3350.00	110.11	0.00	3007.41	3239.89	3134.64	5.54
rc*201-100	34	76	10835.00	607.85	3.98	9747.32	10223.17	10021.54	536.71
rc*202-100	41	104	12615.00	538.33	2.14	11307.92	12074.53	11619.10	556.55
rc*203-100	45	109	13595.00	509.02	0.00	12688.62	13085.98	12902.21	575.03
rc*204-100	54	128	14715.00	412.91	0.00	13137.47	14302.09	13874.64	585.60
rc*205-100	39	98	11710.00	556.55	0.00	10549.46	11153.45	10956.51	542.30
rc*206-100	39	94	12345.00	510.61	0.00	10763.46	11834.39	11301.84	547.48
rc*207-100	45	111	13505.00	506.58	0.00	10659.51	12998.42	11500.63	554.12
rc*208-100	47	111	14630.00	475.71	0.00	12020.63	14154.29	13285.30	569.69
c*101-200	11	24	3390.00	140.85	17.72	3112.30	3231.43	3159.71	6.15
c*102-200	13	29	3405.00	135.87	0.00	3182.07	3269.13	3243.58	9.62
c*103-200	13	22	3460.00	161.68	0.00	3208.05	3298.32	3262.52	11.61
c*104-200	13	23	3490.00	139.58	0.00	3301.64	3350.42	3327.19	10.55
c*105-200	11	21	3460.00	169.02	0.00	3125.64	3290.98	3188.38	6.73
c*106-200	10	17	3460.00	141.03	0.00	3135.31	3318.97	3207.93	6.66
c*107-200	12	22	3445.00	143.78	0.00	3248.46	3301.22	3274.18	7.66
c*108-200	12	22	3490.00	161.30	0.00	3191.49	3328.70	3238.33	7.54
c*109-200	12	21	3480.00	160.88	0.00	3136.14	3319.12	3234.75	8.51
c*110-200	12	18	3500.00	142.20	0.00	3289.09	3357.80	3334.64	10.30
c*201-200	31	72	10880.00	682.97	6.09	9164.09	10190.94	9638.86	500.33
c*202-200	32	70	11365.00	624.12	0.00	9615.53	10740.88	10113.92	445.57
c*203-200	35	65	11810.00	393.28	0.00	10407.44	11416.72	10837.64	575.40
c*204-200	35	60	11150.00	389.38	0.00	10252.68	10760.62	10518.80	573.45
c*205-200	33	74	10910.00	607.42	0.00	9470.51	10302.58	10017.46	555.59
c*206-200	33	61	10890.00	546.79	0.00	9957.51	10343.21	10164.01	570.64
c*207-200	34	67	10725.00	472.12	0.00	9602.50	10252.88	9979.83	475.54
c*208-200	35	81	11710.00	435.80	0.00	10001.07	11274.20	10754.31	554.51
c*209-200	35	69	11030.00	440.60	0.00	10043.72	10589.40	10226.11	522.68
c*210-200	34	70	10910.00	447.46	0.00	9891.45	10462.54	10176.25	575.87
r*101-200	15	23	3500.00	255.35	96.96	2946.62	3147.69	3027.53	28.24
r*102-200	19	33	3410.00	239.26	1.65	3071.55	3169.09	3123.57	41.05
r*103-200	15	22	3500.00	204.24	2.10	3239.29	3293.66	3256.52	18.26
r*104-200	27	42	3470.00	279.98	0.00	3137.93	3190.02	3158.75	209.49
r*105-200	13	18	3500.00	261.93	15.00	3125.37	3223.07	3182.33	10.90
r*106-200	22	33	3500.00	289.28	11.73	3152.95	3198.99	3172.44	49.88
r*107-200	19	31	3490.00	221.94	10.03	3191.93	3258.03	3221.87	24.40
r*108-200	20	27	3500.00	225.89	0.00	3184.72	3274.11	3227.73	209.33
