A LOCAL SEARCH META-HEURISTIC ALGORITHM FOR SOLVING THE CAPACITATED ELECTRIC VEHICLE ROUTING PROBLEM WITH PARTIAL RECHARGING AND MULTIPLE TRIPS

ANDREAS APOLLONAS KOULOPOULOS
SCHOOL OF MECHANICAL ENGINEERING
NATIONAL TECHNICAL UNIVERSITY OF ATHENS

Sector of Industrial Management & Operational Research
Under the supervision of Assistant Professor Dr Athanasios Rentizelas

Page intentionally left blank

Acknowledgements

I would like to sincerely thank Assistant Professor Dr Athanasios Rentizelas for his assistance, professionalism and continuous support. I am thankful to him for guiding me through my first steps into research and motivating me to make this thesis valuable.

In addition, I would like to thank Dr Paraskevas N. Georgiou for his feedback and remarks on this work.

Declaration

I have read and understood the rules on plagiarism and how to properly cite sources contained in the Guide to Writing Dissertations. I declare that, to the best of my knowledge, the content of this Thesis is the product of my work and there are references to all the sources I have used.

The opinions and conclusions contained in this Diploma Thesis are those of the author and should not be interpreted as representing the official positions of the School of Mechanical Engineering or the National Technical University of Athens.

Page intentionally left blank

Abstract

In light of the growing EV penetration and the continuous enhancement of the available charging grid over the last decade, the integration of EVs into common routing problems has become an increasingly significant subject of research. Under proper EV charging and discharging management, present grid capacity can meet the energy needs of a significant number of EVs, directing the transporting industry towards a more sustainable future.

In this thesis, we investigate a new routing problem, which incorporates limited battery capacity for electric vehicles, the ability to partially recharge at certain recharging stations and the possibility of vehicles performing multiple trips within a route. Aiming at illustrating industry-specific scenarios and addressing the need for practical solution approaches, we propose the electric vehicle routing problem with partial recharging and multiple trips (EVRP-PR-MT). In order to compete against benchmark optimal values, the study investigates different solution methods with a focus on meta-heuristics, which better apply to the complexity of EVRP problems. With respect to the EVRP-PR-MT, we formulate a local search (LS) meta-heuristic with intra- and inter-route moves and custom neighbourhood search heuristics regarding recharging stations. We tested our method against two EVRP benchmark sets of small- and large-scale problems accordingly and achieved higher than currently known optimum results on 75% of the instances while maintaining computational times 94% lower on average per instance. Additionally, the LS method managed to eliminate the need for one vehicle in two of the large-scale problems, while maintaining optimal values.

All in all, this thesis presents a new flavour of the EVRP, which adapts to real-life constraints and outperforms current benchmarks in solution quality and efficiency. The EVRP-PR-MT could adapt to scenarios such as last-mile delivery in large city centres, where route efficiency and duration play a crucial role in the competency of logistic systems.

Σύνοψη

Υπό το πρίσμα της αυξανόμενης διείσδυσης των ηλεκτρικών οχημάτων και της συνεχούς βελτίωσης του διαθέσιμου δικτύου φόρτισης κατά την τελευταία δεκαετία, η ενσωμάτωση των ηλεκτρικών οχημάτων σε κοινά προβλήματα δρομολόγησης έχει γίνει ολοένα και πιο σημαντικό αντικείμενο έρευνας. Υπό την κατάλληλη διαχείριση της φόρτισης και της εκφόρτισης των EV, η παρούσα χωρητικότητα του δικτύου μπορεί να καλύψει τις ενεργειακές ανάγκες σημαντικού αριθμού EV, κατευθύνοντας τη βιομηχανία μεταφορών προς ένα πιο βιώσιμο μέλλον.

Στην παρούσα διατριβή, διερευνούμε ένα νέο πρόβλημα δρομολόγησης, το οποίο ενσωματώνει την περιορισμένη χωρητικότητα των μπαταριών των ηλεκτρικών οχημάτων, τη δυνατότητα μερικής επαναφόρτισης σε ορισμένους σταθμούς επαναφόρτισης και τη δυνατότητα των οχημάτων να εκτελούν πολλαπλές διαδρομές εντός μιας διαδρομής. Στοχεύοντας την αποτύπωση πρακτικών προβλημάτων της βιομηχανίας μεταφορών, προτείνουμε το πρόβλημα δρομολόγησης ηλεκτρικών οχημάτων με μερική επαναφόρτιση και πολλαπλές διαδρομές (EVRP-PR-MT). Προκειμένου να ανταγωνιστεί τις βέλτιστες τιμές αναφοράς, η μελέτη διερευνά διαφορετικές μεθόδους επίλυσης με έμφαση στις μετα-ευρετικές μεθόδους, οι οποίες εφαρμόζονται καλύτερα στην πολυπλοκότητα των προβλημάτων EVRP. Όσον αφορά το EVRP-PR-MT, διαμορφώνουμε μια μετα-ευρετική μέθοδο τοπικής αναζήτησης (Local Search) με ευρετικές μεθόδους εντός και μεταξύ των διαδρομών, προσαρμόζοντας παράλληλα ευρετικές μεθόδους αναζήτησης γειτονιάς (neighbourhood search heuristics) όσον αφορά τους σταθμούς επαναφόρτισης. Δοκιμάσαμε τη μέθοδό μας σε δύο σύνολα αναφοράς EVRP με προβλήματα μικρής και μεγάλης κλίμακας αντίστοιχα και επιτύχαμε υψηλότερα από τα γνωστά βέλτιστα αποτελέσματα στο 75% των περιπτώσεων, ενώ διατηρήσαμε υπολογιστικούς χρόνους 94% χαμηλότερους κατά μέσο όρο ανά περίπτωση. Επιπλέον, η μέθοδος LS κατάφερε να εξαλείψει την ανάγκη για ένα όχημα σε δύο από τα προβλήματα μεγάλης κλίμακας, διατηρώντας παράλληλα βέλτιστες τιμές.

Συνολικά, η παρούσα διατριβή παρουσιάζει μια νέα εκδοχή της EVRP, η οποία προσαρμόζεται σε πραγματικούς περιορισμούς και ξεπερνά τα τρέχοντα benchmarks σε ποιότητα λύσεων και αποδοτικότητα. Το EVRP-PR-MT θα μπορούσε να προσαρμοστεί σε σενάρια όπως η παράδοση του τελευταίου χιλιομέτρου (last mile) σε μεγάλα αστικά κέντρα, όπου η αποδοτικότητα και η διάρκεια της διαδρομής παίζουν καθοριστικό ρόλο στην επάρκεια των συστημάτων logistics.

Page intentionally left blank

Table of Contents

Page intentionally left blank

Chapter 1. Introduction

"*In the past the man has been first; in the future the system must be first.*"

- Frederick Taylor

While in the dawn of Industry 5.0, the words of Frederick Taylor still echo in our ears creating a sense of fear and excitement, as the development of the "system" (Operations and Scientific Management) has reached an all-time high by utilising advanced technologies and leveraging disruptive crises (i.e., COVID-19). Now, "the man" finds himself in turmoil where advanced technology, with the use of Artificial Intelligence (AI) and Deep Learning, has emancipated him from being "the first" and fear prevails over the possible implications of such development. At the same time, the global industry continues to strive for optimisation and faster ways of satisfying the never-ending and highly-customized needs of the modern consumer, while of course reducing their environmental footprint. Thus, the focus is eventually turned over to "the man" again, with Industry 5.0 putting him at the centre of the value chain, while also using advanced AI and cognitive computing technologies to reach optimum results in a Cyber-Physical collaborative environment (Adel, 2022). Supply Chain Management, as an important driver of the industry's transformation, doesn't remain stagnant towards Industry 5.0 and attracts a lot of academic and industrial attention.

In parallel, sustainability is becoming a global trend in industrial production. The transportation sector is a major contributor to greenhouse gas (GHG) emissions, accounting for approximately 20% of all carbon dioxide (CO_2) emissions globally, and road transportation accounts for the large majority of those emissions (Albuquerque et al., 2020). A future in which fossil fuels are only fossils, and clean energy (electricity produced by renewable sources) powers the global grid, may seem impossible to grasp with our current efforts. Nevertheless, there have been steps towards this utopia, as in 2022 passenger electric cars are surging in popularity, estimated at 13% of new cars sold in 2022, and projected to reach up to 30% of vehicles sold globally by 2030 (International Energy Agency, 2022). At the same time, endeavours have been taken by regulators around the world to facilitate vehicle electrification for its ability to mitigate GHG emissions, promote sustainable ways of electricity generation, and reduce particulate matter pollution thus benefiting human health (Waraich et al., 2009).

