Department of Mathematics and Computer Science

Ph.D. in Mathematics and Computer Science

XXX Cicle

Green Logistics and Crowd-shipping: Challenges and Opportunities

Scientific Sector: MAT/09 - Operations Research

Coordinator

Prof. Nicola LEONE

Supervisors

Prof.ssa Francesca Guerriero

Prof. Gilbert Laporte

Candidate

Dott.ssa Giusy Macrina

2016/2017

A mia madre, a mio padre, a mia sorella

.. a me

Acknowledgements

Foremost, I would like to thanks my supervisor Prof. Francesca Guerriero, not only for the special support she has given me during these years, but for believing in me, in my abilities and for her unerring guidance.

My sincere thanks goes to Prof. Gilbert Laporte whom I visited at Cirrelt in Montréal. I am extremely grateful for the time he has dedicated to me, for his hospitality, his precious advices and for the excellent example as a successful professor and great person.

Special thanks go to Dr. Luigi Di Puglia Pugliese, he has been a constant presence in my PhD years. Thanks for his great helpfulness, his patience and his suggestions.

Many thanks to Elisa, Annarita, Rosita, Vittorio, Sara, M. Carmela and all my colleagues for making my work place so pleasant.

A heartfelt thanks to Selene, Serena, Gislaine, Gianluca, Andrea, Giacomo, Khalid, Amine, Yaser, Merve and all the friends that I met at Cirrelt of Montréal. It wouldn't have been the same without them.

I would like to give special thanks to Jessica, Costanza, Sonia, Mariagrazia, Santina, Roberta, Daniela, Oriana, Letizia, Graziella, Francesco and to all of my friends for supporting me during these years.

My last big thanks go to my parents, my sister Maria Teresa, my aunt Rosy and Christian. They have motivated and encouraged me throughout this experience, and I am so blessed to have them in my life.

Giusy

Summary in Italian

Negli ultimi anni l'interesse verso le problematiche ambientali, quali, ad esempio, la riduzione delle emissioni di diossido di carbonio, dell'inquinamento acustico e del traffico urbano, è cresciuto notevolmente.

Di conseguenza, lo sviluppo di sistemi di trasporto e di distribuzione efficienti, definiti in modo tale da assicurare il miglior trade-off tra la minimizzazione dei costi e la riduzione delle esternalità negative, rappresenta una sfida importante per molti paesi.

In questo contesto, si sono sviluppati due nuovi filoni nell'ambito della logistica distributiva, che cercano di rispondere in maniera appropriata alle esigenze sopra evidenziate, considerando le nuove prospettive legate a tematiche ambientali, ovvero:

- *Green logistics*, il cui obiettivo è quello di introdurre esplicitamente problematiche ambientali nella logistica tradizionale.
- *Crowd-shipping*, che mira a sfruttare la capacità di carico non utilizzata di veicoli privati, che generalmente transitano nella rete stradale per rispondere ad esigenze personali.

In quest'ottica, l'obiettivo del presente lavoro di tesi è proprio quello di definire, progettare e sviluppare tecniche quantitative innovative ed efficienti per la gestione ottimale di sistemi di trasporto in una ottica "green".

In particolare, vengono affrontate diverse varianti del classico

problema dell'instradamento dei veicoli (vehicle routing problem, VRP), definite in modo tale da rispondere ad esigenze di tipo ambientale, ovvero riduzione dei livelli di inquinamento e dei consumi energetici, generale miglioramento del traffico e della circolazione stradale.

La prima parte della tesi è dedicata alla *Green logistics*. Al fine di ridurre le esternalità negative, i modelli e le procedure adottate per l'instradamento dei veicoli devono considerare fattori ecosostenibili e offrire nuove strategie e soluzioni per il trasporto.

Un possibile approccio è quello di minimizzare le emissioni di sostanze inquinanti, ad esempio quelle legate al diossido di carbonio, considerando i relativi costi direttamente nella definizine della funzione obiettivo. Un diverso approccio consiste nell'utilizzare veicoli alimentati in modo alternativo, ad esempio veicoli elettrici (Electric Vehicles, EVs), al posto dei veicoli tradizionali alimentati a diesel o benzina.

Negli ultimi anni, molte aziende di trasporto hanno introdotto gli EVs nelle loro flotte, soprattutto grazie ai numerosi incentivi statali. È tuttavia opportuno evidenziare che, anche se gli EVs non emettono inquinanti chimici e polveri sottili e sono molto più silenziosi dei veicoli tradizionali, presentano alcuni limiti legati principalmente all'autonomia della batteria, alla scarsa presenza di stazioni di ricarica e ai lunghi tempi di ricarica richiesti.

Tuttavia, recentemente, si é registrato un incremento di investimenti (pubblici e privati) finalizzati alla realizzazione di infrastrutture più adatte all'instradamento dei veicoli elettrici e allo sviluppo di nuove tecnologie. I risultati raggiunti sono molto positivi e pertanto é possibile affermare che gli EVs sono una valida alternativa ai poco eco-friendly veicoli a carburante.

Il primo capitolo di questo elaborato è dedicato all'analisi dei contributi più importanti e recenti pubblicati nell'ambito dei problemi di instradamento dei veicoli di tipo “green” (green VRP, G-VRP).

Nel secondo e nel terzo capitolo vengono descritti due modelli matematici innovativi sviluppati per rappresentare due varianti del G-VRP, definiti in modo tale da prendere in considerazione obiettivi di sostenibilità ambientale.

Entrambi i modelli considerano l'impiego di una flotta mista di veicoli, composta sia da veicoli tradizionali, alimentati a diesel, e sia EVs. Vengono inoltre considerati alcuni vincoli relativi alla vita utile e alla ricarica della batteria. Ricaricare completamente la batteria, infatti, può comportare una degradazione più veloce della stessa, riducendone notevolmente la vita utile; inoltre, ricaricare l'ultimo 10% di una batteria richiede tempi relativamente molto lunghi, incompatibili con la durata delle rotte e i turni di lavoro degli autisti.

I modelli matematici sviluppati tengono in considerazione questi vincoli; in particolare, vengono introdotte delle limitazioni sul minimo e sul massimo stato di carica della batteria, inoltre si assume che i veicoli possano essere ricaricati, anche parzialmente, presso una qualsiasi delle stazioni di ricarica disponibili.

In particolare, nel modello matematico presentato nel secondo capitolo, viene considerato un limite sulle emissioni di agenti inquinanti dei veicoli tradizionali. Si assume che le emissioni di CO_2 siano legate a due fattori: il tipo di veicolo e la quantità di carburante consumato. Quest'ultimo dipende sia dalla distanza percorsa che dalla quantità di merce trasportata. Inoltre si assume che il consumo di energia elettrica sia proporzionale alla distanza percorsa.

Nelle realtà, tuttavia, quest'ultima assunzione non risulta essere verificata. Infatti, il consumo di energia non è proporzionale alla distanza percorsa, ma dipende da altri fattori, quali la velocità, il carico del veicolo e anche le caratteristiche fisiche della strada percorsa. Al fine di tenere in dovuta considerazione tali aspetti, nel terzo capitolo viene introdotta una nuova formulazione del G-VRP che si basa su due modelli innovativi per la valutazione del consumo di energia, uno sviluppato per i veicoli tradizionali e l'altro definito per i veicoli elettrici.

La seconda parte del presente lavoro tesi è invece dedicata al *Crowd-shipping.* Negli ultimi anni, la velocità nell'eseguire consegne dell'ultimo miglio sta assumendo un'importanza sempre maggiore, soprattutto grazie all'aumento della popolarità degli acquisti on-line.

Ciò ha portato le aziende di distribuzione a ricercare soluzioni innovative per organizzare questa tipologia di consegne. In questo contesto, la "sharing economy" ha assunto un ruolo di crescente interesse. Il Crowd-shipping è strettamente legato al concetto di "sharing economy", l'idea fondamentale è, infatti, quella di realizzare in outsourcing alcune attività che generalmente vengono effettuate dalle aziende.

L'utilizzo di veicoli privati, che giornalmente transitano nella rete stradale, per effettuare consegne a domicilio, permette non solo di sfruttare risorse sotto-utilizzate, ma ha anche un impatto positivo sull'inquinamento e sul traffico urbano, grazie al non utilizzo di mezzi pesanti, generalmente usati dalle aziende che effettuano il trasporto.

Sulla base delle considerazioni precedenti, nel quarto capitolo viene analizzata una variante del VRP con finestre temporali, in cui si valuta la possibilità di implementare una strategia di Crowd-shipping.

Si prende in considerazione lo specifico scenario in cui un'azienda non ha a disposizione soltanto la propria flotta di veicoli, ma può fare affidamento ad alcuni autisti occasionali, che decidono di mettere a disposizione i propri mezzi per effettuare alcune consegne, in cambio di un compenso.

In particolare, si propongono due varianti del problema. Nel primo si assume che gli autisti occasionali possano effettuare più di una consegna, nel secondo viene utilizzata la politica dello split and delivery.

I due modelli sono validati su scenari realistici e viene effettuato un confronto tra le strategie proposte e quelle presenti in letteratura.

I risultati sperimentali dimostrano che il Crowd-shipping può essere considerata una strategia di distribuzione molto conveniente, e vantaggi maggiori, in termini di efficienza, si registrano con la politica di split and delivery.

Nel quinto capitolo, infine, viene presentata un'euristica per la risoluzione del problema di VRP con Crowd-shipping e consegne multiple.

Nel capitolo sei vengono riportate le conclusioni del presente lavoro di tesi.

Contents

List of Tables

List of Figures

Introduction

In recent years, we have witnessed a growing interest in environmental problems related to polluting emissions, noise and congestion in the transportation logistics. In this context, developing environmentally friendly and efficient transport and distribution systems, defined in such a way to ensure the best trade-off between cost minimization and negative environmental externalities reduction, represents an important challenge for many countries. In this dissertation we focus on two new branches of logistics:

- *Green logistics*, which aims to bring the environmental dimension in traditional logistics.
- *Crowd-shipping*, which encourages ordinary people to use their underexploited capacity on their cars, bikes, buses and planes to carry parcels for other people on their route.

The main goal is to provide innovative quantitative techniques to efficiently manage the "green" transportation systems, hence we propose several variants of the Vehicle Routing Problem (VRP) by considering new perspectives. The first part of this dissertation is devoted to the *Green logistics.* In order to reduce the negative externalities, routing models and procedures have to consider sustainable factors and offer new transport strategies and solutions. One possible approach is to minimize the polluting emissions by including the emission costs in the objective function. A different approach is to use alternative fuel vehicles (AFVs), in particular electric vehicles (EVs), instead of the conventional ones. In recent years several companies have started using EVs as a result of governmental incentives. While EVs do not produce CO_2 emissions and are more silent than the conventional vehicles, they are constrained by the low autonomy of their battery, the limited number of public charging stations (CSs) and long charging times. The fist chapter 1 is devoted to a brief overview on the main contributions related to the green-vehicle routing problem (G-VRP). In chapters 2 and 3 we modelled and then solved two green VRP (G-VRP) variants by incorporating sustainability goals. We consider a mixed fleet of vehicles, composed of capacitated EVs and conventional diesel vehicles. We assume that partial battery recharges for each electrical vehicle are allowed at any available recharging station. We consider also some realistic issues related to the life span of the battery. Indeed, full recharges can damage battery and the last 10% of recharge requires considerable time. Thus, we also need to constrain the state of charge of the battery. Combining these elements makes the problem different from the other contributions and interesting from a point of view of the realistic applications. In particular in chapter 2 we consider a limitation on polluting emissions for conventional vehicles. We assume that the calculation of CO_2 emissions depends on two factors: the type of vehicle and the type and quantity of fuel consumed. The EVs energy consumption is assumed to be proportional to the traveled distance. Since real-life energy consumption is not a linear function of traveled distance, in chapter 3 we use realistic energy consumption model for ECVs and conventional vehicles which takes into account vehicle speed, gradient and cargo load.

The second part of this dissertation is devoted to the *Crowd-shipping*. In the last years the growing importance of shorter delivery lead times has led the companies to create innovative solutions to organize the last-mile and same-day deliveries. In this context, the "sharing economy" has attracted a great deal of interest. Crowd-sourcing is strictly related with the concept of "sharing economy", and allows activities that usual are performed by a company to be outsourced to a large pool of individuals. In chapter 4 we study a variant of the VRP with Time Windows in which the crowd-shipping is considered. We suppose that the transportation company can make the deliveries by using its own fleet composed of capacitated vehicles and also some occasional drivers (ODs). We consider two different scenarios, in the first one multiple deliveries are allowed for each occasional driver, in the second one we introduce the split and delivery policy. We validate the two mathematical models by considering several realistic scenarios. The results show that the transportation company can achieve important advantages by employing the occasional drivers, which become more significant if the multiple delivery and the split delivery policy are both considered. In chapter 5 we propose a hight performing heuristic for the VRP with ODs and multiple deliveries. Conclusions follow in chapter 6.

Part I

The Green Vehicle Routing Problem

Chapter 1

The Green Vehicle Routing Problem: A survey

This paper presents a survey of the main contributions related to the green-vehicle routing problem (G-VRP). The G-VRP is a variant of the well-known vehicle routing problem, which takes into account the environmental sustainability in freight transportation. We provide a classification of the G-VRP variants and discuss the proposed solution approaches.

Keywords: green logistics; green-vehicle routing; survey.

1.1 Introduction

Green logistics aims to bring the environmental perspective in traditional logistics (see [45]). In recent years governments and business organizations have triggered several green initiatives, as a result of which interest in green logistics has increased as well as the society's environmental awareness. In this perspective, due to the major impact that transportation logistics has on environment, reducing negative externalities in transportation logistics is a priority for many countries. The vehicle routing problem (VRP), introduced by Dantzig and Ramser [12], aims to finding the optimal delivery or collection routes for a fleet of vehicles from a depot to a set of customers. The VRP often includes constraints such as capacity, route length, time windows, precedence relations between customers, etc. (see Laporte [34]). Since the VRP is a central problem in freight transportation, it has been widely studied during the years (see Laporte [33], Laporte [34] and Kumar and Paneerselvam [31]). Recently, several authors have started to study the VRP under a green perspective, by considering environmental effects of routing strategies, use of alternative fuel vehicles, energy minimization, etc. We call this VRP variant green-vehicle routing problem (G-VRP). Lin et al.[38] proposed a survey on G-VRP. They reviewed the main VRP variants, and then focused on green logistics contributions during 2006-2012. They classified the G-VRP in three main classes: green-VRP, pollution routing problem and VRP in reverse logistics. The aim of our work is to survey and classify the G-VRP variants introduced in the 2011-2018 periods and describe the proposed solution approaches for these problems. The remainder of this paper is organized as follows. Section 1.2 presents the survey methodology. Section 1.3 gives an overview of the main contributions to G-VRP with conventional vehicles. In Section 1.4 we review the G-VRP with alternative fuel vehicles variants. Conclusions follow in Section 1.5

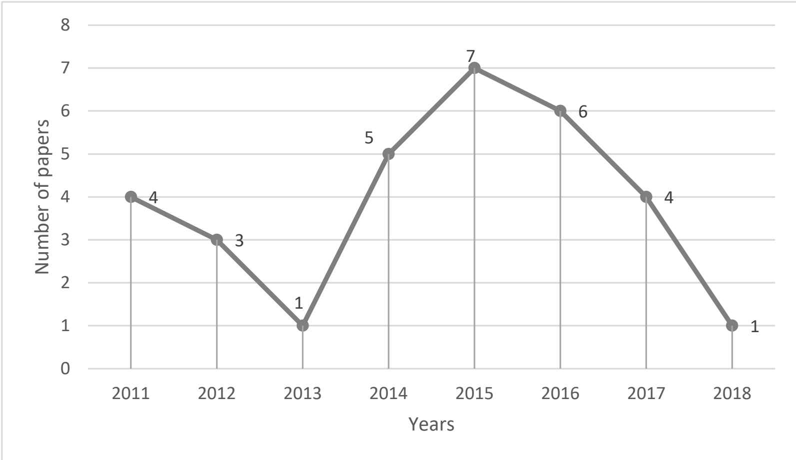
1.2 Survey methodology

In order to review the literature on G-VRP we used several academic databases, including Scopus, Google scholar, ScienceDirect etc., accessed from the university library by using keywords such as green vehicle routing, green logistics, pollution routing problem, etc. We searched papers in journals, books, technical reports and conference proceedings. We also used bibliographies of survey papers and papers on G-VRP. We reviewed 31 papers on the G-VRP in the 2011-2018 period. Figure 1.1 shows the distribution of the surveyed works during these years. A large part of the surveyed works are scientific articles, published in operations research journal such as: *European Journal of Operational Research*, *Operations Research, Transportation Research* (parts B,C and E), *Transportation Science*; three are proceedings, only one is a technical report and one is an unpublished article. We propose a classification scheme based on the different variants of G-VRP presented in the scientific literature. In particular, we identified two main classes: (1) the G-VRP with conventional vehicles and (2) the G-VRP with alternative fuel vehicles. We also identified five sub-categories for the G-VRP with alternative fuel vehicles. We describe the variants and the proposed approaches for these problems. Since the VRP is classified as an NP-hard problem, solving the G-VRP with exact optimization methods may be very difficult. Only four out of 31 works proposed an exact algorithm for the proposed G-VRP variant. a large proportion of the authors propose local search based metaheuristics for their problems, such as adaptive large neighborhood search, variable neighborhood search and tabu search, which lperforming very well on this class of problems.

1.3 Literature review on the G-VRPs with conventional vehicles

Externalities in freight transportation are various. The CO_2 emissions problem is one of the most known and significant, due to the negative impact on the environment and human health. Several authors explicitly consider the CO_2 emissions in their objective function and focus on the minimization of routing cost and polluting emissions. Figliozzi [19] introduced a time-dependent VRP with time windows. The author calculated the amount of fuel spent with the purpose of studing

Figure 1.1: Number of contributes during the years



the impacts of congestion, land use and travel speed on CO_2 emissions. Bektaş and Laporte [6] modeled for the first time the energy consumption of conventional vehicles and explicitly considered the polluting emissions impact. They called this problem the Pollution-Routing Problem (PRP), and presented a non-linear mixed integer mathematical problem for it. This paper lead to several modeling and algorithmic extensions. Demir et al. [13] highlighted the difficulty of solving medium-scale PRPs by using the model presented in Bektaş and Laporte [6]. After introducing an extended PRP, they proposed an effective adaptive large neighborhood search (ALNS) heuristic capable of solving instances with up to 200 nodes. Jabali et al. [24] solved a T-DVRP by tabu search, considering the maximum achievable vehicle speed as a part of the optimization. They considered a two-stage planning horizon: free flow traffic and congestion. They modeled and minimized the emissions per kilometer as a function of speed, and showed that reducting emissions leads to reducing routing costs. Since the minimization of fuel consumption and driving time are conflicting, Demir et al. [14] proposed and solved the bi-objective PRP, in which they jointly minimized the two conflicting factors. The problem was solved via a bi-objective ALNS algorithm embedding a speed optimization procedure. The computational results showed one does not need to increase driving time significantly in order to reduce fuel consumption and CO_2 emissions. Franceschetti et al. [20] considered the Time-Dependent PRP (TDPRP) with time windows, an extension of the PRP which explicitly takes into account traffic congestion. They proposed an integer linear programming formulation and partitioned the planning horizon into two phases, as in Jabali et al. [24]. Tajik et al. [53] introduced uncertain data in the TDPRP with pickups and deliveries. They defined a mixed integer linear program in which the main objective is to minimize the travel distance, the number of vehicles and the polluting emissions. They then introduced a robust counterpart, considering the vehicle speed as an uncertain parameter. Koç et al. [27] introduced the fleet size and mix pollution-routing problem, a PRP variant with a heterogeneous fleet. They solve the problem by means of a hybrid evolutionary metaheuristic. The authors showed the benefit of using a heterogeneous fleet over a homogeneous one. Kramer et. al [30] developed a new hybrid iterated local search (ILS) that integrates a set partitioning procedure and a speed optimization algorithm for the PRP introduced in Bektaş and Laporte [6]. The proposed algorithm highly outperforms the previous available algorithms. Since there exists an optimal speed yielding a minimum fuel consumption (see Demir et al. [13]), hence a minimization of CO_2 emissions, the main goal for the PRP is to optimize the speed for each route. In Kramer et. al [30], the authors consider the same speed on for each arc and assume that the departure time is fixed. Kramer et. al [29] extended the previously work, by introducing variable departure times. Moreover, the speed and departure time are both embedded in the optimization algorithm proposed in Kramer et. al [30]. Table 2.1 summarizes the main papers on the PRP and its variants.

Table 1.1: Summary of the literature on the PRP and its variants

Reference	Algorithm	Math model	Time windows	Time dependency	Pickup & delivery	Uncertain data	Heterogeneous fleet
[6]		•	•				
[19]	Heuristic	•	•	•			
[13]	Heuristic	•	•				
[24]	Heuristic			•			
[20]			•	•			
[14]	Heuristic	•	•				
[53]		•	•		•	•	
[27]	Heuristic	•	•				•
[30]	Heuristic	•	•				
[29]	Heuristic						

1.4 Literature review on the G-VRPs with alternative fuel vehicles

A different approach is to use alternative fuel vehicles (AFVs), especially electric vehicles (EVs), instead of the conventional ones. Governments have started to provide incentives aimed at increasing the commercial use of EVs, (see Pelletier et al. [44]). While EVs do not produce CO_2 emissions and are more silent than the conventional vehicles, they are constrained by the low autonomy of their battery, the limited number of public charging stations (CSs) and long charging times. In the last years, the number of publications on this topic considerably has increased with the interest for green transportation. The authors started to study and propose several G-VRP variants, in particular we have identified and classified five classes of G-VRP with AFVs summarized in Table 1.4. Figure 1.2 shows the trend in publications on the G-VRP and its variants during the last eight years. Looking at Figure 1.2, it is clear that a large number of contributes was published during the years 2014 - 2017 with a growing trend.

Table 1.2: Recent studies of G-VRP during 20112018

G-VRP variants	Papers	Number
G-VRP with AFVs	[11], [17], [41]	3
G-VRP with EVs	[50], [18], [16], [8], [15], [39], [23], [26], [28], [35]	10
Mixed Fleet VRP	[22], [46], [21]	3
G-VRP with EVs and location	[55], [36], [49], [48], [43]	5
G-VRP with EVs and non-linear charging function	[42]	1

Figure 1.2: Number of contributes in green-vehicle routing problems published during the last years

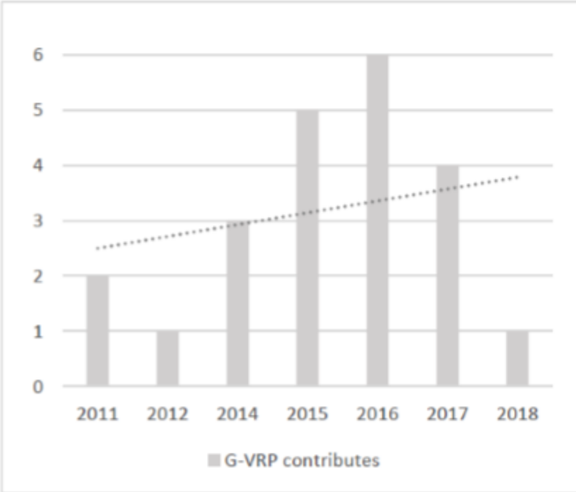


Table 2.2 summarizes the main papers addressing G-VRP and its variants.

1.4.1 Green vehicle routing problem

Conrad and Figliozzi [11] presented the recharging VRP with time windows, where the vehicles have to be charged at some customer locations in order to continue their route. Energy consumption and travelled distance are directly correlated. Recharging is allowed while servicing customers. The authors solve the problem with an iterative construction and improvement heuristic. Erdoğan and Miller-Hooks [17] introduced the Green VRP (GVRP) in which the fleet is composed of AFVs. Fuel consumption is proportioned to travelled distance, and the vehicle fuel tank can be charged at alternative fuel charging stations. The authors developed two constructive heuristics for the problem, with the goal of minimizing the travelled distance. Montoya et al. [41] proposed a multi-space sampling heuristic to solve the GVRP proposed in Erdoğan and Miller-Hooks [17]. The procedure includes three main components: three randomized traveling salesman problem heuristics, a tour partitioning procedure, and a set partitioning formulation. They made several computational tests and compared their approach with those of Erdoğan and Miller-Hooks [17] and of Schneider et al. [50] and concluded that their heuristic is highly competitive, and also the simplest for the GVRP.