r*109-200	18	33	3470.00	297.73	20.24	3118.56	3152.03	3133.37	36.78
r*110-200	22	35	3500.00	290.75	0.00	3091.33	3209.25	3160.60	205.66
r*201-200	69	121	16310.00	1432.87	103.12	14509.79	14774.01	14605.03	689.60
r*202-200	90	149	17395.00	1397.19	50.20	15427.17	15947.61	15731.16	823.50
r*203-200	87	134	17475.00	1038.92	2.33	16236.18	16433.75	16334.77	942.89
r*204-200	87	135	17470.00	862.20	0.00	16429.80	16607.80	16520.75	1121.18

r*205-200	73	145	17105.00	1377.03	72.06	14813.64	15655.91	15207.38	739.40
r*206-200	84	140	17455.00	1164.24	16.20	15859.98	16274.56	16060.07	901.05
r*207-200	92	148	17335.00	942.63	0.00	16102.17	16392.37	16286.48	1057.21
r*208-200	92	145	17435.00	859.91	0.00	16446.00	16575.09	16503.81	1086.50
r*209-200	79	134	17440.00	1395.05	33.82	15730.46	16011.13	15890.10	820.70
r*210-200	81	144	17435.00	1264.17	0.00	16013.63	16170.83	16097.06	896.84
rc*101-200	19	30	3480.00	209.49	20.10	3079.02	3250.41	3171.96	53.08
rc*102-200	15	25	3480.00	246.59	13.41	3067.12	3220.00	3132.09	215.79
rc*103-200	24	39	3485.00	248.95	0.00	3054.54	3236.05	3146.70	204.00
rc*104-200	23	37	3490.00	186.40	0.00	3148.48	3303.60	3255.80	171.94
rc*105-200	15	24	3500.00	204.17	15.61	3194.36	3280.22	3249.80	14.29
rc*106-200	16	27	3500.00	249.41	0.00	3118.01	3250.59	3188.32	25.57
rc*107-200	14	20	3500.00	181.78	0.00	3151.92	3318.22	3229.44	123.45
rc*108-200	19	34	3430.00	243.27	9.33	3048.55	3177.40	3125.59	238.15
rc*109-200	19	31	3475.00	184.77	0.00	3224.76	3290.23	3251.71	17.37
rc*110-200	13	23	3500.00	173.08	0.00	3230.49	3326.92	3261.66	54.28
rc*201-200	66	106	16745.00	1428.83	66.08	14472.35	15250.09	14860.81	648.23
rc*202-200	80	134	17335.00	1117.51	94.58	15786.88	16122.91	15947.22	775.64
rc*203-200	92	154	17485.00	972.15	30.20	16355.26	16482.65	16403.28	1006.54
rc*204-200	100	156	17495.00	703.25	0.00	16659.65	16791.75	16715.89	1177.03
rc*205-200	70	139	17115.00	1181.81	154.26	15030.84	15778.93	15479.14	728.80
rc*206-200	70	123	17375.00	1535.75	94.09	15329.39	15745.16	15471.92	718.11
rc*207-200	85	140	17465.00	1250.36	17.69	15937.42	16196.95	16074.35	794.50
rc*208-200	85	147	17295.00	1192.79	36.89	15755.29	16065.32	15921.72	887.61
rc*209-200	90	149	17425.00	1204.83	22.73	15920.51	16197.44	16040.05	811.47
rc*210-200	85	152	17370.00	888.11	37.65	16116.16	16444.24	16213.47	916.16

Table A.9 Results of VNS*II on instances of Type C

<i>Instance</i>	<i># Cust</i>	<i># Ord</i>	<i>T_profit</i>	<i>T_distance</i>	<i>T_wait</i>	<i>Obj</i>			<i>Time(s)</i>
						<i>min</i>	<i>max</i>	<i>avg</i>	
