TRANSPORTATION OPTIMIZATION FOR THE COLLECTION OF END-OF-LIFE VEHICLES

by

Ahmed KHABOU

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT FOR A MASTER'S DEGREE
WITH THESIS IN AUTOMATED MANUFACTURING ENGINEERING
M.A.Sc.

MONTREAL, JULY 26, 2019

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Amin Chaabane, Thesis Supervisor
Department of Systems Engineering, École de technologie supérieure

Mr. Mustapha Ouhimmou, Thesis Co-supervisor
Department of Systems Engineering, École de technologie supérieure

Mr. Yvan Beauregard, President of the Board of Examiners
Department of Mechanical Engineering, École de technologie supérieure

Mr. Armin Jabbarzadeh, Member of the jury
Department of Systems Engineering, École de technologie supérieure

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON JULY 22, 2019

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGMENT

I would like to take this opportunity to express my gratitude to all the people who have assisted and helped me throughout this project.

First, I would like to offer my gratitude to my director, Mr. Amin Chaabane, and my co-director, Mr. Mustapha Ouhimmou, professors at the Systems Engineering Department, for their constant guidance and feedback as well as for their valuable comments and suggestions that they provided throughout this project.

I would also like to give a special thanks to Mr. Julio Montecinos, a postdoctoral researcher at the Systems Engineering Department, for his help in the development of the heuristic algorithm and Ms. Leila Hanachi, a research intern at the Systems Engineering Department, for her help in doing the sensitivity analysis.

It is impossible to forget the incredible support extended by my family, and my love Azza throughout this experience and I am forever indebted to them. I am very much grateful to my uncle, for his hosting and support along these two years.

OPTIMISATION DU TRANSPORT POUR LA COLLECTE DES VÉHICULES EN FIN DE VIE

Ahmed KHABOU

RESUMÉ

Les entreprises impliquées dans l'achat des véhicules en fin de vie ont d'importants défis à relever en raison du fait que la plupart des véhicules achetées doivent être collectés efficacement afin de réduire leurs coûts de transport. Dans ce projet, nous étudions un problème de logistique inverse d'une entreprise Canadienne qui ramasse des véhicules en fin de vie de différents vendeurs et les accumule à son entrepôt pour la revente des pièces ou pour des fins de recyclage. Ce problème peut être modélisé comme un problème de tournées de véhicules avec différentes contraintes. Bien que des recherches antérieures aient apporté des contributions substantielles pour modéliser et résoudre différentes variantes du problème de tournées de véhicules, le problème spécifique dans ce projet envisage la résolution d'une nouvelle combinaison de contraintes, telles que l'affectation de clients à une flotte privée ou à un transporteur externe, les fenêtres de temps, les routes multiples et les séquences de chargement. Nous proposons un modèle de programmation linéaire mixte en nombres entiers ainsi qu'une heuristique capable de trouver la planification des itinéraires qui minimisent les coûts totaux de transport. Le rendement des méthodes proposées est évalué à l'aide de données recueillies auprès de l'entreprise.

Mots-clés : Logistique inverse, problème de tournées de véhicules, heuristique, fenêtres de temps, flotte hétérogène, routes multiples, plusieurs transporteurs externes, séquences de chargement.

TRANSPORTATION OPTIMIZATION FOR THE COLLECTION OF END-OF-LIFE VEHICLES

Ahmed KHABOU

ABSTRACT

Firms operating in the purchasing of end-of-life vehicles (ELVs) have significant challenges related to the fact that most of the purchased ELVs must be collected efficiently in order to minimize their transportation costs. In this project, we study a reverse logistics problem of a Canadian firm that collects ELVs from a group of dealers and accumulates them at its warehouse for part resale or recycling. This problem can be modeled as a vehicle routing problem (VRP) with different side-constraints. Although prior research has made several contributions to model and solve different variants of the VRP, the specific issue in this project considers solving a VRP with a new combination of constraints, such as customer assignment to the private fleet or an external carrier, time-windows, multi-trip, and loading sequences. We propose a mixed-integer linear programming (MILP) model as well as a heuristic algorithm capable of finding the routes' planning that minimizes the total transportation costs. The performance of the proposed methods is assessed by generated instances using the data obtained from the company.

Keywords: Reverse logistics, vehicle routing problem, heuristic algorithm, time-windows, heterogeneous fleet, multi-trip, multiple external carriers, loading sequences.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ABBREVIATIONS

CVRP	Capacitated Vehicle Routing Problem
ELVs	End-of-Life Vehicles
HVRP	Heterogenous Vehicle Routing Problem
MTVRP	Multi-Trip Vehicle Routing Problem
SECs	Subtour Elimination Constraints
TSP	Travelling Salesman Problem
VRP	Vehicle Routing Problem
VRPPC	Vehicle Routing Problem with private fleet and external carriers
VRPTW	Vehicle Routing Problem with Time-Windows

INTRODUCTION

Traditionally, supply chains have been considered the linear movement of goods through distribution channels from suppliers to manufacturers, wholesalers, retailers, and finally to consumers. Recently, the flow of material has been proven to go in an upstreaming way during the production, distribution and consumptions stages which create a whole new are of logistics management, called as the reverse logistics (Cruz-Rivera & Ertel, 2009).

The role of reverse logistics has been developed a lot such that it now plays a significant part in the success of many different companies and organizations. From an economic point of view, the reverse logistics represent direct incomes from reduced consumption of raw materials, from adding value to recovered material and from cost reduction on waste treatment. For this reason, many companies are encouraged to become active in this new area of management (Schultmann, Zumkeller, & Rentz, 2006).

In recent years, the research field of supply chain management has been extended by tasks referring to reverse logistics flow such as product recovery, refurbishing, or recycling. These tasks constitute the end-of-life phase of products and complement the traditional supply chains by closing the loop to have the so-called, closed-loop supply chains. Intensive research has been focused on supply chain management and reverse logistics, trying to modify supply chains to form closed-loop supply chains (Schultmann et al., 2006).

In this project, we focus on a collection problem of end-of-life vehicles (ELVs), which is faced by a Canadian firm involved in product recovery and recycling. More specifically, we try to optimize the routes' planning for ELV collection using operation research tools. The problem of route planning is known by the research community as the Vehicle routing problem (VRP). The VRP is one of the most widely studied combinatory in operation research. The high interest of the research community in the different variants of VRP is not only motivated by its difficulty as combinatorial optimization problem but also by its practical relevance (Stefan Irnich, Toth, & Vigo, 2014). The main objective of this project is to achieve more efficiency in the reverse logistics activities related to the collection of ELV and to help in the economic success of recycling these types of products.

This report is composed of six chapters, as follows:

The first chapter, research problem, presents the research problem, objectives as well as the research methodology.

The second chapter, literature review, gives an overview of the research literature related to our project. First, we provide a brief review of the relevant VRP variants to our problem. Second, the different solution methods to solve the VRP are presented. Finally, we give a classification of the literature and present its limitations along with our contributions in this project.

The third chapter, experimental data, focuses on the information obtained from a Canadian company involved in the collection of end-of-life vehicles as well as on real instances' generation for the mathematical model and heuristic testing.

The fourth chapter, mathematical model, describes in details the formulation of our mathematical model, including the presentation of main inputs and assumptions. After that, we validate the model and presents its limitations.

The fifth chapter, heuristic development, presents in details our developed heuristic and gives the principal assumptions and steps for its execution.

The sixth chapter, computational results, presents the testing results of the mathematical model and the heuristic using the generated instances.

CHAPTER 1

RESEARCH PROBLEM

In this chapter, we first present the research problem. Then, we describe the research specific objectives. Finally, we give the research methodology followed in this project.

1.1 Research problem

In this project, we consider the reverse supply chain of vehicles, which is integrated within the end-of-life phase of these products (Figure 1.1). In fact, companies are now profitably and legally motivated to incorporate this life phase into their existing supply chain. Take the example of the treatment of ELVs, there is component disassembly and resale which constitutes the profit-oriented motivation, and there is recycling quotas imposed by governments which represent the legally oriented motivation (Cruz-Rivera & Ertel, 2009).

In the literature, several papers studied related problems with ELVs. For instance, Ene & Öztürk (2015) developed a mathematical model for managing reverse flows in ELVs' recovery network. Their main objective was to maximize revenue and minimize pollution at the end of life product operations. Besides, Demirel, Demirel, & Gökçen (2016) developed a mathematical model to optimize the reverse logistics activities of ELVs in Turkey. This model includes the different actors taking part in ELVs' recovery system.

Cin & Kusakcı (2017) did an exhaustive literature review on the logistics networks and modeling in the context of ELVs. They have studied 23 scientific works between 2005 and 2017 related to the field of reverse logistics network design for vehicles that have completed their life cycle. Their analyzations are based on the objective functions, the decision variables, the constraint handling method and the optimization methods used in the various papers. Kuşakcı, Ayvaz, Cin, & Aydın (2019) developed a fuzzy mixed integer location-allocation model for a reverse logistics network of ELVs. This study uses a novel approach and assumes that ELV supply in the network is uncertain. The merit of the proposed mathematical model is proved on a real-world scenario addressing the reverse logistics design problem for ELVs generated in the metropolitan area of Istanbul. The model is developed to be conformed to the existing directives in Turkey. Simic (2019) developed an interval-parameter conditional value-at-risk two-stage stochastic programming model for management of end-of-life vehicles. He conducted a case study to illustrate the potentials and applicability of the formulated model. He concluded that the presented model provides an important and contemporary tool for waste managers. Besides, it could be applicable across vehicle recycling industry that processes dozens of millions of ELVs every year.

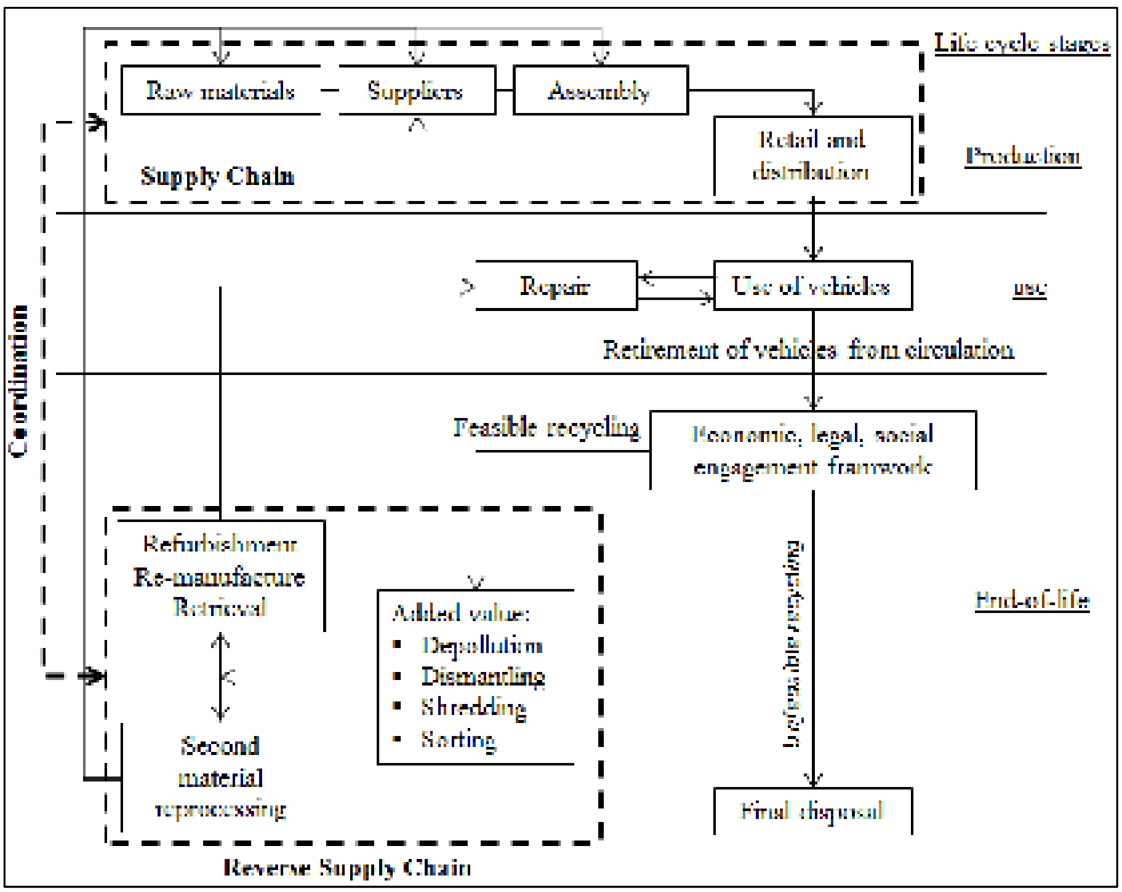

Figure 1.1 Closed loop supply chain for vehicles
Taken from (Cruz-Rivera & Ertel, 2009)

The reverse flow of ELVs consists mainly in the collection of crashed and medium-aged cars from customers and the transportation of these cars for recycling, reuse or disposal. For this reason, we focus on the vehicle routing aspect in the collection networks.

The optimization of transportation costs and distances becomes a crucial issue for many companies operating in the collection activities (Beullens, Van Oudheusden, & Van Wassenhove, 2004). In this context, Schultmann et al. (2006) model the reverse logistic tasks within closed-loop supply chains of the automotive industry in order to enhance ELV recycling in Germany. Reverse logistics modeling is done by vehicle routing planning and solved using a tabu search algorithm. Aras, Aksen, & Tuğrul Tekin (2011) studied the reverse logistics problem of durable goods firm that aims to collect cores from its dealers. They formulate a mathematical model for this problem, and they refer to it as the selective multi-depot VRP with pricing. They solved the problem using a tabu search algorithm which gives good quality results. For more details about the subject of collection and vehicle routing issues in reverse logistics, we refer the reader to the paper of Beullens et al. (2004).

In this project, we consider a real case study of a Canadian company involved in the purchasing and collection of ELVs. The company lives an extreme growth (more than 100,000 ELVs purchased every year), and one of the significant challenges is related to the fact that 80% of the purchased ELVs have to be collected efficiently to minimize the operational costs. At the moment, the process of purchasing and collection of ELVs is managed manually based on the operators’ experience. This leads to an increase in transportation costs, inconvenient delays to suppliers, as well as losing many business opportunities to rival companies. Therefore, there is a need for the development of the suitable optimization techniques to increase transportation efficiency with respect to the company’s business requirements. Indeed, designing the appropriate routes that maximize the recovered vehicles while minimizing the total traveled distance will reduce the complexity related to the ELVs transportation.

1.2 Research objectives

This project considers the vehicle routes planning in the context of reverse logistics. The VRP is a widely studied optimization problem that arises in many practical contexts such as transportation, freight collection, distribution, and reverse logistics. It deals with a whole class of problems that try to find the optimum delivery routes for a fleet of vehicles to serve a number of customers. The extension of the basic VRP by adding different constraints such as capacity and time windows constraints lead to a large number of variants (Stefan Irnich et al., 2014). While the VRP was widely studied in the literature, the specific case study of ELVs collection in this project regroups many constraints together to form a multi-attribute or rich VRP. The main constraints of our problem are:

- Customers assignment: customers may either be served by the company's private fleet or by an external carrier, which serves them directly at a predefined cost.
- Time-widows: the routes should be synchronized with the time-windows already specified in advance.
- A Heterogeneous fixed fleet of vehicles: the fleet has different types of trucks with various capacities and costs.
- Multi-trip: Due to the limited number and capacities of the vehicles, the model should consider performing multi-trip routing.
- Loading sequences: Loading sequences of the different ELVs should be considered in the generation of the routes and the vehicle assignment.

In this project, our main objective is to optimize the collection of ELVs with respect to the above constraints. This objective can be achieved in two steps: First, we need to develop a suitable mathematical model able to respect the specific characteristics of the problem. Second, we need to develop a heuristic algorithm able to obtain good quality results in reasonable computation time.

1.3 Research methodology

The research methodology shows the significant efforts that have been made to achieve our research objectives, and it is divided into four steps as follow:

- Problem analysis and characterization: The first step is the analysis and characterization of our problem through a detailed literature review. This will help us to define the specification and requirements to respect throughout the project.
- Mathematical model development: A mixed integer linear mathematical model that considers the problem specific constraints is developed. It is tested and validated with small instances to make sure that all important requirements are respected.
- Heuristic development: A heuristic solution is developed to solve large instances of the problem in a reasonable computational time. This method can have good-quality solutions (feasible solutions) close to optimality.
- Testing and evaluation: The final step is to test the efficiency of our developed model and the heuristic algorithm. To achieve this, we have collected information from the company to generate different instances for the testing.

CHAPTER 2

LITERATURE REVIEW

This chapter gives an overview of the literature related to our problem. First, since we are dealing with VRPs, our literature review mainly includes the line of research on the different variants of VRP and the main solution methods used by researchers to solve this kind of problems. After that, we make a classification for the most relevant works in the literature to our problem. Finally, we specify the limitations of the literature and we describe our main contributions.

2.1 The vehicle routing problem

Dantzig & Ramser (1959) introduced the ‘‘Truck Dispatching Problem” modeling how a fleet of homogeneous trucks could serve the demand for oil of a number of gas stations from a central hub and with a minimum traveled distance. After five years, Clarke & Wright (1964) generalized this problem to a linear optimization problem as follow: how to serve a set of customers geographically dispersed around the central depot, using a fleet of trucks with varying capacities. This became known as the Vehicle Routing Problem (VRP), one of the most widely studied topics in the field of operations research, supply chain management, graph theory, and computer sciences to optimize transportation, logistics, distribution and delivery systems (Braekers, Ramaekers, & Van Nieuwenhuyse, 2016).

The VRP is the generalization of the Travelling Salesman Problem (TSP) where a salesman wants to visit each of a set of towns exactly once, starting from and returning to his home town in the shortest possible way (Jünger, Reinelt, & Rinaldi, 1995). The TSP and the VRP are among the most widely studied combinatorial optimization problems. A vast number of books deal with these problems such as Gutin & Punnen (2002), Toth & Vigo (2001) and Toth & Vigo (2014). An excellent state of the art classification and review of the VRP is provided by Braekers et al. (2016). Nowadays, VRP problems incorporate real-life constraints such as time windows, number of depots, type of vehicles and many others. A classification of side constraints occurring in real-life VRPs is provided by Van Breedam (1995).

2.1.1 Capacitated VRP

2.1.1.1 Problem statement

The Capacitated Vehicle Routing Problem (CVRP) is the most classical version of VRPs. The problem can be structured on a directed graph $G = (V, A)$ where $V = 0 \cup N = \{1,2, \dots, n\}$ is the set of vertices (or nodes) and $A = \{(i,j) \in V \times V: i \neq j\}$ is the set of arcs. In the CVRP, the transportation requests consist of the distribution of goods from a single depot, denoted as point 0, to a given set of n other points, denoted as customers, $N = \{1,2, \dots, n\}$. The amount that has to be delivered to customer $i \in N$ is the customer's demand, $q_i \geq 0$, with $q_0 = 0$ for the depot. The fleet $K = \{1,2, \dots, |K|\}$ is assumed to be homogeneous, meaning that $|K|$ vehicles are available at the depot, all have the same capacity $Q > 0$. A vehicle moving from node i to node j incurs the travel cost c_{ij} for $(i,j) \in A$. A route is a sequence $r = (i_0, i_1, i_2, \dots, i_s, i_{s+1})$ with $i_0 = i_{s+1} = 0$, in which the set $S = \{i_1, \dots, i_s\} \subseteq N$ of customers is visited. The route r is feasible if the capacity constraint, $q(S) = \sum_{i \in S} q_i \leq Q$, holds and no customer is visited more than once: $i_j \neq i_k \; \forall \; 1 \leq j \leq k \leq s$. Therefore, a solution to a CVRP consists of $|K|$ feasible routes, one for each vehicle $k \in K$, of minimum route cost $c(r)$, such that every vertex is serviced exactly by one vehicle, each route starts and ends at the depot and the total demand serviced by a route does not exceed vehicle capacity (Stefan Irnich et al., 2014). In the following, we present a mixed integer mathematical model for the CVRP.

2.1.1.2 Mathematical model

This formulation uses two binary variables x_{ij}^k and y_{ik}: The first one, x_{ij}^k equals 1 if and only if a vehicle k traverses an arc (i, j) ∈ A, 0 otherwise. The second one, y_{ik} takes 1 if customer i is served by vehicle k and takes 0 otherwise. Moreover, for a customer subset $S \subseteq N$, we define $r(S)$ as the minimum number of vehicle routes needed to serve S. A lower bound often used instead of $r(S)$, is given by $[q(S)/Q]$. The following formulation is taken from Toth & Vigo (2001):

$$\text{Minimize} \sum_{(i,j)\in A, k\in K} C_{ij}\, x_{ij}^{k} \tag{2.1}$$

Subject to

$$\sum_{k\in K} y_{ik} = 1 \quad \forall \text{ i} \in N \tag{2.2}$$

$$\sum_{k\in K} y_{0k} = K \tag{2.3}$$

$$\sum_{j\in V} x_{ij}^{k} = \sum_{j\in V} x_{ji}^{k} = y_{ik} \quad \forall \text{ i} \in V, k \in K \tag{2.4}$$

$$\sum_{i\in S, j\notin S, k\in K} x_{ij}^{k} \geq r(S) \quad \forall \text{ S} \subseteq \text{N}, \text{S} \neq 0 \tag{2.5}$$

$$x_{ij}^{k} \in \{0,1\}, \;\; \forall \text{ (i, j)} \in A, k \in K \tag{2.6}$$

$$y_{ik} \in \{0,1\}, \quad \forall \text{ i } \in V, k \in K \tag{2.7}$$

The objective function (2.1) minimize of the total routing costs. Constraints (2.2) and (2.3) impose that each customer is visited exactly once and that $|K|$ vehicles leave the depot. Constraints (2.4) are the flow connectivity constraints, i.e. the same vehicle enters, and leaves a given customer. Constraints (2.5) serve at the same time as capacity constraints and subtour elimination constraints (SECs): First, consider an infeasible route over the cluster $S \subseteq N$ with a demand $q(S) > Q$. Due to $r(S) > 1$, at least two routes must connect S with its complement $V \setminus S$, so that any capacity-infeasible route is excluded. Second, any subtour over a non-empty subset $S \subseteq N$ (S $\neq$ 0) fulfills $\sum_{i\in S, j\notin S, k\in K} x_{ij}^{k} = 0$. Due to $r(S) \geq 1$ this subtour is also eliminated (Stefan Irnich et al., 2014). Finally, constraints (2.6) and (2.7) impose binary conditions on the decision variables.