1.4.2 Electric vehicle routing problem

Interest in EVs has increased in recent years, due to government incentives and technology progress. Using EVs in good distribution is considered to be a serious alternative to the conventional vehicles, hence several authors started to study the VRP with fleets composed of EVs and its variants. Schneider et al. [50] extended the work of Erdoğan and Miller-Hooks [17] by introducing the Electric-VRP (E-VRP) with time windows and recharging stations (E-VRPTW). EVs can be charged at any of the available CSs, and charging time is related to the state of charge of the battery when the vehicle arrives at the CS. They proposed a hybrid metaheuristic that integrates variable neighborhood search (VNS) with tabu search. Felipe et al. [18] extended the model presented in Erdoğan and Miller-Hooks [17]. They allowed partial recharges and considered multiple charging technologies at CSs. The authors developed a nearest neighbour construction heuristic, as well as a simulated annealing algorithm. Ding et al. [16] extended the E-VRPTW model of Schneider et al. [50] by introducing partial recharges and the pickup and delivery policy. This problem is then solved with a hybrid heuristic which incorporates VNS with a tabu search. Bruglieri et al. [8] studied a variant of E-VRPTW in which the battery charging level is a decision variable. They solved the problem with a VNS branching. Desaulniers et al. [15] developed two branch-price-and-cut algorithms for the E-VRPTW and extended the problem by considering four charging strategies: a single charge or multiple charges per route and fully recharge only, multiple recharges per route and batteries are fully charged, at most one single recharge per route and partial recharges, and multiple partial recharges. Lin et al. [39] extended the E-VRP by considering a heterogenoeus fleet of EVs and the vehicle load effect on battery consumption as in Goeke and Schneider [21]. They solved the model with CPLEX and compared alternative routing strategies using a case study in Austin, Texas. Hiermann et al. [23] introduced the electric fleet size and mix vehicle routing problem with time windows and recharging stations. They considered a heterogeneous fleet of EVs in which each vehicle is characterised by its fixed cost, battery and load capacity, energy consumption and charging rate. Each vehicle can be fully charged at a CS. They proposed a hybrid metaheuristic that combines ALNS to a cyclic neighborhood search and labeling procedures. Keskin and Çatay [26]) formulated the E-VRPTW with partial recharges and solved it by ALNS. Koç and Karaoglan [28] developed a simulated annealing heuristic based on an exact solution approach to solve the G-VRP introduced by Erdoğan and Miller-Hooks [17]. In their formulation, the authors introduced new decision variables in order to allow multiple visits to the CSs without augmenting the networks with dummy nodes. Based from this work, Leggieri and Haouari [35], proposed a new formulation for the E-VRPTW. In order to assess the effectiveness of their approach, the authors solve their model by CPLEX and compared the results with those obtained by the branch-and-cut algorithm of Koç and Karaoglan [28].

1.4.3 Mixed fleet green vehicle routing problem

Gonçalves et al. [22] studied a VRP variant with a mixed fleet composed of EVs and conventional vehicles, as well as pickup and delivery. EVs have a fixed autonomy and charging time, and can be charged at any time during a route. They applied their model to a particular case of a Portuguese battery distributor and studied three different scenarios: in the first one the fleet is composed only of company's conventional vehicles, in the second one they considered a fleet composed of conventional vehicles and uncapacitated EVs, while in the third one they use only EVs. After solving the model with CPLEX and testing the three scenarios, the authors concluded that the use of EVs leads to a significant cost increase due to the initial investment. Sassi et al. [46] formulated the heterogeneous electric vehicle routing problem with time dependent charging costs and a mixed fleet composed by conventional vehicles and EVs. The EVs have different battery capacities and operating costs. An EV can be charged at the available CSs only if it is compatible with the available technologies. Partial recharges and the recharges at the depot are allowed. Charging costs vary according to station and the time of day. The authors solved the problem with a construction heuristic, followed by an inject-eject routine-based local search. A mixed fleet of conventional vehicles and EVs is also considered in Goeke and Schneider [21]. The authors formulated the E-VRP with time windows and mixed fleet, in which the EVs can be charged at the available CSs. Charging times vary according to the battery state of charge when the EV arrives at the CS and charging is always done to maximum battery capacity. The authors propose a realistic energy consumption model which considers speed, vehicle mass and gradient. They model three different objective functions: the first one minimizes the travelled distance, the second one the energy and labor costs, and the third one also includes the cost related to the battery replacement after the depreciation.

1.4.4 Electric vehicle location routing problem

The decisions about the location and technology of the CSs are directly related to the EV routing. The installation and operation costs of the network highly impact on the companies decisions. Yang and Sun [55] introduced the electric vehicles battery swap stations location routing problem whose aim is to determine the locations of battery swap stations (BSSs), as well as the routing plan of EVs. The authors propose two heuristics for the problem: the first one, called SIGALNS, is a four-phase heuristic including a modified sweep heuristic, an iterated greedy algorithm and ALNS; the second one, called TS-MCWS, is a hybrid heuristic that combines tabu search and the Clarke-Wright savings method. The CSs location and the type selection of charging infrastructure are two critical factors for the E-VRPs. Their optimization may have a major impact on logistics costs. Li-ying and Yuanbin [36] introduced the EV multiple charging station location-routing problem with time windows whose aim is to optimize the EV routing plan and the CSs location strategy. In particular, they consider also the possibility of choosing among the different types of charging infrastructures. They propose a hybrid heuristic which incorporates an adaptive variable neighborhood search (AVNS) with a tabu search algorithm. Schiffer and Walther [49] introduced the electric location routing problem with time windows and partial recharging in which the the EVs can be charged at any node in the network with only one type of technology. The authors modeled three objective functions: the first one minimizes the total traveled distance, the second one minimizes the number of used EVs, and the third one minimizes the number of CSs. Schiffer and Walther [48] defined the location routing problem with intra-route facilities which focuses on determining the location of facilities for intermediate stops. The facilities are not depots and do not necessarily coincide with customers. Intra-route facilities allow for intermediate stops on a route in order to keep the vehicle operational. The authors proposed an ALNS heuristic including dynamic programming. Paz et al. [43] proposed the multi-depot electric vehicle location routing problem with time windows in which a homogeneous fleet of EVs is considered. The goal is to determine the number and location of CSs and depots, as well as the number of EVs and their routes. The authors also considered the possibility of charging the EV at the CSs or to swap the battery to the BSSs. Since they considered different charging strategies, they proposed and tested three models: in the first one the conventional partial or complete charges can be done at the depots or at the customer locations, in the second one the batteries can be swapped only at the depots, while in the third one if a charging vertex is activated, then it is a BSS and a customer vertex is activated only for the conventional recharging.

1.4.5 Electric vehicle routing problem with non-linear charging function

All early E-VRP models assumed that the battery state of charge is a linear function of charging time. Since in reality this function is non-linear, Montoya et al. [42] extended the E-VRP by considering a non-linear charging function. They proposed an iterated local search enhaced with a heuristic concentration for the problem. They then conducted several computational experiments by comparing their proposed non-linear charging function to those in the literature, and concluded that a linear function charging may lead to infeasible or expensive solutions.

1.5 Conclusions

In recent years the interest in green logistics increased, hence several authors started to study the vehicle routing problem under a green perspective. In this work we have provided an overview of the green vehicle routing (G-VRP) variants introduced during 2011-2018. We have studied and classified the G-VRP variants and the proposed solution approaches. We have identified two main classes of that problem and we have described the variants and the proposed approaches. From the literature, it is clear that very few papers are devoted to the mixed-fleet G-VRP variant and only one contribution takes into account the non-linear charging function for the electric vehicles. Hence, future research may focus on these two aspects.

Table 1.3: Summary of the literature on the G-VRP variants

Variants	Ref.	Algorithm	Time windows	Fixed charging	Partial recharge	Location of CSs	Multiple technologies	Battery swap	Linear charging	Nonlinear charging	Energy consumption linear to distance	Energy consumption model	Pickup & delivery	Multi-depot	Mixed fleet (EVs and ICCVs)
AFVs VRP	[11]	Heuristic	•						•		•				
	[17]	Heuristic		•							•				
	[41]	Heuristic		•							•				
E-VRP	[50]	Heuristic	•						•		•				
	[18]	Heuristic			•		•		•		•				
	[16]	Heuristic	•		•				•		•		•		
	[8]	Heuristic	•						•		•				
	[15]	Exact	•		•				•		•				
	[39]								•			•	•		
	[23]	Heuristic	•								•				
	[26]	Heuristic	•						•		•				
	[28]	Exact		•							•				
	[35]	Exact		•							•				
Mixed fleet G-VRP	[22]			•							•		•		•
	[46]	Heuristic	•		•				•		•				•
	[21]	Heuristic	•						•			•			•
E-VRP with location	[55]	Heuristic						•			•				
	[36]	Heuristic	•			•	•		•		•				
	[49]	Exact			•				•		•				
	[48]	Heurisic	•		•	•			•		•				
	[43]		•		•	•		•	•		•			•	
E-VRP with non-linear charging function	[42]	Heuristic					•			•	•				

Chapter 2

An iterated local search procedure for the green mixed fleet vehicle routing problem with partial battery recharging and time windows

This work presents a new variant of the Green Vehicle Routing Problem with time windows. We propose an iterative local search heuristic to optimize the routing of a mixed vehicle fleet, composed of electric and conventional (internal combustion engine) vehicles. Since the electric vehicles have a limited autonomy of the battery, we consider the possibility of recharging partially at any of the available stations. In addition, we explicitly take into account a limitation on the pollution emissions for the conventional vehicles. The behaviour of the proposed approach is evaluated empirically on a large set of test instances.

Keywords: green vehicle routing; mixed fleet; pollution routing; iterated local search.

2.1 Introduction

In recent years, we have witnessed a growing interest in environmental problems related to polluting emissions, noise and congestion in the transport industry. In this context, developing environmentally friendly and efficient transport and distribution systems represents an important challenge. As a result, several researchers have begun analysing and studying classical vehicle routing problems (VRPs) from a green perspective, by incorporating sustainability goals with a primary focus on the reduction of environmental externalities. We refer to these problems as Green-VRPs (G-VRPs). The aim of this paper is to present a new G-VRP model in which a mixed fleet of electric and conventional vehicles is considered and which explicitly takes into account the polluting emissions, and we develop an efficient heuristic for the proposed problem. In the following, we review the literature related to the G-VRPs variants concerning sustainable transport issues

2.1.1 Literature review on the G-VRPs with conventional vehicles

In order to reduce the negative externalities, routing models and procedures have to consider sustainable factors and offer new transport strategies and solutions. One possible approach is to minimize the polluting emissions by including the emission costs into the objective function. Figliozzi [19] presented a time-dependent VRP with time windows. The author calculated the amount of fuel spent with the purpose of studing the impacts of congestion, land use and travel speed on CO_2 emissions. Bektaş and Laporte [6] modeled for the first time the energy consumption of conventional vehicles and explicitly considered the polluting emissions impact. They called this problem the Pollution-Routing Problem (PRP), and presented a nonlinear mixed integer mathematical problem for it. This paper lead to several modeling and algorithmic extensions. Thus Demir et al. [13] highlighted the difficulty of solving medium-scale PRPs by using the model presented in Bektaş and Laporte [6]. After introducing an extended PRP, they proposed an effective adaptive large neighborhood search (ALNS) heuristic capable of solving instances with up to 200 nodes. Since CO_2 emissions are directly related to vehicle speed, Jabali et al. [24] solved a T-DVRP by tabu search, considering maximum achievable vehicle speed as a part of the optimization. These authors considered a two-stage planning horizon: free flow traffic and congestion. They modeled and minimized the emissions per kilometer as a function of speed, and highlighted the relationship between reducing emissions and routing cost. Since the minimization of fuel consumption and driving time are conflicting, Demir et al. [14] introduced and solved the bi-objective PRP, in which they jointly minimized the two conflicting factors. Franceschetti et al. [20] considered the Time-Dependent PRP with time windows, an extension of the PRP that explicitly takes into account traffic congestion. Tajik et al. [53] introduced uncertain data in the Time-Dependent PRP with pickup and delivery. They defined a mixed integer linear program in which the main objective is to minimize the travel distance, the number of vehicles and the polluting emissions. They then introduced a robust counterpart, considering vehicle speed as an uncertain parameter. Koç et al. [27] introduced a heterogeneous fleet in the PRP, called the fleet size and mix pollution-routing problem and demonstrated the benefit of using a heterogeneous fleet over a homogeneous one. Kramer et. al [30] developed a new hybrid iterated local search for the PRP addressed in Bektaş and Laporte [6]. Since speed has a major impact on CO_2 emissions, the main objective in the PRP is to optimize vehicle speed for each route. Kramer et. al [30] consider the same speed on each arc and assume that the departure time is fixed. Kramer et. al [29] extended the previous work by introducing variable departure times. Moreover, speed and departure time are both embedded in the optimization algorithm proposed in Kramer et. al [30]. Table 2.1

summarizes the main papers on the PRP and its variants.

Table 2.1: Summary of the literature on the PRP and its variants

Reference	Algorithm	Mathematical model	Time windows	Time dependency	Pickup & delivery	Heterogeneous fleet	Uncertain data
[6]		•	•				
[19]	Heuristic	•	•	•			
[13]	Heuristic	•	•				
[24]	Heuristic			•			
[14]	Heuristic	•	•				
[20]			•	•			
[53]		•	•		•		•
[27]	Heuristic	•	•			•	
[30]	Heuristic	•	•				
[29]	Heuristic						

2.1.2 Literature review on the G-VRPs with alternative fuel vehicles

Since the transport sector has a heavy environmental impact, and usually companies do not compensate for the emission costs, reducing the CO_2 emissions constitutes a challenge for governments. In recent years several companies have started using alternative fuel vehicles (AFVs), especially electric vehicles (EVs), instead of conventional ones, as a result of governmental incentives (see Pelletier et al. [44]). While EVs do not produce CO_2 emissions and are more silent than conventional vehicles, they are constrained by the low autonomy of their battery, the limited number of public charging stations (CSs) and long charging times.

Gonçalves et al. [22] studied a VRP variant with pickup and delivery, and a mixed fleet composed of EVs and conventional vehicles. Charging can be done at any time during a route and each EV has a fixed autonomy and charging time. They applied their model to a particular case of a Portuguese battery distributor and studied three different scenarios: in the first one they considered only the company's conventional fleet, in the second one the fleet is composed of conventional vehicles and uncapacitated EVs, while in the third one they use only EVs. Conrad and Figliozzi [11] introduced the recharging VRP with time windows, where the vehicles have to be charged at some customer locations in order to continue their route. Energy consumption and travelled distance are proportional. Recharging is allowed while servicing customers.

Erdoğan and Miller-Hooks [17] defined the Green VRP (GVRP) in which the fleet is composed of alternative fuel vehicles. The vehicle fuel tank can be charged at alternative fuel charging stations, and fuel consumption is proportional to travelled distance. Schneider et al. [50] extended this work by introducing the Electric-VRP (E-VRP) with time windows (E-VRPTW) and recharging stations, in which EVs can be charged at any of the available CSs. Charging time is not fixed, but is related to the battery state of charge when the vehicle arrives at the CS. Felipe et al. [18], extended the model presented in Erdoğan and Miller-Hooks [17] in a different way. They allowed partial recharges at the stations and considered multiple charging technologies.

Sassi et al. [46] formulated the heterogeneous electric vehicle routing problem with time dependent charging costs and a mixed fleet, in which a set of customers have to be served by a mixed fleet of vehicles composed of conventional vehicles and EVs. The EVs have different battery capacities and operating costs. An EV can be charged at the available CSs only if it is compatible with the available technologies. Partial recharges and recharges at the depot are allowed. Charging costs vary according to station and time of day. A mixed fleet of conventional vehicles and EVs are also considered in Goeke and Schneider [21]. The authors formulate the E-VRP with time windows and mixed fleet, in which the EVs can be charged at the available CSs. Charging times vary according to the battery level when the EV arrives at the CS and charging is always done up to maximum battery capacity. The authors propose a realistic energy consumption model which considers speed, vehicle mass and gradient. They model three different objective functions: the first one minimizes the travelled distance, the second one the energy and labor costs, and the third one also includes the cost related to the battery replacement after the depreciation.

Li-ying and Yuan-bin [36] introduced the EV multiple charging station location-routing problem with time windows whose aim is to optimize the EV routing plan and the CSs location strategy. In particular, they also consider the possibility of choosing among the different types of charging infrastructures. Ding et al. [16] extended the E-VRPTW model of Schneider et al. [50] by introducing partial charging and a pickup and delivery policy. Bruglieri et al. [8] presented a variant of E-VRPTW in which the battery charging level is a decision variable. Desaulniers et al. [15] extended the E-VRPTW by considering four charging strategies: a single charge or multiple charges per route and fully recharge only, multiple recharges per route and batteries are fully charged, at most a single recharge per route and partial recharges, and multiple partial recharges. Lin et al. [39] extended the E-VRP by considering a heterogenoeus fleet of EVs and the vehicle load effect on battery consumption as in Goeke and Schneider [21].

Hiermann et al. [23] introduced the electric fleet size and mix vehicle routing problem with time windows and recharging stations. They considered a heterogeneous fleet of EVs in which each vehicle is characterised by its fixed cost, battery and load capacity, energy consumption and charging rate. Each vehicle can be fully charged at a CS. Keskin and Çatay [26] formulated the E-VRPTW with partial recharges and solved it by means of an ALNS procedure. Koç and Karaoglan [28] proposed a new formulation for the G-VRP introduced by Erdoğan and Miller-Hooks [17], with new decision variables in order to allow multiple visits to the CSs without augmenting the networks with dummy nodes. Based on this work, Leggieri and Haouari [35], proposed a new formulation for the E-VRPTW.

Montoya et al. [41] developed a multi-space sampling heuristic for the G-VRP introduced by Erdoğan and Miller-Hooks [17]. They performed several computational tests and compared their approach with those proposed by Erdoğan and Miller-Hooks [17] and Schneider et al. [50] and concluded that their heuristic is highly competitive, and also the simplest one for the G-VRP. All early E-VRP models assumed that the battery charge level is a linear function of charging time, while in reality it is non-linear. Montoya et al. [42] extended the classical E-VRP by considering a non-linear charging function and proposed an iterated local search enhaced with a heuristic concentration for the problem. They conducted several computational experiments by comparing their proposed non-linear charging function to those used in previous models. They concluded that a linear charging function may lead to infeasible or expensive solutions.

The decisions about the location and technology of the CSs are directly related to EV routing. The installation and operation costs of the network highly impact on the companies' decisions. Yang and Sun [55] introduced the electric vehicles battery swap stations location routing problem whose aim is to determine the locations of battery swap stations (BSSs), as well as the routing plan of EVs. The CS locations and the selection of charging infrastructure types are two critical factors in the E-VRP. Their joint optimization may have a major impact on logistics costs. Paz et al. [43] proposed the multi-depot electric vehicle location routing problem with time windows in which a homogeneous fleet of EVs is considered. The goal is to determine the number and location of CSs and depots, as well as the number of EVs and their routes. The authors also considered the possibility of charging the EV at a CS or to swap the battery at a BSS. Hence they proposed and tested three models: in the first one the conventional partial or complete charges can be done at the depots or at the customer locations, in the second one the batteries can be swapped only at the depots, while in the third one if a charging vertex is activated, then it is a BSS and a customer vertex is activated only for the conventional recharging.

Schiffer and Walther [49] introduced the electric location routing problem with time windows and partial recharging in which the the EVs can be charged at any node in the network with only one type of technology. The authors modeled three objective functions: the first one minimizes the total traveled distance, the second one minimizes the number of EVs, and the third one minimizes the number of CSs. Schiffer and Walther [48] defined the location routing problem with intra-route facilities which focuses on determining the location of facilities for intermediate stops. These facilities are not depots and do not necessarily coincide with customers. Intra-route facilities allow for intermediate stops on a route in order to keep the vehicle operational. Table 2.2 summarizes the main papers on the G-VRP and its variants. For a more complete survey, the reader is referred to Lin et al. [38].

2.1.3 Scientific contribution and organization of this paper

In this paper, we investigate a GVRP variant in which we consider a mixed vehicle fleet composed of ECVs and conventional internal combustion commercial vehicles (ICCVs). The literature on the VRP with a fleet composed of both ECVs and ICCVs is very limited (see Table 2.1.3 for a summary). In the majority of the aforementioned works, the authors suppose that the EVCs battery must be fully recharged. Moreover, while Goeke and Schneider [21] modelled a realistic energy consumption for both the EVs and ICCVs, the other works do not consider the impact of ICCVs polluting emissions. Since the ECVs have a limited autonomy, and a full battery recharge requires a long time, as in Montoya et al. [42] we assume that the ECVs can be partially recharged at any of the available stations. Referring to the ICCVs, we model the pollution emission with a function of both travelled distance and vehicle load. Furthermore, we consider customer time windows and limited vehicle freight capacities. The objective of the model is the minimization of an objective function that takes into account recharging, routing, and activation of ECVs costs. In addition, the overall pollution emissions are maintained within a given limit. In order to solve the problem under investigation, we propose an iterative local search procedure.

The remainder of this paper is organized as follows. Section 3.2 is devoted to the description of a mathematical model for the green mixed fleet vehicle routing problem with partial battery recharging and time windows. Section 2.3 provides a general description of the iterated local search procedure, whereas the computational experiments are reported in Section 2.4. Conclusions follow in Section 2.5.

Table 2.2: Summary of the literature on the G-VRP variants

Ref.	Algorithm	Math model	Time winows	Fixed charging	Partial recharge	Location of CSs	Multiple tech-nologies	Battery swap	Linear charging	Non-linear charging	Energy consump-tion linear to distance	Energy consump-tion model	Pickup & deliv-ery	Multi-depot	Mixed fleet (EVs and ICCVs)
[22]		•		•							•		•		•
[11]	Heuristic	•	•						•		•				
[17]	Heuristic	•		•							•				
[50]	Heuristic	•	•						•		•				
[18]	Heuristic	•			•		•		•		•				
[46]	Heuristic	•	•		•				•		•				•
[21]	Heuristic	•	•						•			•			•
[55]	Heuristic	•						•			•				
[36]	Heuristic	•	•			•	•		•		•				
[16]	Heuristic	•	•		•				•		•		•		
[8]	Heuristic	•	•						•		•				
[15]	Exact	•	•		•				•		•				
[39]		•							•			•	•		
[23]	Heuristic	•	•								•				
[26]	Heuristic	•	•						•		•				
[28]	Exact			•							•				
[41]	Heuristic			•							•				
[42]	Heuristic	•					•			•	•				
[49]	Exact	•			•				•		•				
[48]	Heurisic	•	•		•	•			•		•				
[35]	Exact	•		•							•				
[43]		•	•		•	•		•	•		•			•	

Table 2.3: Summary of the literature on the Mixed Fleet G-VRP and its variants

Ref.	Algorithm	Math model	Time windows	Fixed charging	Partial recharge	Multiple technologies	Linear charging	Energy consumption proportioned to distance	Energy consumption model	Pickup & delivery	Polluting emissions
[22]		•		•				•		•	
[46]	Heuristic	•	•		•		•	•			
[21]	Heuristic	•	•				•		•		
In this paper	Heuristic	•	•		•	•	•	•			•

2.2 The green mixed fleet vehicle routing problem with partial battery recharging and time windows

We formulate our problem as follows. Let $\mathcal{N}$ be the set of customers, and $\mathcal{R}$ the set of recharging stations. We will also need σ copies of recharging stations to account for multiple visits at the same station, where σ is an input parameter. Thus, let $\mathcal{R}'$ be the set of all stations and their copies, i.e. $|\mathcal{R}'| = |\mathcal{R}| \, (1 + \sigma)$. The value $1 + \sigma$ corresponds to the number of times each station can be visited. Let $\mathcal{V} = \mathcal{R} \cup \mathcal{N}$ and $\mathcal{V}' = \mathcal{N} \cup \mathcal{R}'$. The problem is defined on the graph $\mathcal{G}(\mathcal{V}', \mathcal{A})$, where $\mathcal{A} = \{(i, j) : i, j \in \mathcal{V}', i \neq j\}$ is the set of arcs. The depot is a particular element belonging to the set $\mathcal{R}'$, that is the recharging station where vehicle routes start and end. Every customer $i \in N$ has a demand q_i [kg] and a service time s_i [hours]. All customers must be visited by a single vehicle. Each node $i \in \mathcal{V}'$ has a time window $[e_i, l_i]$. For each $(i, j) \in \mathcal{A}$, d_{ij} denotes the distance from i to j [km], while t_{ij} is the travel time from i to j [hours], and c_{ij} denotes the cost [€/km] which depends on the distance traveled. We impose a limit T on the duration of a route [hours], that is, the end of the time window associated with the depot node is set equal to T.

A heterogeneous fleet of vehicles, composed of n^E ECVs and n^C ICCVs, is available. The two types of vehicles (electrical and conventional) are characterized by different loading capacities, denoted as Q^E and Q^C [kg] for the ECVs and ICCVs, respectively. Furthermore, for each ECV let B^E denote the maximum battery capacity [kWh]. The recharging cost w^r is assumed to be constant and the same for all stations. All recharging stations $i \in \mathcal{R}'$ are characterized by a recharging speed ρ_i [kWh per hour]. We denote by π the coefficient of energy consumption, assumed to be proportional to the distance traveled [€/km]. Partial battery recharging is allowed at any recharging station. Referring to the ICCV, we consider a limit on the overall CO_2

emissions [kg].

2.2.1 The emission factor for the conventional vehicles

In order to define a fuel consumption model for the ICCVs, we have to introduce the emission factor. As in Ubeda et al. [54], we assume that the calculation of CO_2 emissions depends on two factors: the type of vehicle and the type and quantity of fuel consumed. In fact, the CO_2 emissions vary according to the type of transport, in particular the mass of the vehicle, the distance traveled and the load carried.

In order to estimate the emission factor, it is important to calculate the fuel conversion factor. For this purpose we use the chemical reaction proposed by Lichty [37]. Once we have calculate the fuel conversion factor, that is 2.61 CO_2/ litre of diesel, it is possible to estimate the emission factor ε. Thus we let define a function, taking into account data related to the average fuel consumption, which depends on the load. The emission factor ε is equal to the emission factor multiplied by the consumption of diesel fuel. Table 2.4 shows the estimation of emission factors for several capacity scenarios for a 10 tonne capacity truck, see Ubeda et al. [54].