c*101-50	12	24	3330.00	129.42	0.00	3153.56	3200.58	3170.18	1.18
c*102-50	12	26	3240.00	110.96	0.00	3074.84	3129.04	3099.44	1.64
c*103-50	12	21	3340.00	113.37	0.00	3191.85	3226.63	3211.28	1.97
c*104-50	12	22	3410.00	130.70	0.00	3239.42	3279.30	3257.86	2.08
c*105-50	11	24	3285.00	201.09	0.00	3053.48	3083.91	3075.66	1.26
c*106-50	11	20	3270.00	165.01	0.00	3047.41	3104.99	3070.75	1.43
c*107-50	12	23	3370.00	153.22	0.00	3181.81	3216.78	3196.43	1.39
c*108-50	10	21	3360.00	136.84	0.00	3055.07	3223.16	3166.26	1.39
c*109-50	12	21	3420.00	114.83	0.00	3276.16	3305.17	3289.71	1.58
c*201-50	31	71	8525.00	397.50	0.00	7981.48	8127.50	8033.86	11.42
c*202-50	34	75	8640.00	287.63	0.00	8067.49	8352.37	8233.78	10.11
c*203-50	33	73	9370.00	400.82	0.00	8865.21	8969.18	8922.61	12.60
c*204-50	34	79	9110.00	276.42	0.00	8595.74	8833.58	8692.81	17.57
c*205-50	32	69	8360.00	347.94	0.00	7795.08	8012.06	7909.98	12.79
c*206-50	33	70	8140.00	387.44	0.00	7647.60	7752.56	7688.94	12.69
c*207-50	34	73	9050.00	298.33	0.00	8613.42	8751.67	8658.20	13.46
c*208-50	33	79	9165.00	307.52	0.00	8743.90	8857.48	8816.36	10.08
r*101-50	5	13	2010.00	112.62	36.44	1860.94	1860.94	1860.94	0.41
r*102-50	9	25	2975.00	132.51	0.00	2767.04	2842.49	2821.65	0.92
r*103-50	10	25	3150.00	127.85	0.00	2928.88	3022.15	2964.35	1.14
r*104-50	10	25	3150.00	128.26	0.00	2940.97	3021.74	2982.80	1.27
r*105-50	6	12	2170.00	151.97	0.00	2018.03	2018.03	2018.03	0.50
r*106-50	8	20	3150.00	142.12	4.84	3003.04	3003.04	3003.04	1.09
r*107-50	10	25	3130.00	129.09	0.00	2918.48	3000.91	2962.80	1.20
r*108-50	10	19	3300.00	126.87	0.00	3126.79	3173.13	3145.57	1.18
r*109-50	8	21	2775.00	129.94	0.00	2645.06	2645.06	2645.06	0.89
r*110-50	9	23	3035.00	126.45	0.00	2868.55	2908.55	2888.55	0.98
r*111-50	10	24	3080.00	128.01	0.00	2889.06	2951.99	2918.61	1.13
r*112-50	10	25	3150.00	124.35	0.00	2827.82	3025.65	2943.12	1.09
r*201-50	25	58	6725.00	580.01	34.79	5994.25	6110.20	6060.75	7.57
r*202-50	33	78	7690.00	598.43	2.34	6862.32	7089.23	6996.95	9.87
r*203-50	38	84	8470.00	538.39	0.00	7629.43	7931.61	7782.93	11.04
r*204-50	43	93	9220.00	565.91	0.00	8462.70	8654.09	8589.06	12.56
r*205-50	33	76	7810.00	550.61	0.00	7133.85	7259.39	7219.94	12.12
r*206-50	41	89	8725.00	583.44	0.00	7752.47	8141.56	7907.05	12.56
r*207-50	41	88	8550.00	579.67	0.00	7828.50	7970.33	7913.39	12.29
r*208-50	46	97	8815.00	538.44	0.00	8074.00	8276.56	8179.10	11.58
r*209-50	37	82	8415.00	578.39	0.00	7596.22	7836.61	7733.09	15.21
r*210-50	38	92	8265.00	542.51	0.00	7434.94	7722.49	7592.70	14.41