1.1 Context and Problem Statement

The main goal of Supply Chain Management is the optimisation of its operations, by beating the competition and providing better service at lower cost with a sustainability remit (Scott et al., 2011). Optimization is performed by solving complex computational problems, routing, and many other instances related to problem-solving (Deb, 2011). There are many ways to solve a problem that needs to be optimized, based on the type of problem. In this thesis, research is performed regarding the optimisation of the electric vehicle routing problem (EVRP), an extension of the vehicle routing problem (VRP). The main purpose of this problem is to design multiple routes with minimal delivery cost, serving a set of customers and operated by a fleet of Electric Vehicles (EVs). As the problem is an NP-hard problem, determining the perfectly optimal solution is difficult to obtain.

In this thesis, we explore a delivery service system that employs an EV fleet to service consumers. We propose the Capacitated Electric Vehicle Routing Problem with Partial Recharging and Multiple Trips (EVRP-PR-MT), which a) take into account the real-life restrictions of commercial EV fleets and logistics; b) allows EVs to charge and discharge their batteries across the planning horizon and c) directly optimises the routes per the number of vehicles utilised.

1.2 Aim

The aim of this thesis is to solve an Electric Vehicle Routing Problem with constraints that illustrate industry-specific scenarios. The modelling itself is not found in the available literature; thus, following the trend of research, focusing more on real-life cases, we aim at solving the EVRP-PR-MT, and testing its performance against common EVRP benchmarks.

1.3 Objectives

The main objective of this research is to develop a model for the EVRP-PR-MT and propose a solution method, able to compete with instances of common capacitated EVRP problems. The quality of our solution, measured as the comparison of our solution to the optimal solution provided by each instance, should at least be comparable to the quality of the known solution for the instance tested against. The computation time depends on the chosen instance but should remain at a reasonable range within each benchmark test.

1.4 Thesis Outline

The thesis is comprised of seven chapters:

- Chapter 1 presents a brief introduction and a general outlook of the thesis.
- Chapter 2 presents a formal introduction to Vehicle Routing Problem and its variants, and the solution methods found in the literature.
- Chapter 3 describes the mathematical formulation of our problem and the proposed method for solving the EVRP-PR-MT.
- Chapter 4 provides the research methodology and evaluates the design of the experimental process
- Chapter 5 presents our solution results compared to EVRP instances and analyses the method’s performance
- Chapter 6 comments on this thesis, its limitations and possible applications, as well as the role of this work as a stepping stone for future research

Chapter 2. Literature Review

In this chapter, we will give an overview of the literature about the VRP and its extensions. In Section 2.1, we start with a walkthrough of the literature research process, in Section 2.2 an introduction to VRP, in Section 2.3 we address the multiple variants of VRP, later in Section 2.4 we review the different solution methods that have been developed for solving VRPs and finally in Section 2.5 we dive into the EVRP problem and its characteristics. Last, in Section 2.6 we summarise the findings from the literature review.

2.1 Literature Research Methodology

The literature review is the preliminary work required to identify and develop an idea or concept for any research. A literature review was conducted in this study to gather knowledge on the issues listed below:

- To have a good grasp of the vehicle routing problem and its variants.
- To gain a good understanding of the intricacies of the electric vehicle routing problem.
- To map the different flavours of EVRP and find the gap in literature where our variant could contribute.
- To obtain knowledge of the multiple solving methods that have been developed or employed to tackle the VRP and the EVRP.

2.1.1 Search Process

A keyword search has been conducted on Google Scholar, SCOPUS and Science Direct. The following search terms were used to find all of the papers in the list of articles for the literature review: “VRP”, “EVRP”, “Savings heuristic”, “Local Search”, “Multi-trip”, “Improvement Heuristics VRP”, “Partial Recharging”. As mentioned earlier, the first screening criteria of the literature found was a historical hierarchy and later was done on the basis of reading the title and abstract of each article.

2.1.2 Inclusion & Exclusion Criteria

The following criteria were used as guidelines through the vast literature in order to filter related papers for our review:

- Only articles on the topics of the TSP, VRP, VRP solutions methods, construction and optimisation heuristics were included in the review.

- Articles discussing relating variants to ours are selected by reading the title and abstract.
- All articles included are in English, as it is the primary language used in this thesis, but articles in other languages were cited only to reference the existence of research on a specific variant of the VRP.
- Only articles with full-text availability were included

2.2 Vehicle Routing Problem

The general objective of the Vehicle Routing Problem is finding the shortest route for multiple vehicles to serve all customers of a given set. Dantzig and Ramser were the first to introduce The Vehicle Routing Problem in 1959 as The Truck Dispatching Problem (Dantzig & Ramser, 1959), and since then, global research keeps presenting different approaches on how to solve the problem.

VRP instances consist of many different constraints, such as vehicle capacity, precedence relations and time window. Consequently, many variants of the VRP exist. Nonetheless, most research has focused on the classic VRP, a fundamental skeleton of the problem that can be shaped to match real-life scenarios and business cases by adding constraints.

The following is a broad definition of VRP. Let $G = (V, A)$ be an undirected graph where $V = \{0, 1, \ldots, n\}$ is the vertex set and $A = \{(i, j): i, j \in V, i \neq j\}$ is the arc set. Vertex 0 symbolises the depot, which has k vehicles of capacity Q. Customers are represented by the other vertices. Every customer i $\in V \setminus \{0\}$ has a positive demand, $q_i \leq Q$. On A, a cost matrix, c_{ij}, is defined. The objective is to find a set of at most k vehicle routes that (i) start and terminate at the depot, (ii) each customer is only visited once by a single vehicle, (iii) each route's aggregated demand does not exceed Q, and (iv) the overall routing cost is reduced.

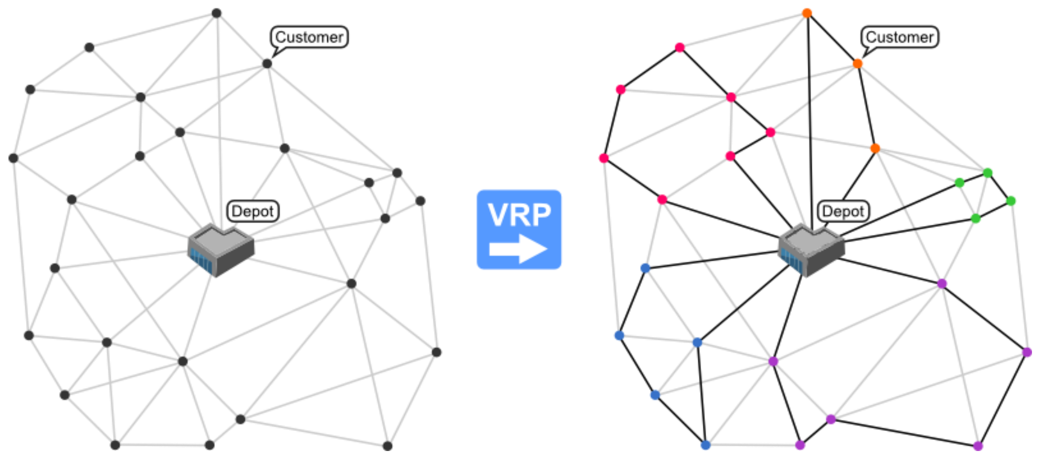

Figure 2.1 A VRP instance, and the suggested solution

Lenstra & Kan, 1981, first classified the VRPs as NP-hard problems, which means that each time we are adding new nodes, i.e., customers, results in an exponential increase in computational complexity.

2.2.1 VRP Notation

Let's go through our notation and its real-world interpretation.

- $G = (V, A)$ is a complete undirected graph

  – Network of routes
- v_0 is the starting node

  – Representing a depot
- $V' = (v_1, . . . , v_n)$ nodes except the initial node

  – Locations of customers
- $A = \{(i, j) : i, j \in V, i \neq j\}$ with associated weight as a cost $c : A \to N^+$

  – A single route between two points bearing a cost, such as distance.
- C is a matrix of edge weights indexed by nodes c_{ij} where $i, j \in V$

  – Matrix of costs between customers
- $R_i \subset V$ is a path that starts and ends at v_0

  – Route visits a subset of customers starting and ending at the depot, it can be referred to it as a delivery plan.
- k number of paths

  – Number of vehicles

- $R = \{ R_1 , . \ . \ . \quad , R_k\}$ is a set of paths

  – All routes (delivery plans) for a curtain VRP instance.
- $\pi \ = \{\pi_1, . \ . \ . \quad , \pi_k\}$ solution for a given instance of VRP.