Load of the vehicle	Weight laden (%)	Consumption (litre/100km)	Emission factor (kg CO_2/km)
Empty	0	29.6	0.77
Low loaded	25	34.0	0.83
Half loaded	50	34.4	0.90
High loaded	75	36.7	0.95
Full load	100	39.0	1.01

Table 2.4: Estimation of emission factors for a truck with 10-tonne capacity

2.2.2 The mathematical model

In order to model our problem we define the decision variables as follows:

$$x_{ij}^E = \begin{cases} 1, & \text{the ECV travels from } i \text{ to } j \\ 0, & \text{otherwise} \end{cases} \quad (i,j) \in \mathcal{A}$$

$$x_{ij}^C = \begin{cases} 1, & \text{the ICCV travels from } i \text{ to } j \\ 0, & \text{otherwise} \end{cases} \quad (i,j) \in \mathcal{A}$$

z_{ij} amount of energy available when arriving at node j from the node i [kWh], $(i,j) \in \mathcal{A}$
g_{ij} amount of energy recharged by the ECV at the node i for travelling to j [kWh], $i \in \mathcal{R}, j \in \mathcal{V}'$
τ_j arrival time of the vehicle to the node j [h], $j \in \mathcal{V}'$
u_i^C amount of load left in the vehicle after visiting node i [kg], $i \in \mathcal{V}'$
u_i^E amount of load left in the vehicle after visiting node i [kg], $i \in \mathcal{V}'$.

Starting from the consideration introduced in Section 2.2.1, we define the emissions function $\varepsilon(u_i^C)$ that depends on the load on the vehicle at node i. For instance, if variable u_i^C assumes a value in the range $[0, 0.25Q^C]$, then $\varepsilon(u_i^C) = 0.77$ (see Table 2.4). The value $\sum_{(i,j)\in\mathcal{A}} \varepsilon(u_i^C) d_{ij} x_{ij}^C$ represents the total pollution emissions.

The mixed integer program that models our problem is as follows.

$$\text{Minimize} \quad w^r \sum_{i\in\mathcal{R}'} \sum_{j\in\mathcal{V}'} g_{ij} + w^a \sum_{j\in\mathcal{V}'} x_{sj}^E + \sum_{(i,j)\in\mathcal{A}} c_{ij} d_{ij} x_{ij}^E + \sum_{(i,j)\in\mathcal{A}} c_{ij} d_{ij} x_{ij}^C \tag{2.1}$$

$$\text{subject to} \quad \sum_{j\in\mathcal{V}'} (x_{ij}^E + x_{ij}^C) = 1 \qquad i \in \mathcal{N} \tag{2.2}$$

$$\sum_{j\in\mathcal{V}'} x_{ij}^E \leq 1 \qquad i \in \mathcal{R}' \tag{2.3}$$

$$\sum_{j\in\mathcal{V}'\setminus\{s\}} x_{ij}^E - \sum_{j\in\mathcal{V}'\setminus\{t\}} x_{ji}^E = 0 \qquad i \in \mathcal{V}' \tag{2.4}$$

$$\sum_{j\in\mathcal{V}\setminus\{s\}} x_{ij}^C - \sum_{j\in\mathcal{V}\setminus\{t\}} x_{ji}^C = 0 \qquad i \in \mathcal{V} \tag{2.5}$$

$$\sum_{j\in\mathcal{V}'} x_{sj}^E \leq n^E \tag{2.6}$$

$$\sum_{j\in\mathcal{V}} x_{sj}^C \leq n^C \tag{2.7}$$

$$\sum_{j\in\mathcal{V}', i\neq t} x_{it}^E + \sum_{j\in\mathcal{V}', i\neq t} x_{it}^C \geq 1 \tag{2.8}$$

$$\sum_{i\in\mathcal{V}' i\neq s} x^E_{si} - \sum_{j\in\mathcal{V}' j\neq t} x^E_{jt} = 0 \tag{2.9}$$

$$\sum_{i\in\mathcal{V} i\neq s} x^C_{si} - \sum_{j\in\mathcal{V} j\neq t} x^C_{jt} = 0 \tag{2.10}$$

$$u^E_j \geq u^E_i + q_j x^E_{ij} - Q^E(1-x^E_{ij}) \qquad i\in\mathcal{V}'\setminus\{s,t\}, j\in\mathcal{V}'\setminus\{s\} \tag{2.11}$$

$$u^C_j \geq u^C_i + q_j x^C_{ij} - Q^C(1-x^C_{ij}) \qquad i\in\mathcal{V}\setminus\{s,t\}, j\in\mathcal{V}\setminus\{s\} \tag{2.12}$$

$$u^E_s = 0 \tag{2.13}$$

$$u^C_s = 0 \tag{2.14}$$

$$\tau_j \geq \tau_i + (t_{ij}+s_i)x^E_{ij} - M(1-x^E_{ij}) \qquad i\in\mathcal{N}, j\in\mathcal{V}' \tag{2.15}$$

$$\tau_j \geq \tau_i + (t_{ij}+s_i)x^C_{ij} - M(1-x^C_{ij}) \qquad i\in\mathcal{V}, j\in\mathcal{V} \tag{2.16}$$

$$\tau_j \geq \tau_i + t_{ij}x^E_{ij} + \frac{1}{\rho_i}g_{ij} - M(1-x^E_{ij}) \qquad i\in\mathcal{R}', j\in\mathcal{V}' \tag{2.17}$$

$$e_j \leq \tau_j \leq l_j \qquad j\in\mathcal{V}' \tag{2.18}$$

$$z_{ij} \leq (z_{hi}+g_{ij}) - \pi d_{ij}x^E_{ij} + M(1-x^E_{ij}) + M(1-x^E_{hi}) \qquad h\in\mathcal{V}',\ i\in\mathcal{V}'\setminus\{s\},\ j\in\mathcal{V}', i\neq j, i\neq h, j\neq h \tag{2.19}$$

$$z_{sj} \leq B^E - \pi d_{sj}x^E_{sj} + M(1-x^E_{sj}) \qquad j\in\mathcal{V}' \tag{2.20}$$

$$g_{ij} \leq B^E - z_{hi} + M(1-x^E_{ij}) + M(1-x^E_{hi}) \qquad i\in\mathcal{R}'\setminus\{s\}, h\in\mathcal{V}', j\in\mathcal{V}' \tag{2.21}$$

$$z_{ij} \geq 0.1B^E \qquad i\in\mathcal{R}', j\in\mathcal{V}' \tag{2.22}$$

$$g_{ij} \leq 0.9B^E \qquad i\in\mathcal{R}', j\in\mathcal{V}' \tag{2.23}$$

$$\sum_{(i,j)\in\mathcal{A}} \varepsilon(u^C_i)d_{ij}x^C_{ij} \leq UB \tag{2.24}$$

$$x^E_{ij}, x^C_{ij} \in \{0,1\}, i\in\mathcal{V}', j\in\mathcal{V}'; u^E_i, u^C_i, \tau_i \geq 0, i\in\mathcal{V}' \tag{2.25}$$
$$g_{ij} \geq 0, i\in\mathcal{R}', j\in\mathcal{V}'.$$

The objective function is the sum of four terms. The first one is the cost [€] of energy recharged at all the recharging stations. The second term takes into account the activation of ECVs. The cost w^a is the vehicle activation cost, which depends on the battery capacity of the vehicle. In particular we assume that $w^a = B^E w^r$, that is the cost of one complete recharge of the vehicle. The third and fourth terms represent the cost of the routes traveled by the ECVs and the ICCVs, respectively.

Constraints (3.10) ensure that each customer is visited exactly once, whereas conditions (3.11) mean that each recharging station can be visited at most once. Constraints (3.12) and (3.13) are the flow conservations constraints, whereas (3.14) and (3.15) ensure that the total number of vehicles used in the solution (electrical and conventional, respectively) does not exceed the fleet size. Constraints (3.17)-(3.18) ensure that the route of each vehicle starts and ends at the depot. Conditions (3.19)-(3.22) represent the capacity constraints, for the ECVs and ICCVs, respectively. Constraints (3.23)-(3.25) define the variables τ, whereas the time windows constraints are specified by (3.26). Constraints (3.27) and (3.28) define the variables z ensuring that the capacity of the battery is not exceeded, and conditions (3.29) are used to represent the partial battery recharging. Constraints (2.22) and (2.23) define the state of charge of the battery. Finally, constraints (3.30) impose a limit on the pollution emissions. In particular,$\varepsilon(u_i^C)$ is the emission function previously introduced. Constraints (3.31) define the domains of variables.

2.3 The Iterated Local Search Heuristic

The proposed heuristic is based on iterated local search (ILS). The general structure of the ILS is detailed in Algorithm 1. We are given a set $\mathcal{N}$ of customers to be served. These are partitioned into two clusters, one served by the ICCVs ($C^{'}$), the other by the ECVs ($E^{'}$). The ILS generates an initial solution η_0, as a set of routes, and while the stopping criterion is not satisfied, a perturbation and the local search procedures are applied. Finally, the best solution η^* is returned, that is, a set of routes with the best total cost.

Algorithm 1 . Iterated local search (ILS)

```
Generate the initial solution η0
Apply the local search procedure
while Stop criterion is not verified do
    Perturbation
    Local search
end while
return best solution η*
```

Initialization Phase In order to generate an initial solution, we propose a constructive heuristic based on Solomon's sequential insertion heuristic SIH [52]. Algorithm 2 presents the general structure of the SIH. This heuristic identifies both the node u^* to be added to the initialized route and the position of the insertion. In order to choose u^*, SIH considers the insertion position for all the unrouted nodes $\mathcal{N}^-$ by evaluating both the insertion cost and the associated time delay to serve the subsequent customers.

Given the use of a heterogeneous fleet in our problem, after the definition of two clusters of customers which will be served by the electric vehicles and conventional vehicles respectively, the constructive heuristic is divided into two different phases. The former is aimed at defining the routes used to serve the customers with the ICCVs (i.e., conventional routes η_c) while in the latter, the routes for the customers served by the ECVs (i.e. electrical routes η_e) are built. Finally, the solution $\eta^{'} = \eta_c \cup \eta_e$ is returned.

Algorithm 2 . Sequential insertion heuristic (SIH)

1. Clustering $(\mathcal{N}) \rightarrow C^{'}, E^{'}$
2. Insertion heuristic $(C^{'}) \rightarrow \eta_c$
if some customers are not served **then**
  update $E^{'} : E^{'} \cup \{\mathcal{N}^{-}\}$
end if
3. Insertion heuristic $(E^{'}) \rightarrow \eta_e$
return solution $\eta^{'} = \eta_c \cup \eta_e$

Clustering Algorithm The main aim of this algorithm is to build two clusters of customers $C^{'}$ and $E^{'}$, which will be served by the electrical and conventional vehicles, respectively. Given the set of customers $\mathcal{N}$ to be served, the procedure determines two subsets $E^{'} \subseteq \mathcal{N}$ and $C^{'} \subseteq \mathcal{N}$, such that $E^{'} \cap C^{'} = \emptyset$ and $E^{'} \cup C^{'} = \mathcal{N}$. In order to define $E^{'}$ and $C^{'}$, let we use of two sets E and C, defined by the start node s. Two scores, called p_i^E $(1 \leq p_i^E \leq 10)$ and p_i^C $(1 \leq p_i^C \leq 10)$, are used in the clustering algorithm. The first score is calculated as

$$p_i^E = 11 - \left(1 + \frac{d_i^E - d_{\min}^E}{d_{\max}^E - d_{\min}^E} \times 9\right), \tag{2.26}$$

where d_i^E is the distance between the customer i and the barycentre b_e of the set E, $d_{\min}^E$ is the distance between b_e and the nearest customer, while $d_{\max}^E$ is the distance between b_e and the farthest customer. The second score is calculated as

$$p_i^C = \lambda(pDist_i^C) + (1 - \lambda)(pQ_i), \tag{2.27}$$

where $0 \leq \lambda \leq 1$, $pDist_i^C = 11 - \left(1 + \frac{d_i^C - d_{\min}^C}{d_{\max}^C - d_{\min}^C} \times 9\right)$, $pQ_i = 11 - \left(1 + \frac{q_i - q_{min}}{q_{\max} - q_{\min}} \times 9\right)$, d_i^C is the distance between the customer i and the barycentre b_c of the set C, $d_{\min}^C$ is the distance between b_c and the nearest customer, $d_{\max}^C$ is the distance between b_c and the farther customer, q_i is the demand of customer i, $q_{\min}$ is the smallest customer demand, and $q_{\max}$ is the largest customer demand.

After the evaluation of the score of each customer, it is possible to define the two clusters. For each cluster, select at each iteration the customers with the maximum score: $i^*_E = \text{argmax}_{i \in N}\{p^E_i\}$, then $i^*_C = \text{argmax}_{i \in N}\{p^C_i\}$. If $i^*_E \neq i^*_C$, i^*_E is assigned to E, while i^*_C to C. Otherwise, if $p^E_{i^*_E} > p^C_{i^*_C}$ i^* is assigned to E, else if $p^E_{i^*_E} \leq p^C_{i^*_C}$ i^*, the node is assigned to C. At the end of each iteration, the barycentre for each cluster is recalculated and the scores for the unassigned nodes are recomputed. The two final clusters $E^{'}$ and $C^{'}$ are obtained by removing customer s from cluster E and C, respectively.

Insertion strategy for conventional routes This heuristic chooses the best customer u^* to be added into the route, by taking into account the increase in the traveled distance and traveled time. The heuristic initializes the route as follows: $Z_k = \{s, i^{'}, t\}$, where $i^{'} \in C^{'}$ is the unserved node with the smallest $l_{i^{'}}$. Let $Z_k = \{s, i_1, i_2, \ldots, i_m\}$ be the current route. For each unserved customer $u \in C^{'}$, calculate the best position inside the current route Z_k as

$$f_1(i(u), u, j(u)) = \min_{p=1,\ldots,m}\{f_1(i_{p-1}, u, i_p)\}, \tag{2.28}$$

where $i(u)$ and $j(u)$ are two adjacent customers into the current route. The customer u^* that will be inserted into the route is the one with the best score:

$$f_2(i(u^*), u^*, j(u^*)) = \max_u\{f_2(i(u), u, j(u))\}, \tag{2.29}$$

where

$$f_2(i(u), u, j(u)) = c_{s,u} - f_1(i(u), u, j(u)). \tag{2.30}$$

Before inserting u^* in the route, it is necessary to verify the feasibility of the new solution. If the insertion is infeasible, the algorithm evaluates the possibility of initializing a new route. Otherwise, the customer will be served by the ECVs.

Insertion strategy for electrical routes If some customers belonging to cluster $C^{'}$ are not served by conventional vehicles, they are inserted in cluster $E^{'}$. For each unserved customer $u \in C^{'}$, the best position in the current route is calculated.

The heuristic initializes the route as follow: $Z_w^E = \{s, i^{'}, t\}$, where $i^{'} \in E^{'}$ is the unserved customers with the smallest $l_{i^{'}}$. Let $Z_w^E = (s, i_1, i_2, \ldots, i_m)$ be the current route. For each unserved customer $u \in C^{'}$ we calculate the best position inside the current route Z_k by using [2.29] and the best node u^* by using [2.30]. If the insertion of u^* satisfies the capacity and time windows constraints, it can be added to the route.

After this step, it is necessary to check the satisfaction of the energy capacity constraints. If necessary, recharge stations may be added to the route. In particular, if it is not possible to reach the next node, because of the low battery charge, the nearest recharge station is added to the route and the vehicle is recharged as much as necessary to reach the next node.

After the insertion of the recharge stations, it is necessary to verify the time windows constraints. If the constraints are respected but some customers are unserved, a new route is initialized. Otherwise, if some constraints are violated, the solution is repaired by removing customers and recharge stations until it becomes feasible. If the heuristic is unable to find a feasible solution, the unserved customers are added to the conventional routes by violating the emission constraints.

Local Search and Perturbation In order to explore the neighborhood, we introduce an improvement heuristic based on local search procedures. The steps are detailed in Algorithm 3. We start from the solution $\eta^{'}$ created by SIH. If $\eta^{'}$ is feasible, we apply the improvement heuristic and we return it as the best final solution η^*. Otherwise, we apply the improvement heuristic with penalty function and we return the best generated feasible solution η^*.

Algorithm 3 . Local search (LS)

η' initial solution generated by SIH
if (η') feasible **then**
  Improvement heuristic $(\eta') \to \eta^*$
else
  Improvement heuristic with penalty function $(\eta') \to \eta^*$
end if
return best solution η^*

Improvement Heuristic A local search method is applied to the initial solution, built by using the constructive heuristic, described in the previous section. It uses the following different strategies:

1. **Change of nodes belonging to the conventional routes**: iteratively, for each conventional route, evaluate the best possible insertion of one of its nodes into the other conventional routes.
2. **Change of nodes belonging to the electrical routes**: iteratively, for each electrical route, evaluate the best possible insertion of its node into the other electrical routes. It is worth observing that the insertion/removal of a node into/from an electrical route can imply also the insertion/removal of new recharge stations into/from the solution.
3. **Change of nodes belonging to the conventional and electrical routes**: for each route, iteratively evaluate the best possible insertion of one of its nodes into the other conventional or electrical routes.

The perturbation is performed by using the same strategies implemented for the local search phase. However, during the perturbation, worsenings of the solutions are accepted in order to better explore the neighborhood. The stopping criterion is satisfied when a fixed maximum number of iterations has been reached.

Improvement heuristic with penalty function This alternative approach allows the generation of infeasible solutions in the initialization phase. In particular, the pollution emission constraints are relaxed and the objective function is modified in order to take the penalty cost into account as

$$z'(\eta) = z(\eta) + \theta e(\eta), \tag{2.31}$$

where $z(\eta)$ is the cost function, θ is the penalty factor, and $e(\eta)$ is the violation of emissions, calculated as

$$e(\eta) = \max\{0, \sum_{(i,j)\in A} \epsilon(u_i^c) d_{ij} x_{ij}^c - UB\}. \tag{2.32}$$

The penalty factor is set equal to 1 and is adjusted at each iteration of the local search as follows: if after one iteration a constraint violation is still verified, the factor is increased by 10%.

Starting from an infeasible solution generated by the constructive heuristic, the ILS efficiently explores the solutions space until a good quality feasible solution has been identified. For both the local search procedure and the perturbation, the improvement strategies previously described is used. At each iteration, the strategy to be applied is randomly chosen. The algorithm terminates after a fixed number of iterations. Among all feasible solutions, the best one η^* is that with the minimum cost.

2.4 Computational study

We now analyse the behaviour of the proposed heuristic. We tested our algorithm on instances inspired from the scientific literature. In Section 2.4.1 we provide a detailed description of these instances. We solved the model with CPLEX 12.5, and implemented the algorithm in Java. We carried out our tests on a PC Intel CoreTM I7-4710 CPU at 2.5 GHz having 16 GB of RAM under Windows 8.1 operating system.

2.4.1 Test instances

The instances used in the computational experiments are the E-VRPTW benchmark instances introduced in [50], based on the well-known VRPTW instances of Solomon [52]. These instances are divided into three classes C, R and RC which differ from one another according to the geographical distribution of the customer locations: a clustered distribution (C), random distribution (R) and a mix of random and clustered structures (RC). Moreover, C1, R1 and RC1 have a short scheduling horizon, while C2, R2 and RC2 have a long scheduling horizon. Schneider et al. [50] applied some modifications to these instance sets in order to yield the E-VRPTW instances: they first determined in a random manner 21 recharging stations and added them to the test instances, second the battery capacity is suitably set, and third the time windows of some customers are recalculated to ensure feasibility. We carried out our tests by considering two sets of instances. The first one contains the small-size instances considered in [50], with five, 10 and 15 customers. The second set is composed of medium-size instances that have been built starting from five of the 100-customer E-VRPTW instances presented in [50], belonging to the classes C1, R1, RC1. In particular, we kept the 21 recharging stations unchanged and we extracted the first 25 and 30 customers, respectively. For each test instance, we generated three different instances by varying the value of the upper bound on the pollution emissions. In particular, we first calculated an estimate $UB_{\max}$ of the emissions in the worst case; the parameter UB was then set equal to $\alpha \cdot UB_{\max}$, where $\alpha = 0.75, 0.50, 0.25$. Thus, we generated three scenarios: hard, medium and soft constrained, based on the allowed emissions.

In what follows, we refer to $test$"C"nn_α to indicate the network *test* (original name) for which nn customers have been considered and an upper bound on the pollution emissions equal to $\alpha \cdot UB_{\max}$ was imposed. Thus, test instance C101C25_75 is the network C101 for which the first 25 customers were chosen and the limit on the pollution emission is set equal to $0.75 \cdot UB_{\max}$.

We fixed the battery capacity equal to 10 KWh for small-size instances and to 20 KWh for medium- and large-size instances, while the vehicle capacity was fixed at 500 Kg. Whereas a battery recharging operation can be achieved in several ways, with different technologies that imply different times and costs, for our computational tests we assumed that all charging stations have the same characteristics and only one technology is available. This means that the vehicles can be charged in any recharging station by spending the same time, at the same cost. In particular, following Felipe et al. [18] in which different technologies are introduced (slow, medium and fast), we chose the medium technology, hence the recharging speed is fixed at 20,000 KWh/h and the cost is unitary. We also defined a number of σ copies of CSs to allow multiple visits to the same CS. In particular, we iteratively solved our model with CPLEX with increasing values of σ. The procedure stops when no improvement on the solution cost is found.

2.4.2 Numerical results

The computational study is divided into two phases: since CPLEX was able to solve only the small-size instances, in the first phase we compare the results obtained by using the proposed ILS with the optimal solution costs obtained with CPLEX, while in the second phase we study the solutions obtained on instances with more than 25 customers and solved by ILS. In Section 2.4.2, we focus on the results obtained on the first set of test instances i.e., the small-size ones. In Section 2.4.2, we test the heuristic on the medium- and large-size instances.

Numerical Results on the small size test instances

To assess the performance of the ILS, we carried out a first phase of the computational testing with the aim of comparing the quality of the solutions yielded by the proposed heuristic with those obtained by solving the model. We imposed a time limit of four hours on the execution time of the solver. We evaluated the performance of the proposed heuristic along two dimensions: solution quality and computational effort.

Table 2.5: Results for the instances with five customers

(a) Results for instances with $\alpha = 0.25$

Test	ILS g_c	ILS Speedup
$C101C5$_0.25	0.04%	1.81
$C103C5$_0.25	0.00%	2.37
$C206C5$_0.25	0.00%	3.03
$C208C5$_0.25	11.63%	2.60
$R104C5$_0.25	0.00%	1.77
$R105C5$_0.25	0.25%	1.37
$R202C5$_0.25	6.51%	1.76
$R203C5$_0.25	1.97%	3.04
$RC105C5$_0.25	0.04%	3.71
$RC108C5$_0.25	4.01%	2.36
$RC204C5$_0.25	1.26%	4.56
$RC208C5$_0.25	3.85%	2.57
Average	2.46%	2.58

(b) Results for instances with $\alpha = 0.50$

Test	ILS g_c	ILS Speedup
$C101C5$_0.50	2.29%	1.25
$C103C5$_0.50	0.00%	2.58
$C206C5$_0.50	0.00%	3.51
$C208C5$_0.50	0.00%	1.97
$R104C5$_0.50	0.00%	1.44
$R105C5$_0.50	0.00%	1.45
$R202C5$_0.50	0.00%	1.29
$R203C5$_0.50	0.00%	3.23
$RC105C5$_0.50	0.00%	5.35
$RC108C5$_0.50	1.96%	3.07
$RC204C5$_0.50	0.00%	3.30
$RC208C5$_0.50	0.06%	3.45
Aerage	0.36%	2.66

(c) Results for instances with $\alpha = 0.75$

Test	ILS g_c	ILS Speedup
$C101C5$_0.75	0.00%	1.95
$C103C5$_0.75	0.21%	2.50
$C206C5$_0.75	0.00%	2.49
$C208C5$_0.75	0.00%	1.70
$R104C5$_0.75	0.00%	1.42
$R105C5$_0.75	0.00%	1.19
$R202C5$_0.75	0.00%	1.23
$R203C5$_0.75	0.00%	3.31
$RC105C5$_0.75	0.00%	4.52
$RC108C5$_0.75	0.00%	3.21
$RC204C5$_0.75	0.00%	2.80
$RC208C5$_0.75	0.00%	2.70
Average	0.02%	2.42

Tables 2.5 to 2.7 present the related computational results. For each test instance, we report in the second column the percentage gap in cost g_c, defined as $g_c = (c^H - c^M)/c^M$, where c^H is the cost provided by the heuristic and c^M is the cost obtained solving the model. In the third column we report the speedup value i.e. the ratio between the computational time required by CPLEX and the computational

Table 2.6: Results for the instances with 10 customers

(a) Results for instances with $\alpha = 0.25$

Test	ILS	
	g_c	Speedup
C101C10_0.25	2.14%	4.28
C104C10_0.25	13.93%	3.82
C202C10_0.25	13.11%	3.69
C205C10_0.25	14.84%	1.00
R102C10_0.25	0.83%	6.30
R103C10_0.25	0.00%	63.74
R201C10_0.25	7.87%	2.73
R203C10_0.25	1.19%	66.69
RC102C10_0.25	0.60%	14.50
RC108C10_0.25	0.00%	37.13
RC201C10_0.25	0.89%	2.40
RC205C10_0.25	0.00%	3.59
Average	4.62%	17.49

(b) Results for instances with $\alpha = 0.50$

Test	ILS	
	g_c	Speedup
C101C10_0.50	0.03%	15.81
C104C10_0.50	0.00%	5.33
C202C10_0.50	4.34%	2.70
C205C10_0.50	0.00%	2.48
R102C10_0.50	0.00%	1.56
R103C10_0.50	0.00%	6.05
R201C10_0.50	0.06%	58.19
R203C10_0.50	0.00%	2.65
RC102C10_0.50	0.00%	14.18
RC108C10_0.50	0.00%	3.66
RC201C10_0.50	0.00%	11.85
RC205C10_0.50	0.00%	1.55
Average	0.37%	10.50

(c) Results for instances with $\alpha = 0.75$

Test	ILS	
	g_c	Speedup
C101C10_0.75	0.00%	3.25
C104C10_0.75	0.00%	7.99
C202C10_0.25	0.00%	2.95
C205C10_0.75	0.00%	1.70
R102C10_0.75	0.00%	2.18
R103C10_0.75	0.00%	1.57
R201C10_0.75	0.00%	5.74
R203C10_0.75	0.00%	55.01
RC102C10_0.75	0.00%	2.81
RC108C10_0.75	0.00%	22.74
RC201C10_0.75	0.00%	3.58
RC205C10_0.75	0.00%	14.55
Average	0.00%	10.34

Table 2.7: Results for the instances with 15 customers

(a) Results for instances with $\alpha = 0.25$

Test	ILS	
	g_c	Speedup
C103C15_0.25	2.13%	297.74
C106C15_0.25	0.00%	0.93
C202C15_0.25	1.42%	92.64
C208C15_0.25	0.00%	7.70
R202C15_0.25	0.27%	264.20
R209C15_0.25	0.00%	2.92
R102C15_0.25	2.15%	931.21
R105C15_0.25	2.27%	37.29
RC103C15_0.25	0.53%	1607.65
RC108C15_0.25	2.26%	17099.64
RC202C15_0.25	0.00%	24.68
RC204C15_0.25	0.00%	2381.20
Average	0.92%	1895.65

(b) Results for instances with $\alpha = 0.50$

Test	ILS	
	g_c	Speedup
C103C15_0.50	0.48%	678.04
C106C15_0.50	2.30%	3.01
C202C15_0.50	1.21%	152.90
C208C15_0.50	0.00%	11.51
R202C15_0.50	0.27%	211.77
R209C15_0.50	0.77%	3.01
R102C15_0.50	2.54%	530.30
R105C15_0.50	9.54%	66.75
RC103C15_0.50	0.53%	760.08
RC108C15_0.50	1.28%	17697.47
RC202C15_0.50	2.58%	40.75
RC204C15_0.50	1.83%	3056.36
Average	1.94%	1934.33

(c) Results for instances with $\alpha = 0.75$

Test	ILS	
	g_c	Speedup
C103C15_0.75	2.13%	297.74
C106C15_0.75	0.00%	0.93
C202C15_0.75	1.42%	92.64
C208C15_0.75	0.00%	7.70
R102C15_0.75	0.27%	264.20
R105C15_0.75	0.00%	2.92
R202C15_0.75	2.15%	931.21
R209C15_0.75	2.27%	37.29
RC103C15_0.75	0.53%	1607.65
RC108C15_0.75	2.26%	17099.64
RC202C15_0.75	0.00%	24.68
RC204C15_0.75	0.00%	2381.20
Average	0.92%	1895.65

overhead of the heuristic.