Using (2.5) as SECs, the number of constraints grows exponentially with the number of nodes, which means that is practically impossible to solve directly the linear programming relaxation of problem (2.1) - (2.7). This problem can be remedied using the MTZ-formulation as introduced by Miller, Tucker, & Zemlin (1960) for the TSP. The new formulation consists of replacing the SECs (2.5) by another set of constraints using additional variables. The additional variables $u = (u_{1k}, \dots, u_{nk})^T$ indicate the accumulated demand u_{ik} already distributed by the vehicle k when arriving at customer i $\in N$. Thus, constraints (2.5) can be replaced by (2.8) and (2.9), which are respectively the MTZ-specific SECs and capacity constraints (Stefan Irnich et al., 2014).

$$u_{ik} - u_{jk} + Q\, x_{ij}^{k} \le Q - q_j \qquad \forall\ (\mathrm{i,j}) \in A, k \in K \tag{2.8}$$

$$q_i \le u_{ik} \le Q \qquad \forall\ \mathrm{i} \in V, k \in K \tag{2.9}$$

Note that $x_{ij}^{k} = 1$ implies $u_{jk} \ge u_{ik} + q_j > u_{ik}$. Hence, the presence of a subtour $(i, j, \dots, i)$ not containing the depot leads to the contradiction $u_{ik} > u_{jk} > \cdots > u_{ik}$. The advantage of the MTZ-constraints is that it has polynomial cardinality of variables and constraints sets (Stefan Irnich et al., 2014).

2.1.2 VRP with Time-Windows

The VRP with Time Windows (VRPTW) is an extension of the CVRP in which the service of each customer must be within a time interval, called a time-window. The depot is represented by the two vertices 0 and $n+1$ with reference to a source vertex and a sink vertex, respectively. Thus, we define the set $N = V \setminus \{0, n+1\}$ as the set of customer vertices. A feasible solution for the VRPTW is obtained by an elementary path from the source to the sink. The converse path, however, may not represent a feasible route as it can violate the time-window (Desaulniers, B.G. Madsen, & Ropke, 2014).

The time window is defined as an interval $[a_i, b_i]$ where a_i and b_i are the earliest possible departure time from node i and the latest possible arrival time at node i, respectively. We also define the travel time, t_{ij}, for each arc $(\mathrm{i}, \mathrm{j}) \in \mathrm{A}$ and the service time s_i for each customer i. The service of each customer must start within the associated time-window and the vehicle must stop at the customer location for s_i time instants. In addition, there is a time variable T_{ik} specifying the start of service time at vertex i when serviced by vehicle k (Toth & Vigo, 2001).

Time windows can be classified into two types (Desaulniers et al., 2014):

- Hard time windows: in which a vehicle that arrives early at a customer must wait until the customer is ready to begin service. Usually, waiting for the service to start does not incur additional costs.
- Soft time windows: in which the time interval may be not respected by the drivers, but this will incur a penalty cost each time the specified window is violated.

2.1.3 Heterogeneous VRP

The Heterogeneous VRP (HVRP) considers a group of vehicles that can differ in capacity, variable and fixed costs, speeds, and the customers that they can access. In HVRP, we have a fleet of vehicles made up of $|P|$ different vehicle types, i.e., the fleet K is partitioned into subsets of homogeneous vehicles $K = K^1 \cup K^2 \cup \dots \cup K^{|p|}$. Each vehicle type $p = 1, \dots, |P|$ has capacity Q_p, and may also have a fixed cost FC_p and a specific traveling cost c_{ij}^{p} along each arc modeling the route. The selection of an appropriate vehicle for each route has an impact on the total cost of the solution (Stefan Irnich, Schneider, & Vigo, 2014).

2.1.4 VRP with loading constraints

The VRP with loading constraints considers the way that the items should be loaded within the vehicles in addition to the routing optimization. Usually, in the CVRP, the demand of customers is expressed by the total weight of items. However, in VRP with loading constraints, we consider the shape and dimensions of items. This gives rise to new variants of VRP such as VRP with two-dimensional Loading constraints (2L-VRP) and VRP with three-dimensional

Loading constraints (3L-VRP). For such cases, restrictions imposing a feasible packing of the goods in the loading space have to be added to the weight constraints. In addition, the order of item's loading and unloading is also important in some cases. This also gives rise to other VRP variants such as VRP with Last-In-First-Out (LIFO) loading and VRP with First-In-First-Out (FIFO) loading (Iori & Martello, 2010).

2.1.5 Multi-trip VRP

The CVRP could also be extended to have a multi-route aspect and become known as multi-trip VRP (MTVRP). While in the CVRP each vehicle can only perform one route, in the MTVRP, vehicles may perform several routes over a planning horizon T. Given some routes with durations $T_1, T_2, \ldots, T_p$, a single vehicle may perform them if $T_1 + T_2 + \cdots + T_p \leq T$ holds. This situation is imposed, especially when the vehicle capacity is limited, or other constraints impose a small number of services per route, such as the number of available vehicles. Hence, feasible solutions with a limited fleet can only be achieved when vehicles are reused (Stefan Irnich et al., 2014).

2.1.6 VRP with private fleet and external carriers

Our problem considers the outsourcing option of some of the dealers to an external carrier if there are not enough private vehicles to serve them. This variant is known as VRP with private fleet and external carrier (VRPPC) or VRP with outsourcing (Archetti, Speranza, & Vigo, 2014). A single-depot routing problem with outsourcing options was first introduced by Chu (2005). The problem considers a private fleet of vehicles with limited capacity and a set of customers with known demand. Each customer can be served either by the private fleet which then incurs travel costs as in standard VRP or outsourced to a common carrier and in such a case, only fixed service costs must be paid. The objective is to minimize the total cost involving fixed costs for vehicles, variable travel costs, and fixed costs for orders performed by the common carrier. Our problem adds new features to the problem of Chu (2005) as it considers multiple external carriers that can have different costs depending on the customer's region. Hence, the cost of the external carrier should be considered in the global routing decision.

2.2 Solution methods

Solution methods for the VRP can be classified as exact, heuristic and metaheuristic algorithms. In recent years, heuristics and metaheuristics became the preferred methods for researchers to solve many variants of VRPs. While they are not always able to prove the optimality of solution they find, heuristics methods are often capable of finding solutions whose quality is good, particularly for real-world applications, which usually have high degrees of complexity (Gendreau & Potvin, 2010).

2.2.1 Exact methods

Exact methods are used to obtain optimal solutions by reducing the solution space and using as a base the developed mathematical model of the VRP. The main exact methods for VRP are branch-and-bound (and their extensions: branch-and-but, branch-and-price, branch-and-cut-and-price), set covering, integer programming and dynamic programming. Since the VRP with or without side-constraints is proven by Lenstra & Kan (1981) to be NP-hard, relatively large instances cannot be solved (in reasonable amount of time) using exact methods and only small instances can be solved to optimality. Therefore, we need to develop a heuristic algorithm to obtain good-quality results in reasonable computation time for large instances (Laporte, Ropke, & Vidal, 2014).

2.2.2 Heuristics

Heuristics are solution methods that can often find feasible solutions relatively quickly with no guarantee regarding solution quality. Thus, the test of these solutions is empirical, and their performances are judged by their computational results. Heuristics can be classified into two main families: constructive heuristics and improvement heuristics. Classification of the main heuristics used for the VRP is presented in Figure 2.1 (Toth & Vigo, 2001). In the following, we explore the main algorithms and concepts used for these solution methods.

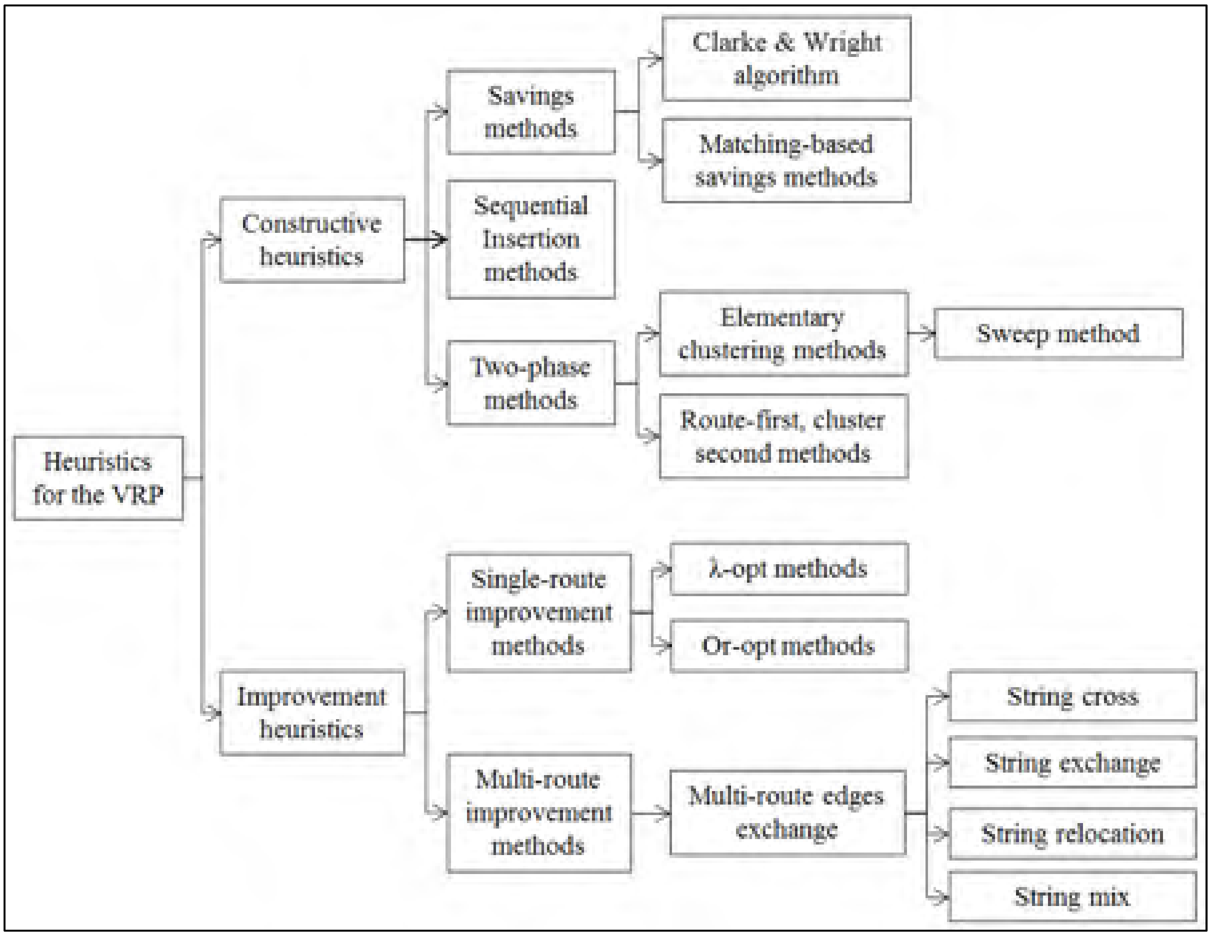

Figure 2.1 Classification of the main heuristic methods for the VRP
Taken from (Toth & Vigo, 2001)

2.2.2.1 Constructive heuristics

Constructive heuristics are usually employed to provide a starting solution to an improvement heuristic. The two most widely used algorithms for constructing VRP solutions are:

- The Clarke & Wright (1964) savings algorithm: which is based on merging existing routes using a saving criterion. More specifically, merging the two routes $(0, \ldots, i, 0)$ and $(0, j, \ldots, 0)$ into a single route $(0, \ldots, i, j, \ldots, 0)$ generates a saving $s_{ij} = c_{i0} + c_{0j} - c_{ij}$.

- The insertion algorithm of Mole & Jameson (1976): which is based on gradually assigning vertices to the vehicles' routes using an insertion cost. This algorithm has many variants such as inserting the vertex yielding the minimum extra distance, that the vertex yielding the smallest sum of distances between two neighbors' nodes or inserting the furthest vertex from the depot.

2.2.2.2 Improvement heuristics

Most of the improvement heuristics are based on the notion of local search, which is a central concept in most successful heuristics for the VRP. Local search algorithms are based on neighborhoods. Let φ be a finite set of feasible solutions to a given VRP instance and let $c: \varphi \rightarrow \mathbb{R}$ be a function that maps from a solution to the cost to this solution. Taking the example of a cost minimization, our goal is to find a solution s^* for which $c(s^*) \leq c(s)$ for all $s \in \varphi$. However, with heuristics, we are willing to find a solution that might be slightly inferior to s^*. Let $P(\varphi)$ be the set of subsets of solutions in φ. We define a neighborhood function as a function $N: \varphi \rightarrow P$ that maps from a solution s to a subset of solutions $N(s)$ called as the neighborhood of s. A solution s is said to be locally optimal with respect to a neighborhood $N(s)$ if $c(s) \leq c(s')$ for all $s' \in N(s)$.

Using the above definitions, we can describe a steepest descent algorithm (see Algorithm 2.1). The algorithm takes an initial solution s as input. At each iteration, it finds the best solution s' in the neighborhood $N(s)$ of the current solution s (line 4). If s' is better than s (line 5) then s' replaces s as the current solution (line 6). Lines 3–8 are repeated as long as s' is an improved solution. When the loop stops, the algorithm returns s as the best solution found.

In the following, we review the main improvement heuristics used for the VRP. These algorithms, which are based on local search, explore solution space using neighborhoods and can be divided into two categories: Single route (or intra-route) improvements in which a single route is changed compared to the initial solution and multi-route (or inter-route) improvements in which solutions are obtained by moving customers between two or more routes (Desaulniers et al., 2014).

Algorithm 2.1 Steepest Descent

Taken from (Desaulniers et al., 2014)

Steepest Descent - Taken from (Desaulniers et al., 2014)

Input: Initial solution $s \in \varphi$

Output: s as the best solution found

Input: Initial solution $s \in \varphi$

done = false

while done ≠ true do

    $s' \in \arg\min_{s'' \in N(s)}(c(s''))$

    if $c(s') \leq c(s)$ then

        $s = s'$

    else

        done = true

return s

➢ **Single-route improvements**

Most VRP intra-route improvements are based on λ-opt mechanism of Shen Lin (1965). Here, λ edges are removed from the route and the λ remaining segments are reconnected in all possible ways. The algorithm identifies any profitable reconnection and implements it. It stops when no further improvement can be obtained and so a local optimum is reached. S. Lin & Kernighan (1973) modified the original procedure and tried to move λ dynamically throughout the search to have better results than the static version. Another popular method, called as Or-opt method (Or, 1976), consists of displacing strings of three, two, or one consecutive vertices to another location.

- **Multi-route improvements**

In this types of improvements, Thompson & Psaraftis (1993) describes a general b-cyclic, k-transfer scheme in which a circular permutation of b routes is considered and k customers from each route are shifted to the next route. Some of the widely particular techniques of this general scheme are 2-opt*, string exchange and string relocation. The 2-opt* is defined similarly to the 2-opt, but its solutions are derived by modifying two routes instead of one. In string exchange, two sub-paths are selected, and their positions are exchanged. Finally, in string relocation, solutions are obtained by relocating a sub-path from one route to another one.

2.2.3 Metaheuristics

Metaheuristics are solution methods that encapsulate several heuristics. They are powerful algorithms that combine the power of local improvement procedures and higher-level strategies to create a process capable of performing a robust search of a solution space. Current metaheuristics for the VRP can be classified into local search algorithms and population-based algorithms. Classification of the main metaheuristics used for the VRP is presented in Figure 2.2. Readers interested in an overview of metaheuristic principles are referred to the handbook of metaheuristics of Gendreau & Potvin (2010).

2.2.3.1 Local search algorithms

These algorithms are based on local search and neighborhood techniques. They start from an initial solution x_1 and move at each iteration t from the current solution x_t to another solution x_{t+1} in its neighborhood $N(x_t)$. These methods are also called as single trajectory because they generate a sequence of solutions that can be seen as a trajectory through the solution space. In this type of algorithms, only the current solutions are used to determine the next one. However, care must be taken to avoid cycling, i.e. have the same solution at each iteration (Laporte et al., 2014).

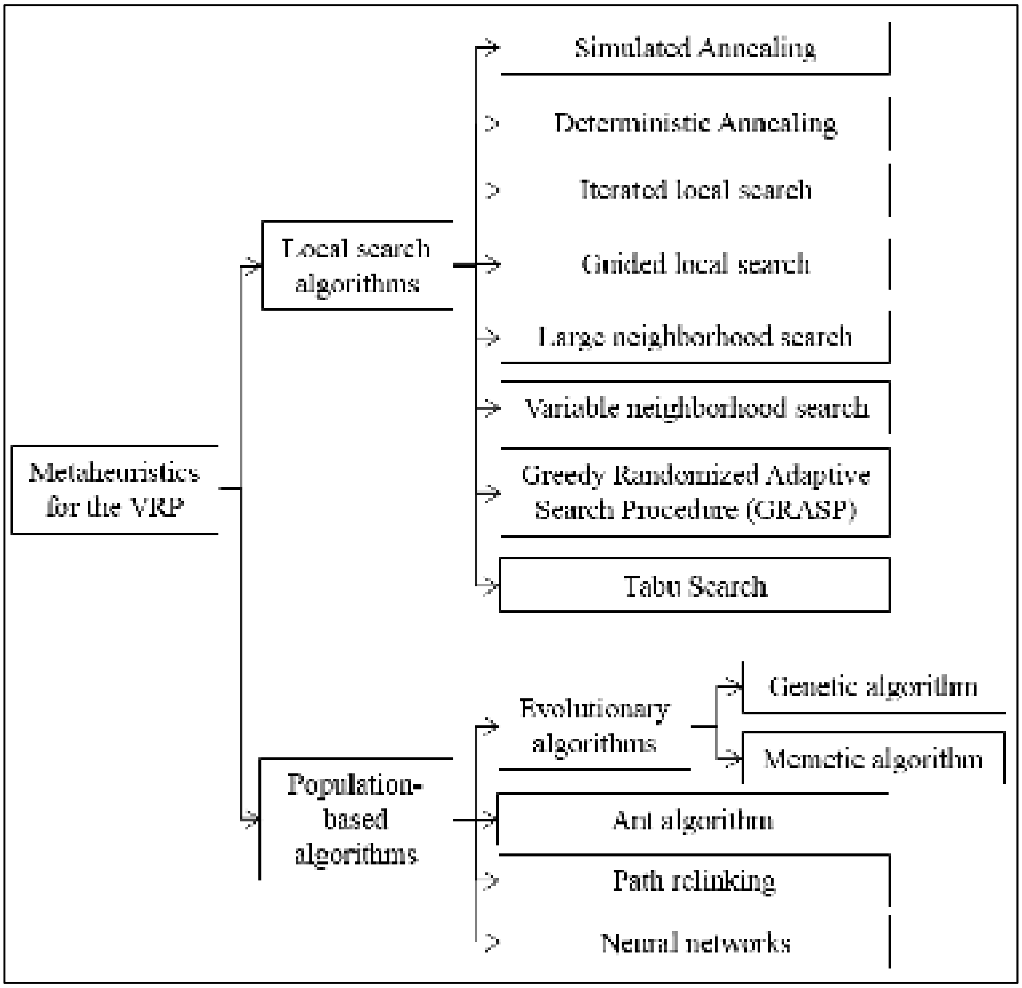

Figure 2.2 Classification of the main metaheuristic methods for the VRP
Taken from (Toth & Vigo, 2014)

2.2.3.2 Population-Based Algorithms

Population-based algorithms are based on the idea of maintaining a pool of solutions, called a population, which evolve at each iteration of the solution process. Unlike the single trajectory algorithms which have been inspired by the necessity of escaping from local optima and avoiding cycling, population-based methods take their inspiration from natural concepts, e.g., the evolution of species for genetic algorithm and the behavior of social insects foraging for ant colony algorithm. These algorithms use a guidance strategy based on a pool of solutions represented as chromosomes for genetic algorithm, or pheromone matrices for ant colony algorithm (Gendreau & Potvin, 2010).

2.3 Classification of the literature

In this section, we classify the main research papers from the literature, that have similar problems to ours, with respect to the VRP variants and to the solution methods.

2.3.1 Classification based on VRP variants

Table 2.1 presents a classification of the relevant research papers in the literature with respect to the VRP variants. We notice that all papers consider the capacitated VRP variant and most of them consider the time-window variant. In addition, some of the works define a maximum limit on the permitted time of the vehicles' route. This is due, for example, to the working shifts of drivers or the maximum allowable time for vehicles to operate. Moreover, most of the articles use a limited private homogenous fleet of vehicles, whereas others include the variant of heterogenous fleet in their works. Furthermore, for the VRP with the outsourcing variant, some papers consider multiple external carriers, whereas the others consider only a single common carrier for all customers. Finally, some authors consider the multi-trip aspect in their problems.

2.3.2 Classification based on solution methods

Table 2.2 presents a classification of the relevant research papers in the literature with respect to the used solution method. In this classification, we review mainly the papers that use heuristics and metaheuristics as solution methods, as we are interested in the different techniques used by researchers to develop our heuristic algorithm. For each research paper, we give the name of the heuristic, the heuristic used to build the initial solution, the different neighborhoods used by the algorithm, and the local search technique.

Table 2.1 Classification of research papers based on VRP variants

Paper	**VRP variant**					
	Capacity	**Time-windows**	**Time limit**	**Private fleet**	**Carriers**	**Multi-trip**
(Petch & Salhi, 2003)	x	x	x	Limited Homogenous		x
(Chu, 2005)	x			Limited Homogenous	Single	
(Bolduc, Renaud, & Boctor, 2007)	x			Limited Heterogeneous	Single	
(Olivera & Viera, 2007)	x		x	Limited Homogenous		x
(Krajewska & Kopfer, 2009)	x	x	x	Limited Homogenous	Multiple	
(Battarra, Monaci, & Vigo, 2009)	x	x	x	Limited Homogenous		x
(Ceschia, Di Gaspero, & Schaerf, 2011)	x	x	x	Limited Heterogeneous	Multiple	
(Potvin & Naud, 2011)	x			Limited Homogenous	Single	
(Stenger, Vigo, Enz, & Schwind, 2013)	x		x	Limited Homogenous	Multiple	
(Wang, Liang, & Hu, 2014)	x	x	x	Limited Homogenous		x
(Despaux & Basterrech, 2014)	x	x	x	Limited Heterogeneous		x
(Anaya-Arenas, Chabot, Renaud, & Ruiz, 2016)	x	x	x	Limited Homogenous		x
(Euchi, 2017)	x			Limited Homogenous	Single	
(Wu, Chu, & Hsu, 2017)	x	x	x	Limited Homogenous	Single	
(Gahm, Brabänder, & Tuma, 2017)	x			Limited Homogenous	Multiple	
(Sun, Wang, Lang, & Zhou, 2018)	x	x	x	Limited Heterogeneous		x
Our work	x	x	x	Limited Heterogeneous	Multiple	x

We notice that different heuristic algorithms are used by researchers such as tabu search, variable neighborhood search, adaptive memory programming and other specific heuristics. Concerning initial solutions, several heuristics are also used such as CS (Customer Selection) and MCW (Modified Clarke and Wright) algorithm, sweep algorithm and different kind of insertion algorithms. Each heuristic uses several techniques for both intra-route and inter-route neighborhoods such as 2-opt, 3-opt, 4-opt*, Or-opt, 1-E (one exchange), 2-E (two exchanges), 1-T (one transfer), 2-T (two transfers), 1-S (one swap), 2-S (two swaps), cyclic-E (cyclic exchange), EC (Ejection Chains), 1-I (one insertion), 2-I (two insertions) and LKH-2 (Lin Kernighan Heuristic). Finally, the main used local search technique are SS (Sequential Search), BI (Best Improvement) and FI (First improvement), Allowing IS (Infeasible Solutions), reduction techniques, adaptive guidance approach and using penalties.