  – Customer locations in series of visits for numerous vehicles.

The total cost of the route Ri that we intend to optimize is the sum of its weights (costs).

$$C(R_i) \coloneqq \sum_{k=0}^{|R_i|} c_{r_k, r_k+1}$$

The VRP solution's overall cost is the sum of each route cost.

$$C(R) \coloneqq \sum_{i=1}^{|R_i|} C(R_i)$$

2.3 VRP Variants

The VRP model is broadly applied in the global transportation industry and supply chain management, and different variants were further investigated since its dawn to better adapt to the specialised needs of each application. Specifically, the VRP was enriched by considering customer characteristics, service quality, system stochasticity, fleet heterogeneity, environmental and energy issues etc. All flavours of VRP can be mutually combined, which is usually the case found in the literature.

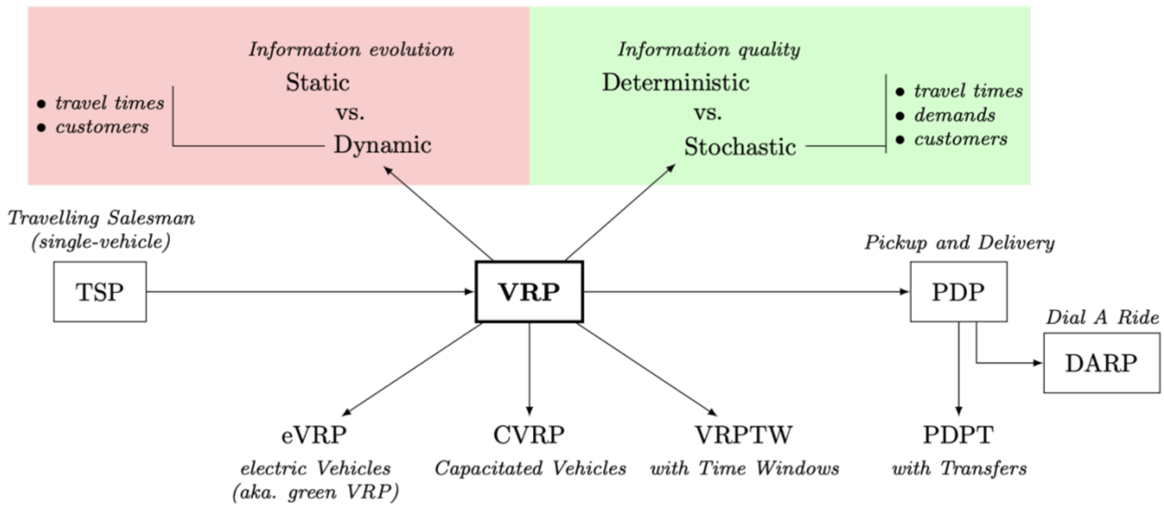

Figure 2.2 Taxonomy of VRPs

2.3.1 Capacitated Vehicle Routing Problem

The Capacitated VRP (CVRP) is an extension of the regular VRP model, by introducing a capacity element for each customer, while the cargo capacity of a single vehicle is assumed to be much smaller than the total customer demands. In the literature, it is often referred to as demand. The customer's demand is represented by $d \in N^{+}$, which might reflect capacity in the form of weight, size, or abstract ideas like a box of products. Furthermore, each vehicle has a specified capacity $Q > 0$.

The CVRP extends the solution feasibility formula by the following capacity constraint.

$$q(R) \coloneqq \sum_{i \in R} d_i \leq Q \qquad (1.3)$$

If the vehicle capacity of the fleet stays the same, we are dealing with CVRP with a homogeneous fleet. A fleet with varying capacities for each vehicle is a heterogeneous fleet.

2.3.2 Vehicle Routing Problem with Time Windows

Robert A. Russell, 1977 was the first to propose an extension to the set of constraints about time intervals in which individual customers should be visited. Customers have assigned time window intervals $[e_i, l_i]$ where $e_i < l_i$. The time interval equals to the time period in which a vehicle should visit the node.

The time window can be implemented as either a hard or soft restriction (Sanghavi et al., 2007). A hard constraint requires the vehicle to visit the node, i.e., the customer, within the time frame specified, or else the solution is not feasible. A hard constraint requires the vehicle to visit the node, i.e., the customer, within the time frame specified, or else the solution is not feasible. Soft restrictions do not absolutely compel the vehicle to visit the customer, but they do impose a penalty for a missed period in the form of a penalty fee. The penalty is incorporated into the cost function, which VRP seeks to decrease (Russell & Urban, 2008).

2.3.3 Pick and Deliver

The Pick and Deliver Problem (PDP) extends the ordinary VRP by combining pick and drop with precedence relationships, where the pickup point must come before the associated delivery location. This kind of VRP is one of the most complicated, posing a challenge to traditional approaches such as optimization heuristic algorithms. The viability of a PDP solution is evaluated by determining if all delivery locations were visited before the pickup point (Dumas et al., 1991).

2.3.4 Static & Dynamic VRP

When solving the vehicle routing problem, we normally assume that all the input data is static and definite. However, in real-world applications, data such as customer demand or travel time are frequently incomplete or inaccurate throughout the planning phase and are only progressively discovered and described.

Static VRP considers the input data constant. The dynamic VRP is aware of information evolution (Psaraftis, 1980), and its purpose is to develop a resilient routing planner capable of solving previously encountered instances subject to minor modifications without having to recalculate the entire instance. After solving a certain instance of a combinatorial optimization problem, it becomes necessary to repeatedly solve many other instances with minor variations from the original instance but without revisiting the entire problem (Bertsimas et al., 1990).

2.3.5 Deterministic & Stochastic VRP

A VRP is considered stochastic (Laporte & Gendreau, 1996), when part of the data act as random variables, and the routes should be designed before the values of these random variables become known. We can extract some hidden information from the probability distribution of the random variables and utilize it to our advantage in the planning process. Because stochastic information is part of the cost function, the newly produced plans will include stochastic information, and the routing decisions may result in different conclusions. On the contrary, deterministic VRP contains no random information that may be used before route execution, and all provided information is known with certainty.

2.3.6 Multi-Trip VRP

Multi-Trip VRP is a variation in which vehicles can make several trips that start and end at the same depot (Fermín Cueto et al., 2021). In the context of the multi-trip element of the problem, we use the term *trip* to refer to an instance when a specific vehicle leaves the depot to visit customers and returns. The term *route*, on the other hand, refers to the actual path that a vehicle takes throughout a trip.

This is a variant that popped out of an empirical case study that Brandão & Mercer, (1998) did for Burton’s Biscuits Ltd, where their simulation showed that by introducing a second trip for vehicles returning in less than 7 hours, “*the number of vehicles could be reduced from 21 to 19 and the unit cost of the deliveries was 5% less”*. The study uses the nearest neighbour insertion procedure to construct the initial route and applies Tabu search with insertion and swap moves.

Petch & Salhi noted in a 2003 article that permitting multi-trips saves companies money on all transportation expenditures. They also emphasize that Multi-Trip VRP may be useful for both tactical and strategic planning, and they hope to get strategic planning insights as a result (Petch & Salhi, 2003). Their solution approach is composed of a multi-phase construction heuristic composed by Yellow’s savings algorithm (Yellow, 1970) phase, a phase of improvement heuristics and a final tour partition approach where available trips are partitioned into small feasible trips using a geographical “route codification”. After obtaining the new solution population, the search is reverted to the previous phase in order to enhance the available solutions.

In the last decade, as research is focused more and more towards real-life cases, we see a plethora of studies where multi-trip VRPs are implemented and solved with advanced heuristics such as local search, adaptive large neighbourhood search and genetic algorithms (Babaee Tirkolaee et al., 2019; Cattaruzza et al., 2014; Grangier et al., 2016).

In this thesis we allow vehicles to end the route and return to the depot, as it allows us to introduce more restrictions and better portray a real-life problem, while still receiving competitive results. In this research the reasons why a route could be broken into two could be for recharging purposes, reloading/restocking the vehicle and incorporating shorter trip durations that could help in the strategic planning of the drivers’ schedule.

2.3.7 Other VRP variants

Stein (1978) proposed the Dial-a-Ride (DARP), which is a special case of dynamic VRP with pick and deliver. Passengers request a ride with an origin and drop location, as well as an optional time window.