The computational results clearly demonstrate the advantage of the proposed heuristic in terms of efficiency. This advantage becomes more evident on large instances. Indeed, the larger the number of the customers, the higher the speedup value achieved. In particular, the ILS is on average 2.55 times faster on the set with five customers, and up to 1908.54 time faster on the set with 15 customers. The ILS overall outperforms in terms of efficiency CPLEX.

It is worth observing that the heuristic is also effective. Looking at Tables 2.5(a) to 2.5(c), it is clear that the ILS is more effective when α is equal to 0.50 and 0.75. Indeed, it finds an optimal solution for the majority of the instances. However, the average on the cost gap is less than 1% for both the sets, while it increases to 2.46% for $\alpha = 0.25\%$.

The results in Tables 2.6(a) to 2.6(c) and 2.7(a) to 2.7(c) exhibit the same trend. Indeed, when $\alpha = 0.75$, the ILS finds optimal solutions for all the instances with 10 customers (see Table 2.6(c)), and the average on the cost gap is less than 1% for the instances with 15 customers (see Table 2.7(c)). When $\alpha = 0.50$, the average on gap is 0.37% and 1.94% for the instances with 10 and 15 customers, respectively.

Table 2.8 summarizes the percentage deviations of the solution costs found by the ILS from the optimal solution values and the values of speedup values with varying values of α. The table clearly shows the efficiency of the proposed algorithm for all sets of instances. The parameter α has a significant impact on the quality of the solution found by the ILS. The best setting is obtained with $\alpha = 0.75$, however the average on percentage gap is less than 2%.

Numerical Results on the medium-size and large-size test instances

In this section we present a description of the results obtained for the instances with more than 25 customers, for detailed results the reader

Table 2.8: Average Speedup and percentage deviations of the solution costs found by the ILS from the optimal solution values with varying α

Test		ILS g_c	ILS Speedup
5 customers	$\alpha = 0.25$	2.46%	2.58
	$\alpha = 0.50$	0.36%	2.66
	$\alpha = 0.75$	0.02%	2.42
Average		0.95%	2.55
10 customers	$\alpha = 0.25$	4.62%	17.49
	$\alpha = 0.50$	0.37%	10.50
	$\alpha = 0.75$	0.00%	10.34
Average		1.66%	12.78
15 customers	$\alpha = 0.25$	0.92%	1895.65
	$\alpha = 0.50$	1.94%	1934.33
	$\alpha = 0.75$	0.92%	1895.65
Average		1.26%	1908.54

is referred to Appendix 2.6. Tables 2.11 to 2.14 show the results obtained for the medium- and large-size instances. In particular, for each table, the first column displays the name of the instance, while the others give the time in seconds and the cost obtained by the ILS. Table 2.9 summarizes the average time required by the ILS to solve the instances. Overall, the ILS finds the solutions within short computation times. Indeed, the ILS solves all instances with 25 and 30 customers within less than 20 seconds, while the instances with 50 customers are solved within less than two minutes, and the large-size instances within about 11 minutes. Looking at the results, it is possible to conclude that the ILS is less time consuming when α is set equal to 0.50 and 0.75. From Tables 2.11 to 2.14 it is evident that the lower the value of α, the higher the average solution cost. This specific behaviour can be explained by observing that when the emissions constraints are tighter, more EVs are used since only a limited number of ICCVs can be considered. Table 2.10 shows the percentage of impact of the EVs use on the total costs. When $\alpha = 0.25$, the number of routed EVs increases and the cost associated with these vehicles is higher than the cost of the conventional ones, in particular, it is more than 90% for all the classes.

Table 2.9: Summary of ILS averages time execution [seconds] for the medium- and large-size instances

Test		Time	Test		Time
25 customers	$\alpha = 0.25$	22.89	50 customers	$\alpha = 0.25$	135.29
	$\alpha = 0.50$	11.45		$\alpha = 0.50$	126.04
	$\alpha = 0.75$	14.06		$\alpha = 0.75$	117.09
Average		16.13	Average		126.14
30 customers	$\alpha = 0.25$	22.52	100 customers	$\alpha = 0.25$	679.97
	$\alpha = 0.50$	20.03		$\alpha = 0.50$	669.98
	$\alpha = 0.75$	18.37		$\alpha = 0.75$	647.97
Average		20.31	Average		665.97

Table 2.10: Electric Vehicle impact cost

Customers	$\alpha = 0.25$	$\alpha = 0.50$	$\alpha = 0.75$
25	93%	69%	27%
30	92%	70%	34%
50	94%	87%	82%
100	96%	91%	87%

2.5 Conclusions

We have introduced, modelled and solved the green mixed fleet vehicle routing problem with partial battery recharging and time windows. We proposed a mathematical model and an iterated local search procedure to solve it. We conducted several computational experiments on modified benchmark instances in order to evaluate the behaviour of the proposed heuristic. Our test results have shown that the developed method can find good quality solutions within a reasonable amount of time.

2.6 Appendix

Table 2.11: Results for the instances with 25 customers

Instance	ILS		Instance	ILS		Instance	ILS	
	Time [s]	Cost		Time [s]	Cost		Time [s]	Cost
C101C25_0.25	23.18	276.00	C101C25_0.50	11.42	256.00	C101C25_0.75	13.94	245.00
C102C25_0.25	22.54	233.00	C102C25_0.50	11.98	244.00	C102C25_0.75	15.38	227.00
C103C25_0.25	20.76	233.00	C103C25_0.50	12.80	225.00	C103C25_0.75	16.30	210.00
C104C25_0.25	22.07	225.00	C104C25_0.50	14.23	219.00	C104C25_0.75	16.56	208.00
C105C25_0.25	19.48	262.00	C105C25_0.50	12.19	259.00	C105C25_0.75	14.72	250.00
R101C25_0.25	19.84	575.00	R101C25_0.50	10.41	571.00	R101C25_0.75	13.30	570.00
R102C25_0.25	20.33	502.00	R102C25_0.50	10.97	501.00	R102C25_0.75	13.83	500.00
R103C25_0.25	21.43	431.00	R103C25_0.50	11.36	431.00	R103C25_0.75	14.50	426.00
R104C25_0.25	22.95	407.00	R104C25_0.50	12.53	410.00	R104C25_0.75	14.70	401.00
R105C25_0.25	20.83	494.00	R105C25_0.50	11.34	493.00	R105C25_0.75	13.41	494.00
RC101C25_0.25	24.07	472.00	RC101C25_0.50	9.84	473.00	RC101C25_0.75	12.27	470.00
RC102C25_0.25	26.27	392.00	RC102C25_0.50	10.95	382.00	RC102C25_0.75	13.03	380.00
RC103C25_0.25	27.25	310.00	RC103C25_0.50	10.25	311.00	RC103C25_0.75	13.27	309.00
RC104C25_0.25	27.38	371.00	RC104C25_0.50	11.28	306.00	RC104C25_0.75	14.20	309.00
RC105C25_0.25	24.93	453.00	RC105C25_0.50	10.25	450.00	RC105C25_0.75	12.19	453.00
Average	22.89	379.92	Average	11.45	372.23	Average	14.06	367.23

Table 2.12: Results for the instances with 30 nodes

Instance	ILS		Instance	ILS		Instance	ILS	
	Time [s]	Cost		Time [s]	Cost		Time [s]	Cost
C101C30_0.25	23.53	280.00	C101C30_0.50	14.48	281.00	C101C30_0.75	13.39	266.00
C102C30_0.25	17.45	259.00	C102C30_0.50	15.36	267.00	C102C30_0.75	13.94	277.00
C103C30_0.25	18.42	274.00	C103C30_0.50	16.13	269.00	C103C30_0.75	14.88	253.00
C104C30_0.25	22.05	238.00	C104C30_0.50	18.09	241.00	C104C30_0.75	16.36	217.00
C105C30_0.25	15.66	278.00	C105C30_0.50	14.86	283.00	C105C30_0.75	13.61	271.00
R101C30_0.25	16.63	660.00	R101C30_0.50	15.19	655.00	R101C30_0.75	13.73	648.00
R102C30_0.25	18.11	576.00	R102C30_0.50	16.59	569.00	R102C30_0.75	15.44	567.00
R103C30_0.25	19.97	464.00	R103C30_0.50	17.75	456.00	R103C30_0.75	16.03	457.00
R104C30_0.25	21.75	434.00	R104C30_0.50	19.52	412.00	R104C30_0.75	17.69	415.00
R105C30_0.25	19.05	532.00	R105C30_0.50	17.67	533.00	R105C30_0.75	15.80	530.00
RC101C30_0.25	26.50	673.00	RC101C30_0.50	24.97	616.00	RC101C30_0.75	22.88	659.00
RC102C30_0.25	28.86	554.00	RC102C30_0.50	26.67	570.00	RC102C30_0.75	24.73	520.00
RC103C30_0.25	30.45	556.00	RC103C30_0.50	27.78	519.00	RC103C30_0.75	25.78	515.00
RC104C30_0.25	31.44	434.00	RC104C30_0.50	29.58	441.00	RC104C30_0.75	27.44	437.00
RC105C30_0.25	27.95	587.00	RC105C30_0.50	25.86	580.00	RC105C30_0.75	23.91	562.00
Average	22.52	453.27	Average	20.03	446.13	Average	18.37	439.60

Table 2.13: Results for the instances with 50 nodes

Instance	ILS		Instance	ILS		Instance	ILS	
	Time [s]	Cost		Time [s]	Cost		Time [s]	Cost
C101C50_0.25	106.36	280.00	C101C50_0.50	99.40	281.00	C101C50_0.75	91.68	266.00
C102C50_0.25	124.21	259.00	C102C50_0.50	118.78	267.00	C102C50_0.75	110.03	277.00
C103C50_0.25	141.17	274.00	C103C50_0.50	135.24	269.00	C103C50_0.75	121.43	253.00
C104C50_0.25	154.66	238.00	C104C50_0.50	148.11	241.00	C104C50_0.75	136.05	217.00
C105C50_0.25	112.60	278.00	C105C50_0.50	105.58	283.00	C105C50_0.75	101.45	271.00
R101C50_0.25	91.70	660.00	R101C50_0.50	85.50	655.00	R101C50_0.75	79.67	648.00
R102C50_0.25	100.03	576.00	R102C50_0.50	93.57	569.00	R102C50_0.75	87.50	567.00
R103C50_0.25	118.16	464.00	R103C50_0.50	111.92	456.00	R103C50_0.75	102.21	457.00
R104C50_0.25	125.72	434.00	R104C50_0.50	116.50	412.00	R104C50_0.75	111.31	415.00
R105C50_0.25	107.05	532.00	R105C50_0.50	100.92	533.00	R105C50_0.75	94.86	530.00
RC101C50_0.25	150.71	673.00	RC101C50_0.50	137.31	616.00	RC101C50_0.75	130.42	659.00
RC102C50_0.25	164.07	554.00	RC102C50_0.50	152.90	570.00	RC102C50_0.75	140.48	520.00
RC103C50_0.25	180.62	556.00	RC103C50_0.50	163.60	519.00	RC103C50_0.75	153.66	515.00
RC104C50_0.25	189.57	434.00	RC104C50_0.50	173.25	441.00	RC104C50_0.75	159.48	437.00
RC105C50_0.25	162.69	587.00	RC105C50_0.50	147.95	580.00	RC105C50_0.75	136.14	562.00
Average	135.29	453.27	Average	126.04	446.13	Average	117.09	439.60

Table 2.14: Results for the instances with 100 nodes

Instance	ILS		Instance	ILS		Instance	ILS	
	Time [s]	Cost		Time [s]	Cost		Time [s]	Cost
C101_0.25	814.25	280.00	C101_0.50	770.15	281.00	C101_0.75	768.76	266.00
C102_0.25	852.94	259.00	C102_0.50	841.27	267.00	C102_0.75	826.15	277.00
C103_0.25	994.01	274.00	C103_0.50	922.22	269.00	C103_0.75	918.57	253.00
C104_0.25	1033.26	238.00	C104_0.50	1004.99	241.00	C104_0.75	963.90	217.00
C105_0.25	862.44	278.00	C105_0.50	841.44	283.00	C105_0.75	799.58	271.00
R101_0.25	392.67	660.00	R101_0.50	363.78	655.00	R101_0.75	357.99	648.00
R102_0.25	392.98	576.00	R102_0.50	397.04	569.00	R102_0.75	388.05	567.00
R103_0.25	532.67	464.00	R103_0.50	507.95	456.00	R103_0.75	477.32	457.00
R104_0.25	589.78	434.00	R104_0.50	543.32	412.00	R104_0.75	515.88	415.00
R105_0.25	521.50	532.00	R105_0.50	414.59	533.00	R105_0.75	456.61	530.00
RC101_0.25	527.03	673.00	RC101_0.50	506.63	616.00	RC101_0.75	508.92	659.00
RC102_0.25	546.05	554.00	RC102_0.50	699.91	570.00	RC102_0.75	655.11	520.00
RC103_0.25	781.71	556.00	RC103_0.50	753.17	519.00	RC103_0.75	691.04	515.00
RC104_0.25	840.80	434.00	RC104_0.50	786.31	441.00	RC104_0.75	735.89	437.00
RC105_0.25	517.44	587.00	RC105_0.50	697.01	580.00	RC105_0.75	655.84	562.00
Average	679.97	453.27	Average	669.98	446.13	Average	647.97	439.60

Chapter 3

The energy-efficient green mixed fleet vehicle routing problem with partial battery recharging and time windows

We investigate a specific version of the Green Vehicle Routing Problem, in which we assume the availability of a mixed vehicle fleet composed of electrical and conventional (internal combustion engine) vehicles. We allow partial battery recharging at any of the available stations. In addition, we use a realistic energy consumption model which takes into account speed, load cargo and gradients. We propose a matheuristic embedded within the large neighborhood search scheme. In a numerical study we evaluate the behaviour of the proposed approach.

Keywords: Green vehicle routing; pollution-routing; integer linear mathematical model; matheuristic.

3.1 Introduction

The planning and management of freight logistics systems have traditionally aimed at improving the transportation efficiency in terms of cost, time and profit. More recently, we have witnessed a growing interest in the environmental aspects of transportation, such as pollution, noise and congestion. In this context, developing environmentally-friendly and efficient transport and distribution systems, defined in such a way to ensure the best trade-off between cost minimization and negative environmental externalities reduction, represents an important challenge.

We consider the problem of managing electrical and conventional vehicles with the aim of reducing the costs derived from the routing and the recharging operations.

In particular, we investigate a vehicle routing problem with a mixed fleet, composed by electrical and conventional diesel vehicles that have different capacities. We propose a realistic energy consumption model and we assume that partial battery recharges for each electrical vehicle are allowed at any available recharging station. Combining these elements makes the problem different from the other contributions and interesting from a point of view of the realistic applications.

We propose a mathematical formulation and we design and implement a matheuristic, which combines the resolution of the proposed model and a restrict subproblem, embedded in the large neighborhood search scheme.

3.1.1 State of the art

Here, we briefly review the most interesting scientific contributions in green logistics (for a complete survey the reader is referred to Lin et al. [38]). We can distinguish between two important categories of problems: Pollution Routing Problems (PRPs) and Green Vehicle Routing Problems (GVRPs). The former aim at minimizing pollution, in particular carbon emissions. The latter make use of alternative fuel vehicles (AFVs) and alternative fuel stations (AFSs). In both cases the main objective is to minimize energy consumption in transportation.

PRP. Bektas and Laporte [6] introduced and modeled the PRP. They explicitly considered the effect of CO_2 emissions, showed the difficulty of solving the PRP to optimality, and mentioned the possibility of several extensions. They proposed a non-linear mixed integer programming formulation to mathematically represent the problem, whose objective is to minimize the cost of greenhouse gas (GHG) emissions, the operational costs of drivers and fuel consumption. This work was extended by Demir et al. [13] who considered several vehicle speeds and proposed and tested an Adaptive Large Neighbourhood Search (ALNS) algorithm. Jabali et al. [24] focused on the Time-Dependent VRP. They presented a model that considers travel time, fuel and CO_2 emissions costs, and proposed a tabu search procedure to solve the problem. Franceschetti et al. [20] studied the Time-Dependent PRP, a PRP extension that takes traffic congestion into account. The authors proposed an integer linear programming formulation and considered a special case called the departure time and speed optimization problem. Tajik et al. [53] investigated the time window pickup-delivery PRP. In this PRP variant, pickup and delivery operations are considered and the vehicle speed is stochastic. The authors solved a mixed integer linear programming model (MILP) and introduced a robust variant. Koç et al. [27] studied a PRP extension that considers a heterogeneous vehicle fleet. They developed and tested a metaheuristic called HEA++.

G-VRP. One of the first articles on the G-VRP is that of Kara et al. [25]. In this work the authors consider a capacitated VRP and propose a linear integer formulation in order to reduce energy consumption.

Gonçalves et al. [22] investigated the VRP with pickups and deliveries. They analysed three different scenarios. The first one is an application of the VRP with pickups and deliveries with a conventional fleet; in the second one the fleet is composed by conventional vehicles and electrical uncapacitated vehicles; in the last one they considered only capacitated electrical vehicles. The authors proposed a MILP model and applied a p-median algorithm in order to decompose the original set of customers. The problem was then solved on each cluster. Erdoğan and Miller-Hooks [17] presented a MILP formulation for the G-VRP. Moreover, they proposed several techniques in order to find a solution that minimizes the total distance traveled, while incorporating stops for the refuelling of AFVs at AFSs. Vehicles are assumed to be uncapacitated and the time window constraints are not taken into account. Customer time windows, demands and capacity constraints were considered by Schneider et al. [50]. The authors focused on the Electrical VRP with time windows (VRPTW) with Recharging Stations (E-VRPTW). Recharging vehicles at any of the available stations is allowed, but the batteries must be fully charged. The authors presented a MIP formulation and proposed a hierarchical objective function of E-VRPTW. The first objective is the minimization of the number of vehicles; the second one is the minimization of the total traveled distance. Their approach is a metaheuristic that combines variable neighbourhood search (VNS) and tabu search. Felipe et al. [18] described the G-VRP with Multiple Technologies and Partial Recharge. Partial battery recharges and overnight depot charging are allowed. The recharging operations can be performed with different technologies, each of them having a different recharging time and cost. They proposed a constructive algorithm based on a greedy generation method, a deterministic local search and a simulated annealing. Ćirović et al. [10] investigated the G-VRP with a heterogeneous fleet composed by environmental friendly and unfriendly vehicles. However, when defining a route, friendly and unfriendly vehicles are considered separately. They use a neuro-fuzzy model to formulate the problem under study. Goeke and Schneider [21] considered a mixed fleet of conventional vehicles and EVs. The authors formulate the E-VRP with time windows and mixed fleet, in which the EVs can be charged at the available CSs. Charging times vary according to the battery level when the EV arrives at the CS and charging is always done up to maximum battery capacity. The authors propose a realistic energy consumption model which considers speed, vehicle mass and gradient. Desaulniers et al. [15] presented four variants of the E-VRPTW. In the first one, batteries must be fully charged and at most one recharge per route is allowed; in the second one multiple recharges are allowed, in the third one only one partial battery recharging per route is allowed, in the last one multiple and partial battery recharges are allowed. The authors developed two branch-and-price-and-cut algorithms in order to solve the problems. Hiermann et al. [23] introduced the electric fleet size and mix vehicle routing problem with time windows and recharging stations. They considered a heterogeneous fleet of EVs in which each vehicle is characterised by its fixed cost, battery and load capacity, energy consumption and charging rate. Each vehicle can be fully charged at a CS. Koç and Karaoglan [28] developed a simulated annealing heuristic based on an exact solution approach to solve the G-VRP introduced by Erdoğan and Miller-Hooks [17]. In their formulation, the authors introduce new decision variables in order to allow multiple visits to the CSs without augmenting the networks with dummy nodes. Based on this work, Leggieri and Haouari [35], proposed a new formulation for the E-VRPTW. In order to assess the effectiveness of their approach, the authors solve the proposed model by using CPLEX and compare the results with those obtained by the branch-and-cut algorithm proposed in Koç and Karaoglan [28].

Since the installation and operation costs of the network highly impact on company's strategies, several authors introduced decisions about location and technology of CSs in the E-VRPs. Yang and Sun [55] introduced the electric vehicles battery swap stations location routing problem whose aim is to determine the locations of battery swap stations, as well as the routing plan of EVs. Li-ying and Yuan-bin [36] focused on the EV multiple charging station location-routing problem with time windows. Schiffer and Walther [49] introduced the electric location routing problem with time windows and partial recharging in which the the EVs can be charged at any node in the network. Schiffer and Walther [48] proposed the location-routing problem with intra-route facilities which focuses on determining the location of facilities for intermediate stops. The facilities are not depots and do not necessarily coincide with customers. Intra-route facilities allow for intermediate stops on a route in order to keep the vehicle operational. Paz et al. [43] defined the multi-depot electric vehicle location routing problem with time windows and a homogeneous fleet of EVs. The authors considered the possibility to charge the EV to the CSs or to swap the battery to the battery swap stations. The goal is to determine the number and location of CSs and depots, as well as the number of EVs and their routes.

The mathematical formulation proposed in our paper can be viewed as an extension of the model presented by Erdoğan and Miller-Hooks [17], which is the first routing model that considers recharging stations. However significant modifications have been introduced in order to represent the specific characteristics of the problem under study. Schneider et al. [50] and Felipe et al. [18] have already extended the model presented in [17]. In the first contribution only complete recharges are allowed, while in the second one the batteries can be partially recharged with different technologies. However, in both papers, it is assumed that the fleet is exclusively composed of electrical vehicles. Here we extend the model in such a way to also handle conventional vehicles. Gonçalves et al. [22] and Goeke and Schneider [21] considered a mixed fleet. However, in both contributions, the batteries must be fully recharged. In addition, in the first of these papers, it is assumed that the electrical vehicles are uncapacitated. Ćirović et al. [10] considered conventional and electrical vehicles which composed the fleet separately. In the problem considered in our paper, we take these two possibilities into account.

3.1.2 Aim and organization of this paper

From this literature review, it is clear that scarce attention has been devoted to the use of a mixed vehicle fleet. The possibility of partial battery recharging is considered if a fleet is composed of only AFVs and the energy consumption is supposed to be proportional to traveled distance. Nobody has combined the four features considered in our paper, namely mixed fleet, partial battery recharging, time windows and realistic energy consumption model. In particular, we modeled a realistic energy consumption function which takes into account speed, load cargo and gradients. We consider also some realistic issues related to the life span of the battery. Indeed, full recharges can damage the battery and the last 10% of recharge requires considerable time. Thus, we also need to constrain the state of charge of the battery.