Table 2.2 Classification of research papers based on solution methods

Paper	**Heuristic/Metaheuristic**				
	Name	**Initial solution**	**Name**		**Local search**
			Intra-route	**Inter-route**	
(Petch & Salhi, 2003)	Multi-phase constructive heuristic	Saving based construction, Bin-packing.		Meiosis, Exchange, 2- opt, 3-opt	SS, guided neighborhood.
(Chu, 2005)	TL–LTL	CS, MCW	1-E and 2-E	1- and 2-E	SS, BI
(Bolduc et al., 2007)	Selection, routing, insertion	MCW		1-E, 2-E, 2-T, 4-opt*	SS, BI
(Olivera & Viera, 2007)	Tabu search and adaptive memory	Sweep algorithm.		1-E, 1-T	Allow IS, reduction technique and route pool updating.
(Krajewska & Kopfer, 2009)	Tabu search	Two-phase construction and insertion.		1-T,1-E and 2-E	SS, Allow IS, Intensification through restarts.
(Battarra et al., 2009)	Vehicle routing heuristic	Sequential and parallel insertion.		2-opt	Adaptive guidance allow IS, critical intervals.

Table 2.2 Classification of research papers based on solution methods (continued)

Paper	Heuristic/Metaheuristic				
	Name	Initial solution	Name		Local search
			Intra-route	Inter-route	
(Ceschia et al., 2011)	Tabu search	Random construction heuristic	1-S	1-T and 1-S	Token-ring search Allow partial IS, tabu dynamics.
(Potvin & Naud, 2011)	Tabu search	CS, least-cost and convex hull insertion		1-T, 2-E, 4-opt* and EC	SS, Allow IS, self-adapting penalties
(Stenger et al., 2013)	Adaptive Variable neighborhood search (VNS)	CS, MCW	2-Opt, Or-Opt	cyclic-E, 1-E and 2-S	SS, FI, Allow IS, virtual vehicle, hill-climbing, roulette wheel selection
(Wang et al., 2014)	Adaptive memory	Optimal splitting.		1-S, 1-T	SS, BI, Route pool updating
(Despaux & Basterrech, 2014)	Simulated annealing	Solomon insertion	Order reversing	1-T, 2-T, 2-E and route partition	ad-hoc local search and allow IS.
(Anaya-Arenas et al., 2016)	Schedule construction heuristic		1-T		Multi-start technique
(Euchi, 2017)	Tabu search	CS, specific insertion.		1-E, 2-E, 2-opt and EC	SS, Allow IS.
(Wu et al., 2017)	VRPTWLTL	MCW	1-S, 1-I, 2-I	1-E, 2-E	Allow partial IS, SS BI
(Gahm et al., 2017)	Variable neighborhood search	CS, MCW	2-opt	LKH-2, 1-E, 2-E, 3-E, 4-E and 5-E	BI and FI, Virtual vehicle, explicit shaking.
(Sun et al., 2018)	Tabu search	Nearest-neighbour	2-E		Allow IS
Our Work	Heuristic algorithm	Routes' construction heuristic	1-S, 1-T	1-S	SS, BI

2.4 Limitations of the literature and contributions

As can be seen from Table 2.1, no paper considers the variant with external carriers and the variant with multi-trips at the same time. In our problem, in addition to the capacity, time-windows, time-limit and heterogeneous fleet constraints, we consider the multi-trip and outsourcing aspects together in the same mathematical model and heuristic development. We also notice that the VRP literature that considers the outsourcing and multi-trip aspects do not generally deal with loading constraints. In our problem, in addition to the mentioned variants, we also treat a specific type of loading constraints that considers the type of cars to be collected and their sequence of loading on the vehicles.

Concerning solution methods, most of the literature considers metaheuristics development. Since our problem is considered to be more restrictive compared to many reviewed works from the literature and given the limited size of our problem, we consider developing a specific heuristic algorithm to deal with the complicated nature of our constraints as well as to obtain a good solution in the smallest computational time. In our algorithm, we use a combination of different concepts and techniques that have already proven to be effective in the literature such as insertion algorithms, different intra-route and inter-route neighborhoods, assignment and scheduling algorithms, etc.

CHAPTER 3

EXPERIMENTAL DATA

In this chapter, we focus on the information collection process and the structure of the collected data from the company as well as the generation of the instances for the mathematical model and the heuristic algorithm.

3.1 Information collection process

In this project, our objective consists of automatizing and optimizing the manual planning of routes of ELVs' collection for a Canadian Company using a mathematical model and a heuristic algorithm. However, before beginning the routes' planning optimization, we need to characterize our problem and generate the different instances for the testing of our solution methods by collecting the necessary data on the transportation network of the firm. To achieve that, a survey was developed to identify the type and class of the VRP to model and its different constraints by collecting the following information:

- Type of offered service.
- The number of warehouses and dealers and their locations.
- Hours of operation of the warehouse and the dealers.
- Work schedule of the vehicles.
- The time required to load and unload the items on the vehicles.
- Characteristics of the fleet of vehicles.
- Characteristics of the third-party logistics service providers.
- Types and status of the items to be transported.
- Quantity of transported items.
- Type of costs taken into consideration by the company.

After obtaining the above information, we have two main steps. The first step is to organize and prepare the raw collected data. This step is important before beginning the development of our solution methods and it takes us a lot of time because the raw data is unorganized and cleaning it from the additional information is time consuming. After that, we use the prepared data to generate the different instances.

3.2 Data structure

In this part, we give the structure of the collected data from the company in order to use it for the instances' generation.

3.2.1 Locations

In this project, we are interested in optimizing the route planning of the company within the Greater Montreal region. Figure 3.1 displays some geolocated dealers and the depot of the company within the Greater Montreal region (area in green) using their zip codes. The firm has several depots located within the province of Quebec. However, we are interested in optimizing the route of each depot separately (a route begins and ends at a given depot after visiting some dealers).

3.2.2 The private fleet of vehicles and external carriers

The company has a fixed fleet of vehicles (auto-carriers) for collection activities at each one of its warehouses. This fleet is composed of different type of vehicles where each type has its own capacity and costs. In addition, the company may need the service of an external carrier if the demand of the day exceeds the capacity of the fleet. Table 3.1 displays an example of internal vehicles and external carriers used by the firm. Considering the capacity of the vehicles, we can divide our fleet into two types of trucks, ones with a capacity of three cars and the others with a capacity of two cars. Vehicles with a capacity of three cars can hold two cars on their platform and the third is towed (Figure 3.2), whereas vehicles with a capacity of two cars can hold only one car on the platform and the other is towed (Figure 3.3).

Figure 3.1 Example of geolocated depot and dealers

Table 3.1 List of internal vehicles and external carriers used by the company

Private fleet		**External carriers**
Truck	**Capacity**	
R-9	3 cars	W.V. AUTO
R-23	3 cars	DURAFLEX
R-27	3 cars	S.S REMORQUAGE
R-30	2 cars	V.I.P REMORQUAGE
R-31	2 cars	REMORQUAGE ICEBERG
T-154	2 cars	MAXIM

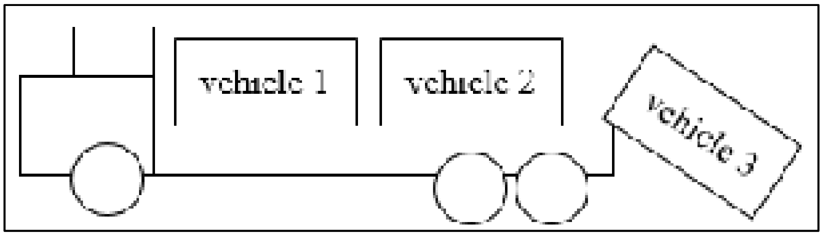

Figure 3.2 A vehicle with a capacity of 3 cars

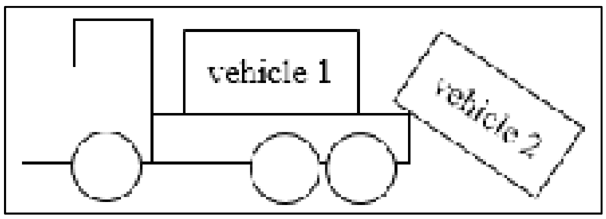

Figure 3.3 A vehicle with a capacity of 2 cars

3.2.3 Internal fleet and carriers' costs

In this project, we consider the minimization of the total cost of collecting cars from the dealers. The total cost can be divided into two main costs as follow:

- The internal cost which is composed of fixed and variable costs. The fixed cost is the cost of activating one internal vehicle from the private fleet to do a route and the variable cost is the unit cost per distance traveled by the internal vehicle. The evaluation of these costs is based on the operation costs of the company, as the depreciation cost of the vehicle, fuel costs, drivers' costs, repair and maintenance costs, insurance costs and finally license costs, and they are determined by the operations manager.
- The external cost which is the cost of using an external carrier to collect the cars. The carriers have a fixed cost for each region, so this cost depends on the region where the dealers will be visited.

Table 3.2 displays the costs defined for the internal vehicles and for the external carriers. For the internal vehicles with a capacity of three cars, they have a variable cost equals to 2,6$/km and a fixed cost equals to 75$, whereas, for the internal vehicles with a capacity of two cars, they have a variable cost equals to 2,4$/km and a fixed cost equal to 65$. The cost of the external carriers generally varies between 50$ and 150$ depending on the region where the customer is visited.

Table 3.2 Costs of internal vehicles and external carriers used by the company

	Type	**Variable cost**	**Fixed cost**
Internal vehicles	1 (capacity = 3)	2,6$/km	75$
Internal vehicles	2 (capacity = 2)	2,4$/km	65$
External carriers	-	[50$,150$]	-

3.2.4 Cars characteristics

In order to respect the loading constraints of the vehicles, we have to consider the characteristics of the cars collected by the company. This is done by building a database in which we classify the cars with respect to their length (or size), status and driveline. The car-size can be small, medium or large, the car status can be for example new, burned, damaged, without wheels, etc., and the driveline can be for example 4x4, 2x4, 4x2, etc. These different characteristics allow us to define the adequate loading sequences that should be respected in the generation of the routes. Table 3.3 gives the correspondence between the car length and size. For instance, a Volkswagen Beetle has a length of 4,28 meters, so it is considered as a small-size car, whereas, a Porsche Cayenne has a length of 4,85 meters, so it is considered as a large-size car.

Table 3.3 Classification of towed cars by size and length

Car size	**Car length**
Small vehicles	[0, 4.33 m [
Mid-size vehicles	[4.33 m, 4.72 m [
Large vehicles	[4.72 m, ∞ [

3.3 Generation of the instances

In our problem, the company needs to collect the cars from the dealers and transport them to the depot every single day. For this reason, one instance corresponds to one day of collection activities and our solutions methods will be used to provide an optimized daily planning. In the following, we give an example of data for one instance, then we give the different generated instances for the testing.

Table 3.4 Example of main data for a generated instance

Data	Example
Number of dealers	5
Number of available vehicles with a capacity of 3 cars	1
Number of available vehicles with a capacity of 2 cars	1
Number of permitted routes per vehicles	3
Number of available externa carriers	2

Table 3.5 Example of data for locations

ID	Name	Demand	Opening	Closing	Loading	Unloading
0	Depot	0	8 a.m.	8 p.m.	-	-
1	Dealer 1	1	8 a.m.	12 p.m.	20 minutes	-
2	Dealer 2	1	12 p.m.	4 p.m.	25 minutes	-
3	Dealer 3	1	12 p.m.	4 p.m.	20 minutes	-
4	Dealer 4	1	4 p.m.	8 p.m.	20 minutes	-
5	Dealer 5	1	4 p.m.	8 p.m.	30 minutes	-
6	Depot	0	8 a.m.	8 p.m.	-	15 minutes

Table 3.4 presents an example of the main data needed for an instance generation which include the number of dealers, number of available vehicles, number of the maximum permitted routes per vehicle and number of the external carriers.

Table 3.5 shows an example of data for the depot and the dealers. We consider a single depot for all generated instances that is duplicated to a start depot and an end depot for testing purposes. We consider that the demand of the dealers is only one car at a time. If a dealer has more than a car to sell, say for instance two cars, then it will be treated as two dealers who are in the same location. The opening and closing times for the depot and the dealers are also presented. For the end depot, we give an example of an unloading time of 10 minutes, whereas for the dealers, the loading time varies depending on the type of the car to collect.

Table 3.6 Example of data for cars

Dealer	**Car length**	**Car status**	**Driveline**
1	Medium	Body	2x4
2	Large	Damaged, Have only front wheels	4x4
3	Small	New, with wheels	4x2
4	Medium	Burned, have only back wheels	4x4
5	Large	Have only front wheels	4x2

Table 3.7 Example of brokers cost

Dealer	**Broker cost**
1	100
2	80
3	95
4	110
5	70

Table 3.8 Example of data for vehicles

Vehicle	Capacity	Working schedule	Fixed cost	Variable cost
1	3	12 hours	75$	2,6 $/km
2	2	12 hours	60$	2,4 $/km

Table 3.6 describes the characteristics of the cars to be collected from the dealers. Table 3.7 specifies the brokers cost for each dealer in case a dealer is visited by this broker. Finally, Table 3.8 depicts the characteristics of the available vehicles for this instance such as their capacities, working schedules, fixed costs and variable costs. Note that the only missing elements in this example are the matrices of distances and travel times between all defined locations.

Table 3.9 Generated instances for the mathematical model

Instances	Dealers	Vehicles	
		$q_v = 3$	$q_v = 2$
I-1	5	1	0
I-2	5	1	1
I-3	7	1	1
I-4	7	1	2
I-5	10	1	0
I-6	10	1	1
I-7	10	1	2
I-8	12	1	0
I-9	12	1	1
I-10	15	2	1
I-11	15	1	1
I-12	15	1	0
I-13	20	1	1

Following the previous example, we generate 43 different instances ranging from small ones with only five dealers to bigger ones with 59 dealers. The first 13 small instances are generated for the testing and validation of the mathematical model as it takes a lot of time to solve this complex problem (Table 3.9). To assess the performance of our heuristic algorithm, we use it to build the routes' planning of the company for November 2018. Table 3.10 depicts the 30 generated instances of November 2018 which are bigger than the ones generated for the testing of the mathematical model. The internal fleet used for the testing of these instances is composed of three vehicles: two vehicles with a capacity of three cars and one vehicle with a capacity of tow cars.

Table 3.10 Generated instances for the heuristic algorithm

Instances	**Dealers**	**Instances**	**Dealers**
D-1	49	D-16	45
D-2	53	D-17	40
D-3	32	D-18	22
D-4	34	D-19	51
D-5	40	D-20	45
D-6	45	D-21	49
D-7	48	D-22	55
D-8	38	D-23	49
D-9	40	D-24	25
D-10	33	D-25	23
D-11	28	D-26	48
D-12	52	D-27	40
D-13	43	D-28	59
D-14	50	D-29	58
D-15	39	D-30	42

CHAPTER 4

MATHEMATICAL MODEL

This chapter is based on our paper presented during the CIGI QUALITA conference (see ANNEX II). We first describe the inputs and the assumptions of our problem and then we present the developed mixed-integer linear mathematical model. Next, we validate the developed model by showing a detailed example of its execution over a generated instance. Finally, we present the limitations of using the model to solve large instances.

4.1 Problem statement

In this part, we propose a mixed-integer linear mathematical model that takes into account the company's specific requirements. The company needs daily planning for their operations. Thus, it must run the optimization every day after having a number of dealers to visit. For that reason, our model is considered as a single-period deterministic model because the number of dealers is known in advance, and the optimization horizon is only one day. Figure 4.1 presents an illustration of our VRP problem. In this example, we have the depot of the company, seven dealers, and one broker. The dealers can be served either by the private vehicle, that goes out from the depot, or the broker, which is an external logistics service provider that deals with the company. Our objective is to determine which dealers will be served by the private fleet and which dealers will be served by the brokers as well as the generation of a set of optimized routes for the private vehicles such that total internal costs plus total external costs are minimized.

Moreover, we need to respect some constraints and hypothesis imposed by the company, as shown in Figure 4.1:

- Each dealer can either be assigned to one broker or visited exactly by one route and one vehicle. In other words, a vehicle cannot visit a dealer two times or a dealer cannot be visited by two vehicles. In the example, we have five dealers serviced by the private fleet, which are 1, 2, 3, 5 and 7, and the dealers 4 and 6 are serviced by the broker.

- We optimize the route only for the private fleet. For example, we do not optimize the route for the broker to collect the cars from dealers 4 and 6. Instead, they are directly assigned to the cheapest broker depending on their locations.
- Each route starts and ends at the same depot after finishing the service for the last dealer. In the example, we have two routes, 0-3-1-0 and 0-2-5-7-0 (0 denotes the depot).
- The sum of the demands of the dealers in any route does not exceed vehicle capacity. This means that we cannot use a vehicle with a capacity of two cars to serve three dealers in one route.
- The total duration of the routes assigned to the same vehicle does not exceed the vehicle working time.
- Each dealer should be visited within a predefined time window. In our example, we have two time-windows which are from 8 a.m. to 12 p.m and from 12 p.m. to 4 p.m.
- The specific loading constraints of each vehicle should be respected. In the example, this is done by respecting the characteristics of the cars such as small, special, regular, 4x4, etc.

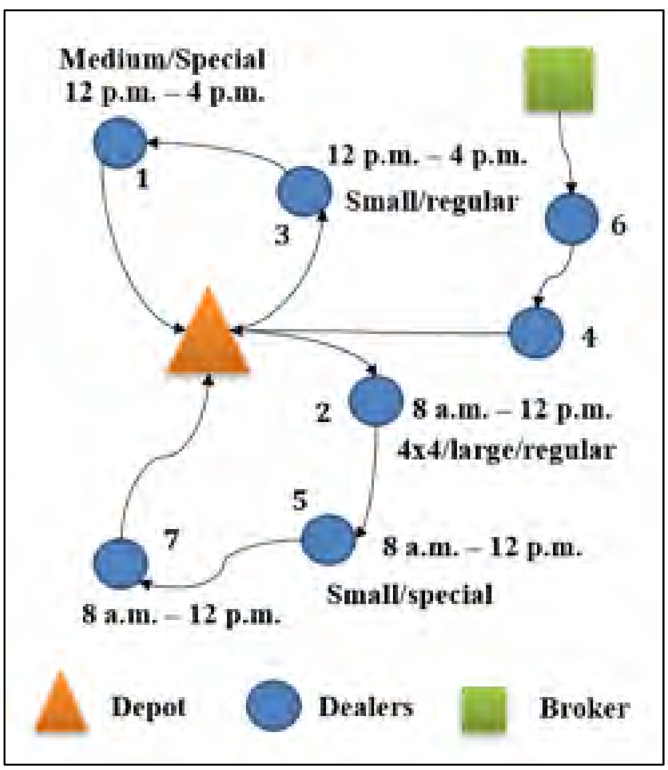

Figure 4.1 Illustration of our VRP problem

4.2 Inputs and assumptions

In this section, we present the main inputs and assumptions to formulate our mathematical model.

4.2.1 Network

Given a complete graph $G = (I, A)$, where $I = \{0,1, \dots n, n+1\}$ is the set of nodes (locations) and $A = \{(i,j): i < n+1; j > 0; i \neq j\}$ is the set of arcs connecting each node. Node 0 corresponds to the depart depot and node $n+1$ corresponds to the end depot which is a duplication of the depart depot, whereas vertices $N = \{1, \dots n\}$ correspond to the n dealers to be visited. A distance $d_{ij} \geq 0$ and a travel time $t_{ij} \geq 0$ are associated with each arc $(i,j) \in A$. Each location $i \in I$ has its service time $s_i \geq 0$ (s_i is the loading time for $i \in N$, $s_0 = 0$ and s_{n+1} is the unloading time) and each location should be visited within a predefined time window $[o_i, c_i]$ $(0 \leq o_i \leq c_i)$ with o_i and c_i are respectively the opening time and the closing time of the time-window for location $i \in I$. The service of the dealers must start within the time-window, but the vehicle may wait at a dealer's location if it arrives before the beginning of the time-window. Note that the time-window for the depot is limited by the work schedule of the vehicle v.

4.2.2 Fleet and routes

The set of vehicles is donated by V. The set is composed of a fixed heterogeneous fleet of vehicles where each vehicle $v \in V$ has a maximum capacity q_v, duration of working day T_v, fixed operating cost f_v for each time a vehicle leaves the depot and variable cost rate per distance unit c_v. In this model, we suppose that we have two vehicle types: We donate the vehicle type with $q_v = 3$ as the set V_1 and the vehicle type with $\mathrm{q_v} = 2$ as the set $\mathrm{V_2}$ and $V_1 \cup V_2 = V$.

Vehicles may perform several routes on the same day. This is due to the limited vehicle capacity q_v and to the limited number of available vehicles. Thus, feasible solutions with a limited fleet of size $|V|$ can only be achieved when vehicles are reused to perform several routes. The set of routes is denoted as R. A route $r \in R$ has a duration T_r and a single vehicle may perform several routes with durations $T_1, T_2, \ldots, T_r$ if $T_1 + T_2 + \cdots + T_r \leq T_v$ holds.

4.2.3 Pickups

The number of cars that need to be collected from a dealer $i \in N$ consists of a_i cars. Each request for pickup can be fulfilled by two transportation options: The first option is to use an internal vehicle. The other transportation option is offered by a set of brokers or external carriers. The set of brokers is donated by B. The assignment of a dealer i to a broker $b \in B$ incurs a cost p_{ib} that depends on the broker to be used and the location of the dealer to be served. In our model, we suppose that the brokers do not have any capacity limits but accept every subcontracted quantity.

4.2.4 Parameters

f_v: Unit vehicle operating cost (fixed cost)
c_v: Cost per unit distance traveled for a vehicle v (variable cost)
q_v : Capacity of vehicle v
T_v: Duration of the working day for a vehicle v
d_{ij}: Distance between location i and location j
t_{ij}: Time required to travel from location i to location j
a_i: Number of cars to be collected from dealer i
s_i : Service time for location i
$[\mathrm{o_i}, c_\mathrm{i}]$: Time window for location i
p_{ib}: Cost of assignment of a dealer i to broker b
d_i: Takes 1 if a dealer i has 4x4 car (also known as Four-Wheel Drive car), 0 otherwise

nw_i: Takes 1 if a dealer i has a body, burnt, damaged or without wheels car, 0 otherwise

g_i: Takes 1 if a dealer i has a large car, 0 otherwise

mid_i: Takes 1 if a dealer i has a medium car, 0 otherwise

M: A large number

4.2.5 Variables

x_{ijrv} : Takes 1 if arc $(i, j) \in A$ is used on route r by vehicle v, 0 otherwise.

y_{irv}: Takes 1 if a dealer i is visited on route r by vehicle v, 0 otherwise.

AT_{irv} : Arrival time of vehicle v for location i on route r.

u_{irv} : Load of the vehicle v before reaching a dealer i on a route r.

z_{ib}: Takes 1 if a dealer i is assigned to the external borker b, 0 otherwise.