r*211-50	42	93	9125.00	570.46	0.00	8419.89	8554.54	8499.46	16.47
rc*101-50	9	19	2800.00	121.47	0.00	2678.53	2678.53	2678.53	0.72
rc*102-50	7	15	3100.00	162.97	0.00	2802.46	2937.03	2872.38	0.77
rc*103-50	9	24	3165.00	148.83	0.00	2871.05	3016.17	2959.31	1.09
rc*104-50	10	23	3320.00	125.25	0.00	3137.87	3194.75	3163.50	1.20
rc*105-50	9	22	3065.00	140.71	0.00	2776.20	2924.29	2821.69	0.90
rc*106-50	9	19	3075.00	136.45	0.00	2911.02	2938.55	2923.15	0.96

rc*107-50	9	16	3240.00	138.20	0.00	3045.75	3101.80	3064.19	1.04
rc*108-50	10	25	3275.00	136.78	0.00	3071.33	3138.22	3105.74	1.29
rc*201-50	27	69	8785.00	634.97	0.00	7990.03	8150.03	8110.64	9.90
rc*202-50	34	73	9790.00	600.30	5.70	8960.09	9184.00	9089.42	8.32
rc*203-50	38	89	10610.00	555.49	0.00	9466.92	10054.51	9736.87	10.47
rc*204-50	45	103	10900.00	506.51	0.00	10241.18	10393.49	10306.18	13.81
rc*205-50	30	72	9435.00	613.30	12.78	8398.47	8808.92	8631.28	9.25
rc*206-50	32	73	10035.00	632.76	0.00	9098.45	9402.24	9270.83	9.30
rc*207-50	33	76	10670.00	609.10	0.00	9830.19	10060.90	9929.00	12.70
rc*208-50	40	92	11255.00	525.39	0.00	10590.36	10729.61	10659.90	17.23
c*101-100	11	20	3315.00	136.79	3.00	3027.76	3175.21	3095.94	2.43
c*102-100	11	18	3425.00	121.63	1.00	3219.76	3302.37	3266.60	3.09
c*103-100	11	21	3440.00	173.62	0.00	3221.24	3266.38	3248.61	3.57
c*104-100	12	21	3425.00	113.51	0.00	3281.05	3311.49	3300.56	3.69
c*105-100	10	19	3385.00	154.27	0.00	3187.38	3230.73	3210.33	2.41
c*106-100	11	23	3340.00	168.53	0.00	3096.45	3171.47	3136.10	2.51
c*107-100	11	20	3420.00	177.42	0.00	3187.27	3242.58	3211.15	2.65
c*108-100	12	21	3410.00	153.73	0.00	3171.04	3256.27	3223.57	3.20
c*109-100	11	20	3415.00	115.58	0.00	3232.60	3299.42	3267.23	3.21
c*201-100	30	64	9755.00	594.24	0.00	9003.44	9160.76	9062.85	18.63
c*202-100	33	72	11480.00	401.10	0.00	9255.41	11078.90	9687.46	24.63
c*203-100	33	70	9915.00	358.56	0.00	9418.54	9556.44	9502.70	30.41
c*204-100	33	70	10095.00	366.24	0.00	9670.21	9728.76	9706.07	38.50
c*205-100	32	72	9995.00	433.22	0.00	9469.12	9561.78	9515.26	25.88
c*206-100	32	64	9785.00	435.17	0.00	9262.32	9349.83	9292.63	25.81
c*207-100	33	70	10025.00	382.48	0.00	9578.11	9642.52	9620.48	28.39
c*208-100	33	67	10040.00	409.08	0.00	9545.58	9630.92	9595.16	29.99
r*101-100	9	24	2810.00	130.65	0.00	2679.35	2679.35	2679.35	1.67
r*102-100	11	24	3380.00	117.90	0.00	3241.28	3262.10	3253.95	3.19
r*103-100	11	23	3275.00	117.00	0.00	3062.59	3158.00	3110.78	3.73
r*104-100	12	23	3365.00	108.72	0.00	3081.29	3256.28	3151.46	4.17