The remainder of this paper is structured as follows. In Section 3.2, we highlight the main characteristics of the problem under study and describe the mathematical model developed for its representation. In particular we use two energy consumption models for the conventional and electric vehicles described in Sections 3.2.1 and 3.2.2 respectively. Section 3.3 describes the algorithm proposed to solve the problem. In Section 3.4 we describe the computational experiments and we present the results. Section 3.5 summarizes the conclusions.

3.2 The energy-efficient green mixed fleet vehicle routing problem with partial battery recharging and time windows

We formulate our problem as follows. Let $\mathcal{N}$ be the set of customers, and let $\mathcal{R}$ be the set of recharging stations. We will also need copies of recharging stations to account for multiple visits at the same stations. Thus, let $\mathcal{R}'$ be the set of all stations and their copies, i.e. $R \subset R'$. Let $\mathcal{V} = \mathcal{R} \cup \mathcal{N}$ and $\mathcal{V}' = \mathcal{R}' \cup \mathcal{N}$. The problem will be defined on the graph $G(V', A)$, where $\mathcal{A}$ is the set of arcs.

The depot 0 is a particular element belonging to the set $\mathcal{R}'$, that is the recharging station where vehicle routes start and its dummy copy $0'$ is the node where the routes end. Each customer $i \in N$ has a demand q_i (in kg) and a service time s_i (in hours). All customers must be visited by a single vehicle. Each node $i \in \mathcal{V}'$ has a time window $[e_i, l_i]$.

For each $(i, j) \in \mathcal{A}$, d_{ij} denotes the distance from i to j [km], while t_{ij} the travel time from i to j [hours].We impose a limit T on the duration of a route [hours], that is, the end of the time window associated with the depot node is set equal to T.

A heterogeneous fleet of vehicles, composed of n^E electrical vehicles and n^C conventional vehicles, is available. The two types of vehicles (electrical and conventional) are characterized by different capacities, denoted as Q^E_{max} and Q^C_{max} [kg] for the electrical and conventional vehicles, respectively, and different curb weight denoted by w^E and w^C respectively. Furthermore, for each electrical vehicle let B denote the maximum battery capacity [kWh], while for each conventional vehicle B^C is the fuel tank maximum capacity [L]. The recharging cost [€/kWh] is equal to ω.

Each recharging station $i \in \mathcal{R}'$ has a charging mode (i.e., slow, moderate, fast), thus it is characterized by a recharging speed ρ_i [kWh per hour]. Partial battery recharging is allowed at any recharging station.

3.2.1 The fuel consumption for the conventional vehicles

We define an energy consumption model by following the ideas presented in [6]. To this end, we first calculate two constant α and β as follows:

$$\alpha = a + g\sin\theta + gC_r\cos\theta, \tag{3.1}$$

with a denoting the acceleration [m/s^2], assumed to be zero, while g denotes the gravitational constant (9.81 m/s^2), the angle of the road is θ, assumed to be zero, and C_r is the coefficient of rolling resistance:

$$\beta = 0.5C_dA\rho, \tag{3.2}$$

where C_d is the drag coefficient, A is the frontal surface Area [m^2], and ρ is the air density [kg/m^3)]. Let u_j be the amount of cargo (in kg) when arriving at node j, the mechanical power is calculated as follows:

$$p_{ij}^M(u_j) = [\alpha(u_j + w) + \beta v_{ij}^2]v_{ij}, \tag{3.3}$$

where w denotes the curb weight of the vehicle (in kg) and v_{ij} is the speed [m/s], assumed to be constant. We use the emission model of Barth et al. [5] and Barth and Boriboonsomsin [4], applied to the PRP by Bektas and Laporte [6], Demir et. al [13], Koç et. al [27] to estimate fuel consumption, thus we convert the mechanical power into fuel consumption. The fuel consumption rate $FR_{ij}(u_j)$ [L/s] is given by

$$FR_{ij}(uj) = \xi(kN_eD_e + p_{ij}^M(u_j)/\eta_e\eta_{dt})/\kappa\Psi, \tag{3.4}$$

where ξ is the fuel-to-air mass ratio, k is the engine friction factor [kJ/rev/L], N_e is the engine speed [rev/s], D_e is the engine displacement [L], $p_{ij}^M(u_j)$ is the mechanical power [kW], η_e and η_{dt} are the efficiency parameter for diesel engines and the drive train efficiency respectively, while κ is the heating value of a typical diesel fuel [kJ/g] and Ψ is a conversion factor from [g/s] to [L/s]. Therefore, the fuel consumption for traversing an arc (i, j) with the cargo u_j can be written as

$$f_{ij}(u_j) = t_{ij} FR_{ij}(uj). \tag{3.5}$$

3.2.2 The energy consumption for the electric vehicles

We calculate the energy consumption $p_{ij}^E(u_j)$ [KW] from a node i to a node j, starting from the mechanical power $p_{ij}^M(u_j)$ described in equation (3.3):

$$p_{ij}^E(u_j) = (p_{ij}^M(u_j)/\eta)t_{ij}, \tag{3.6}$$

where η is the energy efficiency from battery-to-wheels and it is given by η^+ in motor mode (i.e. $p_{ij}^E(u_j)$ is positive and it represents the discharged electric energy) and η^- in recuperating mode (i.e. $p_{ij}^E(u_j)$ is negative and it represents the recuperated electric energy). In particular

$$\eta = \begin{cases} \eta^+ \leq 1, & \text{if } p_{ij}^E(u_j) \text{ is positive, and } 0 \leq p_{ij}^M(u_j) \leq 100 \text{ KW} \\ \eta^- \geq 1, & \text{if } p_{ij}^E(u_j) \text{ is negative, and } -100 \leq p_{ij}^M(u_j) \leq 0 \text{ KW} \end{cases}. \tag{3.7}$$

3.2.3 The mathematical model

In order to model the problem we define the decision variables as follows:

$$x_{ij}^E = \begin{cases} 1, & \text{the electrical vehicle travels from i to j} \\ 0, & \text{otherwise} \end{cases} \quad (i, j) \in \mathcal{A}$$

$$x_{ij}^C = \begin{cases} 1, & \text{the conventional vehicle travels from i to j} \\ 0, & \text{otherwise} \end{cases} \quad (i, j) \in \mathcal{A}$$

z_j^E amount of energy available when arriving at node j [kWh], $j \in \mathcal{V}'$
z_j^C amount of fuel available when arriving at node j [L], $j \in \mathcal{V}'$

$z_{i0'}^{depotE}$ amount of energy available when arriving at depot $0'$ from node i [kWh], $i \in \mathcal{V}'$
$z_{i0'}^{depotC}$ amount of fuel available when arriving at depot $0'$ from node i [L], $i \in \mathcal{V}'$
g_{ij} amount of energy recharged by the electrical vehicle at the node i for travelling to j [kWh], $i \in \mathcal{R}, j \in \mathcal{V}'$
p_{ij}^{E} amount of energy necessary to travels from i to j; $i, j \in \mathcal{V}'$
τ_j arrival time of the vehicle to the node j [h], $j \in \mathcal{V}'$
u_i^C amount of load left in the conventional vehicle after visiting node i [kg], $i \in \mathcal{V}'$
u_i^E amount of load left in the electric vehicle after visiting node i [kg], $i \in \mathcal{V}'$
The Mixed Integer Program that models our problem is reported in what follows.

$$\text{Minimize} \quad \sum_{i\in R'}\sum_{j\in V'} \omega_i^g g_{ij} + \omega^e \sum_{i\in V'\setminus\{0'\}} (B^E - z_{i0'}^{depotE}) + \omega^f \sum_{(i,j)\in A} f_{ij}(u_i^C) + \sum_{(i,j)\in\mathcal{A}} cd_{ij}(x_{ij}^E + x_{ij}^C) \tag{3.8}$$

$$\text{subject to} \quad \sum_{j\in\mathcal{V}'} (x_{ij}^E + x_{ij}^C) = 1, \qquad i \in \mathcal{N} \tag{3.9}$$

$$\sum_{j\in\mathcal{V}'} x_{ij}^E \le 1, \qquad i \in \mathcal{R}' \tag{3.10}$$

$$\sum_{j\in\mathcal{V}'\setminus\{0\}} x_{ij}^E - \sum_{j\in\mathcal{V}'\setminus\{0'\}} x_{ji}^E = 0, \qquad i \in \mathcal{V}' \tag{3.11}$$

$$\sum_{j\in\mathcal{V}\setminus\{0\}} x_{ij}^C - \sum_{j\in\mathcal{V}\setminus\{0'\}} x_{ji}^C = 0, \qquad i \in \mathcal{V} \tag{3.12}$$

$$\sum_{j\in\mathcal{V}'} x_{0j}^E \le n^E \tag{3.13}$$

$$\sum_{j\in\mathcal{V}} x_{0j}^C \le n^C \tag{3.14}$$

$$\tau_j \ge \tau_i + (t_{ij} + s_i)x_{ij}^E - M(1 - x_{ij}^E), \qquad i \in \mathcal{N}, j \in \mathcal{V}' \tag{3.15}$$

$$\tau_j \ge \tau_i + (t_{ij} + s_i)x_{ij}^C - M(1 - x_{ij}^C), \qquad i \in \mathcal{V}, j \in \mathcal{V} \tag{3.16}$$

$$\tau_j \ge \tau_i + t_{ij}x_{ij}^E + \frac{1}{\rho_i} g_{ij} - M(1 - x_{ij}^E), \qquad i \in \mathcal{R}', j \in \mathcal{V}' \tag{3.17}$$

$$e_j \le \tau_j \le l_j, \qquad j \in \mathcal{V}' \tag{3.18}$$

$$u_j^E \ge u_i^E + q_j x_{ij}^E - Q_{max}^E(1 - x_{ij}^E), \qquad i \in \mathcal{V}'\setminus\{0, 0'\}, j \in \mathcal{V}'\setminus\{0\} \tag{3.19}$$

$$u_j^C \ge u_i^C + q_j x_{ij}^C - Q_{max}^C(1 - x_{ij}^C), \qquad i \in \mathcal{V}\setminus\{0, 0'\}, j \in \mathcal{V}\setminus\{0\} \tag{3.20}$$

$$u_0^C = 0 \tag{3.21}$$

$$u_0^E = 0 \tag{3.22}$$

$$z_j^E \le z_i^E - p_{ij}(u_i^E) + B(1 - x_{ij}^E), \qquad i, j \in V'\setminus\{0, 0'\} \tag{3.23}$$

$$z_j^E \le z_i^E + g_{ij} - p_{ij}(u_i^E) + B(1 - x_{ij}^E), \qquad i \in R', j \in V'\setminus\{0, 0'\} \tag{3.24}$$

$$z_0^E = 0.9B \tag{3.25}$$

$$z_{i0'}^{depotE} \le z_i^E - p_{i0'}(u_i^E) + B(1 - x_{i0'}^E), \qquad i \in V'\setminus\{0, 0'\} \tag{3.26}$$

$$z_{i0'}^{depotE} \le z_i^E + g_{i0'} - p_{i0'}(u_i^E) + B(1 - x_{i0'}^E), \qquad i \in R' \tag{3.27}$$

$$
\begin{aligned}
&0 \leq z_{i0'}^{depotE} \leq 0.9B^E x_{i0'}^E, && i \in V' \backslash \{0'\} && (3.28)\\
&0.1B \leq z_j^E \leq 0.9B, && j \in V' \backslash \{0\} && (3.29)\\
&g_{ij} \leq 0.9B - z_i^E + B(1 - x_{ij}^E), && i \in R', j \in V' && (3.30)\\
&p_{ij}^E \geq ([\alpha(u_i^E + w^E) + \beta v_{ij}^2] v_{ij}) t_{ij}/\eta^+, && i, j \in V' && (3.31)\\
&p_{ij}^E \geq ([\alpha(u_i^E + w^E) + \beta v_{ij}^2] v_{ij}) t_{ij}/\eta^-, && i, j \in V' && (3.32)\\
&z_j^C \leq z_i^C - f_{ij}(u_i^C) + B^C(1 - x_{ij}^C), && i, j \in V \backslash \{0'\} && (3.33)\\
&0 \leq z_j^C \leq B^C, && j \in V \backslash \{0\} && (3.34)\\
&z_0^C = B^C && && (3.35)\\
&z_{i0'}^{depotC} \leq z_i^C - f_{i0'}(u_i^C) + B^C(1 - x_{i0'}^C), && i \in V \backslash \{0'\} && (3.36)\\
&0 \leq z_{i0'}^{depotC} \leq 0.9B^C x_{i0'}^C, && i \in V \backslash \{0'\} && (3.37)\\
&x_{ij}^E, x_{ij}^C \in \{0,1\}, i \in \mathcal{V}', j \in \mathcal{V}'; u_i^E, u_i^C, \tau_i \geq 0, i \in \mathcal{V}', && &&\\
&g_{ij} \geq 0, i \in \mathcal{R}', j \in \mathcal{V}'. && && (3.38)
\end{aligned}
$$

The objective function is the sum of three terms. The first one, that is $\sum_{i\in R'}\sum_{j\in V'}\omega_i^g g_{ij}$, is the cost of the energy recharged during the route. In particular ω_i^g is the unit cost of recharge [€/kW] at station i and it depends of the available technology at station i. The second one ($\omega^e \sum_{i\in V'\backslash\{0'\}}(B^E - z_{i0'}^{depotE})$) is the cost of the energy recharged to the depot, the unit cost of energy is ω^e [€/km]. The third one, that is $\omega^f \sum_{(i,j)\in A} f_{ij}(u_i^C)$ is the fuel cost, with ω^f the unit cost of fuel [€/L]. The last one, $\sum_{(i,j)\in\mathcal{A}} cd_{ij}(x_{ij}^E + x_{ij}^C)$ is the travel cost. The constraints (3.9) ensure that each customer is visited exactly once, whereas conditions (3.10) impose that each recharging station can be visited at most once. Constraints (3.11) and (3.12) are the flow conservations constraints, whereas conditions (3.13) and (3.14) ensures that the total number of used vehicles (electrical and conventional, respectively) is less than the available ones. Constraints (3.15)–(3.17) define the variables τ, whereas the time windows constraints are represented by conditions (3.18). Conditions (3.19)–(3.22) represent the capacity constraints, for the electrical and the conventional vehicles. Constraints (3.23) and (3.24) define the variables z^E ensuring that the capacity of the electric vehicles battery is not exceeded and conditions, in particular after visiting a customer and a recharge stations respectively. Constraint (3.25) ensures that the vehicle is full charged at starting node, while constraints (3.26) and (3.27) define the amount of energy available when the vehicles arrive at ending node. Constraints (3.30) are used to represent the partial battery recharging. Constraints (3.32-3.33) linearise the energy consumption as described in Section 3.2.2. Constraints (3.33) set the fuel level equal to the maximum fuel tank capacity reduced by the fuel necessary to traverse the arc, constraints (3.34) and (3.35) restrict the fuel level, while constraints (3.36) and (3.37) define the available amount of fuel and restrict the fuel level for the ending node respectively. Finally, conditions (3.38) define the domains of variables.

3.3 A matheuristic algorithm

We have developed a matheuristic to solve our problem. In particular, we propose a hybrid version of large neighborhood search (HLNS) algorithm introduced by Shaw [51], which iteratively removes and inserts customers from the routes in the solution. We generated an initial feasible solution Γ_{current} by solving the proposed model with CPLEX. We fixed a limit $\bar{t}$ on the execution time. We random applied removal and insertion operators obtaining the solutions Γ_{remove} and Γ_{insert} respectively. Contrary to the majority of contributions on LNS, which work only with feasible solutions, we relaxed the battery capacity constraints for the EVs and we allowed infeasible insertion in the route. Since the electric routes could be infeasible, we applied a repair phase. In order to repair the solution by inserting CSs in the infeasible routes, we solved the E-VRP counterpart of the model presented in Section 3.2.3. The E-VRP takes into account the constraints 3.9–3.11, 3.13, 3.15, 3.17–3.19, 3.22–3.32 , 3.36, and the variables related to the EVs, thus we obtain $\Gamma_{\text{temporary}}$. The procedure was repeated with the best solution found Γ_{best} or an accepted current solution whose cost is minor than cost (Γ_{best}) υ where υ is a tolerance input parameter, until the stopping criteria (i.e. a maximum number of iterations $k^{\max}$) was met.

Algorithm 4 . Hybrid large neighborhood search (HLNS)

Generate the initial solution Γ_{current}
$\Gamma_{\text{current}} \rightarrow \Gamma_{\text{best}}$
while $k < k^{\max}$ **do**
  Apply a removal operator to Γ_{current} and obtain Γ_{remove}
  Apply a insertion operator to Γ_{remove} and obtain Γ_{insert}
  Apply a repair using CPLEX to solve the E-VRP counterpart and obtain $\Gamma_{\text{temporary}}$
  if cost($\Gamma_{\text{temporary}}$) < cost($\Gamma_{\text{best}}$) **then**
    $\Gamma_{\text{temporary}} \rightarrow \Gamma_{\text{best}}$
    $\Gamma_{\text{temporary}} \rightarrow \Gamma_{\text{current}}$
    $k = 0$
  else if cost($\Gamma_{\text{temporary}}$) < cost($\Gamma_{\text{best}}$) v) **then**
    $\Gamma_{\text{temporary}} \rightarrow \Gamma_{\text{current}}$
    $k \leftarrow k + 1$
  else
    $k \leftarrow k + 1$
  end if
end while
return best solution Γ_{current}

3.3.1 Removal and Insertion operators

We now describe our removal and insertion operators. Removal operators remove ζ customers and then place them in a removal list. The value of ζ is selected from an interval $[\zeta^-, \zeta^+]$, where ζ^- and ζ^+ are input parameters. Insertion operators insert ζ customers in the destroyed solution by following several rules. We introduce a temporary tabu status which forbids the insertion (removal) of customers in (from) routes which have been recently removed (inserted) from (in) routes. We introduce a temporary tabu status which forbids the insertion of customers in routes which have been recently removed from routes, as well as the removal of customers which have been recently inserted in routes.

Removal operators

Our ALNS uses the following four destroy operators:

1. **Random removal**: iteratively removes ζ customers from a solution.
2. **Worst distance removal**: iteratively removes the unfavorable customers. The operator sorts all the customers in descending order of cost, where the cost is the sum of distances of the customer from the preceding and succeeding nodes in the route.
3. **Worst time removal**: similar to the worst distance removal, the operator sorts all the customers in descending order of cost, where cost for a node i is calculated as $|\tau_i - e_i|$.
4. **Route removal**: this operator randomly selects a route and remove it from the solution.

Insertion operators

We use three insertion operators. During the insertion of customers in routes, we relax the battery capacity constraints, hence we allow infeasible solutions for EV-routes.

- **Greedy insertion**: iteratively determines the best insertion position for a customer by calculating the insertion cost between two nodes in the route.
- **Greedy new route insertion**: this operator initializes a new route, electric or conventional by evaluating the insertion cost of a customer between the starting and ending nodes.
- **GRASP insertion**: this operator sorts customers in a list of size L in ascending order of insertion cost. Then it selects the next customer to be inserted among the best r^{GRASP} insertions, where r^{GRASP} is a random number in $[0, r^{GRASP}L/2]$.

3.4 Computational study

This section presents the results of our preliminary computational experiments. We tested the implemented algorithm on instances inspired from the scientific literature (see [52] and [50]). We solved the model with CPLEX 12.5, by imposing a time limit of three hours. The HLNS was implemented in Java. All computations were performed on an Intel 2.60 GHz processor and 16 GB of RAM. Tables 3.1 summarizes the parameter setting used for our computational results.

Table 3.1: Setting of conventional and electrical vehicles parameters

Notation	Description	Value
g	Gravitational constant [m/s^2]	9.81
θ	Road angle	0
C_r	Coefficient of rolling resistance	0.01
C_d	Drag coefficient	0.7
A	Frontal surface Area [m^2]	3.912
ρ	Air density [kg/m^3]	1.225
w	Curb weight (kg/m^3]	6350
v	Speed [m/s]	13.88
ξ	Fuel-to-air mass ratio	1
k	Engine friction factor [kJ/rev/L]	0.2
N_e	Engine speed [rev/s]	33
D_e	Engine displacement [L]	5
η_e	Efficiency parameter for diesel engines	0.9
η_{dt}	Drive train efficiency	0.4
κ	Heating value of a typical diesel fuel [kJ/g]	44
Ψ	Conversion factor [g/L]	737
B^C	Fuel tank maximum capacity [kg])	3650
B	Maximum battery capacity [kWh]	80
ρ	Recharging speed [W/min]	0.0083
ω_i^g	Unit cost of recharge at station i [€/kW]	0.4
ω^e	Unit cost of energy [€/kW]	0.17
ω^f	Unit cost of fuel (€/L)	1.3
c	Driver wage [€/km]	0.195

The remainder of this section is organized as follows. In Section 3.4.1 we describe the generation of instances, in Section 3.4.2 we compare our electrical energy consumption model to the classical one in which energy consumption is proportional to the travelled distance. In Section 3.4.3 we present the results of our matheuristic on small-size instances with ten and 15 customers. In Section 3.4.4 we test our HLNS on the medium-size instances

3.4.1 Generations of instances and experimental setting

For each E-VRPTW instance, with customer locations (a_i,b_i), we generate the charging station in the square ($\min_i a_i$, $\min_i b_i$) and upper right hand corner ($\max_i a_i$, $\max_i b_i$) by solving a location problem. Let N be the set of customers defined in Section 3.2 and Y be the set of candidate charging stations. We define the decision variables as follows: $y_j = \begin{cases} 1, & \text{charging station } j \text{ is activated} \\ 0, & \text{otherwise} \end{cases} \quad j \in \mathcal{Y}$

$$x_{ij} = \begin{cases} 1, & \text{customer } i \text{ is served by } j \\ 0, & \text{otherwise} \end{cases} \quad i \in \mathcal{N}, j \in \mathcal{Y}$$

Thus we formulate and solve the following problem:

$$\begin{aligned}
\text{Minimize} \quad & \sum_{i \in R} \sum_{j \in \mathcal{Y}} p_{ij} x_{ij} + c^f \sum_{j \in Y} y_j && && (3.39) \\
\text{subject to} \quad & p_{ij} x_{ij} \le B^E y_j, && i \in \mathcal{N}, j \in \mathcal{Y} && (3.40) \\
& p_{ij} x_{ij} \ge 1, && i \in \mathcal{N} && (3.41) \\
& \sum_{j \in \mathcal{Y} y_j} \ge H && && (3.42) \\
& x_{ij} \in \{0,1\} && i \in \mathcal{N}, j \in \mathcal{Y} && (3.43) \\
& y_j \in \{0,1\} && j \in Y, && (3.44)
\end{aligned}$$

where H is a minimum number of charging stations that we want to activate. We calculate p_{ij} as described in Section 3.2.2. We fix a configuration by considering a partially laden vehicle in recuperating mode. With this configuration, we ensure that each customer may be reached by using at least one charging station.

3.4.2 Proportional consumption model vs Energy consumption model

To assess the importance of the proposed energy consumption model, we have conducted several computational experiments. We have compared solutions obtained by using the proposed electrical energy function, which depends on several realistic factors, with those obtained assuming that electrical energy consumption is proportional to the travelled distance, as commonly used in the literature. The experiments consist of solving to optimality a set of small-size instances with five customers, and comparing solutions in terms of cost and feasibility (i.e. since the proportional model may underestimate the energy consumption, some configurations generated using this model may be infeasible when our realistic energy consumption model is considered. More specifically, constraints (3.23)–(3.24) and (3.26)–(3.30) of the MIP model presented in Section 3.2.3 are not satisfied). We have considered optimal solutions delivered by the MILP proposed in Section 3.2.3. Table 3.2 presents the results. For each instance we report its name, the percentage error in cost ($\mathrm{p_e}$) calculated as $(\mathrm{Obj_{new}} - \mathrm{Obj_{prop}})/\mathrm{Obj_{new}}$, where $\mathrm{Obj_{prop}}$ is the objective function obtained by using the proportional consumption model and $\mathrm{Obj_{new}}$ is the objective function obtained by using our consumption model, the difference in consumption energy ($\mathrm{d_e}$), calculated as the difference between the energy spent calculated with our consumption model and the energy spent calculated with the proportional model. In the last column, we report the infeasible solutions obtained by using the proportional model (no feasible solution NFS).