Note that this formulation makes use of the additional variables u_{irv} to model the Miller–Tucker–Zemlin (MTZ) subtour elimination constraints.

4.3 Mathematical model

In this part, we give the mathematical formulation of our model which includes the objective function and the constraints of our problem.

4.3.1 Objective function

$$\sum_{j \in N \cup \{n+1\}} \sum_{r \in R} \sum_{v \in V} f_v x_{0jrv} + \sum_{i \in N \cup \{0\}} \sum_{\substack{j \in N \cup \{n+1\} \\ j \neq i}} \sum_{v \in V} \sum_{r \in R} c_v\, d_{ij} x_{ijrv} + \sum_{i \in N} \sum_{b \in B} p_{ib} z_{ib} \tag{4.1}$$

The objective function (4.1) minimizes the total cost which is the sum of three parts as follow: The first and second parts calculate the total fixed costs and the total variable costs of the internal vehicles, respectively. The third part calculates the cost of assigning the dealers to the external brokers.

4.3.2 Loading constraints

$$u_{irv} - 1 \leq M(1 - d_i \times y_{irv}) \quad i \in N,\ v \in V,\ r \in R \tag{4.2}$$

$$u_{irv} - 2 \leq M(1 - nw_i \times y_{irv}) \quad i \in N,\ v \in V_1,\ r \in R \tag{4.3}$$

$$u_{irv} - 1 \leq M(1 - nw_i \times y_{irv}) \quad i \in N,\ v \in V_2,\ r \in R \tag{4.4}$$

$$x_{jlrv} \leq M\big(1 - x_{0jrv}\big) \qquad v \in V_1,\ r \in R,\ j,l \in N, i \neq j, g_j = g_l = 1; \text{ or } g_l = 1\ and\ mid_j = 1; \text{ or } g_j = 1\ and\ mid_l = 1 \tag{4.5}$$

Constraints (4.2) states that if a dealer $i \in N$ has a 4x4 car then it should be visited first on a route $r \in R$ using a vehicle $v \in V$. Constraints (4.3) and (4.4) state that if a dealer $i \in N$ has a body, burnt, damaged or without wheels car then it should be visited either first or second on a route $r \in R$ if $v \in V_1$ and first if $v \in V_2$ (i.e. The vehicle should be on the platform of the vehicle). Constraints (4.5) state that vehicle $v \in V_1$ cannot hold two large cars or one large and one medium on the platform at the same time.

4.3.3 Flow and capacity constraints

$$\sum_{v \in V} \sum_{r \in R} y_{irv} + \sum_{b \in B} z_{ib} = 1 \qquad i \in N \tag{4.6}$$

$$\sum_{\substack{i \in N \cup \{0\} \\ i \neq j}} x_{ijrv} = \sum_{\substack{i \in N \cup \{n+1\} \\ i \neq j}} x_{jirv} = y_{jrv} \qquad j \in N, r \in R, v \in V \tag{4.7}$$

$$\sum_{j \in N} x_{0jrv} \leq 1 \qquad r \in R, v \in V \tag{4.8}$$

$$\sum_{i \in N} a_i y_{irv} \leq q_v \quad r \in R, v \in V \tag{4.9}$$

$$\sum_{j \in N} x_{0j(r+1)v} \leq \sum_{j \in N} x_{0jrv} \quad r \in \{1, \dots, |R|-1\}, v \in V \tag{4.10}$$

Constraints (4.6) ensure that a dealer $i \in N$ is either visited exactly once (by one route $r \in R$ and one internal vehicle $v \in V$) or it is assigned to an external broker $b \in B$. Constraints (4.7) are known as the flow conservation constraints which ensure that if a vehicle v visits a location $j \in I$ on route $r \in R$, then it should leave this location after service completion to have a balanced flow. Constraints (4.8) state that at most one vehicle $v \in V$ can go out from the depart depot on a route $r \in R$. Constraints (4.9) ensure that the total demand of the dealers on a route $r \in R$ should not exceed the vehicle capacity. Constraints (4.10) ensure that with respect to a vehicle $v \in V$, its $(r+1)^{th}$ route is realized only if its r^{th} route has been realized.

4.3.4 Subtour elimination constraints

$$u_{irv} - u_{jrv} + q_v x_{ijrv} + (q_v - a_i - a_j) x_{jirv} \leq q_v - a_j \tag{4.11}$$
$$i, j \in N,\ i \neq j,\ r \in R,\ v \in V$$

$$u_{irv} \leq q_v \qquad i \in N,\ r \in R,\ v \in V \tag{4.12}$$

$$u_{irv} \geq a_i \times y_{irv} \qquad i \in N,\ v \in V,\ r \in R \tag{4.13}$$

Constraints (4.11) are the MTZ subtours elimination constraints which are used together with lower and upper bounds on u_{irv} variables. Constraints (4.12) and (4.13) ensure that the load of the vehicle $v \in V$ on route $r \in R$ right after departing from dealer $i \in N$ must be at least equal to the number of cars picked up from that dealer and should not exceed vehicle capacity.

4.3.5 Time constraints

$$AT_{irv} + s_i + t_{ij} - AT_{jrv} \leq T_v \left(1 - x_{ijrv}\right) \tag{4.14}$$
$$i \in N \cup \{0\}, j \in \mathrm{N} \cup \{n+1\},\ \mathrm{r} \in R, \mathrm{v} \in \mathrm{V}$$

$$o_i \times y_{irv} \leq AT_{irv} \leq c_i \times y_{irv} \quad i \in N,\ v \in V,\ r \in R \tag{4.15}$$

$$o_i \leq AT_{irv} \leq c_i \qquad i \in \{0, n+1\}, v \in V,\ r \in R \tag{4.16}$$

$$AT_{0rv} \geq AT_{(n+1)v(r-1)} + s_{(n+1)} \quad v \in V,\ r \in \{2, \dots,\ |R|\} \tag{4.17}$$

$$AT_{(n+1)rv} - AT_{01v} \leq T_v \qquad v \in V,\ r \in R \tag{4.18}$$

Constraints (4.14) calculate the arrival time to location $j \in N \cup \{n+1\}$ after visiting its predecessor $i \in N \cup \{0\}$ and ensure that the arrival time of the location j is greater than the sum of the arrival time of location i plus the traveling time plus the service time of that same location. Constraints (4.15) and (4.16) ensure that the arrival time for location $i \in I$ is within the time-window of that location. Constraints (4.17) ensure that the starting time of the r^{th} route from the depart depot is greater than the arrival time of the $(r-1)^{th}$ route to the end depot plus the unloading time at the end depot. Constraints (4.18) states that the duration of routes performed by vehicle v should not exceed its working time limit.

4.3.6 Variable definition constraints

$$x_{ijrv} = \{0,1\} \quad (i,j) \in \mathrm{A},\ \mathrm{v} \in V, r \in R \tag{4.19}$$

$$y_{irv} = \{0,1\} \quad i \in \mathrm{N}, \mathrm{v} \in V, r \in R \tag{4.20}$$

$$AT_{irv} \geq 0 \qquad i \in I,\ \mathrm{v} \in V, r \in R \tag{4.21}$$

$$u_{irv} \geq 0 \qquad i \in \mathrm{N},\ \mathrm{v} \in V, r \in R \tag{4.22}$$

$$z_{ib} = \{0,1\} \quad i \in \mathrm{N},\ b \in B \tag{4.23}$$

Constraints (4.19) - (4.23) define the variables of the model.

4.4 Model validation

To validate the model, we present the results of its execution over a generated instance in Table 4.1. This example includes 10 dealers and uses 2 vehicles (one vehicle of each type) and no more than 2 routes per vehicle. There are 4 routes in total and no vehicle works more than twelve hours per day. In this example, all 10 dealers are visited by the internal fleet. The arrival time is written below the number of the dealer. Time windows are not presented, but they are all respected. Loading and unloading times are also respected. The column 'Duration' represents the total time of one route (length of the route). Next to the number of dealers, between brackets, we find the type of the car to be collected in order to help us verify that the loading sequences are also respected by the model (L refers to a large size car, M refers to mid-size car, NW refers to a car without wheels, 4x4 refers to a four-wheel drive car).

Table 4.1 Results for model validation

Vehicle	Start	Dealers			End	Duration
1 Route 1	8:00	8 [4x4] 8:53	7 [L] 10:23	2 11:46	12:35	4h35
1 Route 2	12:55	9 [4x4] [M] 13:58	6 [NW] [M] 14:49	10 15:35	16:23	3h28
2 Route 1	12:00	3 [L] 13:21	5 [L] 14:44		15:26	3h26
2 Route 2	15:46	11 [4x4] 16:47	4 [M] 17:33		18:41	2h55

4.5 Limitations of the mathematical model

Since the VRP is known from the literature as an NP-hard problem and our problem is a generalization of the basic VRP variant by adding other side-constraints, then our problem is also considered to be NP-hard. Therefore, when the size of our mathematical formulation increases, it generates a huge number of variables and constraints that cannot be solved to optimality in polynomial computation time. Taking the example of a real generated instance with 20 dealers, 3 routes and 2 vehicles. Since the proposed instance is relatively large, our model can only obtain a feasible solution, within a limit of three hours of computation, with a gap of 29% from the theoretical optimal solution proposed by the solver. This proves the limitation of the mathematical model to solve this kind of complex problems, especially on an industrial scale. Thus, we need to develop a heuristic algorithm to minimize the computation time while guaranteeing good-quality feasible solutions.

CHAPTER 5

HEURISTIC DEVELOPMENT

In this chapter, we present our developed heuristic for this project. First, we give an overview of the heuristic along with our different assumptions. Then, we present the main steps of the heuristic in details.

5.1 Heuristic overview

This heuristic is used to provide good-quality feasible solutions to our problem. It is a problem specific combination and adaption of the procedures described in Gahm et al. (2017) and Anaya-Arenas et al. (2016). Both works did not consider loading constraints and the work of Anaya-Arenas et al. (2016) did not consider the outsourcing aspect. The main difference with the idea of Gahm et al. (2017) lies in how to explore the space of the solutions in the first phase of the algorithm. In Gahm et al. (2017), the solution space is divided from the beginning into two sub-groups, one for the dealers who are going to be served by the internal vehicles and the other for external dealers. However, in our heuristic, we explore the whole space of solutions to search for optimal routes for the private fleet and the potentially unassigned customers are left to be served by the external carriers.

Our heuristic is an iterative procedure composed of three phases that are executed sequentially until all the dealers are assigned to the internal fleet and to the external carriers. It starts with a routes' construction phase in which the dealers are used to generate the set of routes. The second phase is about improving the generated routes in the first phase with the help of a local search procedure using different types of neighborhood structures. The final phase is about the assignment of a subset of the improved routes to the internal vehicles and potentially unassigned dealers are left to be served by the external carriers. Algorithm 5.1 presents the main course of action of our heuristic.

In the moment of heuristic development, we have respected the following assumptions:

- Dealers who are near each other are preferred to build a feasible route, i.e. routes that minimize the total distance and time.
- We maximize the utilization of the fleet, even if the total cost of the solution becomes higher.
- We build routes that comply with the capacities of the available vehicles.
- Because all constraints are considered throughout the algorithm execution, the solution is always feasible

Algorithm 5.1 Main course of action of the heuristic algorithm

Heuristic Algorithm

1 Routes construction phase (See section 5.2.1)

Sort the set of dealers N in ascending order of earliest time window (o_j) of the dealers.

Define R as the set of routes to be generated, AT_{jr} as the arrival time at node $j \in N$ with route $r \in R$, S_r the starting of route $r \in R$ and F_r as the finishing time of route $r \in R$.

1.1 Routes initialization

Let $r = 1$ be an empty route which begins from the depot i.

While there are still routes to generate then,

1.2 Dealers assignment to the routes

While there still eligible dealers in N that can be added to the route r then,

For j in N do,

Select the first j such that it satisfies the constraints of the problem.

Add j to the route r.

Update $AT_{jr} = \max\{AT_{ir} + s_i + t_{ij};\ o_j\}$.

If j is the first visited node by route r, then

Update $S_r = o_j - t_{0j}$.

The route is closed, and the vehicle returns to the depot i.

Update $F_r = AT_{jr} + s_j + t_{jn+1} + s_{n+1}$.

$r = r + 1$.

2 Routes improvement phase (See section 5.2.2)

Define $N(r)$ as the neighborhood of a route $r \in R$.

For r in R do,

While it is still possible to improve r do

$r' \in \arg\min_{r'' \in N(r)}(cost(r''))$

If r is feasible and $cost(r') \leq cost(r)$

then $r = r'$

Return r.

3 Routes assignment phase (See section 5.2.3)

Sort the set of vehicles V into decreasing order of their fixed operating cost.

Define VS_v as the vehicle starting time and VF_v as the vehicle finishing time.

3.1 Routes classification and vehicle selection

Classify the routes in ascending order by the Ratio = Loss / Time.

Let $v = 1$ be the first selected vehicle from V.

While there are still not used vehicles then,

3.2 Routes assignment to the vehicle

Let R be the set of routes obtain by step 2.

While the maximum capacity of v is not reached,

Assign a subset of routes to v using an exact mathematical formulation.

Delete all routes that contain already visited dealers by v from R.

$v = v + 1$.

3.3 Routes assignment to the external carriers

If all vehicles are used and there are still not assigned routes, then the dealers of these routes are assigned to the external carriers.

5.2 Heuristic description

In this part, we describe the main phases of the heuristic which are routes construction, routes improvement and routes assignment.

5.2.1 Routes construction

The routes construction phase considers all available dealers in one day and respects all the constraints of the problem. We donate AT_{jr} as the arrival time at node j with route r. We also define S_r and F_r as the starting and finishing time of route r, respectively. The set of dealers N is sorted in ascending order of the earliest time window (o_j) of the dealers. In this phase, we define a limit on the number of routes to be generated as there are a lot of combinations and the process can be very time-consuming.

Step 1: Routes initialization

We suppose that we are going to generate $|R|$ routes in total. Let $r = 1$ be an empty route which begins from the depot i.

Step 2: Dealers assignment to the routes

Selects the first dealer $j \in N$ to be visited from i that satisfies the following constraints:

i. The dealer j has not been visited by route r.

ii. Time-window constraint:

- It is possible to arrive at dealer j before the end of its time window ($AT_{jr} + s_j + t_{ij} \leq c_j$).

iii. Duration constraint:

- After visiting j at $AT_{jr} = \max\{AT_{ir} + s_i + t_{ij};\ o_j\}$, it is possible to return to the depot.

iv. Loading constraint type 1:

- If a dealer has a 4×4 car then it should be visited first on the route.

v. Loading constraint type 2:

- If $v \in V_2$: If a dealer j has a body, burnt, damaged or without wheels car then it should be visited first on the route.

- If $v \in V_1$: If a dealer j has a body, burnt, damaged or without wheels car then it should be visited either first or second on the route.

vi. Loading constraint type 3:

- If $v \in V_1$ and if j is the second dealer on the route, then j cannot be either large nor medium car if the first dealer has a large car, and j cannot be large if the first dealer has a medium car.

If j satisfies the above constraints, then it will be added to the route r. If two dealers have the same opening time-window (o_j), then the one who minimizes the total distance of the route is chosen. Update AT_{jr} as the earliest possible arrival time to dealer j, $AT_{jr} = \max\{AT_{ir} + s_i + t_{ij};\ o_j\}$. If j is the first visited node by route r, then the starting time of route r is set in such a way that the vehicle arrives at j at the beginning of its time window o_j ($S_r = o_j - t_{0j}$). Also, the visit time of node j is set as $AT_{jr} = o_j$. j becomes the current position in the route r and the next potential visit is evaluated using step 2. When none of the dealers in N are eligible to be added to the route, the route is closed, and the vehicle returns to the depot. Update the route finishing time, $F_r = AT_{jr} + s_j + t_{jn+1} + s_{n+1}$ (we consider that the time of the route finishes with the unloading operation at the depot). If there are still routes to generate, go to step 1 to initiate a new route $r = r + 1$. Otherwise, the algorithm goes to the next phase of routes improvement.

5.2.2 Routes improvement

After the routes' construction phase, we obtain a set of routes where each one of them has a total travel cost. The total travel cost of a route is the total cost per unit distance. Thus, the objective of this phase is to optimize the generated routes by minimizing their distances. To achieve this, an iterative local improvement procedure is developed using different neighborhood structures.

5.2.2.1 Neighborhood structures

Neighborhood structures are based on transformation rules, which define the transformation of a solution to obtain another solution. A feasible solution r in our case is a route obtained by the first phase. The neighborhood $N(r)$ of a current solution r is composed of all feasible solutions that can be obtained by applying to r one of the specified intra-route and inter-routes moves defined below. If the move leads to distance reduction, the feasibility of the neighboring solution is verified.

- **Intra-route neighborhoods**

Two intra-route moves are used in this algorithm: intra-swap move for swapping two dealers belonging to the same route and intra-shift move for changing the position of a dealer in the same route. Figure 5.1 presents an example of intra-swap neighborhood: Given a route r, the intra-swap move is obtained by replacing arcs (2, 4) and (1, 3) with arcs (1, 2) and (3, 4). Figure 5.2 presents an example of intra-shift neighborhood: For each node 1 and a route r, the intra-shift move corresponds to its insertion after node 3, is obtained by removing arcs (0, 2), (3, 1) and (1, 4) and replacing them with arcs (0, 1), (1, 2) and (3, 4).

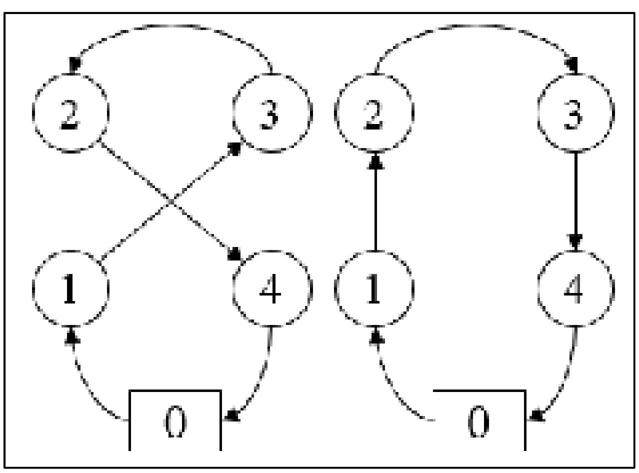

Figure 5.1 Intra-swap neighborhood
Taken from (Chu, 2005)

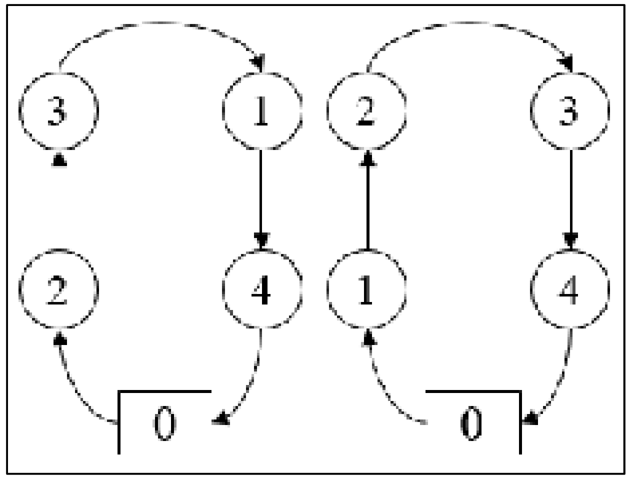

Figure 5.2 Intra-shift neighborhood
Taken from (Chu, 2005)

- **Inter-routes neighborhood**

One inter-route move is used in this algorithm: inter-swap move for swapping two dealers belonging to two different routes. Figure 5.3 presents an example of inter-swap neighborhood: For each node 2 on route 1, the inter-swap move corresponds to its exchange with node 5 on route 2, are obtained by removing arcs (1, 5), (5, 3), (4, 2) and (2, 6), and replacing them with arcs (1, 2), (2, 3), (4, 5) and (5, 6).

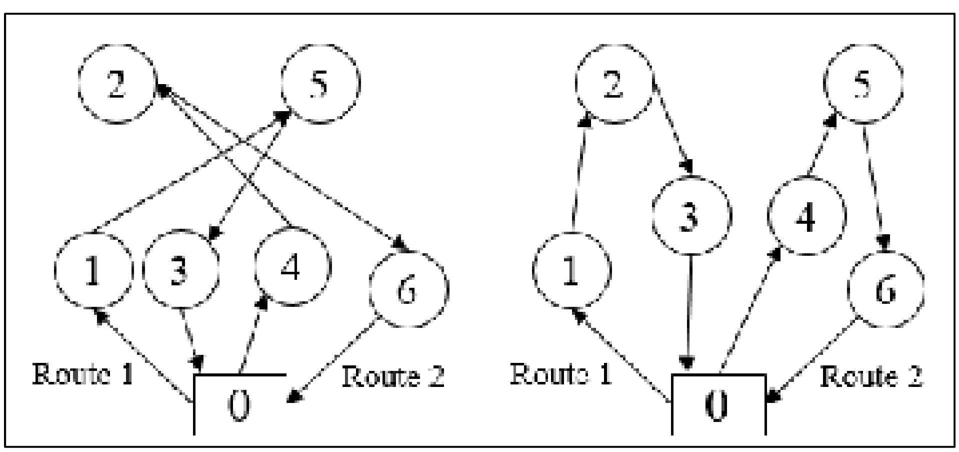

Figure 5.3 Inter-swap neighborhood
Taken from (Chu, 2005)

5.2.2.2 Local search procedure

Our local search procedure involves executing the inter-routes and the two intra-route neighborhoods sequentially using all possible permutations of the available routes. The improvement strategy used in this algorithm is the best-improvement strategy. This strategy evaluates all possible solutions defined by each of the neighborhood structure and returns the best of these solutions. To extract and insert dealers from and into a route, we use a local-search mechanism that extracts randomly chosen dealers and inserts them individually. To insert the dealers, the feasible position with the lowest cost is calculated for insertion. At each local search iteration, a neighborhood solution is generated from the current solution using one of the described neighborhoods. Neighbor solution is tested for cost improvement and feasibility and if the conditions are met, then it will be accepted as the new solution. This process continues until no improvement for the different neighborhoods is found. This phase generates a set of new optimized routes to be used in the routes' assignment phase.

5.2.3 Routes assignment

The goal of this phase is the assignment of the improved routes to the vehicles and the external carriers. This phase has both the character of a job scheduling problem and an assignment problem. The vehicles can be viewed as a set of heterogeneous machines and the routes can be viewed as the jobs that need to be executed by the machines. The goal is to find valid planning for one day for the vehicles respecting the constraints of the problem. The result planning must indicate which route should be carried out by which vehicle, when the routes should begin, and finish as well as what is the routes execution sequence for each vehicle.

In this problem, we have two objectives: The first one is to assign as much as possible of routes to the vehicles by minimizing the spare time. The spare time is defined as the time when the vehicle is not used. The second objective is to minimize the cost of the schedule by assigning the routes that have the least cost. Since the cost of the brokers is usually lower than the costs of using the company's fleet and with the condition of using the internal fleet to its maximum capacity, we use a penalty cost for non-using the vehicles to their maximum capacity in a given day.