r*105-100	11	26	3255.00	115.71	0.00	3081.24	3139.29	3114.91	2.59
r*106-100	12	24	3035.00	108.33	0.00	2625.06	2926.67	2792.57	3.36
r*107-100	12	28	3360.00	99.34	0.00	3187.88	3260.66	3229.24	3.50
r*108-100	12	25	3260.00	92.77	0.00	3091.13	3167.23	3124.56	4.34
r*109-100	12	23	3325.00	105.67	0.00	3095.33	3219.33	3157.59	3.07
r*110-100	12	24	3255.00	89.20	0.00	3068.94	3165.80	3129.89	3.65
r*111-100	12	23	3375.00	102.55	0.00	3203.52	3272.45	3251.18	3.72
r*112-100	13	25	3335.00	88.99	0.00	3201.85	3246.01	3224.99	3.88
r*201-100	37	88	10335.00	564.39	20.34	9611.47	9750.27	9662.33	35.34
r*202-100	48	100	10660.00	508.55	4.16	9902.73	10147.29	10043.90	52.78
r*203-100	51	110	11890.00	485.99	0.00	11258.88	11404.01	11323.21	63.99
r*204-100	55	117	13120.00	448.15	0.00	12451.58	12671.85	12548.98	75.20
r*205-100	45	99	11910.00	545.49	0.00	10977.25	11364.51	11222.17	50.20
r*206-100	50	105	11390.00	493.69	0.00	10351.13	10896.31	10569.16	61.77
r*207-100	52	111	11745.00	471.81	0.00	10992.66	11273.19	11087.32	72.71
r*208-100	57	122	11975.00	429.40	0.00	11342.35	11545.60	11452.80	85.31
r*209-100	45	105	11750.00	535.82	0.00	10885.58	11214.18	11068.72	57.10
r*210-100	49	110	11200.00	509.07	0.00	10482.63	10690.93	10585.72	57.62
r*211-100	53	112	12410.00	446.25	0.00	11369.47	11963.75	11641.73	65.17

rc*101-100	10	23	3100.00	110.08	0.00	2959.92	2989.92	2980.35	2.23
rc*102-100	10	19	3385.00	137.18	0.00	3170.97	3247.82	3196.43	2.46
rc*103-100	10	20	3295.00	127.67	0.00	3121.90	3167.33	3149.01	2.74
rc*104-100	11	18	3380.00	125.76	0.00	3217.63	3254.24	3243.33	3.56
rc*105-100	9	18	3375.00	142.81	0.00	3178.78	3232.19	3207.06	2.50
rc*106-100	11	22	3360.00	127.42	0.00	3095.68	3232.58	3156.64	2.65
rc*107-100	11	20	3360.00	128.28	0.00	3138.40	3231.72	3187.14	3.08
rc*108-100	12	27	3280.00	110.11	0.00	2941.92	3169.89	3085.59	3.05
rc*201-100	34	75	10755.00	607.85	3.98	9571.53	10143.17	9839.65	29.38
rc*202-100	41	94	11045.00	544.74	0.00	10093.27	10500.26	10372.34	43.73
rc*203-100	48	101	12365.00	479.29	0.00	11341.58	11885.71	11627.85	59.63
rc*204-100	52	111	13295.00	435.94	0.00	12291.64	12859.06	12525.09	66.28
rc*205-100	38	88	11110.00	558.36	0.00	10206.48	10551.64	10455.75	33.73
rc*206-100	40	95	11330.00	545.73	0.00	10472.21	10784.27	10643.90	37.21
rc*207-100	42	92	11200.00	537.56	0.00	10208.42	10662.44	10460.15	42.05
rc*208-100	47	99	12610.00	475.71	0.00	11618.61	12134.29	11941.46	54.71
c*101-200	11	21	3305.00	174.84	0.54	3057.56	3129.62	3086.73	4.34
c*102-200	13	25	3390.00	135.87	0.00	3199.80	3254.13	3231.24	7.19