Split Delivery VRP is a variation in which customers can be visited more than once. This might be useful for large-capacity deliveries or stocking fulfilment centres (Dror & Trudeau, 1989).

Multi Depot VRP is a simplified version of the vehicle routing problem with PDP, where pick can only take place at designated depot sites. Because of this, Multi Depot VRPs are less complicated than the vehicle routing problem with pick and deliver.

2.4 Solution Methods

The classic VRP and its derivatives have been proven to be NP-hard problems. Over the last few decades, significant scientific efforts have been undertaken to overcome these solving difficulties. In general, there are four types of solution techniques: exact methods, classical heuristics, meta-heuristics and reinforcement learning based approaches. Exact algorithms were the first to solve VRPs and are known to solve the issue optimally. However, since exact optimisation employs enumeration, it converges slowly and is unable to handle problems of reasonable sizes with a consistent success rate in a reasonable amount of time. The most popular exact algorithms can tackle problems with up to 100 vertices (Baldacci et al., 2008), although real instances frequently exceed this size. Therefore, research has focused on heuristics (Laporte, 2007). When opposed to exact methods, approximate heuristics are more suited for very large-scale routing problems. Heuristics are also easy to modify (i.e., by adding limitations), which is necessary for realistic conditions. Simple heuristics include both route construction and improvement approaches that grow the route one node at a time until a full route is produced while simultaneously attempting to enhance the solution for a more efficient global optimum. Meta-heuristics identify their first solution and then look for a better global optimum solution.

2.4.1 Exact Algorithms

Although there are exact methods to solve the VRP, they have a limited ability to handle instances of larger sizes in reality. The three primary approaches in this area are direct tree search methods, dynamic programming, and integer linear programming.

Direct tree search

Direct tree search techniques solve the VRP by generating routes progressively using a branch and bound tree. Their algorithm reformats arcs with branches; branches are formed by including or omitting an arc from the solution (Christofides & Eilon, 1969). As a result, broad search trees with minimal depth are produced. These algorithms could only solve simple or small instances. The addition of two approaches capable of obtaining sharp lower limits (Christofides N. et al., 1981) significantly improved the efficiency of these algorithms.

Dynamic programming

An optimization technique called dynamic programming (DP) segregates complicated problems into several smaller problems. At each level, a piece of the partial solution is added as these subproblems are solved. The DP method produces an optimum solution for the original problem when an optimal solution to a (sub)problem can be identified exclusively based on the optimal solutions of its subproblems. This is called the *Principle of Optimality* (van Hoorn, 2016).

Integer linear programming

Numerous approaches fall into this category as a result of the fact that the majority of research on exact algorithms for the VRP has been conducted in the area of integer linear programming (Laporte et al., 1984). Among integer linear programming techniques, vehicle flow formulations are by far the most frequently employed.

2.4.2 Classic heuristics

Construction heuristics and improvement heuristics are two forms of classic heuristics. The primary distinction between these two is that improvement heuristics begin with a feasible solution that they seek to enhance, whereas construction heuristics do not.

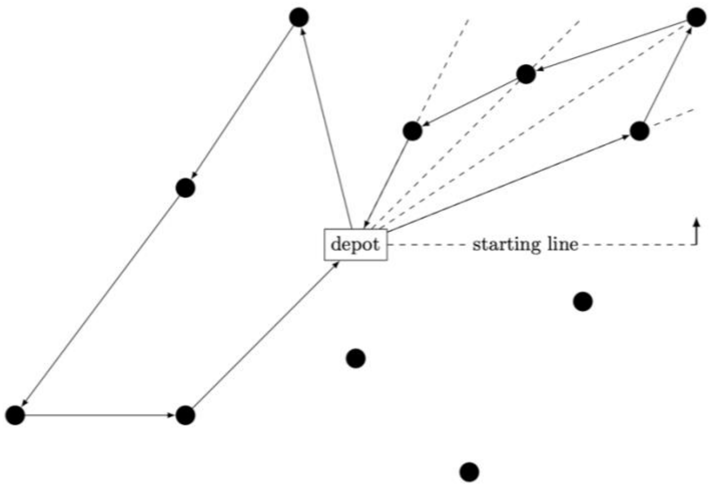

Figure 2.3 A sweep algorithm example

Construction heuristics

The most well-known construction heuristic for solving a VRP is the savings method proposed by Clarke & Wright in 1964. The fundamental idea is to merge routes based on the cost reductions that result from this process. Initial routes are established from the depot to the customer and back. The method then computes the potential savings $s_{ij} = c_{0i} + c_{0j} - c_{ij}$ by adding the arcs $(i, 0)$ and $(j, 0)$ and eliminating the arc (i, j), if this results in a positive saving and a feasible merged route. The combination that has the highest savings is then implemented. This approach is repeated until no further profitable and feasible savings are attainable.

To initialize our solution in this thesis, we implement a savings heuristic variation (Section 3.3.1), with added restrictions for maximum energy capacity and maximum route duration, to generate a solution that could be easily adapted to industry-specific problems, such as driver work schedules.

Improvement heuristics

Improvement heuristics begin with a feasible path. Afterwards, they aim to improve the route by making either intra-route or inter-route improvements. Inter-route heuristics move one or more consumers between routes, whereas intra-route heuristics move one or more customers inside a route.

The 2-opt heuristic, which Croes (1958) first described, is an intra-route optimization technique that offers a 2-optimal route; one that cannot be improved by swapping two arcs. To achieve this, crossings (arcs that overlap) are removed from the routes since crossings are never the best option in a traditional VRP with a symmetric cost matrix. However, the optimum route could involve crossings in other VRP variations due to limitations like time constraints. Similar techniques include Or-opt (Or, 1976), which involves moving a string of vertices in a row while maintaining the orientation of the original path.

Several alternative inter-route improvement heuristics, such as ejection chains (Glover, 1992), λ-interchange (Osman, 1993), and CROSS (Taillard et al., 1997), have been developed.

In this thesis, a set of improvement heuristics are being used in combination with neighbourhood-search operators to reach the optimum (Section 3.3.2).

2.4.3 Meta-heuristics

Meta-heuristics explores the solution space significantly more thoroughly than standard heuristics. In meta-heuristics, inferior or infeasible motions can also be accepted to escape from local minima. This field is divided into three categories: local search, population search, and neural networks.

Local Search

Local search metaheuristics work within a single-point-based search framework, with the goal of iteratively improving a solution in hand over time with respect to a single objective by employing solution perturbation strategies known as move operators and move acceptance methods that begin with an initially generated solution (Jackson et al., 2018). In this thesis, we use the local search meta-heuristic built by Rasku, Kärkkälnen and Musliu (2019), and customized it to meet the requirements of EVRP with Partial Recharging and Multiple Trips, while also adding inter-route and intra-route moves and recharging station (RS) removal and insertion operators.

Tabu search (TS) is a local metaheuristic search method that is widely utilized in mathematical optimization. Local search algorithms are frequently trapped in suboptimal areas. TS improves the performance of these approaches by forbidding previously visited solutions or others by user-defined restrictions. Taillard et al. (1997), who worked on a TS heuristic for VRP with time-windows, introduced the core concept of tabu search. A neighbourhood of the present solution is generated in the TS by an exchange mechanism that swaps customer sequences between two routes.

Population Search

Population search methods simulate natural selection (i.e., genetic algorithms). Mutating the attributes of candidate solutions from a population of solutions causes the population to progress towards a new generation of presumably better solutions. As mentioned by Mester & Bräysy, 2007 and Vidal et al., 2014, genetic algorithms are frequently combined with local search.

A lot of attention has been towards Ant Colony Optimisation (ACO) for solving VRPs. ACO illustrates the behaviour of real ants along with enhanced abilities such as memory of past occurrences and knowledge about the distance to other locations (Bell & McMullen, 2004). Basically, edges that occur frequently in good solutions are remembered as good edges and are more likely to be used in later solutions. The first attempts to implement

ACO to the travelling salesman problem (TSP) were done by Colorni, Dorigo and Maniezzo, (1991), and later it was extended to the VRP, (Bullnheimer et al., 1999) showing promising results. This led to ant colony optimisation methods being implemented for many variants of the classical VRP (Corne et al., 1999).