This phase is composed of three steps that are executed sequentially until all the routes are assigned to the internal fleet and the external carriers. Given the set of routes R, obtained by the routes' improvement phase. At each iteration, a subset of the routes is assigned to one vehicle $v \in V$ until the maximum capacity of the vehicle is reached. The process stops when all the fleet's vehicles are used. The potential remaining dealers are automatically assigned to the external carriers.

In order to use the internal fleet to its maximum capacity and to speed up the resolution time, we use an exact mathematical formulation for the construction of the schedule of each vehicle. In this problem, each route $r \in R$ has a duration which is equal to the sum of the travel times between each node of the route plus the loading times at the dealers plus the unloading time at the depot. An early starting time which is equal to the opening of the time-window of the first dealer. A late starting time which is equal to the minimum between the closing of the time-window of the first dealer and the closing of the time-window of the last dealer minus the traveling time from the first dealer to the last dealer on the route minus the loading times of the previously visited dealers. Each route must be completed within a time window larger than its processing time. Finally, there is different types of routes (routes of two dealers or routes of three dealers).

The number of vehicles is fixed. Each vehicle can handle only one route at a time and a route should be completed before starting another one. Each vehicle has defined workday limits which means it only works during this time. As well, vehicles have a depot from which they start and end their routes. Working overtime is not allowed due to increased work costs for the company. Furthermore, we consider that vehicles may have spare time between the execution of the routes or at the beginning and the ending of their schedule. We want to precise also that if the selected vehicle has a capacity of three places, then it is possible to assign routes of three or two dealers with a priority to routes of three dealers. However, if the selected vehicle has a capacity of two places, then it is only possible to assign routes of two dealers.

In the following, we present the steps of our third and final phase for this heuristic.

Step 1: Routes classification and vehicle selection

In this phase, we try to have a limited number of routes from the improved routes of the previous phase in order to speed up the assignment model. To achieve that, we classify the improved routes in ascending order by a ratio defined as Ratio = Loss / Time. The Loss is defined as Loss = Total travel cost - Total broker cost. This ratio will help us to choose the cheapest and fastest routes for the assignment model as well as prioritize the assignment of the best-quality routes to the vehicles. After that, we sort the set of vehicles V into decreasing order of their fixed operating cost. In fact, we consider the largest available vehicle $v \in V$ at each iteration. Let $v = 1$ be the first selected vehicle from the ordered set of vehicles.

Step 2: Routes assignment to the vehicle

This step is about the construction of the schedule of the selected vehicle using an exact mathematical formulation. In this problem, we consider a set $T = \{1, \dots, t\}$ of available time-slots of one vehicle, a set $R = \{1, \dots, r\}$ of the routes to be assigned to the time-slots and a set $N = \{1, \dots, n\}$ of dealers. For each route $i \in R$ is associated with a loss c_i (Loss = Total travel cost - Total broker cost), an early arrival time e_i, a late arrival time l_i, a duration d_i. We also have a binary parameter y_{ki} which states if dealer $k \in N$ is visited by route $i \in R$ or not. In addition, we use a penalty cost q which is defined as the cost of non-using the time-slots at their maximum capacities. Furthermore, we consider four variables as follow:

- ✓ x_{ij}: Equals 1 if route $i \in R$ is assigned to time slot $j \in T$, 0 otherwise.
- ✓ s_j: Starting time of a time-slot $j \in T$.
- ✓ f_j: Finishing time of a time-slot $j \in T$.
- ✓ p_j: Spare time of a time-slot $j \in T$.

Equations (5.1) - (5.12) present the mathematical formulation of the problem:

$$Minimize \sum_{\mathrm{i \in R}} \sum_{\mathrm{j \in T}} c_i x_{\mathrm{ij}} + \sum_{\mathrm{j \in T}} q\, p_{\mathrm{j}} \tag{5.1}$$

$$\sum_{\mathrm{j \in R}} x_{\mathrm{ij}} \leq 1 \;\; i \in R \tag{5.2}$$

$$\sum_{\mathrm{i \in R}} x_{\mathrm{ij}} \leq 1 \;\; j \in T \tag{5.3}$$

$$\sum_{\mathrm{i \in R}} \sum_{\mathrm{j \in T}} y_{ik} x_{\mathrm{ij}} \leq 1 \;\; k \in \mathrm{N} \tag{5.4}$$

$$\sum_{\mathrm{i \in R}} x_{\mathrm{ij}}\, e_{\mathrm{i}} + \left(1 - \sum_{\mathrm{i \in R}} x_{\mathrm{ij}}\right) s_0 \leq s_j \leq \sum_{\mathrm{i \in R}} x_{\mathrm{ij}}\, l_{\mathrm{i}} + \left(1 - \sum_{\mathrm{i \in R}} x_{\mathrm{ij}}\right) f_m \quad j \in T \tag{5.5}$$

$$f_j = s_j + \sum_{\mathrm{i \in R}} x_{\mathrm{ij}}\, d_{\mathrm{i}} + p_j \quad j \in T \tag{5.6}$$

$$s_{j+1} \geq f_j \;\; j \in T \tag{5.7}$$

$$s_1 \geq 8 \tag{5.8}$$

$$f_t \leq 20 \tag{5.9}$$

$$s_j \geq 0 \;\; j \in T \tag{5.10}$$

$$f_j \geq 0 \;\; j \in T \tag{5.11}$$

$$x_{\mathrm{ij}} = \{0,1\} \;\; i \in R\,, j \in T \tag{5.12}$$

The objective function (5.1) minimize the total loss of the assigned routes to the time-slots plus the total cost of non-using the time-slots at their maximum capacities. Constraints (5.2) and (5.3) ensure that a route can be only assigned to only one time-slot and a time-slot could have only one route, respectively. Constraints (5.4) ensure that a dealer is visited only once in all the time-slots. Constraints (5.6) and (5.6) calculate, respectively, the starting time and the finishing time of a time-slot $j \in M$. Constraints (5.7) states that the starting time of the $(j+1)^{th}$ time-slot is longer than or equal to the finishing time of the j^{th}. Constraints (5.9) and (5.9) ensure that the starting time of the first slot is bigger than 8 a.m. and the finishing time of the last slot is less than 8 p.m. Constraints (5.10) - (5.12) defines the variables of the problem.

After solving the mathematical model, we have an optimized vehicle schedule which contains as mush as possible of routes, within the allowed working time limit, and with a minimized cost. Thus, we update the calculated vehicle starting time $VS_v = s_0$ and the vehicle finishing time $VF_v = f_m$. Delete all the routes that contain the already visited dealers from the set of routes R. If there are still routes not assigned to any vehicle, a new vehicle v is considered from the list of vehicles and this phase is repeated until all vehicles are used.

Step 3: Routes assignment to the external carriers

If all vehicles are used and there are still not assigned routes, then the dealers of these routes are assigned, using a least cost assignment, to the external carriers.

5.3 Heuristic validation

To validate the heuristic, we present the results of its execution over some generated instances in Table 5.1 and we compare its performance to the performance of the mathematical model. The details of the instances can be found in Table-A I-1 of the ANNEX I. The used fleet here is composed of two vehicles from each type (V_1 has a capacity of 3 cars and V_2 has a capacity of 2 cars). The first column in the table shows the name of the instance and the number of dealers in each one. The next three columns present the results of the mathematical model, the heuristic algorithm without using a penalty cost and the heuristic algorithm using a penalty cost. The three columns include the routes obtained for each instance, the dealers assigned for the external carriers and total cost of the solution. Besides, the two columns of the heuristic include the percentage gap of their solutions from the optimal solution of the mathematical model. The routes include only the visited dealers and each route is displayed with the vehicle that performs it (for example V_2 R_1 means that vehicle number 2 performs route number 1).

Table 5.1 Model performance vs. heuristic performance

Instance	Mathematical Model	Heuristic without penalty	Heuristic with a penalty
J-1 (5 dealers)	Carriers: 1, 2, 3, 4, 5 Total cost = 335 $	Carriers: 1, 2, 3, 4, 5 Total cost = 335 $ Gap = 0%	Carriers: 1, 2, 3, 4, 5 Total cost = 335 $ Gap = 0%
J-2 (10 dealers)	V_1 R_1: 7–2–10 Carriers: 1, 3, 4, 5, 6, 8, 9 Total cost = 633,4 $	Carriers: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 Total cost = 650 $ Gap = 2,6%	V_1 R_1: 7–10 V_2 R_1: 8–4 Carriers: 1, 2, 3, 5, 6, 9 Total cost = 778,8 $ Gap = 23%
J-3 (15 dealers)	V_1 R_1: 1–13–15 V_2 R_1: 7–11 Carriers: 2, 3, 4, 5, 6, 8, 9, 10, 12, 14 Total cost = 919,2 $	V_1 R_1: 7–13–15 Carriers: 1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 14 Total cost = 964,9 $ Gap = 5%	V_1 R_1: 7–13–2 V_1 R_2: 1–14–15 V_2 R_1: 4–10 Carriers: 3, 5, 6, 8, 9, 11, 12 Total cost = 1064,7 $ Gap = 15,8%
J-4 (20 dealers)	V_1 R_1: 16–7–11 Carriers: 1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 13, 14, 15, 16, 18, 19, 20 Total cost = 1205,6 $	V_1 R_1: 7–16–19 V_2 R_1: 4–18 Carriers: 1, 2, 3, 5, 6, 8, 9, 10, 11, 12, 13, 14, 15, 17, 20 Total cost = 1237,3 $ Gap = 2,6%	V_1 R_1: 13–7–16 V_1 R_2: 2–19–15 V_1 R_3: 18–14 V_2 R_1: 8–4 V_1 R_2: 10–20 Carriers: 1, 3, 5, 6, 9, 11, 12, 17 Total cost = 1407,1 $ Gap = 16,7%

Figure 5.4 Routes of the first vehicle of the instance J-4 using the heuristic

Before beginning our analysis, we want to note that the solution of the instance (J-4) using the mathematical model is feasible as opposed to the solutions of the other three instances which are optimal. This is because we have limited our computation time to only one hour and it takes more time for the model to obtain an optimal solution for the instance with 20 dealers.

From Table 5.1, we can validate our heuristic, as the performance of the algorithm without a penalty cost is as good as the performance of the mathematical model. In fact, the percentage gaps of the heuristic's solutions from the optimal solutions are small: 0% for J-1, 2,6% for J-2, 5% for J-3 and 2,6% for J-4. Also, we can validate that the heuristic works better for bigger instances because for the instance J-4, even if the cost of the heuristic solution is slightly higher, we obtain a better-quality solution (better fleet utilization) for less computation time. Concerning the heuristic performance with a penalty cost, we notice that the gaps from the optimal solutions are higher (23% for J-2, 15,8% for J-3 and 16,7% for J-4) as the penalty forces the algorithm to consider more alternatives in order to use the internal fleet better. Note that the gap can be reduced by allowing more time to the algorithm to be executed.

Finally, Figure 5.4 shows an example of routes' planning on google maps of the first vehicle of the instance J-4 using the heuristic algorithm with a penalty. The presented routes of this vehicle are: R_1: 13–7–16, R_2: 2–19–15 and R_3: 18–14. Thus, the whole trip is: 0–13–7–16–0–2–19–15–0–18–14–0, where 0 is the depot of the company located in H7E4P2 (the red point on the map).

CHAPTER 6

COMPUTATIONAL RESULTS

This chapter is devoted to the numerical experiments carried out using the generated instances in order to assess the performance of the mathematical model and the heuristic algorithm.

6.1.1 Used material for the development

For mathematical model development, we use LINGO15.0 solver and for heuristic development, we use Python 2.7.13 as a programming language. In addition, we call either CPLEX12.8 or ECOS_BB or CBC for the resolution of the exact problem defined in the assignment phase of the heuristic, depending on the availability of the solver in the machine. Our used desktop computer for the testing has the following characteristics: Windows 7 Enterprise 2009 (64 bits) as the operating system, Intel Core, i7-2600, 3.4GHerz as a processor and 16GB for the memory RAM.

6.1.2 Testing parameters

To give a realistic estimation of the distances and the traveling times between the different locations, we use the Open Source Routing Machine (OSRM). OSRM is an implementation of a high-performance routing engine for shortest paths in road networks. It combines sophisticated routing algorithms with open and free road network data of the Open Street Map (OSM) project. OSRM is able to compute and output a shortest path between any origin and destination. It uses Dijkstra's algorithm to calculate the short paths along with different speed-up techniques to optimize the computational time. OSRM uses basic vehicle profile to calculate the traveling time and use different approximation techniques to adjust the average speed of the vehicle on each segment of the route ("Open Source Routing Machine," 2018). However, it does not consider the variation of the traveling times during the day for the same path caused by the different rush hours, especially in the early morning or the late afternoon. To have a better estimation of the traveling time and to take congestion into consideration, we analyzed real-time routing data from google maps in different hours of the day and compared them with the data obtained with OSRM. Then, we did a linear regression to adjust the OSRM data along with the collected google maps data. Using OSRM, we were able to obtain distances with an average gap of $\pm 11\%$ from the google maps distances and travel times with an average gap of $\pm 26\%$ from the google maps traveling times. Using the linear regression technique, we were able to approximately generate real travel times between the different location of the problem by decreasing the gap from $\pm 26\%$ to approximately $\pm 15\%$.

Each vehicle has a working schedule of 12 hours from 8 a.m. to 8 p.m. and the dealers have different predefined time-windows. The loading time is considered to be 20 minutes for each car plus 10 minutes extra-loading time for cars that have specific conditions and are difficult to collect. The unloading time at the depot is considered to be 10 minutes for each car. For vehicles of type v_1, f_v= 75$ (fixed cost) and c_v= 2,6 $/km (variable cost) meanwhile for vehicles of type v_2, f_v= 60$ and c_v= 2,4 $/km. We consider seven brokers who have different costs depending on the region to be visited. For each postal code, we choose the external carriers that have the lowest towing cost. Thus, if a dealer is geolocated within a specific postal code, then it will be served by a carrier that has the lowest towing cost. Finally, the parameters of the car, i.e. length, status and driveline, are generated for each collected car following the data provided by the company.

6.1.3 Mathematical model testing

Table 6.1 presents the tests done on the 13 small generated instances using the mathematical model within the allowed time limit of 3 hours or equivalently 10800 seconds. Each instance is tested three times using different groups of dealers all of them located in the Greater Montreal region. The column 'Cost ($)' represents the average cost obtained by the different test results. The average 'Gap (%)' is calculated by the following formula: GAP (%) = |Objective value – best objective| / (best objective) × 100. The best objective used to calculate the gap is the best theoretical value that can be obtained by the solver. The column 'CPU(s)' represent the average computational time in seconds.

Table 6.1 Test results using the mathematical model

Instances	Dealers	Vehicles		Routes	Cost($)	GAP(%)	CPU(s)
		q_v=3	q_v=2				
I-1	5	1	0	1	181	0	0,6
I-2	5	1	1	1	202	0	1,4
I-3	7	1	1	1	347	0	2,1
I-4	7	1	2	1	305	0	6
I-5	10	1	0	1	572	0	23
I-6	10	1	1	2	547	0	58
I-7	10	1	2	1	615	0	129
I-8	12	1	0	3	646	0	838
I-9	12	1	1	2	685	0	7365
I-10	15	2	1	1	767	9	10800
I-11	15	1	1	2	785	10	10800
I-12	15	1	0	3	718	11	10800
I-13	20	1	1	3	1215	29	10800

We notice that the mathematical model can obtain optimal solutions only for the small instances with 5, 7, 10 and 12 dealers, respectively, within the allowed time limit. For the instances with 15 dealers, the model can only obtain feasible solutions within the allowed time limit. The average percent deviation from the best objective of these solutions is between 9% and 11%. For the instance with 20 dealers, the gap increases to 29%. We limit our testing to the instances with 20 dealers as we notice that the gaps are going to be much higher and more computational time is needed to obtain good results.

6.1.4 Heuristic testing

For the testing of our heuristic algorithm, we use the 30 generated instances of November 2018 and we compare the results of the manual planning with the results of our algorithm. While in the mathematical model, we fix the number of routes in advance, the heuristic finds the optimal number of feasible routes that could fit into the vehicle's schedule. This is done to use the internal fleet at its maximum capacity. In addition, based on the collected information from the company, we estimate that the average cost of using an internal vehicle for one day is equal 450$. Thus, we calculate our penalty cost for non-using an internal vehicle at its maximum capacity $q = 450\$/720\ minutes = 0{,}625\ \$/minute$, meaning that each minute a vehicle is not used, a cost of $0{,}625\$$ is incurred.

Table 6.2 presents the optimized routing results for November 2018 at the warehouse of Laval using the heuristic algorithm. The first column displays the name of the 30 generated instances for the testing. The second column shows the number of dealers for each day. The next column 'routes for each vehicle' presents for each vehicle, the number of routes done in a given day. The used fleet is composed of two vehicles with a capacity of three places (v_1 and v_2) and one vehicle with the capacity of two places (v_3). The next two columns show the number of internal and external dealers, respectively. These two parameters will help us in determining the percentage of using the internal fleet in the routing. Finally, the total travel cost and the brokers' cost are presented in the last two columns. The sum of the two latter costs gives us the total cost of the solution.

The heuristic resolution time is between 20 minutes to 40 minutes for each day. The part that is the most time consuming is the routes generation and improvement as there are many routes combinations, whereas the assignment of the different routes takes only a few seconds with a limit of 1 minute. Note that it is almost impossible to report the exact computation time for the algorithm as python is an interpreted language and so other tasks performed by the computer can influence the speed of the algorithm execution.

Table 6.3 shows the difference between the results of manual planning versus the results of the optimization method. Considering the manual planning, we notice that 20% of dealers were served using internal vehicles and 80% are served using the external brokers. However, using the optimization algorithm, we notice that 54% of dealers are served internally and 46% are served by the external carriers. This means that the utilization of the internal fleet is increased as required by the company.

Table 6.2 Test results using the heuristic algorithm

Instances	Dealers	Routes for each vehicle			Internal dealers	External dealers	Travel cost ($)	Broker cost ($)
		v_1	v_2	v_3				
D-1	49	4	3	2	25	24	1468,5	1555
D-2	53	3	3	3	24	29	1349,3	1970
D-3	32	2	2	1	14	18	876,5	1215
D-4	34	3	3	2	22	12	1398,8	805
D-5	40	4	2	3	24	16	1644,8	1105
D-6	45	3	1	4	19	26	1231,7	1730
D-7	48	4	3	2	25	23	1435,4	1600
D-8	38	3	3	2	22	16	1324,1	995
D-9	40	3	3	2	22	18	1221,6	1255
D-10	33	4	2	2	21	12	1296,8	780
D-11	28	3	1	1	14	14	870,9	965
D-12	52	4	4	2	28	24	1409,2	1590
D-13	43	3	3	2	22	21	1475,9	1255
D-14	50	4	4	2	28	22	1400,9	1450
D-15	39	3	3	3	23	16	1416,4	1070
D-16	45	4	3	4	29	16	1675,4	1025
D-17	40	3	3	1	20	20	1306,3	1305
D-18	22	3	1	2	16	6	1224,4	380
D-19	51	4	2	4	26	25	1398,5	1670
D-20	45	3	3	2	22	23	1417,4	1370
D-21	49	4	3	2	25	24	1381,2	1690
D-22	55	4	4	3	30	25	1796,5	1670
D-23	49	3	3	2	22	27	1316,4	1855
D-24	25	3	1	1	14	11	893,5	730
D-25	23	3	2	2	19	4	1243,5	265
D-26	48	4	4	2	28	20	1581,3	1350
D-27	40	3	3	2	22	18	1315,9	1190
D-28	59	3	3	2	22	37	1215,1	2450
D-29	58	4	4	5	34	24	1893,7	1675
D-30	42	2	3	3	21	21	1161,4	1390
Total	1275	100	82	70	683	592	40641,3	39355

We also notice that the percentage of using each one of the vehicles is increased from 8% to 23% for v_1, from 12% to 19% for v_2 and from 0,3% to 11% for v_2. In addition, we notice that the average collected cars per day are also increased for each of the vehicles. Indeed, we can collect up to 10 cars per day using v_1 instead of 3, we can collect up to 8 cars per day using v_2 instead of 4 and we can collect up to 4 cars per day using v_3 instead of almost not using the vehicle. Furthermore, we remark that the utilization of the vehicles is decreased from v_1, which is the most used one, then v_2, then v_3, which is the least used vehicle. This is because our heuristic works sequentially and affect a higher number of dealers to the first vehicles to be used. This can be solved by adding a balancing aspect between the utilization of each vehicle to the algorithm. However, due to the limited time of the project and the irrelevance of this constraint to the company, we chose not to implement it. Finally, we observe that the total cost for the whole month is minimized by 16091,7$ from 96088$ to 79996,3$ which constitute a saving of 16,7%.

Table 6.3 Manual planning vs. Optimization method results

	Manual planning	**Optimization method**
Total number of collected cars	1275	1275
Total number of internally collected cars	259 (20%)	683 (54%)
Total number of externally collected cars	1016 (80%)	592 (46%)
Total collected cars by v_1	100 (8%)	299 (23%)
Total collected cars by v_2	148 (12%)	244 (19%)
Total collected cars by v_3	4 (0,3%)	140 (11%)
Average internally collected cars per day	8,6	22,8
Average externally collected cars per day	33,9	19,7
Average collected cars per day by v_1	3,3	10,0
Average collected cars per day by v_2	4,9	8,1
Average collected cars per day by v_3	0,1	4,7
Total cost ($)	96088	79996,3

Another thing we have observed when executing the heuristic using a ten-time bigger penalty cost scenario, is that even if the fleet utilization becomes higher, with 62% utilization percentage instead of 54%, the total cost of the solution also becomes much higher, to be at 91291,3 $, which constitutes a saving of only 5% compared to the manual planning scenario. The cost is higher because instead of the cheaply assigning the dealers to the external carriers, the internal vehicles are now doing expensive routes (routes with longer travel times and less dealers). Thus, in order to minimize the total cost and have optimized routes, it is better sometimes to assign some dealers to the external carriers even if an internal vehicle is available.

Furthermore, to better explore the effect of the penalty cost on the solution, we have used three different instances (small, medium and large instances) to study the variation of the percentage of the internal fleet usage (Figure 6.1) and the variation of the time percentage usage of the same fleet with respect to the penalty increase (Figure 6.2).