Table 3.2 clearly shows that all the values of energy consumption calculated by the proportional consumption model are underestimated. $\mathrm{Obj_{nofunction}}$ is always lower than $\mathrm{Obj_{function}}$, due to the

Table 3.2: Comparison of our proposed energy consumption model and proportional energy consumption model

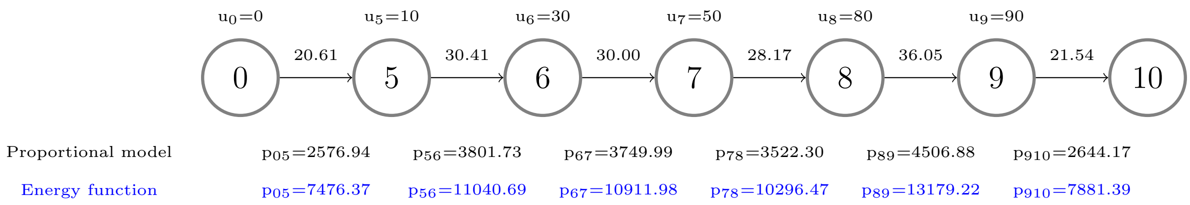
Test	p_e	d_e	NFS
MF_C101_5	9%	-	NFS
MF_C103_5	9%	35021.41	
MF_C206_5	9%	-	NFS
MF_C208_5	9%	36364.84	
MF_R104_5	5%	18993.37	
MF_R105_5	6%	24255.30	
MF_R202_5	9%	30196.18	
MF_R203_5	9%	-	NFS
MF_RC105_5	4%	26238.40	
MF_RC108_5	3%	26140.77	
MF_RC204_5	9%	-	NFS
MF_RC208_5	9%	38839.69	

inaccurate calculation of electrical energy consumption. On average, the percentage error in cost is about 7%, and four solutions out of 12 are infeasible. Table 3.3 reports an example of the solution obtained for MF_C101_5 by using the two different models. To obtain a feasible solution, we have relaxed the battery constraints in the proposed MILP and we have calculated the energy consumption on each arc, as showed in Figure 3.1. In particular, on each arc, we report the distance [km] and on each node i the collected cargo u_i [kg]. We also report the energy spent on each arc [kWh], calculated by the proportional consumption model and by our proposed one. The results clearly demonstrate that the proportional consumption model underestimates the values of energy spent. On average, on each arc the real energy spent is 66% more than the energy calculated with the proportional model. In conclusion, using a proportional consumption model simplifies the problem but can lead to solutions that are infeasible or have objective functions that are, on average, about 7% cheaper than the realistic ones.

Table 3.3: Solution obtained for MF_C101_5

Test	Energy Consumption model	Total Cost	Energy spent	#electrical
MF_C101_5	Proportional	70.27	20832,03	1
	Energy function	85.66	60740.73	1

Figure 3.1: Rapresentation of solution obtained for MF_C101_5



3.4.3 Evaluation of the HLNS performance

In order to assess the performance of our proposed LNS, we first solved the problem using CPLEX, and we then compared the results with those obtained with the HLNS. We evaluate the performance of the proposed heuristic along two dimensions: solution quality and computational effort. We used the parameter setting presented in Table 3.4.

Table 3.4: Parameters setting for instances with ten and 15 customers

	10 customers	15 customers
$k^{\max}$	5	10
$\bar{t}$ [seconds]	[10–20]	[10–50]
n^{C}	1	1
n^{E}	1	1

Tables 3.5 and 3.6 summarize the results obtained for instances with ten and 15 customers respectively. The first column shows the name of instances, in the second one g_{cost} represents the percentage gap in cost defined as $g_{cost} = (c^H - c^M)/c^M$, where c^H is the cost provided by the heuristic and c^M is the cost obtained solving the model. In the third column we report the speedup value i.e. the ratio between the

computational overhead of the heuristic and the computational time required by CPLEX. The entries in bold are those for which CPLEX finds an optimal solution.

The results presented in Tables 3.5 and 3.6 clearly demonstrate the advantage, in terms of efficiency, of the proposed matheuristic. Overall, the algorithm is less time consuming than CPLEX. The LSN is on average about 905 and 80 times faster than CPLEX for instances with ten and 15 customers respectively. It is worth observing that our matheuristic is also effective, the gap on cost for this class of instances is on average less than 1%. In particular, for those instances with ten customers, the HLNS finds the same solution as CPLEX for five instances and outperforms CPLEX for one instance. The average on the cost gap is 0.9%. For instances with 15 customers our matheuristic outperforms CPLEX finding the best solution for two instances: "MF_R10215" and "MF_R20215", and the average on the cost gap is 0.7%.

Table 3.5: Computational results for instances with ten customers

Test	g_cost	Speedup
MF_C101_10	**0.0%**	**1018.75**
MF_C104_10	**0.0%**	**1136.48**
MF_C202_10	0.0%	5259.74
MF_C205_10	1.1%	226.99
MF_R102_10	**0.4%**	**96.34**
MF_R103_10	**0.9%**	**696.86**
MF_R201_10	**5.5%**	**637.35**
MF_R203_10	**2.6%**	**93.14**
MF_RC102_10	-3.1%	115.80
MF_RC108_10	**0.0%**	**31.55**
MF_RC201_10	0.0%	1262.14
MF_RC205_10	**3.8%**	**284.28**
Average	0.9%	904.95

Table 3.6: Computational results for instances with 15 customers

Test	g_cost	Speedup
MF_C103_15	**5.2%**	**138.55**
MF_C202_15	4.6%	29.49
MF_C208_15	2.5%	88.58
MF_R102_15	-7.2%	4.19
MF_R105_15	0.1%	76.05
MF_R202_15	-4.4%	177.89
MF_RC103_15	2.0%	76.58
MF_RC202_15	2.0%	64.64
MF_RC204_15	1.7%	62.92
Average	0.7%	79.88

3.4.4 Numerical results on the medium-size instances

Now we present the computational results on instances with 20, 25, 30 and 35 customers. We used the parameter setting presented in Table 3.7.

Table 3.7: Parameters setting for medium-size instances

	20	25	30	35
$k^{\max}$	10	10	10	10
$\bar{t}$ [seconds]	[10–50]	[20–50]	[20–100]	[20–100]
n^{C}	1	1	2	2
n^{E}	1	1	2	2

Table 3.8 summarize the results obtained for medium-size instances. The first column displays the name of the instance, the second one the computational time [ms], the third one the total cost, the fourth and fifth columns show the number of conventional and electrical vehicles respectively. We also report, in the last line of each class of instances, the average for all the statistics. Overall, the HLNS finds solutions within a reasonable amount of time. Indeed, it solves instances with 20, 25 and 35 customers in about two minutes, and with 30 in about seven minutes. On average, our matheuristic solve the medium-size instances in about four minutes.

Table 3.8: Computational results for medium-size instances

Test	Time	Objective	#conventional	#electrical
MF_C101_20	250929	64.40	0	1
MF_C102_20	110550	86.04	1	1
MF_C103_20	150830	65.12	0	1
MF_C104_20	10147	65.33	1	0
MF_C105_20	130547	74.50	1	1
Average	130600.6	71.07	0.6	0.8
MF_C101_25	50207	89.81	1	0
MF_C102_25	250539	89.17	1	1
MF_C103_25	250665	85.08	1	1
MF_C104_25	50179	76.33	1	0
MF_C105_25	250777	91.76	1	1
Average	170473.4	86.43	1	0.6
MF_C101_30	491303	127.62	1	1
MF_C102_30	371751	99.63	1	1
MF_C103_30	611694	120.50	1	1
MF_C104_30	250645	111.25	1	1
MF_C105_30	451099	95.68	1	1
Average	435298.4	110.93	1	1
MF_C101_35	271169	193.67	2	1
MF_C102_35	161392	141.85	1	1
MF_C103_35	130821	149.66	1	1
MF_C104_35	271653	161.35	1	1
MF_C105_35	20337	143.13	2	0
Average	171074.40	157.92	1.4	0.80

3.5 Conclusions

In this work we have investigated a new and realistic green vehicle routing problem variant. In particular, we have considered a mixed fleet composed of electrical and conventional vehicles, with time windows associated with each customer and partial battery recharging. We have modeled a realistic energy consumption, which takes into account several real-life parameters. We have defined a mixed integer program whose aim is to route the fleet of vehicles in order to serve all the customers satisfying the time window constraints, minimizing the transportation and the recharging costs. To highlight the importance of the proposed energy consumption model, we have shown that models which considers energy consumption proportional to travelled distance may lead to infeasible solutions. We have then proposed a matheuristic based on large neighborhood search for the problem. In order to validate the model and to assess the performance of our proposed heuristic, we have solved the model with CPLEX for small instances. Overall, the matheuristic is less time consuming than CPLEX. We have also shown that our HLNS can solve medium-size instances within a reasonable amount of time.

Acknowledgement

This work was partly funded by the Canadian Natural Sciences and Engineering Research Council under grant 2015-06189. This support is gratefully acknowledged.

Part II

Vehicle routing problem for urban last-mile logistics

Chapter 4

The Vehicle Routing Problem with Occasional Drivers and TimeWindows

Abstract

In this paper, we study a variant of the Vehicle Routing Problem with Time Windows in which the crowd-shipping is considered. We suppose that the transportation company can make the deliveries by using its own fleet composed of capacitated vehicles and also some occasional drivers. The latter can use their own vehicle to make either a single delivery or multiple deliveries, for a small compensation. We introduce two innovative and realistic aspects: the first one is that we consider the time windows for both the customers and the occasional drivers; the second one is the possibility for the occasional driver to make multiple deliveries. We consider two different scenarios, in particular, in the first one multiple deliveries are allowed for each occasional driver, in the second one the split delivery policy is introduced. We propose and validate two different mathematical models to describe this interesting new setting, by considering several realistic scenarios. The results show that the transportation company can achieve important

advantages by employing the occasional drivers, which become more significant if the multiple delivery and the split delivery policy are both considered

Keywords: Vehicle Routing Problem; Crowd-shipping; Occasional Drivers.

4.1 Introduction

In the last years the growing importance of shorter delivery lead times has led the companies to create innovative solutions to organize the last-mile and same-day delivery. In this context, the "sharing economy" has attracted a great deal of interest. Sharing assets and capacities can enhance the use of resource and become a new opportunity to pursue the efficiency in transportation issue. One of the innovative solution is the crowd-shipping, i.e. ordinary people bring items for other people en-route to their destination. The rapid growth in on-line retailing has encouraged the retailers to develop innovative solutions of last-mile delivery. Walmart, DHL and Amazon are among those big retailers who started to use the crowd-shipping and its potential. Walmart, in 2013, announced it was working on a plan to outsource some of its deliveries to its on-line customers.

"MayWays" is the pilot last-mile crowd-shipping service of DHL in Stockholm. Thus, people in Stockholm, mostly students, can use a smartphone app in order to see the uploaded requests. Whereupon, they can decide if they want to pick up the package at a DHL facility and deliver it to the final destination. In 2015, Amazon launched its new service of crowd-shipping, called Amazon Flex, and nowadays it is already used in more than 30 cities in the world. People use the Amazon Flex app to become a delivery partner and set their own schedule.

In this context, crowd-shipping seems to be a new transport solution that offers several potential benefits. The crowd-shipping strategy exploits the personal vehicles capacity that usually travel on roads. This can be useful to reduce the need for separate freight deliveries and it allows to exploit the whole capacity of a vehicle, that is not fully used. There is also a "green" aspect that can be taken into account, in particular, the sharing of the vehicles can lead to the reduction of the pollutant emissions, the energy consumption, the noise and the traffic. In Arslan et al. [2], the authors analyze the potential benefits of crowd-sourced delivery. They present a complete literature review of the recent contributions dealing with crowd-sourcing. They consider a peer-to-peer platform, taking into account the possibility to use both traditional vehicles and ad-hoc vehicles. They also present a rolling horizon framework and an exact solution approach to solve the routing planning problem. In this work, we propose a variant of the Vehicle Routing Problem with Time Windows (VRPTW), starting from the work presented in Archetti et. al [1], in which the crowd-shipping is considered. In Archetti et. al [1], the authors propose a new problem called VRPOD (Vehicle Routing Problem with Occasional Drivers). In the VRPOD, the transportation company can make the deliveries not only by using its own fleet composed by capacitated vehicles, but also by making use of the services of some occasional drivers (ODs). The latter can use their own vehicle to make a single delivery, for a small compensation calculated by evaluating the deviation from their predefined route. They propose an integer programming formulation for the VRPOD and develop a multi-start heuristic, which combines variable neighbourhood search and tabu search. We introduce two innovative aspects to the problem proposed in Archetti et. al [1]. The first one is that we consider time windows constraints for both the customers and the occasional drivers (VRPODTW); indeed, it is merely improbable that an occasional driver is all the time available to make the delivery to the customers. The second one is the possibility for the ODs to make not only a single delivery. As a matter of fact, if on the occasional driver's way there is more than one customer to be served, and the constraints related to time and load are satisfied, the multiple delivery is allowed. The proposed mathematical models are described in Section 4.2. The computational experiments are presented in Section 4.3. Finally, section 4.3.3 summarizes the conclusions.

4.2 The Vehicle Routing Problem with Occasional Drivers

We model the problem on a complete directed graph $G = (N, A)$, with node set $N = C \cup \{s, t\} \cup V$, where C is the set of customers while s is the origin node and t is destination node for the classic vehicles. Let A be the set of arcs. K is the set of available ODs while V is the set of v_k destinations associated with the ODs. Each arc $(i, j) \in A$ has a cost c_{ij} and a time $t_{i,j}$ associated with it. Note that both c_{ij} and t_{ij} satisfy the triangle inequality. Each node $i \in C \cup V$ has a time windows defined as $[e_i, l_i]$. Each customer $i \in C$ has a demand d_i. Q is the capacity of the classic vehicles, P is the number of available classic vehicles, while Q_k is the capacity of OD $k \in K$. Let x_{ij} be a binary variable that is equal to 1 if a classical vehicle traverses the arc (i, j), and 0 otherwise. For each node $i \in N$ let y_i be the available capacity of the vehicle after visiting customer i, while s_i is the arrival time of the vehicle to the customer i. Moreover r_{ij}^k is a binary variable that is equal to 1 if the OD k traverses the arc (i, j), 0 otherwise. Let f_i^k indicate the arrival time of OD k to the customer i and let w_i^k be the available capacity of OD k after visiting customer i. At first we consider the scenario in which multiple deliveries for the OD are allowed and we called this version: VRPODTWmd. The VRPODTWmd can be formulated as follows:

$$
\begin{aligned}
\text{Minimize} \quad & \sum_{i \in C \cup \{s\}} \sum_{j \in C \cup \{t\}} c_{ij} x_{ij} + \sum_{k \in K} \sum_{i \in C \cup \{s\}} \sum_{j \in C} \rho c_{ij} r_{ij}^k - \sum_{k \in K} \sum_{j \in C} c_{sv_k} r_{sj}^k && && (4.1)\\
\text{subject to} \quad & \sum_{j \in C \cup \{t\}} x_{ij} - \sum_{j \in C \cup \{s\}} x_{ji} = 0, && \forall i \in C && (4.2)\\
& \sum_{j \in C} x_{sj} - \sum_{j \in C} x_{jt} = 0 && && (4.3)\\
& y_j \geq y_i + d_j x_{ij} - Q(1 - x_{ij}), && \forall j \in C \cup \{t\}, i \in C \cup \{s\} && (4.4)\\
& y_s \leq Q && && (4.5)\\
& s_j \geq s_i + t_{ij} x_{ij} - \alpha(1 - x_{ij}), && \forall i \in C, j \in C && (4.6)\\
& e_i \leq s_i \leq l_i, && \forall i \in C && (4.7)\\
& \sum_{j \in C} x_{sj} \leq P && && (4.8)\\
& \sum_{j \in C \cup \{v_k\}} r_{ij}^k - \sum_{h \in C \cup \{s\}} r_{hi}^k = 0, && \forall i \in C, k \in K && (4.9)\\
& \sum_{j \in C \cup \{v_k\}} r_{sj}^k - \sum_{j \in C \cup \{s\}} r_{jv_k}^k = 0, && \forall k \in K && (4.10)\\
& \sum_{k \in K} \sum_{j \in C \cup \{v_k\}} r_{sj}^k \leq |K| && && (4.11)\\
& \sum_{j \in C} r_{sj}^k \leq 1, && \forall k \in K && (4.12)\\
& w_j^k \geq w_i^k + d_i r_{ij}^k - Q_k(1 - r_{ij}^k), && \forall j \in C \cup \{v_k\}, i \in C \cup \{s\}, k \in K && (4.13)\\
& w_s^k \leq Q_k, && \forall k \in K && (4.14)\\
& f_i^k + t_{ij} r_{ij}^k - \alpha(1 - r_{ij}^k) \leq f_j^k, && \forall i \in C, j \in C, k \in K && (4.15)\\
& f_i^k \geq e_{v_k} + t_{si}, && \forall i \in C, k \in K && (4.16)\\
& f_{v_k}^k \leq l_{v_k}, && \forall k \in K && (4.17)\\
& f_i^k + t_{iv_k} r_{iv_k}^k - \alpha(1 - r_{iv_k}^k) \leq f_{v_k}^k, && \forall i \in C, k \in K && (4.18)\\
& e_i \leq f_i^k \leq l_i, && \forall i \in C && (4.19)\\
& \sum_{j \in C \cup \{t\}} x_{ij} + \sum_{h \in C \cup \{v_k\}} \sum_{k \in K} r_{ih}^k = 1, && \forall i \in C && (4.20)\\
& x_{ij} \in \{0, 1\}, && \forall (i,j) \in A && (4.21)\\
& r_{ij}^k \in \{0, 1\}, && \forall (i,j) \in A, k \in K && (4.22)\\
& 0 \leq y_i \leq Q, && \forall i \in C \cup \{s, t\} && (4.23)\\
& 0 \leq w_i^k \leq Q_k, && \forall i \in C \cup \{s, v_k\}, k \in K && (4.24)\\
& f_i^k \geq 0, && \forall i \in C \cup \{s, v_k\}, k \in K && (4.25)
\end{aligned}
$$

The objective function 5.1 aims to minimize the total costs. The first term is the transportation cost associated with the vehicles. The second term is the cost of compensation of the OD k for the delivery service with $\rho \geq 0$, the third one is the cost of the OD k when it does not perform the delivery service. Constraints 5.2-5.8 are linked to the classical vehicles. Constraint 5.2-5.3 are the flow constraints. Constraints 5.4 guarantee the fulfilment of demand at customer vertices. Constraints 5.5 restrict the initial cargo load level to the maximum capacity of a vehicle. Constraints 5.6 allow to determine the arrival time at node j, while constraints 5.7 guarantee arrival within the time window at each node. Constraint 5.8 imposes a maximum number of available vehicles. Constraints 5.9-5.19 are linked to the ODs. Constraints 5.9-5.10 are the flow constraints. Constraints 5.11-5.12 guarantee a limit on the number of available ODs and the number of departs from the depot. Constraints 5.13-5.14 are the capacity constraints. Constraints 5.15 allow to determine the arrival time at node j. Constraints 5.16-5.17 are the time windows constraints and they also define the time in which the ODs are available to make the deliveries, while constraints 5.18 allow to determine the arrival time at the destination node v_k. Constraints 5.19 assure that each customer is served within its time windows. Constraint 5.20 guarantees that each customer is visited at most once, by either a classic vehicle or an OD. We also formulate a second VRPOD variant, called VRPODTWsd in which we consider a split delivery policy for the ODs. Thus, the assumption that each customer is visited only once by the ODs is relaxed (constraint 5.20). We introduce a new variable o_i^k that indicates the quantity of demand d_i delivered by the OD $k \in K$ to the customer $i \in C$. In order to define the VRPODTWsd, starting from VRPODTWmd, constraints 5.13 and 5.20 are modified as follows:

$$w_j^k \geq w_i^k + o_i^k - Q_k(1 - r_{ij}^k), \qquad \forall j \in C \cup \{v_k\}, i \in C \cup \{s\}, k \in K \tag{4.26}$$

$$\sum_{j \in C \cup \{t\}} x_{ij} + \sum_{h \in C \in \{v_k\}} \sum_{k \in K} r_{ih}^k \geq 1, \qquad \forall i \in C \tag{4.27}$$

It is also necessary to introduce new constraints for modelling the split delivery policy for the ODs:

$$\sum_{k \in K} o_i^k + d_i \sum_{j \in C \cup \{s\}} x_{ji} = d_i, \qquad \forall i \in C \tag{4.28}$$

$$\sum_{i \in C} o_i^k \leq Q_k, \qquad \forall k \in K \tag{4.29}$$

$$o_i^k \geq 0, \qquad \forall i \in C, k \in K \tag{4.30}$$

4.3 Computational experiments

This section presents the results of computational experiments performed in order to validate the proposed models. The main goal is to demonstrate the potential benefits that can be obtained by using the crowd-shipping in a realistic scenario. With this purpose we take into account the state-of-art mathematical model proposed in Archetti et. al [1], we add to this problem the time windows (VRPODTW) and we find the optimal solution by solving it with a commercial solver. Whereupon, we solve our proposed models and compare the obtained results. We divided the comparative analysis into two phases, in the first one the results obtained solving the VRPODTW is compared to those obtained by solving the VRPODTWmd. In the second phase, we present a comparative analysis of the results obtained by solving the VRPODTWmd and those obtained with the split delivery policy, the VRPODTWsd. The models were implemented and solved with the commercial solver CPLEX 12.5 and run on a computer with an Intel Core i5 processor at 2.70 GHz and 4GB of RAM. We first describe the generated instances and after the results.

4.3.1 Generation of VRPOD instances

The instances, used to assess the behaviour of proposed models in terms of solutions quality, are based on the classical Solomon VRPTW instances (see Solomon [52]). As well known, these instances are divided into 3 classes C, R and RC that differ for the geographical distribution of the customer locations: a clustered distribution (C), random distribution (R) and a mix of both (RC). Each class is divided into two subclasses, the first one (C1, R1, RC1) has a short scheduling horizon, while the second one (C2, R2, RC2) has a long scheduling horizon. We create a set of 36 small instances randomly choosing 5, 10 and 15 customers and 3 OD destinations. To obtain the problem tests for the VRPOD, given a VRPTW instance with the customers locations identified by the coordinates (x_i,y_i), we randomly generate 3 destinations for the ODs, in the square with lower left hand corner ($\min_i x_i$, $\min_i y_i$) and upper right hand corner ($\max_i x_i$, $\max_i y_i$), (see Archetti et. al [1]). After we randomly generate a reasonably time window.

4.3.2 Comparative analysis

We now present a comparative analysis of the results, divided into two phases. At first we take into account the results of the VRPODTW and those obtained by solving the VRPODTWmd, whereupon, we introduce the results obtained solving the VRPODTWsd. We use the settings reported in the tables 4.1 and 4.3 for the first part of experiments and for the second one, respectively. Table 4.2 presents the comparison results for each VRPODTW instance against VRPODTWmd. Each table has 4 columns, for each version of the model. The first one shows the name of the instance, the second one the cost of the solution, in the third one #CD is the number of the classical drivers, while #OD, in the fourth one, the number of the ODs. In the last column the "GAP" on the cost is calculated as follows: GAP=

$$\frac{\text{Obj VRPOD - Obj VRPODTWmd}}{\text{Obj VRPOD}}.$$

Table 4.1: Parameters setting

# customers	$\|C\|$	P	Q	$\|K\|$	Q_1	Q_2	Q_3	ρ
	parameters							
5	5	1	100.0	3	30.0	30.0	40.0	1.1
10	10	2	100.0	3	30.0	30.0	40.0	1.1
15	15	3	100.0	3	40.0	40.0	60.0	1.1

Table 4.2: Results for the VRPODTW and VRPODTWmd

(a) Results for instances with 5 customers

	VRPODTW			**VRPODTWmd**			
Test	Cost	# CD	# OD	Cost	# CD	# OD	**GAP**
C101C5	139.7	1	2	124.9	1	2	11%
C103C5	110.3	1	3	106.6	1	2	3%
C206C5	159.6	1	1	138.4	1	2	13%
C208C5	113.3	1	2	91.1	0	3	20%
R104C5	83	1	2	83	1	2	0%
R105C5	91.5	1	3	77.5	1	3	15%
R202C5	125.8	1	2	125.8	1	2	0%
R203C5	125.3	1	3	93.4	1	3	25%
RC105C5	134	1	2	126.6	1	2	6%
RC108C5	210.5	1	2	164	1	2	22%
RC204C5	107	1	2	107	1	2	0%
RC208C5	122.6	1	3	76	1	3	38%
AVG	126.8833			109.525			**13%**

(b) Results for instances with 10 customers

	VRPODTW			**VRPODTWmd**			
Test	Cost	# CD	# OD	Cost	# CD	# OD	**GAP**
C101C10	261.7	2	3	250.6	2	3	4%
C104C10	223.7	2	3	196.9	1	3	12%
C202C10	181.4	2	3	172.3	1	3	5%
C205C10	184.7	2	2	172.9	1	3	6%
R102C10	169.7	2	3	134.0	1	3	21%
R103C10	150.0	2	0	122.7	1	2	18%
R201C10	154.6	2	3	146.3	2	3	5%
R203C10	149.7	1	3	132.9	1	3	11%
RC102C10	294.2	2	3	276.0	1	3	6%
RC108C10	276.2	2	3	235.2	2	2	15%
RC201C10	242.8	2	2	226.3	1	2	7%
RC205C10	270.1	2	2	260.3	2	2	4%
AVG	213.2333			193.8667			**10%**

(c) Results for instances with 15 customers

	VRPODTW			**VRPODTWmd**			
Test	Cost	# CD	# OD	Cost	# CD	# OD	**GAP**
C103C15	296.6	2	2	264.8	2	1	11%
C106C15	222.4	2	2	173.8	1	3	22%
C202C15	322.0	3	2	285.2	2	2	11%
C208C15	270.9	3	1	255.9	2	2	6%
R102C15	305.4	3	2	279.9	2	2	8%
R105C15	279.6	3	2	244.0	2	2	13%
R202C15	345.4	3	1	320.9	2	2	7%
R209C15	276.4	3	1	240.1	2	2	13%
RC103C15	336.3	3	2	248.4	2	2	26%
RC108C15	377.0	3	1	334.8	2	2	11%
RC202C15	362.5	3	1	281.5	2	2	22%
RC204C15	357.4	3	2	326.6	2	2	9%
AVG	312.7			271.3			**13%**

The computational results show the VRPODTWmd model outperforms VRPODTW in terms of solution quality. The "Gap" is equal to 13% for the instances with 5 and 15 node and 10% for the instances with 10 customers. The reduction of the cost is due to the use of the ODs, that allowed to make multiple deliveries. The solutions of the VRPODTW model use a number of classical drivers greater than the one used in the VRPODTWmd's solutions (35% more) and the cost of the solution is higher. While, solving VRPODTWmd allows to involve the 9.52 % of ODs more than VRPODTW. However, it is possible to highlight that, even if the same configuration of vehicles is obtained, the solutions obtained with VRPODTWms are more competitive than those obtained with VRPODTW. E.g. in the solutions of the instance "RC208C5" both the models consider one classical driver and two occasional drivers, however, the cost for the VRPODTW solution is about the 60% higher. Overall, the use of ODs allowed to make multiple deliveries optimizes the total costs and reduces the use of classical vehicles.