Neural Networks

Neural networks operate in a manner that is similar to organic neural systems like the brain. When it comes to vehicle routing, this idea has only attracted little attention, but over the last decade, with the aid of deep learning techniques and other heuristics, its popularity has dramatically increased. (Nazari et al., 2018) (Amir et al., 2013) (Chen et al., 2020). This line of research finds a lot of stepping ground towards industry-specific problems where, instead of the objective function, there are multiple other parameters and preferences ingrained in the years-long experience of the route planners and the drivers.

"Drivers, for instance, have familiarity with certain neighbourhoods and knowledge of the state of roads, and often consider the best places for rest and lunch breaks. This knowledge is difficult to formulate and balance when operational routing decisions have to be made". (Mandi et al., 2021)

Such and other examples are pointing out the need of incorporating past solutions into the planning process and later in the optimisation phase.

2.5 Electric Vehicle Routing Problem

The electric vehicle routing problem (EVRP) followed the green vehicle routing problem proposed by Erdoĝan & Miller-Hooks, (2012) for alternative fuel vehicles, as a particular variant, which was one of the first studies to present recharging stations as dedicated points. These points have to be visited to extend the vehicle's driving range. As previously stated, the G-VRP fleet is made up of alternative fuel vehicles, with the objective of minimizing the total distance travelled. The authors follow a complete refuelling policy. The Modified Clarke and Wright Savings heuristic and the Density-Based Clustering Algorithm are introduced as constructive heuristics.

Schneider et al., 2014 proposed the electric vehicle routing problem with time windows (EVRPTW) by expanding the framework specifically for electric vehicles. They make the assumption of a linear charging time, dependent on the EV's battery level when they arrive at the stations, rather than Erdoĝan & Miller-Hooks, (2012) who use a constant replenishing time. A similar model has been implemented in this thesis, incorporating a partial charge policy which reduces delays in the total route duration.

Researchers began to examine the topic under more complex modelling for EV batteries beginning with J. Lin et al., (2016). They suggest a framework for the EVRP that considers the effects of EV load, speed, and battery charging speed on battery usage, diverse fleet, cost of battery deterioration, and braking energy recovery. (Goeke & Schneider, 2015; Touati-Moungla & Jost, 2011; S. Zhang et al., 2020; Z. Zhao et al., 2020).

Because of the problem's intricacy, most of the available research uses meta-heuristics as solution approaches. Only a few articles discuss the exact methods for the EVRP, and the work of Desaulniers et al., (2016) is one of them. For four EVPTW variants that cover full/partial recharge and the number of station visits for a specific route, a column-generating approach is proposed (Schneider et al., 2014).

The heuristic and meta-heuristic methods outlined in Section 2.3 may be used for the EVRP with minimal adjustments to the station nodes and charging schedule due to similarities between the VRP and the EVRP. For instance, a construction heuristic based on the Clarke-and-Wright saving heuristic (Clarke & Wright, 1964) is proposed in Erdoĝan & Miller-Hooks, (2012) study. If the merger of two routes would violate the distance constraint, a station is added at the spot, increasing the total distance travelled. In order to relocate the station, if one exists, to its prime location along the route without altering the sequence in which customers are visited, Felipe et al., (2014) constructed a new neighbourhood operator called recharge relocation. Similar operators, such as station insertion and removal (Keskin & Çatay, 2016) are described in other publications and also used in this study.

To summarize, using meta-heuristics is realistically a one-way path to the best solutions since the EVRP problem is more complex than the VRP versions suggested by existing research.

2.5.1 EVRP with Multiple Trips

There is little literature to be found on EVRP with multi-trips, which does not come as a surprise, as the specific VRP variant is usually studied in real-life case studies, and EVs have not yet fully penetrated the transport market to a significant volume. The first literature that combined the features of EVRP and Multiple Trips was introduced by M. Zhao & Lu, (2019), which, foreseeably, studies a real-world routing problem, proposed by a Chinese logistics company. The proposed solution method is a combination of ALNS and integer programming (IP), which uses common insertion and removal heuristics.

In 2022, more interesting case studies were conducted on EVRPs with multi-trips, proposing metaheuristic methods and solutions via genetic algorithms.

Mahmoud et al., (2022), proposed a Ruin & Recreate meta-heuristic for solving the electric vehicle routing problem with time windows and multiple trips (EVRP-TW-MT). The Ruin & Recreate meta-heuristic is a large neighbourhood search which follows the steps of: a) initial route construction, b) partial node removal ("*Ruin*" step), c) configuring new points to insert "*ruined*" nodes ("*Recreate*" step). The purpose of this study was to implement the aforementioned method in a case study on sustainable last-mile logistics, in the city of Lyon, France.

Wang et al., (2022), presented a hybrid variable neighbourhood search (Hybrid VNS) algorithm for solving the multi-trip and heterogeneous-fleet electric vehicle routing problem (MTHF-EVRP), at the 2022 Journal of System Simulation. They aimed at creating a model that could be adapted to real business practice and managed to generate optimum solutions in a very short time, highlighting the efficiency of the Hybrid VNS algorithm.

J. Zhang & Zhang, (2022), stress the need for research on applications of electric buses (EBs) to short-notice evacuations. The current research suggests that their use is obsolete in an evacuation preparation stage due to the inefficiency of the available power grid, but neglects the potentiality of coordinated EBs. Thus, they propose a multi-trip EB routing framework and compare it with conventional liquefied natural gas (LNG) buses in terms of efficiency and operational cost. The algorithm developed is a genetic algorithm with improved recombination (GA-IR), which outperforms conventional genetic algorithms and can be proven to improve evacuation efficiency, if combined with an increase in charging stations and their strategic position allocation.

2.6 Literature Review Results

In Table 1 and Table 2 we summarise the studied literature in order to give some perspective to the research trends through time.

In Table 1, we present a summary of the literature on the general context of the VRP, in chronological order. This table alone serves as an index of how the VRP has evolved in terms of variants and solution methods. We can clearly see that soon after Dantzig and Ramser (1959) first introduced the VRP problem, variants started to pop out, and many solution methods were employed in parallel to their development. As better algorithms were becoming available, linear programming and exact approaches started to fade from the literature as early as the 2000s.

In Table 2, we showcase the literature on the general context of EVRP, which we reviewed for the purpose of our research. As the EVRP is by definition a harder problem than the VRP, it doesn't come as surprise to find only one article proposing an exact algorithm (Desaulniers et al., 2016), while the rest incorporate more powerful solving methods. It is also obvious that Table 2 is proportionally smaller in size than Table 1, which reflects the lack of attention that the EVRP has received from the academic community. As seen in Figure 2.4, according to the SCOPUS database, since the initial introduction of EVRP (or GVRP) by Erdogan, only 67 articles on the topic of EVRP have been published, of which only 5 have received around 100 citations, while the rest remain at low numbers. At the same time, we notice that specifically on the EVRP there is a faster emergence of variants and also research is being developed on top of industry-specific cases, which is also verified by Figure 2.5, showcasing documents published per year on the EVRP.

While the academic focus does not lean towards the EVRP, the transportation industry makes important contributions to the field of VRP in general and drives the research activities in this area. In the last 20 years, we can also see problem variants addressing very specific industry constraints and, at the same time, an advancement in industry-related case studies, as seen in both Table 1 and Table 2. This highlights the important role that the transportation industry has portrayed in recent years in the development of research in terms of introducing new variants and adapting solution methods to real-life needs

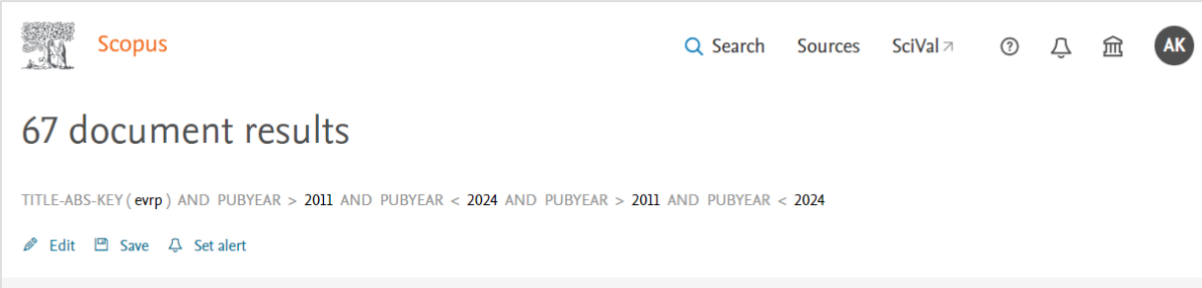

Figure 2.4 Number of documents published on the topic of EVRP from 2012 till 2023 (according to the SCOPUS database)