c*103-200	13	23	3400.00	160.39	0.00	3199.08	3239.61	3216.60	7.90
c*104-200	11	20	3445.00	120.97	0.00	3317.72	3324.03	3320.24	8.18
c*105-200	11	22	3310.00	181.16	0.00	3098.86	3128.84	3110.18	4.79
c*106-200	10	20	3400.00	156.57	0.00	3148.31	3243.43	3197.88	5.03
c*107-200	12	22	3395.00	148.93	0.00	3212.02	3246.07	3234.01	5.59
c*108-200	11	21	3365.00	154.71	0.00	3165.15	3210.29	3184.37	5.87
c*109-200	13	27	3360.00	120.62	0.00	3156.14	3239.38	3200.75	6.59
c*110-200	13	27	3425.00	135.59	0.00	3252.14	3289.41	3271.43	8.02
c*201-200	34	70	9560.00	522.12	0.00	8985.41	9037.88	9021.62	41.36
c*202-200	32	62	11500.00	624.12	0.00	9420.58	10875.88	9759.66	58.33
c*203-200	35	66	10060.00	366.58	0.00	9658.68	9693.42	9675.08	81.17
c*204-200	36	69	10105.00	344.82	0.00	9695.58	9760.18	9735.82	79.91
c*205-200	34	71	10025.00	479.09	0.00	9362.89	9545.91	9443.75	49.94
c*206-200	35	70	10010.00	407.75	0.00	9548.56	9602.25	9569.26	65.55
c*207-200	35	71	10060.00	438.94	0.00	9503.11	9621.06	9542.83	60.77
c*208-200	35	77	11660.00	433.22	0.00	9637.66	11226.78	10472.55	61.57
c*209-200	35	69	10040.00	440.60	0.00	9448.94	9599.40	9544.49	65.64
c*210-200	35	63	10105.00	415.08	0.00	9592.04	9689.92	9645.25	70.72
r*101-200	13	23	2850.00	240.36	32.73	2485.85	2576.91	2519.00	7.56
r*102-200	19	27	3410.00	252.87	0.00	3040.90	3157.13	3106.03	11.39
r*103-200	25	30	3445.00	254.85	0.00	2678.33	3190.15	3073.33	14.08
r*104-200	21	26	3445.00	187.24	0.00	2656.38	3257.76	2999.50	17.83
r*105-200	14	23	3395.00	256.01	0.00	3091.71	3138.99	3110.23	8.61
r*106-200	22	31	3435.00	233.74	0.00	3145.56	3201.26	3166.73	14.95
r*107-200	24	28	3445.00	254.47	0.00	3079.97	3190.53	3138.17	16.46
r*108-200	24	28	3440.00	221.68	0.00	2708.10	3218.32	2994.01	16.43
r*109-200	21	28	3420.00	287.73	15.68	3086.27	3116.59	3100.65	11.45
r*110-200	22	30	3445.00	250.41	0.00	3132.48	3194.59	3156.40	15.68
r*201-200	65	119	14260.00	1387.33	35.84	12601.54	12836.83	12718.95	161.22
r*202-200	80	113	14270.00	1209.29	30.28	12897.27	13030.43	12954.46	268.23
r*203-200	88	126	14400.00	954.33	0.00	13396.90	13445.67	13428.53	373.49
r*204-200	103	135	14425.00	880.61	0.00	13483.99	13544.39	13504.21	552.05

r*205-200	80	127	14305.00	1343.87	20.38	12730.73	12940.75	12836.63	194.74
r*206-200	90	130	14310.00	1204.30	0.00	12974.38	13105.70	13022.53	328.52
r*207-200	89	131	14435.00	939.50	0.00	13421.71	13495.50	13466.89	475.39
r*208-200	94	126	14435.00	764.54	0.00	13585.40	13670.46	13641.85	505.01
r*209-200	86	130	14350.00	1366.11	25.82	12740.41	12958.07	12875.59	274.22