Table 4.3: parameters setting

	parameters							
# customers	$\lvert C\rvert$	P	Q	$\lvert K\rvert$	Q_1	Q_2	Q_3	ρ
5	5	2	60.0	3	5.0	10.0	15.0	1.1
10	10	2	75.0	3	10.0	10.0	15.0	1.1
15	15	3	75.0	3	10.0	15.0	15.0	1.1

The tables 4.4 present the comparison results for each VRPODTWmd instance against VRPODTWsd. The results of tables 4.4 clearly underline that the use of the split delivery strategy results competitive in terms of effectiveness. On average, a cost reduction of about 10% is observed. The possibility to split the deliveries increases the number of ODs used by the VRPODTWsd, with a consequent cost saving, i.e. the 62.50% more than those used by the VRPODTWmd. Also for these experiments, when the same configuration of vehicles is used in the solutions, often VRPODTWsd outperforms VRPODTWmd. E.g. for the instances "C208C5", "C101C10" and "C208C15" in which the same number of classical and occasional drivers are used in the solutions, the "GAP" is equal to 8%. In summary, the presented models outperform the literature model in terms of effectiveness. The com-

Table 4.4: Results for the VRPODTWmd and VRPODTWsd

(a) Results for instances with 5 customers

	VRPODTWmd			**VRPODTWsd**			
Test	Cost	# CD	# OD	Cost	# CD	# OD	**GAP**
C101C5	213.1	2	1	195.3	2	2	8%
C103C5	159.0	2	0	153	2	2	4%
C206C5	202.3	1	1	175.4	1	2	13%
C208C5	154.2	1	2	141.4	1	2	8%
R104C5	166.0	2	0	162.3	2	2	2%
R105C5	155.1	2	1	149.3	2	3	4%
R202C5	170.0	2	0	156.8	2	2	8%
R203C5	179.0	2	1	174.2	1	3	3%
RC105C5	228.0	2	0	177.1	2	2	22%
RC108C5	203.5	1	1	202.5	1	2	0%
RC204C5	173	2	1	118.3	1	3	32%
RC208C5	189.8	2	1	174.2	2	3	8%
AVG	182.8			164.9			**9%**

(b) Results for instances with 10 customers

	VRPODTWmd			**VRPODTWsd**			
Test	Cost	# CD	# OD	Cost	# CD	# OD	**GAP**
C101C10	353.3	2	3	324.7	2	3	8%
C104C10	315.8	2	3	276	2	3	13%
C202C10	295.9	2	3	227.8	2	3	23%
C205C10	206.2	2	1	188	2	2	9%
R102C10	208.9	2	1	198.6	2	3	5%
R103C10	185.7	2	2	167	2	3	10%
R201C10	263.8	2	2	209.6	2	3	21%
R203C10	204.7	2	1	172.3	2	3	16%
RC108C10	434.6	2	3	393.9	2	3	9%
RC102C10	396.6	2	2	391	2	3	1%
RC201C10	333.4	2	2	327.2	2	3	2%
RC205C10	340.8	2	2	340.8	2	2	0%
AVG	294.9			268.1			**10%**

(c) Results for instances with 15 customers

	VRPODTWmd			**VRPODTWsd**			
Test	Cost	# CD	# OD	Cost	# CD	# OD	**GAP**
C103C15	318.4	3	1	318.4	3	1	0%
C106C15	253.3	3	2	247.8	3	2	2%
C202C15	413.6	3	3	404.8	3	2	2%
C208C15	154.2	1	2	141.4	1	2	8%
R102C15	360.8	3	2	315.6	3	3	13%
R105C15	309.6	3	2	299.6	3	3	3%
R202C15	392.4	3	2	379.2	3	2	3%
R209C15	345.5	3	2	328.6	3	2	5%
RC103C15	360.9	3	2	356.9	3	2	1%
RC108C15	488.2	3	3	465.5	3	3	5%
RC202C15	479.9	3	3	444.5	3	3	7%
RC204C15	346.0	3	0	345.7	3	2	0%
AVG	351.9			337.3			**4%**

putational experiments highlight the benefits reached when the ODs are used to make deliveries, which become more interesting when the split delivery policy is considered.

4.3.3 Conclusions

We have proposed two innovative variants for the VRP. We take into account the possibility that a company may use the service provided by some ODs. The ODs are available to make some deliveries for a small compensation. The main goal has been to investigate the achievable potential benefits by introducing the crowd-sourcing in the VRP. The results of our computational experiments are very encouraging. We demonstrated that the use of the ODs may improve the routing plan, generating an interesting cost saving. The possibility to make multiple deliveries and the split delivery policy allows to exploit the whole capacity of the ODs. This work can be viewed as a base for several future works. There are more aspects that can be taken into account. For example the “green” aspect of this strategy. In fact, the use of the ODs reduces the pollutant emissions and the traffic congestion. The ODs perform travels that ordinarily already take place, thus, there is a reduction of routed vehicles and distance travelled. There is also the possibility to deliver the goods by bicycles or public transport. In conclusion, crowd-shipping allows the company to outsource the “last mile” deliveries to ordinary citizens and this may be an opportunity but also a risk. The company may provide a convenient and efficient delivery service. However, misuse the crowd-shipping implies giving more responsibility to the ODs, and it is intrinsically a risk.

Chapter 5

A variable neighborhood search for the vehicle routing problem with occasional drivers and time windows

Abstract

This paper presents a Variable Neighborhood Search (VNS) algorithm for a vehicle routing problem (VRP) variant with a crowd-sourced delivery policy. We consider a heterogeneous fleet composed of conventional capacitated vehicles and occasional drivers, i.e. ordinary people who accept to deviate from their route to deliver items to other people, for a small compensation. The objective is to minimize total costs, that is conventional vehicles costs plus occasional drivers compensation. To assess the performance of our heuristic, we compare the results obtained by using the proposed procedure with the optimal solution costs obtained by solving the model with CPLEX. The VNS is highly effective and is able to solve large-size instances within short computational times.

Keywords: Vehicle Routing, Crowd-shipping, Variable Neighbor-

hood Search

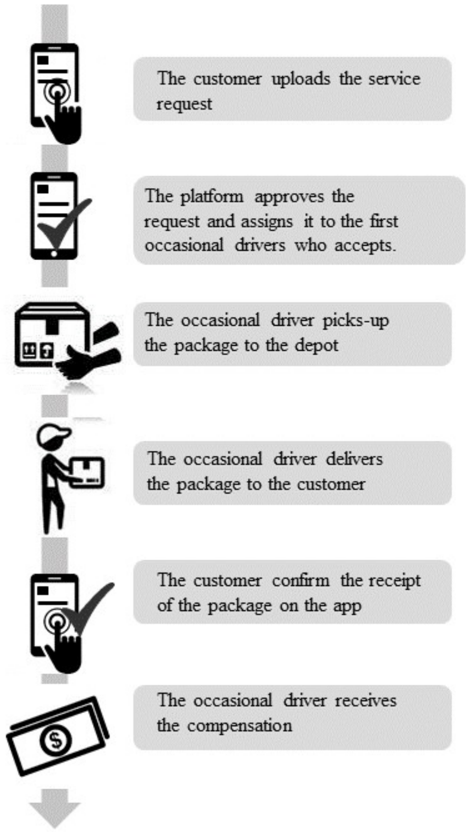
5.1 Introduction

The rapid and substantial growth of the on-line retailers and the continuous search for new ways of speeding up deliveries and of overcoming the traditional problems of last-mile and same-day deliveries have generated some interest in crowd-shipping. While crowd-sourcing is strictly related with the concept of "sharing economy", and allows activities that usually are performed by a company to be outsourced to a large pool of individuals, crowd-shipping, i.e. crowd-sourcing delivery, is a new opportunity to pursue transportation efficiency by exploiting underused assets. The reader is referred to [9] for a review on crowd-logistic, and to [47] for a complete survey on global trends in transportation and city logistics. The idea of crowd-shipping is as follows: an individual (whom we call occasional driver) who is travelling on its route, accepts to deviate from it to deliver items to other individuals, for a small compensation. Thus, in this context the exploited resources are commonly cars which are often underused assets. The advantages of crowd-shipping are numerous and are not only related to economic issues:

- Costs saving: generally the compensation for the occasional drivers is less than the standard drivers' salary.
- No infrastructures: crowd-shipping does not require any special infrastructure.
- Flexibility: while the traditional deliveries are fixed and planned in advance, for crowd-shipping fast delivery is the key factor.
- Less environmental impacts: sharing vehicles can lead to a reduction in polluting emissions, energy consumption, noise and traffic congestion.

The crowd-source of delivery requires the use of a crowd-shipping platform which connects occasional drivers with customers. In particular, when a customer buys some goods and requires a fast service delivery, she has to submit the on-line service request via a phone/computer application. If a customer applies and accepts crowd-shipping delivery, the platform then sends the required information needed to make the delivery. Once the delivery is done, the occasional driver receives the compensation (Figure 5.1 resumes the crowd-shipping process).

Figure 5.1: Crowd-shipping process



Several large on-line retailers have started to implement and use platforms for the crowd-shipping such as Walmart, DHL and Amazon. In 2013, Walmart announced its plan to outsource some of its deliveries by asking its in-store customers to deliver one or more orders placed by on-line customers, for a small compensation (see Barr & Wohl [3]). In other words, if a customer is in a Walmart store for her shopping and she has some free time, she may decide to deliver orders for other shoppers on her way home or to her destination. DHL experimented with crowd-shipping in Stockholm during September to December 2013 with a pilot last-mile service named MyWays (see Landa [32]). Using a specific smartphone app, the service connected individuals, mostly students, who asked for flexible deliveries with those offering to transport packages along their normal routes. The experiment was positive and many customers receiver or delivered packages by using MyWays. In June 2015, Amazon launched Amazon Flex (see Besinger [7]), its new crowd-shipping service that is still used in more than 30 cities in the world. To become an occasional driver for Amazon it is necessary to have an Amazon account, some prerequisites, and to install the application. In particular, Amazon Flex offers several delivery opportunities based on the time within which a package has to be delivered: three or more hours, one or two hours, less than one hour. An occasional driver may also choose among two ways to pick up delivery blocks: by choosing dates on a calendar, in which case the platform sends delivery offers for that dates, or by checking for available blocks on Amazon Flex home screen.

Crowd-shipping is a quite recent topic, and the relevant literature is limited. Arslan et al. [2] analyzed the potential benefits of crowd-shipping. They presented a literature review of the recent contributions dealing with crowd-sourcing. They considered a peer-to-peer platform, taking into account the possibility of using traditional vehicles and ad hoc vehicles. They presented a rolling horizon framework and an exact algorithm to solve the route planning problem. Archetti et al. [1] introduced the vehicle routing problem with occasional drivers (VRPOD). In this work, the authors suppose that the company can make deliveries not only by using its own fleet of capacitated vehicles, but also by resorting to occasional drivers (ODs). These authors proposed an integer programming formulation for the

VRPOD, and then developed a multi-start heuristic, which combines tabu search and variable neighborhood search (VNS). Starting from the VRPOD model of Archetti et al. [1], Macrina et al. [40] introduced three innovative aspects to the problem. These authors first considered time windows constraints for customers and ODs. Second, they allowed multiple deliveries for ODs. Third, they modelled the split & delivery policy for ODs. They tested and compared the proposed models with that of Archetti et al. [1]. Their computational results showed the benefits of allowing multiple deliveries and of using the split & delivery policy.

Based on this work, here we implement a VNS heuristic for the VRPOD with time windows and multiple deliveries. We carry out several computational tests on different size networks and we assess the effectiveness of the proposed algorithm. The remainder of the paper is organized as follows. In Section 5.2 we model the VRPOD with time windows and multiple deliveries (VRPODTWmd) proposed in Macrina et al. [40]. In Section 5.3 we describe our VNS heuristic for the VRPODTWmd. In Section 5.4 we describe the computational experiments and we present the results. Section 5.5 summarizes the conclusions.

5.2 The vehicle routing problem with occasional drivers and time windows

Let C be the set of customers, let s be the origin node and t the destination node for the classical vehicles, i.e. those belonging to the company. Let K be the set of available ODs and V the set of v_k destinations associated with the ODs. We define the node set as $N = C \cup \{s, t\} \cup V$. We model the problem on a complete directed graph $G = (N, A)$, where A is the set of arcs. Each arc $(i, j) \in A$ has a cost c_{ij} and a travel time t_{ij} associated with it. Note that both c_{ij} and t_{ij} satisfy the triangle inequality. Each node $i \in C \cup V$ has a time window $[e_i, l_i]$, and each customer $i \in C$ has a demand d_i. Q is the capacity of the classical vehicles, P is the number of available classical vehicles and Q_k is the capacity of OD $k \in K$. Let x_{ij} be a binary variable equal to 1 if and only if a classical vehicle traverses arc (i, j). For each node $i \in N$ let y_i be the available capacity of a classical vehicle after visiting customer i, and let s_i be the arrival of a classical vehicle at customer i. Moreover r_{ij}^k is a binary variable equal to 1 if and only if OD k traverses arc (i, j). Let f_i^k indicate the arrival time of OD k at the customer i, and let w_i^k be the available capacity of the vehicle associated with OD k after visiting customer i. Table 5.1 summarizes the notation. We formulate the VRPODTWmd as follows:

$$\text{Minimize} \quad \sum_{i \in C \cup \{s\}} \sum_{j \in C \cup \{t\}} c_{ij} x_{ij} + \sum_{k \in K} \sum_{i \in C \cup \{s\}} \sum_{j \in C} \rho c_{ij} r_{ij}^k - \sum_{k \in K} \sum_{j \in C} c_{sv_k} r_{sj}^k \tag{5.1}$$

$$\text{subject to} \quad \sum_{j \in C \cup \{t\}} x_{ij} - \sum_{j \in C \cup \{s\}} x_{ji} = 0 \qquad i \in C \tag{5.2}$$

$$\sum_{j \in C} x_{sj} - \sum_{j \in C} x_{jt} = 0 \tag{5.3}$$

$$y_j \geq y_i + d_j x_{ij} - Q(1 - x_{ij}) \qquad j \in C \cup \{t\}, i \in C \cup \{s\} \tag{5.4}$$

$$y_s \leq Q \tag{5.5}$$

$$s_j \geq s_i + t_{ij} x_{ij} - \alpha(1 - x_{ij}) \qquad i \in C, j \in C \tag{5.6}$$

$$e_i \leq s_i \leq l_i \qquad i \in C \tag{5.7}$$

$$\sum_{j \in C} x_{sj} \leq P \tag{5.8}$$

$$\sum_{j \in C \cup \{v_k\}} r_{ij}^k - \sum_{h \in C \cup \{s\}} r_{hi}^k = 0 \qquad i \in C, k \in K \tag{5.9}$$

$$\sum_{j \in C \cup \{v_k\}} r_{sj}^k - \sum_{j \in C \cup \{s\}} r_{jv_k}^k = 0 \qquad k \in K \tag{5.10}$$

$$\sum_{k \in K} \sum_{j \in C \cup \{v_k\}} r_{sj}^k \leq |K| \tag{5.11}$$

$$\sum_{j \in C} r_{sj}^k \leq 1 \qquad k \in K \tag{5.12}$$

$$w_j^k \geq w_i^k + d_i r_{ij}^k - Q_k(1 - r_{ij}^k) \qquad j \in C \cup \{v_k\}, i \in C \cup \{s\}, k \in K \tag{5.13}$$

$$w_s^k \leq Q_k \qquad k \in K \tag{5.14}$$

$$f_i^k + t_{ij} r_{ij}^k - \alpha(1 - r_{ij}^k) \leq f_j^k \qquad i \in C, j \in C, k \in K \qquad (5.15)$$

$$f_i^k \geq e_{v_k} + t_{si} \qquad i \in C, k \in K \qquad (5.16)$$

$$f_{v_k}^k \leq l_{v_k} \qquad k \in K \qquad (5.17)$$

$$f_i^k + t_{iv_k} r_{iv_k}^k - \alpha(1 - r_{iv_k}^k) \leq f_{v_k}^k \qquad i \in C, k \in K \qquad (5.18)$$

$$e_i \leq f_i^k \leq l_i \qquad i \in C \qquad (5.19)$$

$$\sum_{j \in C \cup \{t\}} x_{ij} + \sum_{h \in C \cup \{v_k\}} \sum_{k \in K} r_{ih}^k = 1 \qquad i \in C \qquad (5.20)$$

$$x_{ij} \in \{0, 1\} \qquad (i, j) \in A \qquad (5.21)$$

$$r_{ij}^k \in \{0, 1\} \qquad (i, j) \in A, k \in K \qquad (5.22)$$

$$0 \leq y_i \leq Q \qquad i \in C \cup \{s, t\} \qquad (5.23)$$

$$0 \leq w_i^k \leq Q_k \qquad i \in C \cup \{s, v_k\}, k \in K \qquad (5.24)$$

$$f_i^k \geq 0 \qquad i \in C \cup \{s, v_k\}, k \in K \qquad (5.25)$$

Table 5.1: Parameters and decision variables of the VRPODmd model

s	origin node
t	destination node for classical vehicles
C	set of customers
K	set of available occasional drivers
V	set of v_k destinations for the occasional drivers
A	set of arcs
c_{ij}	travel cost from node i to node j
t_{ij}	travel time from node i to node j
$[e_i, l_i]$	time windows of node i
d_i	demand of customer i
Q	capacity of classical vehicles
P	number of classical vehicles
Q_k	capacity of occasional driver k
x_{ij}	binary decision variable indicating if arc $(i, j) \in A$ is traversed by a classical vehicle
y_i	decision variable specifying the available capacity of the classical vehicle after visiting customer i
s_i	decision variable specifying the arrival time of the classical vehicle to customer i
r_{ij}^k	binary decision variable indicating if arc $(i, j) \in A$ is traversed by the occasional driver k
f_i^k	decision variable specifying the arrival time of the occasional driver k at customer i
w_i^k	decision variable specifying the available capacity of the occasional driver k after visiting customer i

The objective function (5.1) minimizes the total costs. The first term is the transportation cost associated with classical vehicles. The second term is the compensation cost of OD k for the delivery service, with $\rho \geq 0$, the third one is the cost of OD k when it does not perform the delivery service. Constraints (5.2) to (5.8) are linked to the classical vehicles. In particular, constraints (5.2) to (5.3) are the flow conservation constraints. Constraints (5.4) guarantee the fulfilment of demand at customer vertices. Constraints (5.5) restrict the initial cargo load level to the maximum capacity of a vehicle. Constraints (5.6) allow the determination of the arrival time at node j, while constraints (5.7) guarantee an arrival within the time window at each node. Constraint (5.8) imposes a maximum number of available vehicles. Constraints (5.9) to (5.19) are linked to the ODs. Constraints (5.9) to (5.10) are the flow conservation constraints. Constraints (5.11) to (5.12) impose a limit on the number of available ODs and on the number of departures from the depot. Constraints (5.13) to (5.14) are the capacity constraints. Constraints (5.15) compute the arrival time at node j. Constraints (5.16) to (5.17) are the time windows constraints and also define the time at which the ODs are available to make deliveries, while constraints (5.18) compute the arrival time at the destination node v_k. Constraints (5.19) mean that each customer is served within its time window. Constraints (5.20) guarantee that each customer is visited at most once, either by a classical vehicle or by an OD.

5.3 Variable Neighborhood Search for the VRPODTWmd

This section details our variable neighborhood search (VNS) for the VRPODTWmd. Algorithm 5 presents the VNS scheme. First, we generate an initial solution δ, then we apply a *Shaking* phase to perturb δ order to explore the neighborhoods and improve the solution.

Initial solution The initialization procedure is an insertion heuristic. The starting tour is composed of the origin and destination nodes. The heuristic inserts a new node in the tour in the best feasible position, i.e. where it causes the least increase in the tour cost. We adapt the insertion heuristic to the VRPODTWmd by taking into account the heterogeneity of the fleet. Thus, at first we try to serve the farthest

Algorithm 5 Variable neighborhood search

Input set of neighbourhood N^h, for $h = 0, ..., h_{max}$
Initialization Initial solution δ
while $h \leq h_{\max}$ and $k \leq k_{\max}$ **do**
  $\delta' \leftarrow$ **Shaking**(δ)
  $\delta'' \leftarrow$ **VND** (δ')
  if $f(\delta'') < f(\delta)$ **then**
    $\delta \leftarrow \delta''$;
    $h \leftarrow 0$
  else
    $h \leftarrow h + 1$
  end if
  $k \leftarrow k + 1$
end while
return δ

customers with ODs and then the unserved customers with traditional drivers. Since the initial solution may be infeasible, we apply a repair phase in which the local search moves defined for the VND are applied until a feasible solution is generated.

Local search operators In order to generate the neighborhoods we use seven different Local Search (LS) moves:

1. 2-Opt: This operator removes two arcs (i, j) and (u, v) in the same route r (classical or OD) or in two distinct routes r and r' (classical or OD), and reconnects the path(s) created using arcs (i, u) and (j, v).
2. Move Node: This operator removes one node i from a route r and inserts it in another r' in the best feasible position. We implemented four variants: classical to classical, classical to OD, OD to OD, OD to classical.
3. Swap Inter-Route: This operator removes one node i from a route r and one node j from another route r', $r \neq r'$, and inserts i into r' and j into r in the best feasible positions. We implemented

four variants: classical to classical, classical to OD, OD to OD, OD to classical.

4. Swap Intra-Route: Given a route r changes the position of two nodes i and j with the respect to the constraints. We implemented two variants: classical and OD.

5. New Route: This operator initializes a new route r'. It removes one node i from a route r and inserts i in r'. We implemented two variants: classical and OD.

6. Remove and Insert: This is a particular neighbourhood, which applies all the four variants of "Move Node" moves.

Shaking The main goal of the shaking phase is to perturb the current solution. Thus, we randomly select and apply two different LS operators and we allow the current solution to worsen. To improve the shaking phase, we introduce a semi-random choice. In particular, after the first iteration, we assign a score to each LS move. At the end of each VNS iteration, if there is an improvement on solution cost, we increment the scores of the shaking moves, otherwise the scores are reduced. The LS operators are randomly chosen among those with the best scores.

Variable neighborhood descent Generally, in the VNS scheme the shaking process is followed by an LS phase, in which the h_{max} different LS operators are applied with the purpose of improving the current solution. We propose a VND described in Algorithm 6.

5.4 Computational study

This section presents the results of our computational experiments. All computations were performed on an Intel 2.60 GHz processor and

Algorithm 6 Variable neighborhood descent

Input the set of neighborhood N^h, for $h = 0, ..., h_{max}$
Initialization initial solution δ
$improved \leftarrow 0$, $k \leftarrow 0$
while $improved = false$ and $k \leq k_{\max}$ **do**
  $h \leftarrow 0$
  while $h \leq h_{\max}$ **do**
    $\delta' \leftarrow N^h(\delta)$
    if $f(\delta') < f(\delta)$ **then**
      $\delta \leftarrow \delta'$
      $improved \leftarrow 1$
    else
      $h \leftarrow h + 1$
    end if
  end while
  $k \leftarrow k + 1$
end while
return δ

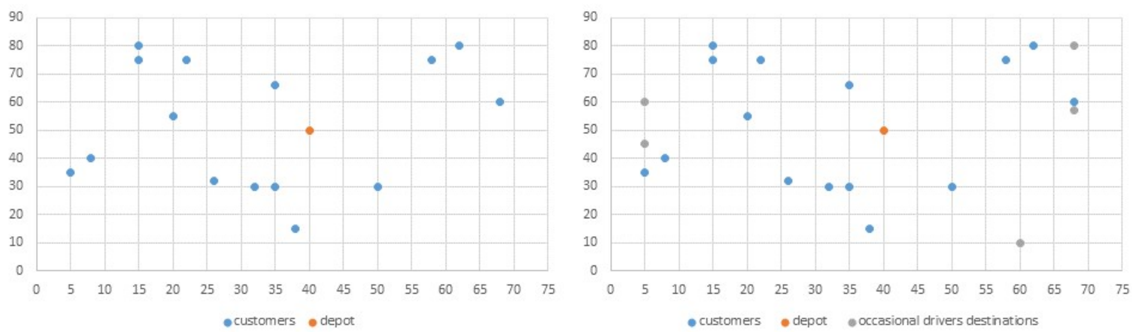
16 GB of RAM. The VNS was implemented in Java. The rest of this section is organized as follows. We first describe the strategy used to generate the VRPODTWmd instances (Section 5.4.1), we then present the numerical experiments. We divided the computational study into two parts. The first part aims to assess the performance of the VNS heuristic (Section 5.4.2). With this purpose, we attempt to solve the problem with CPLEX. Since CPLEX is unable to solve instances with more than 15 customers and 10 ODs, we conduct a second part of tests, in which medium and large size instances are solved by VNS (Section 5.4.3).