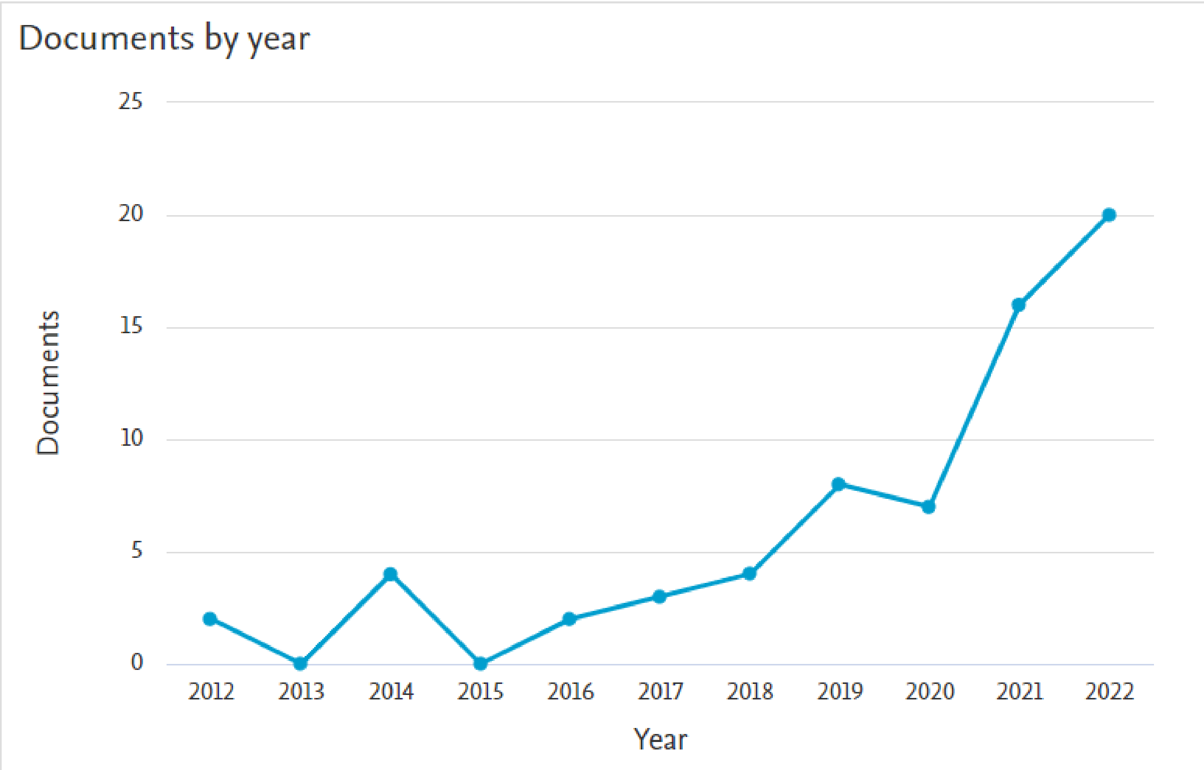

Figure 2.5 Number of documents published per year on the EVRP (according to the SCOPUS database)

Literature	Problem Variant	Solution Approach	Case	Set of
Dantzig & Ramser, (1959)	Classic VRP	Linear Programming	-	-
Clarke & Wright, (1964)	VRP	Savings Algorithm	-	-
Christofides & Eilon,	VDP	Tree Search Algorithm	-	yes
Wren & Holliday, (1972)	VRP	Improvement Heuristics	-	yes
S. Lin & Kernighan, (1973)	TSP	Improvement Heuristics (2-opt)	-	-
Taillard et al., (1997)	VRPSTW	Tabu Search	-	yes
Robert A. Russell, (1977)	VRPTW	Lin and Kernighan heuristic	-	yes
Psaraftis, (1980)	DAR	Dynamic Programming	-	-
Christofides N. et al., (1981)	VRP	Tree Search Algorithm	-	yes
Laporte et al., (1984)	DCVRP	Linear Programming	-	-
Dumas et al., (1991)	PDPTW	Dynamic Programming	-	Yes
Savelsbergh, (1992)	VRPTW	Improvement Heuristics	-	-
Laporte & Gendreau,	SVRP	Various Heuristics	-	-
Brandão & Mercer, (1998)	MTVRP	Tabu Search	yes	yes
Bullnheimer et al., (1999)	VRP	Ant Colony Optimisation	-	yes
Corne et al., (1999)	VRPTW	Ant Colony Optimisation	-	yes
Helsgaun, (2000)	TSP	Improvement Heuristics (n-opt)	-	yes
Petch & Salhi, (2003)	MTVRP	Multi-Phase Heuristic	-	yes
Toth & Vigo, (2003)	VRP	Improvement Heuristics (k-	-	yes
Bell & McMullen, (2004)	VRP	Ant Colony Optimisation	-	yes
Bräysy & Gendreau, (2005)	VRPTW	Improvement Heuristics	-	yes
Mester & Bräysy, (2007)	CVRP	Local & Population Search	-	yes
Baldacci et al., (2008)	CVRP	Exact Algorithm	-	yes
Russell & Urban, (2008)	STVRPSTW	Tabu Search	-	yes
Amir et al., (2013)	Dynamic VRP	Neural Networks	-	yes
Vidal et al., (2014)	MAVRP	Local & Hybrid Genetic Search	-	yes
Cattaruzza et al., (2014)	MTVRP	Genetic Algorithm	-	yes
Grangier et al., (2016)	2E-MTVRP-SS	Adaptive Large Neighbourhood	-	yes
Nazari et al., (2018)	VRP	Neural Networks	-	yes
Rasku et al., (2019)	CVRP	Local Search Heuristics	-	yes
Babaee Tirkolaee et al.,	MTVRPTW	Local Search Heuristics	yes	yes
Chen et al., (2020)	CVRPTW	Neural Networks on LNS	-	yes
Mandi et al., (2021)	VRP	Neural Networks	yes	-

Table 1. Literature summary of the VRP variants

Literature	Problem Variant	Solution Approach	Case Study	Set of Instances
Touati-Moungla & Jost, (2011)	EVRP	Dijkstra's Algorithm	-	-
Erdoĝan & Miller-Hooks, (2012)	GVRP	Improvement Heuristics	-	-
Schneider et al., (2014)	EVRPTW	Hybrid Tabu Search	-	yes
Felipe et al., (2014)	GVRP-PRMF	Local Search	-	yes
Goeke & Schneider, (2015)	EVRPTWMF	Adaptive Large Neighbourhood Search	-	-
J. Lin et al., (2016)	EVRPTW	Mixed Integer Linear Programming	yes	yes
Desaulniers et al., (2016)	EVRPTW	Exact Algorithm	-	-
Keskin & Çatay, (2016)	EVRPTW-PR	Adaptive Large Neighbourhood Search	-	yes
M. Zhao & Lu, (2019)	EVRP-MT	Adaptive Large Neighbourhood Search	yes	yes
S. Zhang et al., (2020)	FEVRPTW	Adaptive Large Neighbourhood Search	-	yes
Z. Zhao et al., (2020)	EVRP	Ant Colony Optimisation	yes	yes
Mahmoud et al., 2022)	EVRPTW-MT	Improvement heuristic	yes	-
Wang et al., (2022)	EVRPMF-MT	Hybrid Variable Neighbourhood Search	-	yes
J. Zhang & Zhang, (2022)	EVRP-MT	Genetic Algorithm	yes	-

Table 2. Literature summary of the EVRP variants

Chapter 3. The EVRP-PR-MT

The most significant factors impacting commercial EVs competitiveness according to the study of Davis & Figliozzi, (2013), are route feasibility, minimum fleet size, minimum travelled distance, charging level, purchase costs and planning horizon. Following the objective function of the classic VRP and inspired by the above, this research study presents a detailed model of the EVRP that takes into account realistic operational constraints (i.e., vehicle capacity and multiple trips) and captures the most important parameters for optimal routing where electric vehicles exist. In this chapter, we attempt to present a comprehensive mathematical model for the EVRP-PR-MT that encompasses the real-world processes and constraints that a corporation should consider when routing and scheduling an electric vehicle fleet.