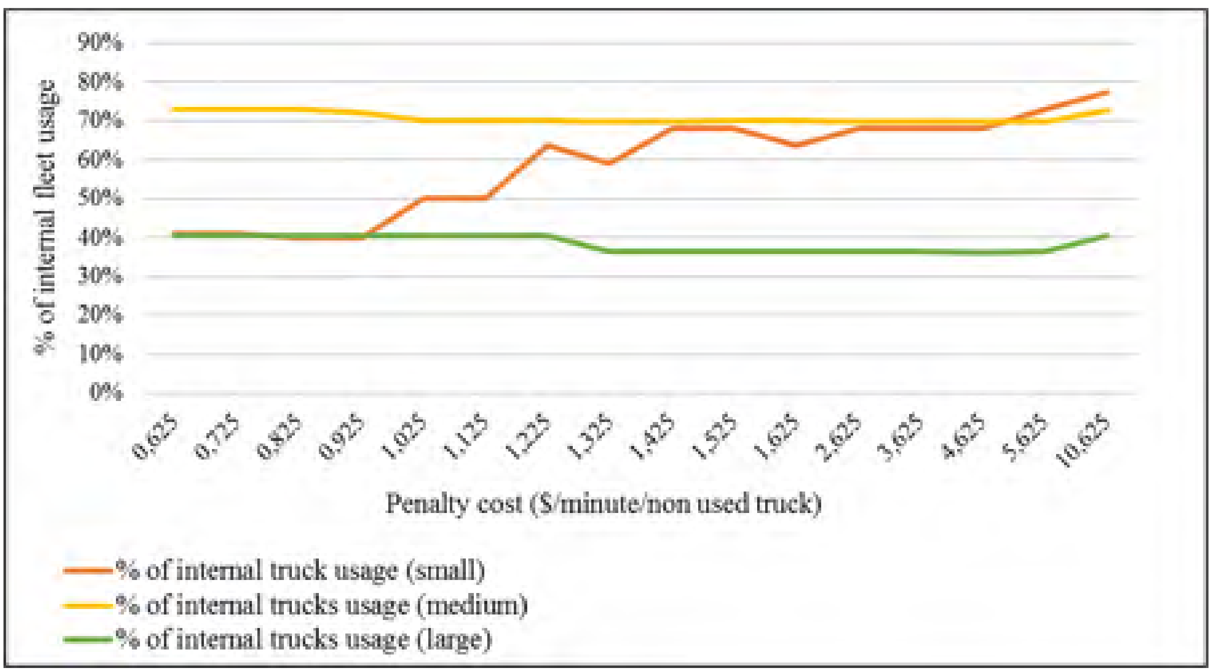

Figure 6.1 Variation of the percentage of using the internal fleet with respect to the penalty cost for three instances

In Figure 6.1, we notice that for medium and large instances the percentage of internal fleet usage is approximately constant when the varying the penalty cost, with a value of 70% and 40% respectively for the medium and large instances. However, we notice for the small instance that increasing the penalty cost led to an increase in the percentage utilization of the internal fleet. This can be explained by the fact that for the large and medium instances, we don't have a room for further optimization as the maximum capacity of the fleet is easily reached in all cases. In Figure 6.2, we notice that increasing the penalty cost led to an increase in the time use percentage of the internal fleet for the three instances. We notice also that there is a gap between the small instance and the large and medium instances. This is because there is fewer number of dealers to visit by the internal fleet for the small instance compared to the two other ones. Finally, we can see that the fleet is almost 100% used for the large instance with nearly no effect when varying the penalty cost. This explains another time that the penalty does not have a big impact on the result when dealing with large instances and the algorithm is always capable of using the internal fleet at its maximum capacity in this case.

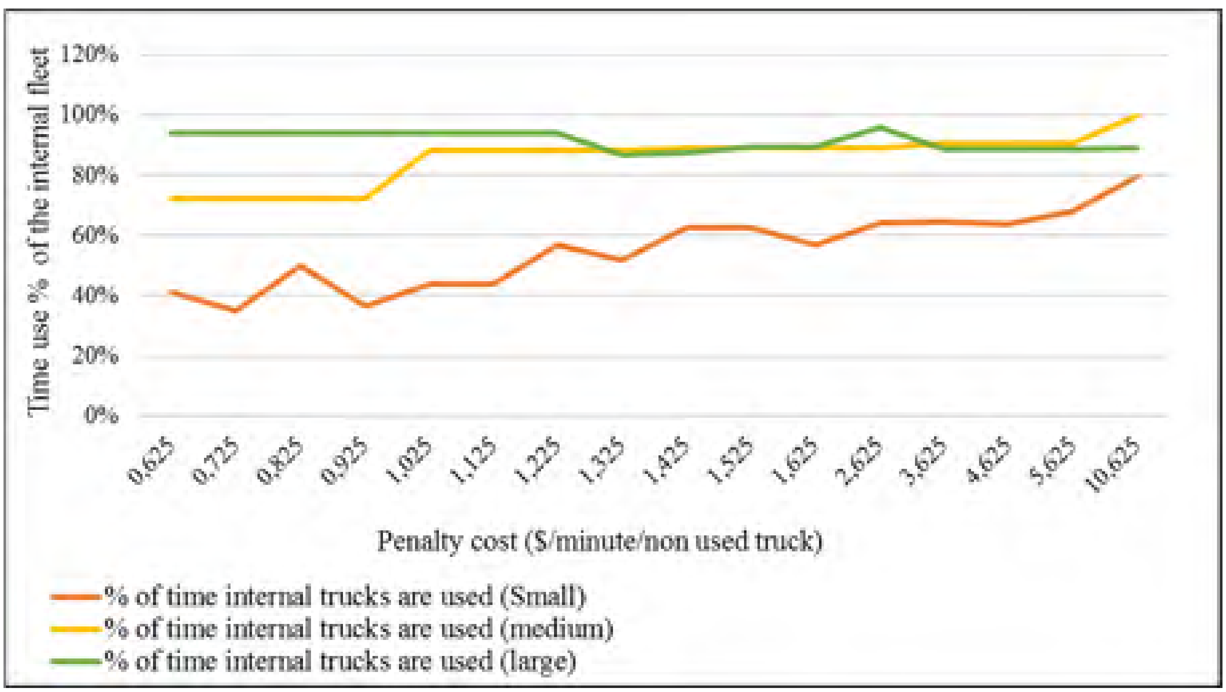

Figure 6.2 Variation of the time percentage of using the internal fleet with respect to the penalty cost for 3 instances

Next, we try to modify some of our initial hypothesis in order to see the outcomes of these changes and test the stability of our heuristic. To achieve that, we have conducted a sensitivity analysis using the instance D-10. The details of this instance are presented in Table-A I-2 of the ANNEX I. In this analysis, we proposed 10 new scenarios (Table 6.4) and each one is characterized by a change in one parameter (the first scenario is the original one with no changes in his parameters). The column ‘Broker extra cost’ donates the change in the initial brokers’ costs. The column ‘Vehicles’ shows the number and capacity of the vehicles used in each scenario. For example, [3, 3, 2] means that we use three vehicles which have capacities of 3 cars, 3 cars and 2 cars, respectively. The final column ‘Vehicles costs’ displays the fixed and variable costs for each type of vehicles.

Table 6.4 Scenarios for sensitivity analysis

Scenario	Dealers	Broker extra cost	Vehicles	Vehicles costs
1 (original)	33	0	[3, 3, 2]	{2: (60 $, 2.4$), 3: (75$, 2.6$)}
2	33	+10$	[3, 3, 2]	{2: (60$, 2.4$), 3: (75$, 2.6$)}
3	33	- 10$	[3, 3, 2]	{2: (60$, 2.4$), 3: (75$, 2.6$)}
4	33	0	[3, 3, 2]	{2: (65$, 2.4$), 3: (80$, 2.6$)}
5	33	0	[3, 3, 2]	{2: (60$, 2.7$), 3: (75$, 2.9$)}
6	33	0	[3, 3, 2]	{2: (55$, 2.4$), 3: (70$, 2.6$)}
7	33	0	[3, 3, 2]	{2: (60$, 2.1$), 3: (75$, 2.3$)}
8	33	0	[3, 2]	{2: (60$, 2.4$), 3: (75$, 2.6$)}
9	33	0	[3, 3, 2, 2]	{2: (60$, 2.4$), 3: (75$, 2.6$)}
10	33	0	[3, 2, 2]	{2: (60$, 2.4$), 3: (75$, 2.6$)}
11	33	-	[2, 2]	{2: (60$, 2.4$), 3: (75$, 2.6$)}

For example, {2: (60 $, 2.4$), 3: (75$, 2.6$)} means that the vehicle with a capacity of two cars has a fixed cost equals to 60$ and a variable cost equals to 2,4$/km, and the vehicle with a capacity of three cars has a fixed cost equals to 75$ and a variable cost equals to 2,6$/km.

In Table 6.5, we present the results of the proposed scenarios. First, for the scenarios 2 and 3, we notice that adding 10$ for brokers' costs increases the fleet utilization by 15%, but the solution is a little bit deteriorated as the total cost increases by 1,1%. On the contrary, reducing the brokers' costs by 10$ decreases the fleet utilization by 6%, but the solution cost is improved by 1,4%. Thus, we show again the importance of using cheap brokers sometimes in the place of using the fleet which can be expensive. Besides, we see from scenarios 4 and 6 that changing the fixed costs leads to an improvement of the solution. Indeed, increasing the fixed cost leads to consider cheaper broker in the solution and decreasing them leads to a cheaper use of the fleet.

Table 6.5 Results of the sensitivity analysis

Scenario	**Routes**				**Total routes**	**Internal dealers**	**Fleet utilization gap**	**Travel cost gap**	**Total cost gap**
	v_1	v_2	v_3	v_4					
1	4	2	2	-	8	21	-	-	-
2	3	3	4	-	10	26	15%	15,7%	1,1%
3	3	2	2	-	7	19	-6%	-6,1%	-1,4%
4	3	2	2	-	7	19	-6%	-6,0%	-0,5%
5	4	2	1	-	7	19	-6%	-4,6%	1,6%
6	4	2	2	-	8	21	0%	-0,7%	-1,9%
7	3	4	1	-	8	23	6%	5,7%	-7,0%
8	4	3	-	-	7	17	-12%	-13,4%	-0,8%
9	4	2	2	3	11	27	18%	19,5%	4,0%
10	4	3	4	-	11	25	12%	12,6%	5,0%
11	5	3	-	-	8	16	-15%	-13,3%	3,6%

In addition, we notice from scenarios 5 and 7 that changing the variable costs affect more the solution than varying the fixed costs. In fact, the increase in the variable costs (scenario 5), leads to a fleet utilization reduction by 6% and a cost augmentation by 1,6%, whereas, the decrease in variable costs (scenario 7), leads to a contrary effect as the fleet utilization grows by 6% and the cost improves by 7%. This is explained by the fact that the distances of the routes are more relevant to consider in the minimization of the total cost than the type of vehicle to use for each trip.

Moreover, we remark that eliminating a vehicle (scenario 8) decreases the fleet utilization by 12% and the number of served dealers by 4. However, adding a vehicle (scenario 9) increases the fleet utilization by 18% and the number of served dealers by 7. Interestingly, we observe that when changing the capacity of the second vehicle from three cars to two cars (scenario 10), the fleet utilization is improved. The reason is that the fleet can do more routes of two dealers that doing routes of three dealers as the routes with two dealers are more flexible. But this is not always true as other instances might have fewer flexible schedules and more constraints to satisfy. Finally, using two vehicles with a capacity of 2 cars (scenario 11), we see that the number of routes did not change compared to the original scenario with three vehicles, but the number of visited dealers is reduced by 5. From these scenarios, we recommend the company to keep its actual private fleet and try to balance its assignments between the fleet and the external carriers in a way that minimizes their operational costs. Another suggestion is, if the company wants to modify its private fleet, then it should consider adding one vehicle with a capacity of three cars as this is the best scenario with respect to the fleet utilization and the cost of the solution.

In conclusion, we can say that our heuristic algorithm performs well and obtain good-quality results that satisfy the business requirement of the company. In fact, the heuristic optimizes the fleet utilization (54%) in comparison with manual routing (20%). In addition, it minimizes the total routing cost by 11% with an optimized assignation of the dealers between the private fleet (54%) and the external carriers (46%). From the sensitivity analysis, we can conclude that changing the initial hypothesis led to variations in the final solution. This can help the decision-makers in choosing the best scenario for their routes' planning. Also, we can see that our heuristic has stable performance and obtain the predicted results for the proposed scenarios. Finally, in comparison with the mathematical model, the heuristic performs much better for solving larger instances in shorter computation time as it can solve a whole day of routing with a big number of dealers in only one hour.

CONCLUSION

In this project, we studied a reverse logistics vehicle routing problem of a Canadian company that collects the end-of-life vehicles (ELVs) from a group of dealers and accumulates them at its warehouse for parts resale or waste recycling. At the moment, the planning process of ELVs collection is managed manually based on the operators' experience. The recent increase in the number of ELVs to collect leads to an increase in transportation costs and inconvenient delays to suppliers. Therefore, there was a need for the development of the appropriate optimization techniques to minimize the transportation costs and reduce the complexity of routes planning related to the collection of ELVs.

This project helped us to make decisions about the following main issues: (i) what is the minimum cost to serve all dealers using the internal fleet and the external carriers, (ii) What dealers should be served by the internal fleet and what dealers should be served by the external carriers and, (iii) in which sequence should the vehicles visit the dealers. This problem can be considered as multi-attribute or rich VRP since it regroups many variants of the classical VRP problem into one complex problem such as capacitated VRP, VRP with time-windows, heterogeneous VRP, VRP with loading constraints, multi-trip VRP, and VRP with private fleet and external carriers. To the best of our knowledge, this problem was not considered until now in the literature, as all the current versions of VRPs do not consider multi-trip aspect, private fleet and multiple external carriers' assignments and loading constraints together in the same mathematical model or heuristic algorithm. Besides, although the cars' length has been included in auto-carrier loading optimization, the specific characteristics of ELVs (body, burned, without wheels, etc.) have not been considered before in the VRP literature.

We propose a mixed-integer linear programming formulation which can be solved by a state-of-the-art commercial solver such as LINGO for small instances. We tested our mathematical model using 13 small-generated instances ranging from instances with 5 dealers to instances with 20 dealers. We notice that the model can obtain optimal solutions only for the small instances with 5, 7, 10 and 12 dealers, respectively, within three hours of computation time.

For the instances with 15 dealers, the LIINGO solver can only obtain feasible solutions within the allowed time limit of three hours. The average percent deviation from the best objective of these solutions is between 9% and 11%. For the instance with 20 dealers, the gap increases to 29% from the theoretical optimal solution proposed by the solver. This proves the limitation of the exact solution to solve this kind of complex problems, especially on an industrial scale. Thus, we develop a heuristic algorithm to minimize the computation time while guaranteeing good-quality feasible solutions even for large instances.

To assess the performance of our heuristic, we conducted a case study that consists of optimizing the routes' planning of the company for November 2018. We used the 30 different instances of the month and we compared the results of the manual planning with the results of our algorithm. The heuristic took between 20 minutes and 40 minutes to solve each instance depending on the number of dealers to be served. Concerning the results, we noticed that the utilization of the internal fleet is increased by 34% compared to manual planning currently used by the firm. In addition, the assignment of the dealers between the private fleet and the external carriers is also optimized as it goes from 20% and 80%, respectively for the dealers visited by the private fleet and the external carriers, to 54% and 46%. Also, the total cost, which is the sum of the travel cost plus the external carriers' cost, is minimized by 16091,7$ for the 30 days which constitutes a saving of 16,7%.

Finally, we performed a sensibility analysis by proposing different scenarios in which we change some of the initial hypothesis imposed by the company. In each scenario, we change the value of one parameter to see the outcome of this variation on the final solution and test the stability of our heuristic. The analysis proves the excellent performance of our algorithm as it obtains the predicted results and it can help the company to know the degree of flexibility of their restrictive hypothesis. Also, based on some scenarios, we recommend the company not to use the fleet to its maximum capacity and should consider outsourcing more customers to the external carriers to minimize the total transportation cost. This is because, at some point, the internal vehicles will begin in doing expensive routes (routes with longer travel times and fewer dealers) and it is more advantageous for the company, in this case, to deal with the carriers who usually have cheaper prices.

Before finishing, we present some limitations of our study, which can be explored in future researches. First, our problem considers only one single depot in the generation of the routes. We can extend the model to consider a multi-depots problem in which vehicles start and end their routes at different depots. In addition, we may consider that the vehicles do not return to the depot after servicing the last customer on the route, which gives us the idea of using the open VRP variant. Second, in our study, we do not optimize the routes of the external carriers. This problem can be addressed by considering the outsourced dealers in the generation of our routes. Third, we can take into account the environmental perspective in our problem by incorporating the energy conception and the pollutant emissions of the vehicles in the routing. This variant is known as green VRP.

Furthermore, our problem is considered to be a single-period deterministic problem as we perform daily planning and the demand is known in advance. In future research, we can explore a multi-period model that can optimize the routing of several days at the same time. Also, we can extend the model to be dynamic or stochastic instead of deterministic. In dynamic VRP, some data is not known in advance but becomes available during operation. If some data are not known in advance but are described by a random variable with a given probability distribution, the VRP is stochastic.

As for solution methods, we can consider using a known metaheuristic algorithm such as Tabu Search and integrate our developed heuristic into the metaheuristic for solution construction and improvement. This will help us in the future to consider more complex problems and solve larger instances.

ANNEX I

INSTANCES FOR HEURISTIC VALIDATION AND SENSITIVITY ANALYSIS

We present here the instances J-1 (5 dealers), J-2 (10 dealers), J-3 (15 dealers) and J-4 (20 dealers) for the heuristic validation in Table-A I-1. Also, we display the instance D-10 for the sensitivity analysis in Table-A I-2. For both tables, the columns 'Start date' and 'End date' are the time windows. The column '4x4' shows if the car has a four-wheel-drive driveline. The column 'Car type' displays the type of the car: G (big), M (medium) and S (small). The column 'Can be towed' states that if it is not obligatory to use the platform of the vehicle to collect the car. The column 'Broker price' shows the minimum broker cost that can be assigned to that dealer if it is served by an external carrier.

Table-A I-1 Instances for heuristic validation

Instance	Dealer	Towing Zip	Start Date	End Date	4x4	Car type	Can be towed	Broker Price
	1	H7P 5H5	08:00:00	12:00:00	NO	G	YES	65
	2	H7A 0C3	08:00:00	12:00:00	NO	G	YES	80
	3	J7M 1J5	12:00:00	16:00:00	NO	G	YES	70
	4	J6Z 1K8	12:00:00	16:00:00	NO	M	NO	60
J-1	5	H9H 2R9	16:00:00	20:00:00	NO	G	YES	60
	6	J7R 6G4	08:00:00	12:00:00	NO	G	YES	50
	7	H1R 1X3	08:00:00	12:00:00	NO	M	NO	65
	8	J5Z 3A4	12:00:00	16:00:00	YES	M	YES	65
	9	J0N 1P0	12:00:00	16:00:00	NO	G	NO	60
J-2	10	H9S 2E6	16:00:00	20:00:00	NO	M	NO	75
	11	H1Y 2L3	12:00:00	16:00:00	NO	G	YES	65
	12	J7P 3N6	12:00:00	16:00:00	NO	G	YES	50
	13	H1A 1C9	08:00:00	12:00:00	NO	S	YES	65

Table-A I-1 Instances for heuristic validation (continued)

Instance	Dealer	Towing Zip	Start Date	End Date	4x4	Car type	Can be towed	Broker Price
	14	J7K 3C2	16:00:00	20:00:00	NO	S	YES	65
J-3	15	J7K 3Y5	16:00:00	20:00:00	NO	G	YES	75
	16	H2C 2C8	08:00:00	12:00:00	NO	M	YES	70
	17	J7R 4K3	12:00:00	16:00:00	NO	G	YES	75
	18	H7J 1G7	16:00:00	20:00:00	NO	M	YES	60
	19	H3W 2E9	16:00:00	20:00:00	NO	S	YES	50
J-4	20	H9A 2N2	16:00:00	20:00:00	NO	M	YES	50

Table-A I-2 Instance D-10 for sensitivity analysis

Dealer	Towing Zip	Start Date	End Date	4x4	Car type	Can be towed	Broker Price
1	H4E3B9	08:00:00	12:00:00	NO	S	YES	80
2	H8Z3E1	08:00:00	12:00:00	NO	M	YES	75
3	H1L4N3	08:00:00	12:00:00	NO	M	YES	65
4	H7H1A6	08:00:00	12:00:00	NO	S	YES	50
5	H1Z3A3	08:00:00	12:00:00	NO	M	YES	65
6	H8Z2X9	08:00:00	12:00:00	NO	M	YES	75
7	H1E2X1	08:00:00	12:00:00	NO	M	YES	65
8	H3A3G5	08:00:00	12:00:00	NO	G	YES	80
9	H9H5N3	08:00:00	12:00:00	NO	S	YES	75
10	H1B2N2	08:00:00	12:00:00	NO	G	YES	65
11	H7X2S2	08:00:00	12:00:00	NO	M	YES	60
12	H3S1B8	08:00:00	12:00:00	NO	M	NO	70
13	H7R4N1	09:30:00	12:00:00	NO	G	YES	60

Table-A I-2 Instance D-10 for sensitivity analysis (continued)

Dealer	Towing Zip	Start Date	End Date	4x4	Car type	Can be towed	Broker Price
14	H1G5J5	12:00:00	16:00:00	NO	M	YES	65
15	H4E3W4	12:00:00	16:00:00	NO	S	NO	80
16	H1G1K4	12:00:00	16:00:00	NO	S	YES	65
17	J7K1C5	12:00:00	16:00:00	NO	G	NO	60
18	H7P1L5	12:00:00	16:00:00	NO	M	YES	60
19	H4N3H9	12:00:00	16:00:00	NO	S	NO	70
20	H1Z3M4	12:00:00	16:00:00	NO	G	VRAI	65
21	J6W3S9	12:00:00	16:00:00	NO	S	YES	65
22	H7N3L2	12:00:00	16:00:00	NO	M	YES	50
23	J7H1H3	12:00:00	16:00:00	NO	G	YES	65
24	H3N2J4	12:00:00	16:00:00	NO	G	YES	70
25	J6X2N9	12:00:00	16:00:00	NO	M	YES	70
26	H2P1Z3	16:00:00	20:00:00	NO	S	YES	65
27	H4M1S4	16:00:00	20:00:00	NO	S	YES	70
28	H1H0A3	16:00:00	20:00:00	NO	G	YES	65
29	J7A1P2	16:00:00	20:00:00	NO	M	YES	60
30	H1M2X4	16:00:00	20:00:00	NO	G	YES	65
31	H7L4W7	16:00:00	20:00:00	NO	M	YES	50
32	J7M1W5	16:00:00	20:00:00	NO	M	NO	80
33	H7C2C8	16:00:00	20:00:00	NO	G	NO	50

ANNEX II

CIGI QUALITA CONFERENCE PAPER

In this annex, we present our conference paper presented during the CIGI QUALITA conference of 2019. The reference of the paper is as follow:

Khabou, A., Chaabane, A., & Ouhimmou, M. (June, 2019). *Solving a reverse logistics routing problem for the collection of end-of-life vehicles.* Paper presented at CIGI QUALITA conference, Montreal, Canada.