r*210-200	88	124	14365.00	1307.62	0.47	12967.71	13056.91	13024.20	340.33
rc*101-200	19	28	3415.00	219.03	10.40	2618.25	3185.57	3019.98	8.38
rc*102-200	15	19	3435.00	215.13	0.00	3113.63	3219.87	3167.30	14.43
rc*103-200	22	29	3420.00	189.85	0.00	3211.71	3230.15	3220.85	19.44
rc*104-200	24	31	3430.00	145.59	0.00	3262.18	3284.41	3270.63	22.65
rc*105-200	16	23	3445.00	236.44	6.49	3159.82	3202.07	3188.19	8.50
rc*106-200	20	28	3430.00	225.48	0.00	3091.40	3204.52	3174.80	9.85
rc*107-200	15	24	3410.00	160.38	0.00	3138.63	3249.62	3182.39	9.84
rc*108-200	22	30	3425.00	261.38	0.00	2654.29	3163.62	2856.91	12.13
rc*109-200	20	28	3425.00	238.28	1.78	2657.05	3184.94	3040.36	11.07
rc*110-200	18	25	3435.00	195.96	0.00	3199.57	3239.04	3215.11	13.27
rc*201-200	67	107	14380.00	1507.84	126.41	12633.70	12745.75	12686.65	129.33
rc*202-200	86	135	14305.00	1170.76	1.41	13010.00	13132.83	13076.46	225.18
rc*203-200	94	137	14395.00	863.98	0.00	13432.57	13531.02	13475.55	408.31
rc*204-200	90	126	14450.00	583.18	0.00	13740.32	13866.82	13792.78	567.88
rc*205-200	81	118	14265.00	1423.33	11.97	12687.49	12829.70	12745.90	182.77
rc*206-200	81	134	14035.00	1420.10	14.36	12397.89	12600.54	12505.72	175.82
rc*207-200	79	125	14380.00	1114.72	0.00	12976.95	13265.28	13118.84	235.45
rc*208-200	88	143	14055.00	1061.68	0.00	12636.01	12993.32	12872.37	306.98
rc*209-200	86	135	14305.00	1116.87	24.63	12947.26	13163.50	13091.87	258.21
rc*210-200	92	139	14320.00	881.42	0.00	13261.04	13438.58	13343.56	343.12

Table A.10 Results of the column generation heuristic on MCMCm-VRPTW instances (Type C)

<i>Instance</i>	<i>Cost_1</i>	<i>Cost_3</i>	<i> K </i>	<i>Costs Gap%</i>	<i>Time(s)</i>
c*101-50	475,94	449,82	6	5,81	50,64
c*102-50	607,67	549,76	7	10,53	525,96
c*103-50	587,10	540,46	6	8,63	87,90
c*104-50	560,27	485,39	6	15,43	682,82
c*105-50	503,81	474,04	6	6,28	170,26
c*106-50	523,63	523,63	6	0,00	130,08
c*107-50	621,46	492,01	6	26,31	388,88
c*108-50	565,10	494,37	7	14,31	531,90
c*109-50	564,55	524,65	7	7,61	502,01
c*201-50	849,05	849,05	6	0,00	130,68
c*202-50	723,11	723,11	5	0,00	142,87
c*203-50	915,57	652,71	3	40,27	3806,32
c*204-50	713,23	628,53	4	13,48	4203,45
c*205-50	940,59	882,46	4	6,59	268,82
c*206-50	900,12	900,12	7	0,00	218,95
c*207-50	776,65	712,72	3	8,97	2696,07
c*208-50	723,16	709,88	6	1,87	235,53
r*101-50	1207,02	1148,62	12	5,08	40,30
r*102-50	1097,47	1002,74	10	9,45	126,71
r*103-50	930,15	850,32	9	9,39	160,46
r*104-50	756,30	697,19	7	8,48	360,80
r*105-50	1071,29	1030,21	9	3,99	72,77
r*106-50	959,65	824,07	8	16,45	172,53