3.1 Problem Description

We define the electric vehicle routing problem with partial recharging and multiple trips on the general premises of Cortés-Murcia's (2019) problem formulation. We define a complete directed graph $G = (V', A)$ with a set of vertices $V' = \{V \cup F' \cup \{0, N+1\}\}$ and a set of arcs given by $A = \{(i,j) | i, j \in V', i \neq j\}$. Let $V = \{1, \dots, N\}$ be the set of customers, $F = \{0, \dots, M\}$ be the set of recharging stations (RS) and F' be the set including dummy vertices that represent the multiple visits to vertices of F. Vertices 0 and $N+1$ denote the depot as a departing and arriving node respectively.

There is a fixed fleet size of P homogeneous EVs, $P = \{p_1, \dots, p_e\}$, with cargo load capacity of Q, a driving range D (due to limited battery capacity) and a charging time t_{ci}. Each customer i has an associated demand q_i and a service time s. During customer service, all vehicles must remain at the customer locations for a certain time period (service time). Similarly, vehicles remain stationed at recharging stations for t_{ci} while charging. Energy consumption of an EV travelling through an arc $(i,j) \epsilon A$ is determined by the arc distance d_{ij} and the consumption rate γ.

The EVRP-PR-MT seeks to determine a route plan for satisfying customers' demands while minimizing the sum of travel costs and charging costs, with each route departing from the depot visiting all customers and coming back to the depot. Multiple trips, $K = \{k_1, \dots, k_l\}$ are allowed in a route, for charging and reloading cargo purposes. The total travel distance of a vehicle after the last charge should not exceed its maximal driving range D.

The time spent at the RS during intra-route charging is represented by the positive variable t_{ci}, which is assumed linear and depends both on the inverse recharging rate r and the difference between the required energy and state of charge (SoC) on arrival at $i \in F'$.

3.2 Mathematic Formulation

We present a mixed integer linear programming formulation for the EVRP-PR-MT. For every arc in $(i,j) \epsilon A$ the Boolean decision variable x_{ij} is equal to 1 if arc (i,j) is traversed, 0 otherwise. Moreover, for the set of arcs $\{(i,j)|i \in F', j \in V\}$ the Boolean decision variable z_{ij} is defined. It is equal to 1 if the customer j is visited from the recharging station i, 0 otherwise. Variable u_i defines the remaining cargo and y_i defines the SoC on arrival at vertex $i \in V'$. Variable w_i is the amount of energy recharge at recharging station $i \in F'$. Finally, the recharging time t_{ci} is computed as $r * w_i$. Table 3 summarises the sets, variables, and parameters of the model.

Using this notation, the EVRP-PR-MT can be formulated as the following integer program, with the objective function to minimize:

$$\min \sum_{(i,j)\in A} d_{ij} \sum_{k\in K} \sum_{p\in P} x_{ijp}^k \tag{1}$$

Subject to:

$$\sum_{j\in V'} x_{ij}^k + \sum_{j\in F'} z_{ij}^k = 1 \quad \forall i \in V, \forall k \in K \tag{2}$$

$$\sum_{j\in V' i\neq j} x_{ij}^k \leq 1 \quad \forall i \in F', \forall k \in K \tag{3}$$

$$\sum_{j\in V'} x_{ij}^k - \sum_{j\in V'} x_{ji}^k = 0 \quad \forall i \in V', i \neq j, \forall k \in K \tag{4}$$

$$u_i \leq Q \quad \forall i \in V' \tag{5}$$

$$u_i - q_i x_{ij}^k + Q\left(1 - x_{ij}^k\right) \leq u_j \quad \forall i,j \in V', i \neq j, \forall k \in K \tag{6}$$

Sets and parameters	
V	Set of customer vertices
A	Set of arcs
F	Set of recharging stations vertices
F'	Set of dummy vertices that represents the visits to RS on F
V'	Set of nodes, recharging visits and depot nodes
K	Set of trips vertices
P	Set of homogenous EVs
$0, N+1$	Depot nodes
Q	Load Capacity
D	Driving range
B	Battery Capacity
q_i	Demand of customer i
s	Service time
d_{ij}	Distance between vertices i and j
γ	Consumption rate
r	Inverse recharging rate
a	Vehicle Speed
Decision variables	
IC_i	Initial SoC that is required by the vehicle that departs from the depot and arrives at node i
t_{ci}	Time spent at the RS_i during the intra-route charging
u_i	Remaining cargo on arrival at node i
w_i	Amount of energy recharged at RS_i
x_{ij}	Boolean variable indicating if arc (i,j) is traversed
y_{ij}	SoC on arrival at node i
z_{ij}	Boolean variable indicating if the customer j is visited from RS_i

Table 3. Variable and parameter definitions of the EVRP-PR-MT model

$$u_i - q_i z_{ij}^k + Q\left(2 - x_{ij}^k - z_{ij}^k\right) \le u_j \qquad \forall i,j \in V', i \neq j, \forall k \in K \tag{7}$$

$$y_i^k - \gamma d_{ij}^k x_{ij}^k + B\left(1 - x_{ij}^k\right) \ge y_j^k \qquad \forall i \in V, \forall j \in V', i \neq j, \forall k \in K \tag{8}$$

$$y_i^k + w_i^k - \gamma d_{ij}^k x_{ij}^k + B\left(1 - x_{ij}^k\right) \ge y_j^k \qquad \forall i \in F', \forall j \in V', i \neq j, \forall k \in K \tag{9}$$

$$0 \le y_i^k \le B \qquad \forall i \in V', \forall k \in K \tag{10}$$

$$0 \leq y_i^k + w_i^k \leq B \qquad \forall i \in F', \forall k \in K \tag{11}$$

$$\sum_{j \in V} z_{ij}^k \leq 1 \qquad \forall i \in F', \forall k \in K \tag{12}$$

$$x_{ij}^k \in \{0,1\} \qquad \forall i, j \in V', i \neq j, \forall k \in K \tag{13}$$

$$z_{ij}^k \in \{0,1\} \qquad \forall i \in F', j \in V, \forall k \in K \tag{14}$$

$$u_i^k, IC_i^k, y_i^k \geq 0 \qquad \forall i \in V', \forall k \in K \tag{15}$$

$$t_{ci}, w_i^k \geq 0 \qquad \forall i \in F', \forall k \in K \tag{16}$$

Based on the above, the objective function (1) is to minimize the total distance travelled. Constraints (*2*) state that all customers have to be visited once while recharging stations and dummy vertices cannot be visited more than once, constraints (*3*). Flow conservation constraints (*4*) guarantee for each vertex that the number of incoming arcs is equal to the number of outgoing arcs. Vehicle capacity is restricted by constraints (*5*). Load flow and fulfilment of demand are represented by constraints (*6*) for all the vertices and by constraints (*7*) for those vertices visited after visiting a recharging station, where the demand of a customer visited, affects the remaining load. The battery level at a vertex following a customer visit is set by constraints (*8*) while constraints (*9*) set the battery level at a vertex after a recharging station visit. Constraints (*10*) and (*11*) guarantee that battery capacity is respected. Constraints (*12*) sets the maximum number of customers that could be served after each visit at a recharging station. Constraints (*13*)–(*16*) specify the types and ranges of the decision variables.

3.3 Algorithm

The EVRP-PR-MT is an NP-hard problem and is an extension of the classic VRP. Our goal is to solve large instances in a reasonable amount of time, and it is well understood that using meta-heuristics is an effective technique to deal with these types of problems. Accordingly, our solution could be divided into two phases: a construction phase, where an initial solution is generated using a savings heuristic, and the improvement phase, where the solution is optimised with a local search framework reiterating improvement moves.

3.3.1 Construction Phase

As mentioned in Section 2.3.2 the initial routes are constructed through the parallel savings algorithm proposed by Clarke & Wright, (1964). The parallel savings algorithm begins with each customer being serviced individually by a route. Then, for each iteration, the best viable merging is made until no more mergers exist in the savings list. Because the routes are built in parallel, each merging decreases the number of routes by one. The savings are calculated by the method proposed by Clarke & Wright, (1964) for merging two routes by connecting customers i and j with:

$$s(i,j) = d_{0i} + d_{0j} - d_{ij}$$

The distance saved by merging the two routes by connecting i and j with an edge, assuming that i and j are linked to the depot (marked by 0) with an edge prior to the merge process, is described by the value of the savings $s(i,j)$.

Aside from the savings function, the ordering of the savings values is important. Typically, the merging with the highest saving value is applied first by the savings algorithm. A merging is also subject to the following constraints:

i. Customers i and j are not on the same route,
ii. The merged routes should have an edge from i and j to the depot, and
iii. The merge will not breach capacity or maximum route cost thresholds.