Solving a reverse logistics routing problem for the collection of end-of-life vehicles

Khabou Ahmed[1], Chaabane Amin[2], Ouhimmou Mustapha[3]

[1,2,3]*Department of automated manufacturing engineering*
Ecole de Technologie Supérieure
Montreal, Canada
[1] Ahmed.Khabou.1@ens.etsmtl.ca, [2] amin.chaabane@etsmtl.ca , [3] Mustapha.Ouhimmou@etsmtl.ca

***Abstract*— Firms operating in the purchasing of end-of-life vehicles (ELVs) have significant challenges related to the fact that most the purchased ELVs must be collected efficiently to maximize their operational costs. In this paper, we study the reverse logistics problem of such a firm that aims to collect ELVs from its dealers and we called as the reverse logistics vehicle routing problem (RLVRP). We propose a mixed-integer linear programming (MILP) model to solve the RLVRP. Although prior research has made some crucial contributions to model and solve the VRP, the specific case study in this paper combines different types of constraints such as customer assignment to the private fleet or an external carrier, time-windows, multi-trip and loading sequences. Numerical experiments were carried out on real case data collected from a Canadian company operating in the collection of ELVs.**

Keywords— Reverse logistics, vehicle routing problem, time-windows, heterogenous fleet, multi-trip, multiple external carriers, loading sequences.

1. Introduction

Traditionally, supply chains have been considered as the linear movement of good through distribution channels from suppliers to manufacturers, wholesalers, retailers, and finally to consumers (Cruz-Rivera & Ertel, 2009). In recent years, the research field of supply chain management has been extended by tasks referring to reverse logistics flow such as product recovery, refurbishing or recycling. These tasks are part of the end-of-life phase of products and complement the traditional supply chains by closing the loop to have the so called, closed-loop supply chains (Schultmann et al., 2006).

Nowadays, the role of reverse logistics has been developed a lot such that it now plays a major part in the success of many companies. From an economic point of view, the reverse logistics represent direct incomes from reduced consumption of raw-materials, from adding value to recovered material and from cost reduction on waste treatment. This is encouraged different companies to become active in this new area of management (Schultmann et al., 2006).

In this paper, we consider a part of the reverse supply chain of vehicles which is the collection of ELVs. More specifically, we try to optimize the routes planning for ELVs collection. The problem of route planning is known by the research community as the vehicle routing problem (VRP). The VRP is one of the most widely studied combinatory problem in operation research. The high interest of the research community in the different variants of VRP is not only motivated by its difficulty as combinatorial optimization problem but also by its practical relevance. Large number of real-world applications has shown that solving the VRP yields to substantial savings in transportation costs (Stefan Irnich et al., 2014).

Our objective is to propose a suitable model for route planning of ELVs collection that takes into account the following company-specific constraints:

- Customers assignment: customers may either be served by the company's private fleet or by an external carrier, which serves them directly at a predefined cost.
- Time-widows constraints : the routes should be synchronized with the time windows already specified in advance.
- Heterogeneous fixed fleet of vehicles: the fleet has different types of vehicles (auto-carriers) with different capacities.
- Multi-trip: Due to the limited number and capacities of auto-carriers, the model should consider performing multi-trip routing.
- Loading constraints: loading sequences of the different ELVs should be considered in the generation of the routes and the vehicle assignment.

The rest of the paper is organized as follows. A brief literature review on the used variants of vehicle routing problem in this paper is provided in Section 2. A mathematical programming formulation is given in Section 3. Section 4 reports the results of the computational tests. Finally, conclusion is in Section 5.

2. literature review

In this paper, we focus on a collection problem of ELVs called as reverse logistics vehicle routing problem (RLVRP), which is faced by a firm involved in product recovery. The firm is mainly interested in the transportation of the ELVs from dealers to the depot using capacitated vehicles. This problem can be considered as rich or multi-attribute VRP as it regroups many variants of the classical VRP into one single problem. Thus, our literature review mainly includes the line of research on these different variants of VRP.

(Dantzig & Ramser, 1959) introduced the ‘‘Truck Dispatching Problem” modeling how a fleet of homogeneous trucks could serve the demand for oil of a number of gas stations from a central hub and with a minimum traveled distance. After five years, (Clarke & Wright, 1964) generalized this problem to a linear optimization problem as follow: how to serve a set of customers geographically dispersed around a central depot, using a fleet of trucks with varying capacities. This became known as the Vehicle Routing Problem, one of the most widely studied topics in the field of operations research. The VRP is the generalisation of the Travelling Salesman Problem (TSP) where a traveling salesman wants to visit each of a set of towns exactly once, starting from and returning to his home town in the shortest possible way (Jünger et al., 1995).

The most classical version of VRP, known as the capacitated VRP (CVRP), consists of finding a set of routes of minimum cost such that every customer is serviced exactly by one vehicle, each route starts and ends at the depot and the total demand serviced by a route does not exceed vehicle capacity (Fig. 3.1). Nowadays, VRP problems incorporate real-life constraints such as time windows, number of depots, nature of demand, type of vehicles and many others. A classification of side constraints occurring in real-life VRPs is provided by (Van Breedam, 1995). A very large number of papers and books deal with the TSP and VRP such as (Gutin & Punnen, 2002), (Toth & Vigo, 2001) and (Toth & Vigo, 2014). An excellent state of the art classification and review of the VRP is provided by (Braekers et al., 2016).

The VRP with time-windows (VRPTW) is an extension of the CVRP in which the service of each customer must be within a time interval, called a time-window. The depot is represented by the two nodes 0 and n+1 with reference to a source node and a sink node, respectively. A feasible solution for the VRPTW is obtained by an elementary path from the source to the sink. The converse path however may not represent a feasible route as it can violate the time-windows (Desaulniers et al., 2014).

Another extension to the CVRP, known as the heterogeneous VRP (HVRP), considers a group of vehicles that can differ in capacity, variable costs, fixed costs and speed. In HVRP, we have a fleet of vehicles made up of |P| different vehicle types, i.e. the fleet K is partitioned into subsets of homogeneous vehicles $K = K^1 \cup K^2 \cup \ldots \cup K^{|P|}$. Each vehicle type $p = 1,\ldots,|P|$ has capacity Q_p, and may also have a fixed cost FC_p and a specific traveling cost c^p_{ij} along each arc modelling the route of our problem (Stefan Irnich et al., 2014). A comprehensive review of the existing work on HVRPs is presented by (Koç, Bektaş, Jabali, & Laporte, 2016).

The CVRP could be also extended to have a multi-route aspect and become known as multi-trip VRP (MTVRP). While in the CVRP each vehicle can only perform one route, in the MTVRP, vehicles may perform several routes over a planning horizon T. Given some routes with durations T1, T2,…,Tp, a single vehicle may perform them if $T1 + T2 + \ldots + Tp \leq T$ holds. This situation is imposed especially when the vehicle capacity is limited, or other constraints impose a small number of services per route such as the number of available vehicles. Hence, feasible solutions with a limited fleet of size |K| can only be achieved when vehicles are reused (Stefan Irnich et al., 2014). (Chabot, 2015) studied a multi-attribute VRP in the context of biomedical simples' transportation. He provided a mathematical formulation considering different VRP constraints such as time-windows and maximum time limit for a route along with the multi-trip aspect. His objective was to minimize the total fixed cost plus the total variable cost of the vehicles. Our model considers new constraints that have not been studied in (Chabot, 2015) such as customer assignments to a private fleet or an external carrier, and the loading sequences. Moreover, our objective function considers the cost of using the external carriers in addition to the fixed and variable costs of the vehicles.

Finally, our problem considers the outsourcing option of some of the dealers to an external carrier if there are not enough private vehicles to serve them. This variant is known as VRP with private fleet and external carrier (VRPPC) or VRP with outsourcing (Archetti et al., 2014). A single-depot routing problem with outsourcing options was first introduced by (Chu, 2005). The problem considers a private fleet of vehicles with limited capacity and a set of customers with known demand. Each customer can be served either by the private fleet which then incurs travel costs as in standard VRP or outsourced to a common carrier, and in such a case only fixed service costs must be paid. The objective is to minimize the total cost involving fixed costs for vehicles, variable travel costs, and fixed costs for orders performed by the common carrier. Our problem adds new features to the problem of (Chu, 2005) as it considers multiple external carriers that can have different costs depending on the customer's region. Hence, the cost of the external carrier should be considered in the global routing decision. In addition, the external carrier is called only after utilizing the maximum capacity of the company's internal fleet.

3. Mathematical model

In the following, we describe the inputs and the requirements of our problem and we present the developed mathematical model for the RLVRP. We use the term vehicle to denote an auto-carrier, the term car to donate the transported item on the vehicle, the term dealer to denote a customer who want to sell his car, and the term broker to donate the external carrier that we will use to collect some of the cars.

3.1 Network

Given a complete graph G = (I, A), where $I = \{0, 1, \ldots, n, n+1\}$ is the set of nodes (locations) and $A=\{(i, j): i < n+1; j > 0; i \neq j\}$ is the set of arcs connecting each node. Node 0 corresponds to the depart depot and node n+1 corresponds to the end depot which is a duplication of the depart depot, whereas vertices $N = \{1,\ldots, n\}$ correspond to the n dealers to be visited. A distance $d_{ij} \geq 0$ and a travel time $t_{ij} \geq 0$ are associated to each arc $(i, j) \in A$ and respectively satisfy the triangle inequality i.e. $t_{ij} + t_{jk} \geq t_{ik}$ and $d_{ij} + d_{jk} \geq d_{ik}$. We suppose also that $d_{ij} = d_{ji}$ and $t_{ij} = t_{ji}$,$\forall$ $(i, j) \in A$. Each location $i \in I$ has its service time $s_i \geq 0$ (s_i is the loading time for $i \in N$, s_0=0 and $s_{(n+1)}$ is the unloading time) and each location should be visited within a prespecified time window $[o_i, c_i]$ $(0 \leq o_i \leq c_i)$ with o_i and c_i are respectively are the

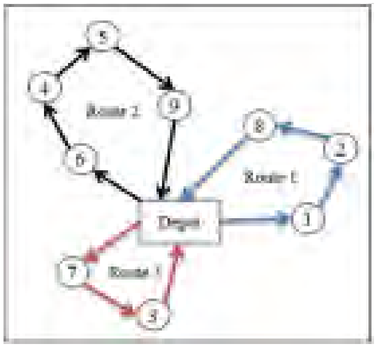

Fig. 3.1. Example of a classic VRP with 9 clients and 3 routes

opening time and the closing time of the time-window for location $i \in I$. Customer service must start within the time-window, but the vehicle may wait at a customer location if it arrives before the beginning of the time window. Note that the time-window for the depot is limited by the work schedule of the vehicle v which is from 8 a.m to 8 p.m that correspond to a total of twelve hours each day.

3.2 Fleet and routes

The set of vehicles is donated by V. The set is composed of fixed heterogeneous fleet of vehicles where each vehicle $v \in V$ has a maximum capacity q_v, duration of working day T_v, fixed operating cost f_v for each time a vehicle leaves the depot and variable cost rate per distance unit c_v. In this model, we suppose that we have two vehicle types: We donate the set of vehicle types with $q_v= 3$ as the set $V_1 \subseteq V$ (Fig. 3.2) and the set of vehicle type with $q_v= 2$ as the set $V_2 \subseteq V$ (Fig. 3.3). Vehicles of type v_1 can hold two cars on their platform and the third is towed, whereas vehicles of type v_2 can hold only one car on the platform and the other is towed.

Vehicles may perform several routes in the same day. This is due to the limited vehicle capacity q_v and to the limited number of available vehicles. Thus, feasible solutions with a limited fleet of size $|V|$ can only be achieved when vehicles are reused to perform several routes if necessary. The set of routes is denoted as R. A route $r \in R$ has a duration T_r and a single vehicle may perform several routes with durations $T_1, T_2,\ldots, T_r$ if $T_1 + T_2+\ldots+ T_r \leq T_v$ holds.

3.3 Pickups

The number of cars that need to be collected from a dealer $i \in N$ consists of a_i cars (we suppose that $a_i = 1$, $i \in N$ and $a_0 = a_{(n+1)} = 0$). Each car is characterised by a length (large, medium or small), a specific driveline (4x4, 2x4, 4x2, etc.), a car status (damaged, new, burned, etc.) and a wheel's status (with wheels, without wheels, have only front wheels, etc.). These different statuses allow us to define the adequate loading sequences that should be respected in the generation of the routes. In our problem, we have three loading constraints that should be respected:

- A 4x4 car should be on the first position for both vehicle types v_1 and v_2.
- A body, burnt, damaged or without wheels car should be on the platform of the vehicle (either first or second position if $v \in V_1$ or on first position if $v \in V_2$)
- A vehicle v_1 cannot hold two large cars (or one large and one medium) on the platform at the same time.

Each request for pickup can be fulfilled by two transportation options: The first option is to use an internal vehicle. The other transportation option is offered by a set of brokers (logistic service providers or freight forwarding companies). The set of brokers is donated by B. The assignment of a dealer i to a broker $b \in B$ incur a cost p_{ib} that depends on the broker to be used and the location of the dealer to be served. In our model, we suppose that the brokers do not have any capacity limits but accepts every subcontracted quantity and that the internal fleet should be used to its maximum capacity before giving the opportunity to brokers to collect the cars.

3.4 Requirements

Our problem calls for the assignment of dealers to be served either by an internal vehicle or an external broker as well as the determination of a set of optimized routes for the internal vehicles such that the total internal costs plus total external costs are minimized, and the following conditions are satisfied:

- Each dealer can either be assigned to one broker or visited exactly by one route and one vehicle.
- Each route starts and ends at the same depot after finishing the service for the last dealer.
- The sum of the demands of the dealers in any route does not exceed vehicle capacity.
- The total duration of the routes assigned to the same vehicle does not exceed the vehicle working time.
- Each dealer should be visited with a prespecified time window.
- The specific loading constraints of each vehicle should be respected.

3.5 Parameters

f_v: Unit vehicle operating cost (fixed cost)

c_v: Cost per unit distance traveled for a vehicle v (variable cost)

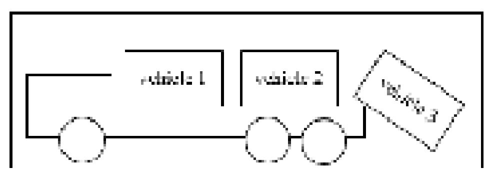

Fig. 3.2. Example of vehicle of type v_1

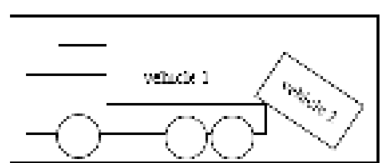

Fig. 3.3. Example of vehicle of type v_2

q_v: Capacity of vehicle v

T_v: Duration of the working day for a vehicle v

d_{ij}: Distance between location i and location j

t_{ij}: Time required to travel from location i to location j

a_i: Number of cars to be collected from dealer i

s_i: Service time for location i

$[o_i, c_i]$: Time window for location i

p_{ib}: Cost of assignment of a dealer i to a broker b

d_i: Takes 1 if a dealer i has 4x4 car (known also as Four-Wheel Drive car), 0 otherwise

nw_i: Takes 1 if a dealer i has a body, burnt, damaged or without wheels car, 0 otherwise

g_i: Takes 1 if a dealer i has large car, 0 otherwise

mid_i: Takes 1 if a dealer i has medium car, 0 otherwise

M: A large number

3.6 Variables

x_{ijrv}: Takes 1 if arc (i, j) ∈ A is used on route r by vehicle v, 0 otherwise.

y_{irv}: Takes 1 if a dealer i is visited on route r by vehicle v, 0 otherwise.

AT_{irv}: Arrival Time of vehicle v for location i on route r.

u_{irv}: Load of the vehicle v right after departing from location i on the route r.

z_{ib}: Takes 1 if a dealer i is assigned to to the external borker b, 0 otherwise.

Note that this formulation makes use of the additional variables u_{irv} to model the Miller–Tucker–Zemlin (MTZ) subtour elimination constraints introduced by (Miller et al., 1960) for the TSP. The advantage of the MTZ-formulation is that it has $O(n^2)$ variables and constraints.

3.7 Model

Minimize

$$\sum_{j\in N\cup\{n+1\}}\sum_{r\in R}\sum_{v\in V} f_v x_{0jrv} \tag{1}$$

$$+\sum_{i\in N\cup\{0\}}\sum_{\substack{j\in N\cup\{n+1\}\\ j\neq i}}\sum_{v\in V}\sum_{r\in R} c_v\, d_{ij} x_{ijrv} + \sum_{i\in N}\sum_{b\in B} p_{ib} z_{ib}$$

Subject to:

$$\sum_{v\in V}\sum_{r\in R} y_{irv} + \sum_{b\in B} z_{ib} = 1 \qquad i\in N \tag{2}$$

$$\sum_{\substack{i\in N\cup\{0\}\\ i\neq j}} x_{ijrv} = \sum_{\substack{i\in N\cup\{n+1\}\\ i\neq j}} x_{jirv} = y_{jrv} \qquad j\in N, r\in R, v\in V \tag{3}$$

$$\sum_{j\in N} x_{0jrv} \le 1 \qquad r\in R, v\in V \tag{4}$$

$$\sum_{i\in N} a_i y_{irv} \le q_v \qquad r\in R, v\in V \tag{5}$$

$$\sum_{j\in N} x_{0j(r+1)v} \le \sum_{j\in N} x_{0jrv} \qquad r\in\{1,\dots,|R|-1\}, v\in V \tag{6}$$

$$u_{irv} - u_{jrv} + q_v x_{ijrv} + (q_v - a_i - a_j)x_{jirv} \le q_v - a_j \tag{7}$$

$i, j\in N,\ i\neq j,\ r\in R,\ v\in V$

$$u_{irv} \le q_v \qquad i\in N,\ r\in R,\ v\in V \tag{8}$$

$$u_{irv} \ge a_i \times y_{irv} \qquad i\in N,\ v\in V,\ r\in R \tag{9}$$

$$AT_{irv} + s_i + t_{ij} - AT_{jrv} \le T_v\,(1 - x_{ijrv}) \qquad i\in N\cup\{0\},\ j\in N\cup\{n+1\},\ r\in R, v\in V \tag{10}$$

$$o_i \times y_{irv} \le AT_{irv} \le c_i \times y_{irv} \qquad i\in N,\ v\in V,\ r\in R \tag{11}$$

$$o_i \le AT_{irv} \le c_i \qquad i\in\{0, n+1\}, v\in V,\ r\in R \tag{12}$$

$$AT_{0rv} \ge AT_{(n+1)v(r-1)} + s_{(n+1)} \qquad v\in V,\ r\in\{2,\dots,|R|\} \tag{13}$$

$$AT_{(n+1)rv} - AT_{01v} \le T_v \qquad v\in V,\ r\in R \tag{14}$$

$$u_{irv} - 1 \le M(1 - d_i \times y_{irv}) \qquad i\in N,\ v\in V,\ r\in R \tag{15}$$

$$u_{irv} - 2 \le M(1 - nw_i \times y_{irv}) \qquad i\in N,\ v\in V_1,\ r\in R \tag{16}$$

$$u_{irv} - 1 \le M(1 - nw_i \times y_{irv}) \qquad i\in N,\ v\in V_2,\ r\in R \tag{17}$$

$$x_{jlrv} \le M(1 - x_{0jrv}) \tag{18}$$

$v\in V_1,\ r\in R,\ j,l\in N, i\neq j, g_j = g_l = 1$

or $g_l = 1$ *and* $mid_j = 1$ or $g_j = 1$ *and* $mid_l = 1$

$$x_{ijrv} = \{0,1\} \qquad (i,j)\in A,\ v\in V, r\in R \tag{19}$$

$$y_{irv} = \{0,1\} \qquad i\in N, v\in V, r\in R \tag{20}$$

$$AT_{irv} \ge 0 \qquad i\in I,\ v\in V, r\in R \tag{21}$$

$$u_{irv} \ge 0 \qquad i\in N,\ v\in V, r\in R \tag{22}$$

$$z_{ib} = \{0,1\} \qquad i\in N,\ b\in B \tag{23}$$

Objective function (1) minimizes the total cost which is the sum of three parts as follow: The first and second parts calculates the total fixed costs and the total variable costs of the internal vehicles, respectively. The third part calculates the cost of assigning the dealers to the external brokers.

Constraints (2) ensure that a dealer i ∈ N is either visited exactly once (by one route r ∈ R and one internal vehicle v ∈ V) or it is assigned to an external broker b ∈ B. Constraints (3) are known as the flow conservation constraints which ensure that if a vehicle v visits a location j ∈ I on route r ∈ R, then it should leave this location after service completion to have a balanced flow. Constraints (4) state that at most one vehicle v ∈ V can go out from the depart depot on a route r ∈ R. Constraints (5) ensure that the total demand of the dealers on a route r ∈ R should not exceed the vehicle capacity. Constraints (6) ensure that with respect to a vehicle v ∈ V, its $(r+1)^{th}$ route is realized only if its r^{th} route has been realized. Constraints (7) are the MTZ subtours elimination constraints which are used together with lower and upper bounds on u_{irv} variables. Constraints (8) and (9) ensure that the load of the vehicle v ∈ V on route r ∈ R right after departing from dealer i ∈ N must be at least equal to the number of cars picked up from that dealer and should not exceed vehicle capacity. Constraints (10) calculate the arrival time to location j ∈ N ∪ {n+1} after visiting its predecessor i ∈ N ∪{0} and ensure that the arrival time of the location j is greater than the sum of the arrival time of location i plus the traveling time and service time of that same location. Constraints (11) and (12) ensure that the arrival time for location i ∈ I is within the time-window of that location. Constraints (13) ensure that the starting time of the r^{th} route from the depart depot is greater then the arrival time of the $(r-1)^{th}$ route to the end depot plus the unloading time at the end depot. Constraints (14) states that the duration of routes performed by vehicle v should not exceed its working time limit. Constraints (15) states that if a dealer i ∈ N has a 4x4 car then it should be visited first on a route r ∈ R using a vehicle v ∈ V. Constraints (16) and (17) state that if a dealer i ∈ N has a body, burnt, damaged or without wheels car then it should be visited either first or second on a route r ∈ R if v ∈ V_1 or first if v ∈ V_2 (i.e. The vehicle should be on the platform of the vehicle). Constraints (18) state that vehicle v ∈ V_1 cannot hold two large cars or one large and one medium on the platform at the same time. Finally, constraints (19) - (23) define the variables of the model.

4. Computational results

In this section, we first describe how we generate the instances of the problem. Then, we report the results of our computations for the developed model. The model was solved using LINGO 15.0 solver.

4.1 Instances generation

To generate our instances, we have three major parameters: the number of dealers, the number of vehicles for each type and the number of routes that can be performed by each vehicle. Using these parameters, we generate 13 different instances ranging from small ones with only 5 dealers to bigger ones with 20 dealers as can be seen in Table 4.1. The depot and the dealers are located within the province of Quebec and particularly in the great area of Montreal. They are directly geolocated using their postal code with the help of the software Supply Chain Guru. Hence, the coordinates and the matrix distance are also generated automatically for the testing. The time matrix is calculated from the distance matrix by dividing the distances by the average speed of the vehicle on the route which is considered to be 60 km/hour. The loading and unloading times are generated from a discrete uniform distribution in the intervals [20, 40] and [15, 30] minutes, respectively. There are three time-windows AM (morning) from 8 a.m. to 12 p.m., PM (afternoon) from 12 p.m. to 4 p.m. and EV (evening) from 4 p.m. to 8 p.m. The time limit of the working day of vehicles is fixed at 12 hours.