5.4.1 Generation of Instances

We conducted several numerical experiments on different size instances. At first we generated the VRPODTWmd instances based on the classical Solomon VRPTW instances [52]. We created a set of 36 small instances randomly choosing five, 10 and 15 customers and three and five

OD destinations. We also created 30 medium instances with 25 and 50 customers, and 15 large instances with 100 customers. To obtain the test instances for the VRPODTWmd, given a VRPTW instance with the customers locations identified by the coordinates (x_i, y_i), we randomly generated the destinations for the ODs, in the square with lower left hand corner $(\min_i\{x_i\}, \min_i\{y_i\})$ and upper righthand corner $(\max_i\{x_i\}, \max_i\{y_i\})$, (see Archetti et. al [1]). We then randomly generated a reasonably time window. Figure 5.2 provides an example of an instance generation.

Figure 5.2: Generation of instance C103C15



5.4.2 Evaluation of the VNS perfomance

In order to assess the performance of our VNS, we first solved the problem to optimality using CPLEX and we then compare the results with those obtained by VNS. We used the parameters setting reported in Table 5.2 and we set $k_{\max} = 50$ and *improve* $= 15$.

Table 5.2: Parameters setting for small instances

#customers	|C|	P	|K|	Q	Q_k
5	5	3	3	80	[10;25]
10	10	3	3	80	[10;30]
15	15	3	5	80	[15;35]

Tables 5.3 to 5.5 compare the solutions obtained for small size instances. For each table and each algorithm (VNS and resolution with CPLEX), the first column displays the number of available ODs and the number of customers, the second one the name of the instance, the third one is the total cost, the fourth and the fifth columns show the number of classical and OD vehicles used respectively. The sixth one displays the optimality gap on cost, calculated as $\mathrm{g}_{\mathrm{cost}} = (\mathrm{Objective}_{\mathrm{VNS}} - \mathrm{Objective}_{\mathrm{CPLEX}})/\mathrm{Objective}_{\mathrm{CPLEX}}$. The VNS looks very effective, indeed, it finds an optimal solution for the majority of instances and the optimality gap is on average less than 1.5%. In particular, Table 5.3 highlights the efficiency of the VNS for instances with five customers. The proposed algorithm finds the optimal solutions for all instances. As shown in Table 5.4, for instances with 10 customers, the average on gap is equal to 1.1 %, the VNS finds optimal solutions for about the 50% of the instances and, for the other instances, it produces solutions with small optimality gaps. Table 5.5 exhibits the same trend for instances with 15 customers. Indeed, on average, the optimality gap is equal to 2.1%. It is worth observing that our approach is also more efficient than CPLEX. As shown in Table 5.6, which reports the time [ms] spent by the VNS and CPLEX to solve instances with five, 10 and 15 customers, the VNS is able to find efficient solutions within short times. The VNS clearly outperforms CPLEX.

5.4.3 Numerical results on the medium-size and large-size instances

We now present the computational results on instances with 25, 50 and 100 customers. We used the parameters setting presented in Table 5.7 and we set $k_{\max} = 200$ and $improve = 150$.

Tables 5.8 to 5.10 summarise the results obtained for this set of instances. For each table, the first column displays the number of available ODs and the number of customers, the second one the name

Table 5.3: Results for five customers and three occasional drivers instances

		VNS			OPT			
	Name	Total cost	#classical	#occasional	Total cost	#classical	#occasional	g_{cost}
$\|K\| = 3$	C101C5	145.4	1	2	145.4	1	2	0.000%
$\|C\| = 5$	C103C5	111.6	1	2	111.6	1	2	0.000%
	C206C5	159.6	1	1	159.6	1	1	0.000%
	C208C5	131.9	1	1	131.9	1	1	0.000%
	R104C5	107.2	1	2	107.2	1	2	0.000%
	R105C5	143.1	2	1	143.1	2	1	0.000%
	R202C5	129.5	2	1	129.5	2	1	0.000%
	R203C5	170.1	1	1	170.1	1	1	0.000%
	RC105C5	143.1	1	2	143.1	1	2	0.000%
	RC108C5	164.0	1	2	164.0	1	2	0.000%
	RC204C5	107.0	1	2	107.0	1	2	0.000%
	RC208C5	124.9	1	2	124.9	1	2	0.000%
	Average	136.45	1.16	1.58	136.45	1.167	1.58	0.00%

Table 5.4: Results for 10 customers and three occasional drivers instances

		VNS			OPT			
	Name	Total cost	#classical	#occasional	Total cost	#classical	#occasional	g_{cost}
$\|K\| = 3$	C101C10	288.0	2	2	283.2	3	2	0.017%
$\|C\| = 10$	C104C10	249.3	2	2	242.4	2	3	0.028%
	C202C10	176.4	2	3	175.4	2	3	0.006%
	C205C10	184.7	2	2	184.7	1	2	0.000%
	R102C10	176.4	2	3	166.4	1	2	0.060%
	R103C10	154.0	2	1	154.0	2	1	0.000%
	R201C10	185.2	2	2	185.2	3	2	0.000%
	R203C10	132.9	1	3	132.9	1	3	0.000%
	RC102C10	337.8	3	2	331.8	2	2	0.018%
	RC108C10	330.1	2	2	330.1	2	2	0.000%
	RC201C10	231.1	2	2	231.1	2	2	0.000%
	RC205C10	260.3	2	2	260.3	2	2	0.000%
	Average	225.51	2.00	2.17	223.12	1.91	2.16	0.01%

Table 5.5: Results for 15 customers and five occasional drivers instances

		VNS			OPT			
	Name	Total cost	#classical	#occcasional	Total cost	#classical	#occasional	g_{cost}
$\|K\| = 5$	C103C15	207.4	1	5	206.5	1	5	0.004%
$\|C\| = 15$	C106C15	173.9	2	5	169.4	2	5	0.027%
	C208C15	304.7	3	3	304.7	3	3	0.000%
	C202C15	338.3	2	4	332.9	2	5	0.016%
	R102C15	300.0	3	4	297.8	3	3	0.007%
	R105C15	223.2	2	5	215.8	1	5	0.034%
	R202C15	339.7	3	2	324.4	3	4	0.047%
	R209C15	249.0	2	5	239.4	3	3	0.040%
	RC103C15	343.7	3	4	341.6	3	3	0.006%
	RC108C15	252.6	2	5	248.6	2	5	0.016%
	RC202C15	357.6	3	4	356.3	3	5	0.004%
	RC204C15	356.5	2	3	341.1	2	3	0.045%
	Average	287.22	2.333	4.09	281.54	2.33	4.08	0.02%

Table 5.6: Computational times required to solve instances with five, 10 and 15 customers

name	VNS	CPLEX		VNS	CPLEX		VNS	CPLEX
C101C5	60.00	55.00	C101C10	104.00	81.00	C103C15	209.00	3831.00
C103C5	25.00	21.00	C104C10	76.00	279.00	C106C15	128.00	502.00
C206C5	17.00	18.00	C202C10	29.00	47.00	C208C15	44.00	2851.00
C208C5	21.00	24.00	C205C10	3.00	69.00	C202C15	53.00	385.00
R104C5	53.00	21.00	R102C10	14.00	85.00	R102C15	8.00	193.00
R105C5	3.00	14.00	R103C10	4.00	285.00	R105C15	9.00	169.00
R202C5	29.00	44.00	R201C10	41.00	53.00	R202C15	9.00	5776.00
R203C5	25.00	27.00	R203C10	4.00	83.00	R209C15	113.00	3788.00
RC105C5	19.00	13.00	RC102C10	48.00	94.00	RC103C15	8.00	2450.00
RC108C5	10.00	17.00	RC108C10	41.00	195.00	RC108C15	33.00	13925.00
RC204C5	7.00	38.00	RC201C10	27.00	32.00	RC202C15	8.00	458.00
RC208C5	16.00	20.00	RC205C10	9.00	32.00	RC204C15	7.00	871428.00
Average	23.75	26.00	Average	33.33	111.25	Average	52.42	75479.67
Speed-up	1.09		Speed-up	3.34		Speed-up	1439.99	

Table 5.7: parameters setting for medium- and large-size instances

#customers	|C|	P	|K|	Q	Q_k
25	25	5	10	100	[20;40]
50	50	8	15	200	[20;40]
100	100	10	30	400	[20;40]

of the instance, the third one the computational time [ms], the fourth one the total cost, the fifth and sixth columns show the number of classical and OD routed vehicles respectively. We also report, in the last line, the average for all the statistics. Overall, the VNS finds solutions within short computational times. Indeed, the VNS solve instances with 25 nodes in about one second, with 50 nodes in about nine seconds, and with 100 nodes in about 25 seconds (about three time less than the time spent by CPLEX to solve instances with 15 customers). Table 5.8 shows that the solutions generated use, on average, about six ODs out of 10 available and about three classical vehicles out of five. Table 5.9 exhibits the same trend. Indeed, the solutions use on average eight out of 15 ODs. The use of ODs becomes more interesting on instances with 100 nodes. Looking at Table 5.10, it is clear that the majority of the generated solutions use almost all the available ODs.

Table 5.8: Results for instances with 25 customers

	Name	Time	Total cost	#classical	#occasional
|K| = 10	C101C25	2495	344.9	4	4
|C| = 25	C102C25	1855	372	5	2
	C103C25	467.0	365.4	3	8
	C104C25	1109.0	463.2	2	9
	C105C25	2506.0	417.4	3	7
	R101C25	725.0	327.4	3	7
	R102C25	658.0	309.5	1	10
	R103C25	220.0	337.5	2	8
	R104C25	1102.0	321.8	2	8
	R105C25	78.0	315.2	2	9
	RC101C25	2345.0	594	5	3
	RC102C25	504.0	570.3	5	2
	RC103C25	419.0	478.4	4	7
	RC104C25	1040.0	486.8	4	6
	RC105C25	2309.0	565.7	5	3
	Average	1188.8	417.9	3.3	6.2

Table 5.9: Results for instances with 50 customers

	Name	Time	Total cost	#classical	#occasional
|K| = 15	C102C50	681.0	527.6	3	13
|C| = 50	C103C50	1042.0	480.1	3	9
	C104C50	13757.0	468.4	4	5
	C105C50	2497.0	501.4	3	10
	R101C50	3018.0	780.6	5	14
	R102C50	7552.0	783.1	5	8
	R103C50	2874.0	608.8	4	6
	R104C50	69357.0	555.9	3	8
	R105C50	3219.0	761.8	5	6
	RC101C50	827.0	529	3	13
	RC102C50	1940.0	714	4	7
	RC103C50	3354.0	677.3	5	3
	RC104C50	3769.0	693.3	4	7
	RC105C50	11830.0	824.3	6	3
	Average	8979.8	636.1	4.1	8.0

5.5 Conclusions

We have proposed a variable neighborhood search heuristic for the vehicle routing problem with occasional drivers and time windows. In order to assess the performance of our proposed heuristic, we have solved the model with CPLEX for small instances. We have then conducted a comparative analysis. The results have shown that our heuristic is highly performing in terms of effectiveness and efficiency.

Table 5.10: Results for instances with 100 customers

	Name	Time	Total cost	#classical	#occasional
$\lvert K\rvert = 30$	C101	753.0	1369.6	6	28
$\lvert C\rvert = 50$	C102	5899.0	1472.3	6	26
	C103	1054.0	1245.8	6	26
	C104	1752.0	1250.8	7	25
	C105	1438.0	1468.4	7	27
	R101	3.0	1408.4	10	27
	R102	61755.0	1346.7	8	8
	R103	84458.0	1150.6	7	7
	R104	206499.0	1002.4	5	7
	R105	4269	1396.3	7	29
	RC101	9004.0	1457.1	7	25
	RC102	14717.0	1584.9	8	16
	RC103	25933.0	1214.4	7	7
	RC104	7553.0	1212.8	6	22
	RC105	247.0	1283.1	9	30
	Average	28355.6	1324.2	7.1	20.7

Overall, the heuristic is less time consuming than CPLEX. We have also shown that it can solve large-size instances within short computational times.

Part III

Conclusions

Chapter 6

Conclusions and future works

In this dissertation we have studied two innovative logistics area: *Green logistics* and *Crowd-shipping*. The first part of the work has been devoted to the green vehicle routing problem (G-VRP), a new vehicle routing problem VRP variant which takes into account sustainability goals. In Chapter 1 we have provided an overview of the main contributions in G-VRPs in the 2011-2018 period. Then, we have studied some innovative G-VRP variants. In particular, we have assumed the availability of a mixed vehicle fleet, composed of electric and conventional (internal combustion engine) vehicles and we have introduced and solved two variants:

The green mixed fleet vehicle routing problem with partial battery recharging and time windows (Chapter 2). We have introduced a mixed fleet G-VRP variant in which we explicitly have taken into account a limitation on the polluting emissions for the conventional vehicles. The energy consumption for the electric vehicles has been supposed to be proportional to the traveled distance, and partial recharges have been allowed. We have proposed an iterative local search heuristic to optimize the routing of the vehicles. We have evaluated the behaviour of the proposed algorithm on a large set of instances.

The energy-efficient green vehicle routing problem with mixed fleet, *partial battery recharging and time windows* (Chapter 3). We have modified the model presented in Chapter 2 by considering some realistic issues. In particular we have modeled a realistic energy function to evaluate the energy consumption, in which we have taken into account vehicle speed, gradient and cargo load. We have proposed a matheuristic to solve the problem. We have carried out a preliminary computational study.

The second part of the dissertation has been devoted to the VRP with occasional drivers (ODs) in which the crowd-shipping has been considered. We have supposed that a transportation company can make deliveries by using its own fleet composed of capacitated vehicles and also some ODs (i.e. ordinary people decide to make either a single delivery or multiple deliveries, by making a deviation on their ordinary route, for a small compensation). In Chapter 4 we have modeled two variants of the problem: in the first one we consider the possibility for the ODs to make multiple deliveries, in the second one we introduced the split and delivery policy. We have tested the models and we have compared the proposed configurations with the literature one. The results have shown that the transportation company can achieve important advantages by employing the occasional drivers, which become more significant when multiple delivery and split and delivery policies have been jointly considered. In Chapter 5 we have proposed a variable neighborhood search for the vehicle routing problem with occasional drivers, time windows and multiple deliveries for the ODs. We have solved instances with up to 100 customers and 15 ODs in a short computation time.

6.1 Direction for future works

While the use of electric vehicles leads to the decrease of polluting emissions and noise, several realistic issues related to energy consumption model, battery, charging and infrastructures have to be considered. Several works are based on unrealistic assumptions. Contrary to the majority of contributions in electric-VRP, which assume energy consumption proportional to the traveled distance, actually it depends on several factors such as speed and load cargo. In addition, CSs charging functions are not linear, as supposed in several works. We have proposed a G-VRP variant in Chapter 3 which incorporates a realistic energy consumption model. Future research should be conducted to extend the results presented in Chapter 3, by consider considering the non-linearity of charging function.

In the last years the growing importance of shorter delivery lead times has led the companies to create innovative solutions to organize the last-mile and same-day delivery. In this context, the crowd-shipping is an innovative strategy to pursue the efficiency, and a new way to take into account the transportation planning sustainability. As we have shown, the use of ODs can lead to interesting advantages. However, more aspects have to be taken into account. We have presented a static model, in reality the ODs availability can vary over the time as well as customers orders. Thus, a possible direction for the future work on VRP with crowd-shipping is to consider the dynamic nature of the problem.

Bibliography

[1] C. Archetti, M. W. P. Savelsbergh, and M. G. Speranza. The vehicle routing problem with occasional drivers. *European Journal of Operational Research*, 254:471–480, 2016.

[2] A. M. Arslan, N. Agatz, L. Kroon, and R. Zuidwijk. Crowdsourced delivery: A dynamic pickup and delivery problem with ad-hoc drivers. $https://papers.ssrn.com/sol3/papers.cfm?abstract\_id = 2726731$, 2015.

[3] A. Barr and J. Wohl. Exclusive: Wal-Mart may get customers to deliver packages to online buyers. REUTERS - Business, 2013.

[4] M. Barth and K. Boriboonsomsin. Energy and emission impacts of a freeway-based dynamic eco-driving system. *Transportation Research Part D*, 14(6):400–410, 2009.

[5] M. Barth, T. Younglove, and G. Scora. Development of a heavy-duty diesel modal emissions and fuel consumption model. Technical Report UCB-ITS-PRR, California PATH program, Insitute of Transportation Studies, University of California at Berkeley, 1 2005.

[6] T. Bektaş and G. Laporte. The pollution-routing problem. *Transportation Research Part B*, 45:1232–1250, 2011.

[7] G. Bensinger. Amazon's next delivery drone: You. Wall Street Journal, 2015.

[8] M. Bruglieri, F. Pezzella, O. Pisacane, and S. Suraci. A variable neighborhood search branching for the electric vehicle routing problem with time windows. *Electronic Notes in Discrete Mathematics*, 47:221–228, 2015.

[9] H. Buldeo Rai, S. Verlinde, J. Merckx, and C. Macharis. Crowd logistics: an opportunity for more sustainable urban freight transport? *European Transport Research Review*, 9(39), 2017.

[10] G. Ćirović, D. Pamuçar, and Bozanić D. Green logistic vehicle routing problem: Routing light delivery vehicles in urban areas using a neuro-fuzzy model. *Expert Systems with Applications*, 41:4245–4258, 2014.

[11] R. G. Conrad and M. A. Figliozzi. The recharging vehicle routing problem. Doolen, T., Van Aken, E., 2011. Industrial Engineering Research Conference. Reno, Nevada.

[12] G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, 6(1):80–91, 1959.

[13] E. Demir, T. Bektaş, and G. Laporte. An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, 223:346–359, 2012.

[14] E. Demir, T. Bektaş, and G. Laporte. The bi-objective pollution-routing problem. *European Journal of Operational Research*, 232:464–478, 2014.

[15] G. Desaulniers, F. Errico, S. Irnich, and M. Schneider. Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, 64:1388–1405, 2016.

[16] N. Ding, R. Battay, and C. Kwon. Conflict-free electric vehicle routing problem with capacitated charging stations and partial recharge. https://www.chkwon.net/papers, 2015.

[17] S. Erdoğan and E. Miller-Hooks. A green vehicle routing problem. *Transportation Research Part E*, 48(1):100–114, 2012.

[18] A. Felipe, M. T. Ortuño, G. Righini, and G. Tirado. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transportation Research Part E*, 71:111–128, 2014.

[19] M. A. Figliozzi. The impacts of congestion on time-definitive urban freight distribution networks CO_2 emission levels: Results from a case study in Portland, Oregon. *Transportation Research Part C*, 19(5):766–778, 2011.

[20] A. Franceschetti, D. Honhon, T. Van Woensel, T. Bektaş, and G. Laporte. The time-dependent pollution-routing problem. *Transportation Research Part B*, 56:265–293, 2013.

[21] D. Goeke and M. Schneider. Routing a mixed fleet of electric and conventional vehicles. *European Journal of Operational Research*, 245:81–99, 2015.

[22] F. Gonçalves, S. R. Cardoso, S. Relvas, and A.P.F.D. Barbosa-Póvoa. Optimization of a distribution network using electric vehicles: A VRP problem. pages 18–20. 15th Congresso Nacional da Associação Portuguesa de Investigação Operacional, 2011.

[23] G. Hiermann, J. Puchinger, S. Ropke, and R. F. Hartl. The electric fleet size and mix vehicle routing problem with time windows and recharging stations. *European Journal of Operational Research*, 252:995–1018, 2016.

[24] O. Jabali, T. Van Woensel, and A.G. de Kok. Analysis of travel times and CO_2 emissions in time-dependent vehicle routing. *Production and Operations Management*, 21(6):1060–1074, 2012.

[25] I. Kara, B. Y. Kara, and M. K. Yetis. Energy minimizing vehicle routing problem. in: Dress a., xu y., zhu b. (eds). *Combinatorial*

optimization and applications, 4616:62–71, 2007. Berlin, Heidelberg: Springer.

[26] M. Keskin and B. Çatay. Partial recharge strategies for the electric vehicle routing problem with time windows. *Transportation Research Part C*, 65:111–127, 2016.

[27] Ç. Koç, T. Bektaş, O. Jabali, and G. Laporte. The fleet size and mix pollution-routing problem. *Transportation Research Part B*, 70:239–254, 2014.

[28] Ç. Koç and I. Karaoglan. The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing*, 39:154–164, 2016.

[29] R. Kramer, N. Maculan, A. Subramanian, and T. Vidal. A speed and departure time optimization algorithm for the pollution-routing problem. *European Journal of Operational Research*, 247:782–787, 2015.

[30] R. Kramer, A. Subramanian, T. Vidal, and L. A. F. Cabral. A matheuristic approach for the pollution-routing problem. *European Journal of Operational Research*, 243:523–539, 2015.

[31] N. S. Kumar and R. Paneerselvam. A survey on the vehicle routing problem and its variants. *Intelligent Information Management*, 4:66–74, 2012.

[32] R. Landa. *Thinking Creatively in the Digital Age.* Nimble, 1^{st} edition, 2015.

[33] G. Laporte. The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3):345–358, 1992.

[34] G. Laporte. What you should know about the vehicle routing problem. *Naval Research Logistics*, 54(8):811–819, 2007.

[35] V. Leggieri and M. Haouari. A practical solution approach for the green vehicle routing problem. *Transportation Research Part E*, 104:97–112, 2017.

[36] W. Li-ying and S. Yuan-bin. Multiple charging station location-routing problem with time window of electric vehicle. *Journal of Engineering Science and Technology Review*, 8(5):190–201, 2015.

[37] L. C. Lichty. *Combustion Engine Processes*. McGraw-Hill Education, New York, 1967.

[38] C. Lin, K.L. Choy, G.T.S. Ho, S.H. Chung, and H.Y. Lam. Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications*, 41(4):1118–1138, 2014.

[39] J Lin, W. Zhou, and O. Wolfson. Electric vehicle routing problem. In *Transportation Research Procedia*, pages 508–521, Tenerife (Canary Islands), Spain, June 17-19 2009. The 9th International Conference on City Logistics.

[40] G. Macrina, L. Di Puglia Pugliese, F. Guerriero, and D. Laganà. The vehicle routing problem with occasional drivers and time windows. In A. Sforza and C. Sterle, editors, *Optimization and Decision Science: Methodologies and Appliations*, volume 217 of Springer Proceedings in Mathematics & Statistics, pages 577–587, Cham, Switzerland, 2017. ODS, Sorrento, Italy, Springer.

[41] A. Montoya, C. Guéret, J. E. Mendoza, and J. G. Villegas. A multi-space sampling heuristic for the green vehicle routing problem. *Transportation Research Part C*, 70:113–128, 2016.

[42] A. Montoya, C. Guéret, J. E. Mendoza, and J. G. Villegas. The electric vehicle routing problem with nonlinear charging function. *Transportation Research Part B*, 103:87–110, 2017.

[43] J. C. Paz, M. Granada-Echeverri, and J. W. Escobar. The multi-depot electric vehicle location routing problem with time win-

dows. *International Journal of Industrial Engineering Computations*, 9:123–136, 2018.

[44] S. Pelletier, O. Jabali, and G. Laporte. Goods distribution with electric vehicles: Review and research perspectives. *Transportation Science*, 50(1):3–22, 2016.

[45] H.N. Psaraftis. *Green Trasportation in Logistics: The Quest fo Win-Win Solutions*, volume 226 of *International Series in Operations Research & Management Science*. Springer International Publishing Forthcoming, 2016.

[46] O. Sassi, W. R. Cherif, and A. Oulamara. Vehicle routing problem with mixed feet of conventional and heterogenous electric vehicles and time dependent charging costs. Technical report, 2014. https://hal.archives-ouvertes.fr/hal-01083966.

[47] M. W. P. Savelsbergh and T. Van Woensel. City logistics: challenges and opportunities. *Transportation Science*, 50(2):579–590, 2016.

[48] M. Schiffer and G. Walther. An adaptive large neighborhood search for the location routing problem with intraroute facilities. *Transportation Science*, 2017. doi: https://doi.org/10.1287/trsc.2017.0746. Forthcoming.

[49] M. Schiffer and G. Walther. The electric location routing problem with time windows and partial recharging. *European Journal of Operational Research*, 260:995–1013, 2017.

[50] M. Schneider, A. Stenger, and D. Goeke. The electric vehicle routing problem with time windows and recharging stations. *Transportation Science*, 48(4):500–520, 2014.

[51] P. Shaw. Using constraint programming and local search methods to solve vehicle routing problems. In M. Maher and J.F. Puget,

editors, *Principles and Practice of Constraint Programming*, volume 1520 of *Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, 1998.

[52] M. M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35:254–265, 1987.

[53] N. Tajik, R. Tavakkoli-Moghaddama, B. Vahdani, and S. Meysam Mousavic. A robust optimization approach for pollution routing problem with pickup and delivery under uncertainty. *Journal of Manufacturing Systems*, 33:277–286, 2014.

[54] S. Ubeda, J. Faulin, A. Serrano, and F. J. Arcelus. Solving the green capacitated vehicle routing problem using a tabu search algorithm. *Lecture Notes in Management Science*, 6:141–149, 2014.

[55] J. Yang and H. Sun. Battery swap station location-routing problem with capacitated electric vehicles. *Computers & Operations Research*, 55:217–232, 2015.