r*107-50	830,76	731,94	7	13,50	200,72
r*108-50	712,46	646,09	6	10,27	314,94
r*109-50	910,81	838,97	8	8,56	112,69
r*110-50	874,91	755,46	8	15,81	195,28
r*111-50	888,50	743,34	7	19,53	261,44
r*112-50	793,93	650,49	6	22,05	270,55
r*201-50	1183,13	1183,13	5	0,00	1240,63
r*202-50	1084,73	1084,73	6	0,00	197,05
r*203-50	1047,37	901,77	4	16,15	1503,09
r*204-50	824,79	726,46	2	13,54	2436,25
r*205-50	1044,29	1044,29	2	0,00	141,66
r*206-50	968,29	968,29	5	0,00	3534,46
r*207-50	996,76	866,43	5	15,04	3242,05
r*208-50	755,22	669,04	3	12,88	3596,01
r*209-50	936,23	936,23	4	0,00	159,82
r*210-50	1045,33	1002,96	3	4,22	2438,43
r*211-50	873,71	770,54	3	13,39	3925,61
rc*101-50	1013,08	1013,08	9	0,00	93,83
rc*102-50	969,45	960,65	9	0,92	188,34
rc*103-50	918,25	918,25	8	0,00	350,38
rc*104-50	659,89	659,89	7	0,00	90,86
rc*105-50	1029,46	979,74	9	5,07	128,17
rc*106-50	830,29	830,29	7	0,00	138,82
rc*107-50	837,19	837,19	7	0,00	146,01

rc*108-50	807,62	807,62	7	0,00	325,95
rc*201-50	1282,57	1282,57	6	0,00	2037,57
rc*202-50	1371,52	1307,00	4	4,94	2079,93
rc*203-50	1145,00	1115,93	3	2,61	2249,43
rc*204-50	994,39	814,45	5	22,09	1695,12
rc*205-50	1224,00	1224,00	5	0,00	2359,35
rc*206-50	1068,91	1036,94	4	3,08	2127,61
rc*207-50	1003,28	935,15	3	7,29	2218,56
rc*208-50	999,19	953,47	5	4,80	15032,82
c*101-100	1036,11	1036,11	12	0,00	646,82
c*102-100	1219,78	1148,85	13	6,17	1416,79
c*103-100	1301,27	1105,80	13	17,68	2467,05
c*104-100	1252,15	1072,47	12	16,75	3617,64
c*105-100	1083,09	1083,09	14	0,00	1235,14
c*106-100	1083,57	1083,57	13	0,00	1171,01
c*107-100	1108,78	1046,48	12	5,95	1002,92
c*108-100	1073,01	1069,27	13	0,35	1514,59
c*109-100	1079,54	999,26	11	8,03	1705,29
r*101-100	1911,51	1792,76	20	6,62	517,55
r*102-100	1785,92	1644,33	18	8,61	837,42
r*103-100	1572,12	1417,78	14	10,89	1508,74
r*104-100	1200,88	1087,11	11	10,47	2304,59
r*105-100	1583,32	1493,81	15	5,99	949,89
r*106-100	1453,83	1350,25	14	7,67	1515,09
r*107-100	1281,30	1145,90	11	11,82	1844,61
r*108-100	1108,22	1038,94	10	6,67	2275,34
r*109-100	1377,39	1271,05	13	8,37	1313,59
r*110-100	1248,21	1172,84	12	6,43	1459,80
r*111-100	1271,45	1099,21	11	15,67	1587,82
r*112-100	1131,61	1050,85	11	7,69	1800,67
rc*101-100	1867,86	1746,87	16	6,93	906,46
rc*102-100	1674,80	1667,05	15	0,46	1389,15
rc*103-100	1539,08	1458,67	13	5,51	1592,79
rc*104-100	1337,63	1278,21	11	4,65	1926,15
rc*105-100	1747,15	1660,05	15	5,25	1224,27
rc*106-100	1608,07	1479,23	13	8,71	1457,54
rc*107-100	1467,21	1440,35	14	1,86	1320,82
rc*108-100	1344,33	1344,33	13	0,00	1247,52