Table 4.1. GENERATED INSTANCES

Instances	Dealers	Routes	Vehicles	
			$q_v = 3$	$q_v = 2$
I-1	5	1	1	0
I-2	5	1	1	1
I-3	7	1	1	1
I-4	7	1	1	2
I-5	10	1	1	0
I-6	10	2	1	1
I-7	10	1	1	2
I-8	12	3	1	0
I-9	12	2	1	1
I-10	15	1	2	1
I-11	15	2	1	1
I-12	15	3	1	0
I-13	20	3	1	1

The fixed and variable costs for each type of the vehicles are calculated by the company and include the vehicle depreciation cost, maintenance cost, insurance cost, fuel cost, driver cost, etc. For vehicles of type v_1, f_v= 75$ (fixed cost) and c_v= 2,6 $/km (variable cost) meanwhile vehicles of type v_2, f_v= 60$ and c_v= 2,4 $/km. Finally, the number of brokers depends on the number of vehicles to be collected i.e. the broker is used when needed, and the broker cost depend on the region to be visited (the cost usually varies between 50$ and 150$ per car).

4.2 Results

In this part, we present detailed results of a single instance for model validation and then we provide the testing results of all generated instances of the problem.

4.3 Model validation

To validate the model, we present the results of an example from instance I-6 in Table 4.2. This example includes 10 dealers and uses 2 vehicles (one vehicle for each type) and no more than 2 routes per vehicle. There are 4 routes in total and no vehicle works more than twelve hours per day. In this example, all 10 dealers are visited by the internal fleet since the model try to use the internal trucks at their maximum

Table 4.2. RESULTS OF THE INSTANCE I-6

Vehicle	Start	Dealers			End	Duration
1 Route 1	8:00	8 [4x4] 8:14	7 [L] 8:52	2 9:15	9:45	1h45
1 Route 2	10:00	9 [4x4] [M] 10:28	6 [NW] [M] 10:35	10 11:17	12:00	2h
2 Route 1	8:00	3 [L] 8:21	5 [L] 8:44		9:28	1h28
2 Route 2	9:43	11 [4x4] 10:23	4 [M] 11:09		12:00	2h17

capacities before assigning the dealers to an external broker. The arrival time is written below the number of the dealer. Time windows are not presented to save space, but they are all respected. Loading and unloading times are also respected. The column 'Time' represent the total time of one route (length of the route).

4.4 Model testing

Table 4.3 presents the tests done within the allowed time limit of 3 h or equivalently 10800 seconds using LINGO 15.0 solver. Each instance is tested five times using different groups of dealers all of them located in the Montreal-Laval region. The column 'Cost ($)' represent the average cost obtained by the different test results. The 'Gap (%)' is calculated by the following formula: Gap (%) = |Objective value – best objective| / (best objective) × 100. The column 'CPU(s)' represent the average computational time in seconds.

Finally, the column 'NoD' represents the number of dealers visited by the internal fleet (the remaining nodes are automatically assigned to the external brokers).

We notice that LINGO can obtain optimal solutions only for the small instances with 5, 7, 10 and 12 dealers, respectively, within the allowed time limit. For the instances with 15 dealers, the model can only obtain feasible solutions within the allowed time limit.

The average percent deviation from the best objective of these solutions is between 9% and 11%. For the instance with 20 dealers, the gap begins to increase to 29% which explains the limitation of the mathematical models to solve this kind of complex problems especially on an industrial scale.

5. Conclusion

In this paper, we study the reverse logistics vehicle routing problem (RLVRP) of a company that collects end-of-life vehicles from a group of dealers and accumulates them at its warehouse for part resale or for recycling. This study helped us to make decisions about the following three issues: (i) What is the minimum cost to serve all dealers using the internal fleet and the external carriers, (ii) What dealers should be served by the internal fleet and what dealers should be served by the brokers and, (iii) How many vehicles from the private fleet should be used for collection and in what sequence should they visit the dealers. This problem can be considered as multi-attribute or rich VRP since it regroups many variants of the classical VRP problem into one complex problem. To the authors' knowledge, this problem was not considered before in the literature, as all the existing versions of VRPs does not involve VRP richness aspect with private fleet and multiple external carriers' assignments.

We propose a mixed-integer linear programming formulation for the problem which can be solved by a state-of-the art commercial solver such as LINGO. However, since the VRP with or without side-constraints is proven to be NP-hard (Lenstra & Kan, 1981), relatively large instances cannot be solved in this way and only small instances can be solved to optimality. Therefore, as a continuation of this study, we need to develop a heuristic algorithm to obtain good-quality results in a reasonable computation time for large instances.

Table 4.3. TEST RESULTS FOR THE GENERATED INSTANCES

Instances	Dealers	Cost($)	Gap(%)	CPU(s)	NoD
I-1	5	381	0	0,6	3
I-2	5	402	0	1,4	5
I-3	7	547	0	2,1	6
I-4	7	505	0	6	7
I-5	10	772	0	23	3
I-6	10	747	0	58	10
I-7	10	815	0	129	9
I-8	12	846	0	838	9
I-9	12	885	0	7365	10
I-10	15	967	9	10800	8
I-11	15	985	10	10800	10
I-12	15	918	11	10800	9
I-13	20	1415	29	10800	15

6. References

Archetti, C., Speranza, M. G., & Vigo, D. (2014). Chapter 10 Vehicle Routing Problems with Profits. Dans *Vehicle Routing - Problems, Methods, and Applications* (2nd éd., pp. 273-297). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/hotlink/toc/id:kpVRPMAE06/vehicle-routing-problems

Braekers, K., Ramaekers, K., & Van Nieuwenhuyse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering, 99*, 300-313. doi: 10.1016/j.cie.2015. 12. 007

Chabot, T. (2015). *Résolution des problèmes de tournées de véhicules pour le transport des échantillons biomédicaux au Québec* (master's dissertation, Université Laval, Québec, Canada). Repéré à https://corpus.ulaval.ca/jspui/handle/ 20.500.11794/26365

Chu, C.-W. (2005). A heuristic algorithm for the truckload and less-than-truckload problem. *European Journal of Operational Research, 165*(3), 657-667. doi: https://doi.org/10.1016/j.ejor.2003.08.067. Repéré à http://www.sciencedirect.com/science/article/pii/S0377221704000888

Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Oper. Res., 12*(4), 568-581. doi: 10.1287/opr.12.4.568

Cruz-Rivera, R., & Ertel, J. (2009). Reverse logistics network design for the collection of End-of-Life Vehicles in Mexico. *European Journal of Operational Research, 196*(3), 930-939. doi: 10.1016/j.ejor.2008.04.041

Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Manage. Sci., 6*(1), 80-91. doi: 10.1287/mnsc.6.1.80

Desaulniers, G., B.G. Madsen, O., & Ropke, S. (2014). Chapter 5 The Vehicle Routing Problem with Time Windows. Dans *Vehicle Routing - Problems, Methods, and Applica-tions* (2nd éd., pp. 117-159). Philadelphia, PA, USA: Society for Industrial and Applied Mathem-atics. Repéré à https://app.knovel.com/web/toc.v/cid:kpVRPMAE06/viewerType:toc/slug:vehicle-routing-problems

Gutin, G., & Punnen, A. (2002). *The Traveling Salesman Problem and Its Variations* (Vol. 12).

Irnich, S., Schneider, M., & Vigo, D. (2014). Chapter 9 Four Variants of the Vehicle Routing Problem. Dans *Vehicle Routing - Problems, Methods, and Applications* (2nd éd., pp. 241-271). Philadephia, PA, USA: Society for Industrial and Applied Mathematics. Retriev-ed from Repéré à https://app.knovel.com/hotlink/toc/id:kpVRPMAE06/vehicle-routing-problems

Irnich, S., Toth, P., & Vigo, D. (2014). Chapter 1 The family of vehicule routing problems. Dans *Vehicle Routing - Problems, Methods, and Applications* (pp. 1-33). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/web/toc.v/cid:kpVRPMAE06/viewerType:toc/root_slug:vehicle-routing-problems

Jünger, M., Reinelt, G., & Rinaldi, G. (1995). chapter 4 The traveling salesman problem. Dans *Handbooks in Operations Research and Management Science* (Vol. 7, pp. 225-330). doi: 10.1016/S0927-0507(05)80121-5

Koç, Ç., Bektaş, T., Jabali, O., & Laporte, G. (2016). Thirty years of heterogeneous vehicle routing. *European Journal of Operational Research, 249*(1), 1-21. doi: https://doi.org/ 10.1016/j.ejor.2015.07.020. Repéré à http://www.sciencedirect.com/science/article/pii/S0377221 715006530

Lenstra, J. K., & Kan, A. H. G. R. (1981). Complexity of vehicle routing and scheduling problems. *Networks, 11*(2), 221-227. doi: doi:10.1002/net.3230110211. Rep-éré à https://onlinelibrary.wiley.com/doi/abs/10.1002/net.3230110211

Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer Programming Formulation of Traveling Salesman Problems. *J. ACM, 7*(4), 326-329. doi:10.1145/321043.321046

Schultmann, F., Zumkeller, M., & Rentz, O. (2006). Modeling reverse logistic tasks within closed-loop supply chains: An example from the automotive industry. *European Journal of Operational Research, 171*(3), 1033-1050. doi: 10.1016/j.ejor.2005.01.016

Toth, P., & Vigo, D. (2014). *Vehicle Routing - Problems, Methods, and Applications* (2nd éd.). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/hotlink/toc/id:kpVRPMAE06/vehicle-routing-problems

Toth, P., & Vigo, D. (Éds.). (2001). *The vehicle routing problem*. Philadelphia, PA, USA: Society for Industrial and Applied Mathem-atics. doi: https://doi.org/10.1137/1 978089 8718515

Van Breedam, A. (1995). Vehicle Routing: Bridging the gap between theory and practice. *Belgian Journal of Operations Research, Statistics and Computer Scien-ce, 35*(1), 63-80.

BIBLIOGRAPHY

Anaya-Arenas, A. M., Chabot, T., Renaud, J., & Ruiz, A. (2016). Biomedical sample transportation in the province of Quebec: a case study *International Journal of Production Research, 54*(2), 602-615. doi: https://doi.org/10.1080/00207543.2015.10 18455.

Aras, N., Aksen, D., & Tuğrul Tekin, M. (2011). Selective multi-depot vehicle routing problem with pricing. *Transportation Research Part C: Emerging Technologies, 19*(5), 866-884. doi: https://doi.org/10.1016/j.trc.2010.08.003

Archetti, C., Speranza, M. G., & Vigo, D. (2014). Chapter 10 Vehicle Routing Problems with Profits. Dans *Vehicle Routing - Problems, Methods, and Applications* (2nd éd., pp. 273-297). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/hotlink/toc/id:kpVRPMAE06/vehicle-routing-problems

Battarra, M., Monaci, M., & Vigo, D. (2009). An adaptive guidance approach for the heuristic solution of a minimum multiple trip vehicle routing problem. *Computers & Operations Research, 36*(11), 3041-3050. doi: https://doi.org/10.1016/j.cor.2009.02.008. Repéré à http://www.sciencedirect.com/science/article/pii/S0305054809000410

Beullens, P., Van Oudheusden, D., & Van Wassenhove, L. N. (2004). Collection and Vehicle Routing Issues in Reverse Logistics. Dans *Reverse Logistics* (pp. 95-134). doi: https://doi.org/10.1007/978-3-540-24803-3_5

Bolduc, M.-C., Renaud, J., & Boctor, F. (2007). A heuristic for the routing and carrier selection problem. *European Journal of Operational Research, 183*(2), 926-932. doi: https://doi.org/10.1016/j.ejor.2006.10.013. Repéré à http://www.sciencedirect.com/science/article/pii/S0377221706010460

Braekers, K., Ramaekers, K., & Van Nieuwenhuyse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering, 99*, 300-313. doi: https://doi.org/10.1016/j.cie.2015.12.007

Ceschia, S., Di Gaspero, L., & Schaerf, A. (2011). Tabu search techniques for the heterogeneous vehicle routing problem with time windows and carrier-dependent costs. *Journal of Scheduling, 14*(6), 601-615. doi: https://doi.org/10.1007/s10951-010-0213-x. Repéré à https://doi.org/10.1007/s10951-010-0213-x

Chabot, T. (2015). *Résolution des problèmes de tournées de véhicules pour le transport des échantillons biomédicaux au Québec* (master's dissertation, Université Laval, Québec, Canada). Repéré à https://corpus.ulaval.ca/jspui/handle/20.500.11794/26365

Chu, C.-W. (2005). A heuristic algorithm for the truckload and less-than-truckload problem. *European Journal of Operational Research, 165*(3), 657-667. doi: https://doi.org/10

.1016/j.ejor.2003.08.067. Repéré à http://www.sciencedirect.com/science/article/pii/S-0377221704000888

Cin, E., & Kusakcı, A. O. (2017). A Literature Survey on Reverse Logistics of End of Life Vehicles. *Southeast Europe Journal of Soft Computing, 6*(1). doi: http://dx.doi.org/10.21533/scjournal.v6i1.132

Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Oper. Res., 12*(4), 568-581. doi: https://doi.org/10.1287/opre.12.4.568

Cruz-Rivera, R., & Ertel, J. (2009). Reverse logistics network design for the collection of End-of-Life Vehicles in Mexico. *European Journal of Operational Research, 196*(3), 930-939. doi: https://doi.org/10.1016/j.ejor.2008.04.041

Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. *Manage. Sci., 6*(1), 80-91. doi: https://doi.org/10.1287/mnsc.6.1.80

Demirel, E., Demirel, N., & Gökçen, H. (2016). A mixed integer linear programming model to optimize reverse logistics activities of end-of-life vehicles in Turkey. *Journal of Cleaner Production, 112*, 2101-2113. doi: https://doi.org/10.1016/j.jclepro.2014.10.079. Repéré à http://www.sciencedirect.com/science/article/pii/S0959652614011226

Desaulniers, G., B.G. Madsen, O., & Ropke, S. (2014). Chapter 5 The Vehicle Routing Problem with Time Windows. Dans *Vehicle Routing - Problems, Methods, and Applications* (2nd éd., pp. 117-159). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/web/toc.v/cid:kpVRPMAE06/vehicle-routing-problems

Despaux, F., & Basterrech, S. (2014). A study of the Multi-Trip vehicle routing problem with time windows and heterogeneous fleet. Dans *2014 14th International Conference on Intelligent Systems Design and Applications* (pp. 7-12). doi: https://doi.org/10.1109/ISDA.2014.7066280

Ene, S., & Öztürk, N. (2015). Network modeling for reverse flows of end-of-life vehicles. *Waste Management, 38*, 284-296. doi: https://doi.org/10.1016/j.wasman.2015.01.007. Repéré à http://www.sciencedirect.com/science/article/pii/S0956053X15000446

Euchi, J. (2017). The vehicle routing problem with private fleet and multiple common carriers: Solution with hybrid metaheuristic algorithm. *Vehicular Communications, 9*, 97-108. doi: https://doi.org/10.1016/j.vehcom.2017.04.005. Repéré à http://www.sciencedirect.com/science/article/pii/S2214209616301449

Gahm, C., Brabänder, C., & Tuma, A. (2017). Vehicle routing with private fleet, multiple common carriers offering volume discounts, and rental options. *Transportation*

Research Part E: Logistics and Transportation Review, 97, 192-216. doi: https://doi.org/10.1016/j.tre.2016.10.010. Repéré à http://www.sciencedirect.com/science/article/pii/S1366554516300837

Gendreau, M., & Potvin, J.-Y. P. (2010). *Handbook of metaheuristics*. doi: https://doi.org/10.1007/978-1-4419-1665-5. Repéré à https://link.springer.com/book/10.1007 2F978-1-4419-1665-5

Gutin, G., & Punnen, A. (2002). *The Traveling Salesman Problem and Its Variations* (Vol. 12).

Iori, M., & Martello, S. (2010). Routing problems with loading constraints. *TOP, 18*(1), 4-27. doi: https://doi.org/10.1007/s11750-010-0144-x

Irnich, S., Schneider, M., & Vigo, D. (2014). Chapter 9 Four Variants of the Vehicle Routing Problem. Dans *Vehicle Routing - Problems, Methods, and Applications* (2nd éd., pp. 241-271). Philadephia, PA, USA: Society for Industrial and Applied Mathematics. Retrieved from Repéré à https://app.knovel.com/hotlink/toc/id:kpVRPMAE06/vehicle-routing-problems

Irnich, S., Toth, P., & Vigo, D. (2014). Chapter 1 The family of vehicle routing problems. Dans *Vehicle Routing - Problems, Methods, and Applications* (pp. 1-33). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/web/toc.v/cid:kpVRPMAE06/viewerType/root_slug:vehicle-routing-problems

Jünger, M., Reinelt, G., & Rinaldi, G. (1995). Chapter 4 The traveling salesman problem. Dans *Handbooks in Operations Research and Management Science* (Vol. 7, pp. 225-330). doi: https://doi.org/10.1016/S0927-0507(05)80121-5

Koç, Ç., Bektaş, T., Jabali, O., & Laporte, G. (2016). Thirty years of heterogeneous vehicle routing. *European Journal of Operational Research, 249*(1), 1-21. doi: https://doi.org/10.1016/j.ejor.2015.07.020. Repéré à http://www.sciencedirect.com/science/article/pii/S0377221715006530

Krajewska, M. A., & Kopfer, H. (2009). Transportation planning in freight forwarding companies: Tabu search algorithm for the integrated operational transportation planning problem. *European Journal of Operational Research, 197*(2), 741-751. doi: https://doi.org/10.1016/j.ejor.2008.06.042. Repéré à http://www.sciencedirect.com/science/article/pii/S0377221708005080

Kuşakcı, A. O., Ayvaz, B., Cin, E., & Aydın, N. (2019). Optimization of reverse logistics network of End of Life Vehicles under fuzzy supply: A case study for Istanbul Metropolitan Area. *Journal of Cleaner Production, 215*, 1036-1051. doi: https://

doi.org/10.1016/j.jclepro.2019.01.090. Repéré à http://www.sciencedirect.com/science/article/pii/S0959652619301015

Laporte, G., Ropke, S., & Vidal, T. (2014). Chapter 4 Heuristics for the Vehicle Routing Problem. Dans *Vehicle Routing - Problems, Methods, and Applications* (2nd éd., pp. 87-116). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/toc/id:kpVRPMAE06/vehicle-routing-problems

Lenstra, J. K., & Kan, A. H. G. R. (1981). Complexity of vehicle routing and scheduling problems. *Networks, 11*(2), 221-227. Repéré à https://onlinelibrary.wiley.com/doi/abs/10.1002/net.3230110211

Lin, S. (1965). Computer Solutions of the Traveling Salesman Problem. *Bell System Technical Journal, 44*(10), 2245-2269. Repéré à https://onlinelibrary.wiley.com/doi/abs/10.1002/j.1538-7305.1965.tb04146.x

Lin, S., & Kernighan, B. W. (1973). An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Oper. Res., 21*(2), 498-516. doi: https://doi.org/10.1287/opre.21.2.498

Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer Programming Formulation of Traveling Salesman Problems. *J. ACM, 7*(4), 326-329. doi: https://doi.org/10.1145/321043.321046

Mole, R. H., & Jameson, S. R. (1976). A Sequential Route-building Algorithm Employing a Generalised Savings Criterion. *Journal of the Operational Research Society, 27*(2), 503-511. doi: https://doi.org/10.1057/jors.1976.95

Olivera, A., & Viera, O. (2007). Adaptive memory programming for the vehicle routing problem with multiple trips. *Computers & Operations Research, 34*(1), 28-47. doi: https://doi.org/10.1016/j.cor.2005.02.044. Repéré à http://www.sciencedirect.com/science/article/pii/S0305054805001346

Open Source Routing Machine. (2018). Repéré à https://fr.wikipedia.org/wiki/Open_Source_Routing_Machine

Or, I. (1976). *Traveling salesman-type combinatorial problems and their relation to the logistics of regional blood banking* (Thesis (Ph. D.), NorthWestern University, Ann Arbor, Mich. : Xerox University Microfilms). Repéré à https://trove.nla.gov.au/work/16946163?q&versionId=19887544

Petch, R. J., & Salhi, S. (2003). A multi-phase constructive heuristic for the vehicle routing problem with multiple trips. *Discrete Applied Mathematics, 133*(1), 69-92. doi: https://doi.org/10.1016/S0166-218X(03)00434-7. Repéré à http://www.sciencedirect.com/science/article/pii/S0166218X03004347

Potvin, J.-Y., & Naud, M.-A. (2011). Tabu search with ejection chains for the vehicle routing problem with private fleet and common carrier. *Journal of the Operational Research Society, 62*(2), 326-336. doi: https://doi.org/10.1057/jors.2010.102

Schultmann, F., Zumkeller, M., & Rentz, O. (2006). Modeling reverse logistic tasks within closed-loop supply chains: An example from the automotive industry. *European Journal of Operational Research, 171*(3), 1033-1050. doi: https://doi.org/10.1016/j.ejor.2005.01.016

Simic, V. (2019). Interval-Parameter Conditional Value-at-Risk Two-Stage Stochastic Programming Model for Management of End-of-Life Vehicles. *Environmental Modeling & Assessment*. doi: https://doi.org/10.1007/s10666-018-9648-9

Stenger, A., Vigo, D., Enz, S., & Schwind, M. (2013). An Adaptive Variable Neighborhood Search Algorithm for a Vehicle Routing Problem Arising in Small Package Shipping. *Transportation Science, 47*(1), 64-80 Repéré à https://pubsonline.informs.org/doi/abs/10.1287/trsc.1110.0396

Sun, Y., Wang, D., Lang, M., & Zhou, X. (2018). Solving the time-dependent multi-trip vehicle routing problem with time windows and an improved travel speed model by a hybrid solution algorithm. *Cluster Computing*. doi: https://doi.org/10.1007/s10586-018-2637-6

Thompson, P. M., & Psaraftis, H. N. (1993). Cyclic transfer algorithms for multivehicle routing and scheduling problems. *Oper. Res., 41*(5), 935-946. doi: https://doi.org/10.1287/opre.41.5.935

Toth, P., & Vigo, D. (2014). *Vehicle Routing - Problems, Methods, and Applications* (2nd éd.). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. Repéré à https://app.knovel.com/hotlink/toc/id:kpVRPMAE06/vehicle-routing-problems

Toth, P., & Vigo, D. (Éds.). (2001). *The vehicle routing problem*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics. doi: https://doi.org/10.1137/1.9780898718515

Van Breedam, A. (1995). Vehicle Routing: Bridging the gap between theory and practice. *Belgian Journal of Operations Research, Statistics and Computer Science, 35*(1), 63-80.

Wang, Z., Liang, W., & Hu, X. (2014). A metaheuristic based on a pool of routes for the vehicle routing problem with multiple trips and time windows. *Journal of the Operational Research Society, 65*(1), 37-48. doi: https://doi.org/10.1057/jors.2013.4

Wu, C.-S., Chu, C.-W., & Hsu, H.-L. (2017). A heuristic algorithm of vehicle routing problem with time windows and less-than-truckload carrier selection. *Journal of Marine Science and Technology, 25*(2), 129-141. doi: https://doi.org/10.6119/jmst-014-1223-1