3.3.2 Improvement Phase

In the improvement phase of the algorithm, we implement the local search framework proposed by Rasku, Kärkkälnen and Musliu (2019), which iteratively improves the solution by exploring neighbouring solutions. In this thesis, we apply *intra-route* and *inter-route* moves, along with custom recharging station operators such as worst RS removal and RS insertion, which are explained later in this chapter. The move operators generate neighbouring solutions by modifying one or more attributes of the solution. Here, attribute might refer to arcs linking two customers, for example. When a neighbouring solution is found, it is compared to the existing solution. If the neighbouring solution is better, it replaces the present solution, and the search continues.

The two acceptance strategies used in our research are first-accept (FA) and best-accept (BA). The first-accept approach chooses the first neighbour who meets the acceptance requirement. The best-accept method looks at all neighbours who meet the criterion and chooses the best one (Bräysy & Gendreau, 2005).

Intra-route heuristics change one or more consumers' positions inside a route. The intra-route heuristics applied in this thesis are: 2-opt move, 3-opt move, relocate move and 2-opt exchange.

Lin & Kernighan (1973) first presented the *2-opt intra-route* refinement solution for the travelling salesman problem. As illustrated in Figure 3.1, the heuristic first selects two non-consecutive edges that belong to the same route at random. The two selected edges are removed from the original path, and their points are re-joined to form a new path.

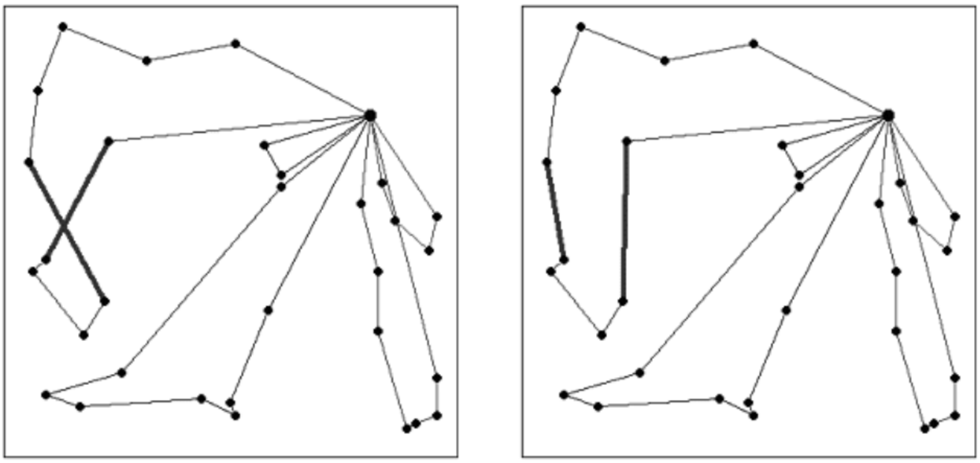

Figure 3.1 Movement of the 2-opt intra-route heuristic

The *3-opt* heuristic chooses three links to eliminate and reconnect for improvement, rather than two as 2-opt heuristics do. Previous research has demonstrated that the power of a 3-opt operation is about similar to three 2-opt operations (Helsgaun, 2000), meaning that the 3-opt heuristic is more efficient and effective than the 2-opt heuristic in identifying the best solution. It should be noted that the 3-opt heuristic is not a simple extension of the 2-opt heuristic: there is only one method to connect any two links picked by a 2-opt heuristic. On the contrary, after picking three connections from a preliminary tour, there are seven methods to reconnect them. A 3-opt heuristic should decide the optimum reconnecting type, making the creation of efficient 3-opt heuristics exceedingly difficult (Figure 3.2).

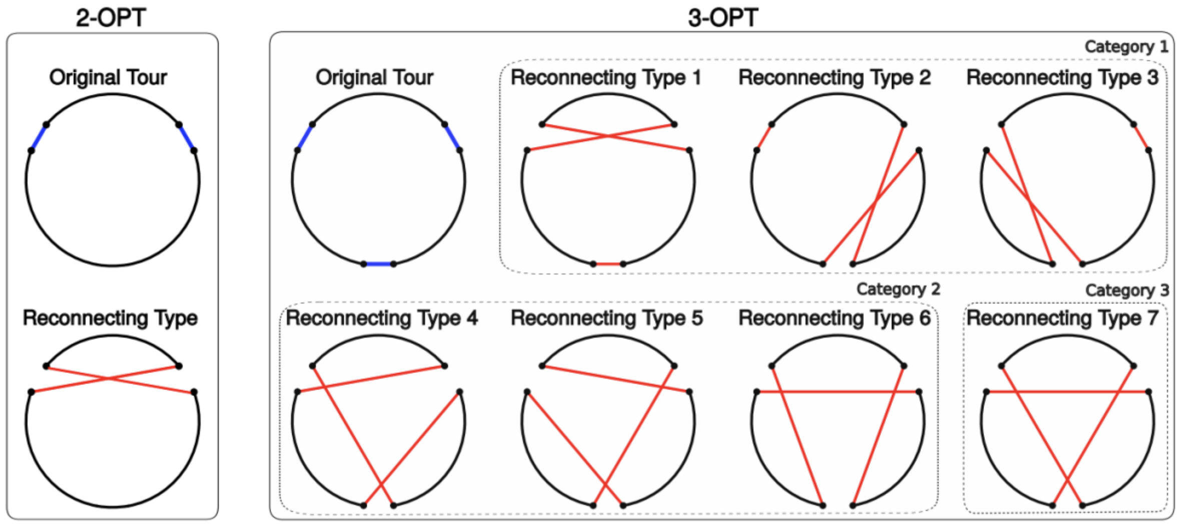

Figure 3.2 Illustration of 2-opt and 3-opt heuristics in use

In intra-move *relocate* a selected customer is removed from its current position in its route and moved to a different position in the same route (Karakostas et al., 2022). Figure 3.3 illustrates an example of the Intra-route Relocate neighbourhood, applied to the pair of customers (5, 4).

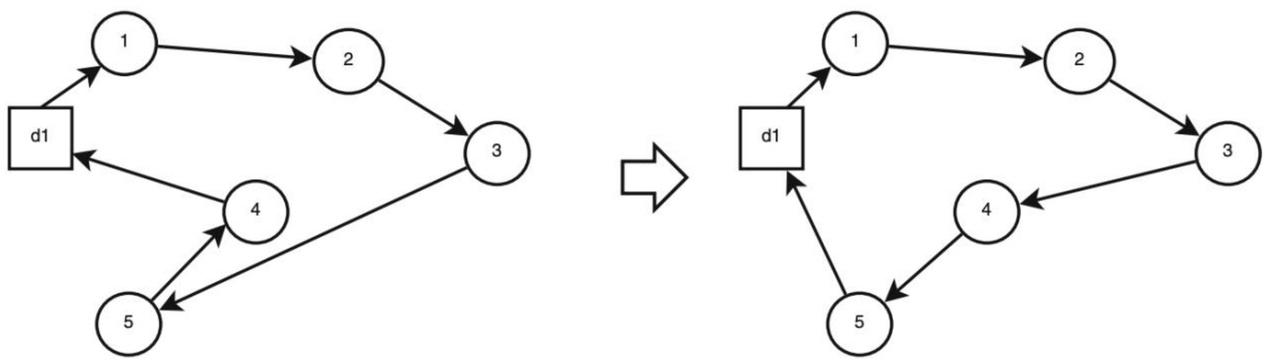

Figure 3.3 Example of the intra-route relocate move

A *k-exchange* move, in general, entails the deletion of (up to) k arcs from the present solution and the production of k new ones to generate the future solution. The complexity of exhaustively examining the k-exchange neighbourhood of a solution is $O(n^k)$, so in practical local search methods, the value of k rarely exceeds 3 or 4, because this would lead to excessive computational times (Toth & Vigo, 2003).

Inter-route heuristics switch one or more customers between routes. The inter-route heuristics applied in this thesis are: 2-opt movement, 1-point move, 2-point move, insertion move, redistribute more, and chain exchange move.

The *inter-route 2-opt* method of refinement is, in fact, an expansion of the 2-opt intra-route heuristic (Tavares et al., 2009). While in the 2-opt intra-route heuristic two edges belonging to the same route are selected randomly, in the 2-opt inter-route heuristic two edges that necessarily belong to distinct routes are selected randomly (Figure 3.4). The result of this exchange can lead to a route whose vehicle capacity is disrespected. The new solution is not evaluated in this scenario.
The stopping criterion selected is reached by finding the minimum route distance from every iteration.

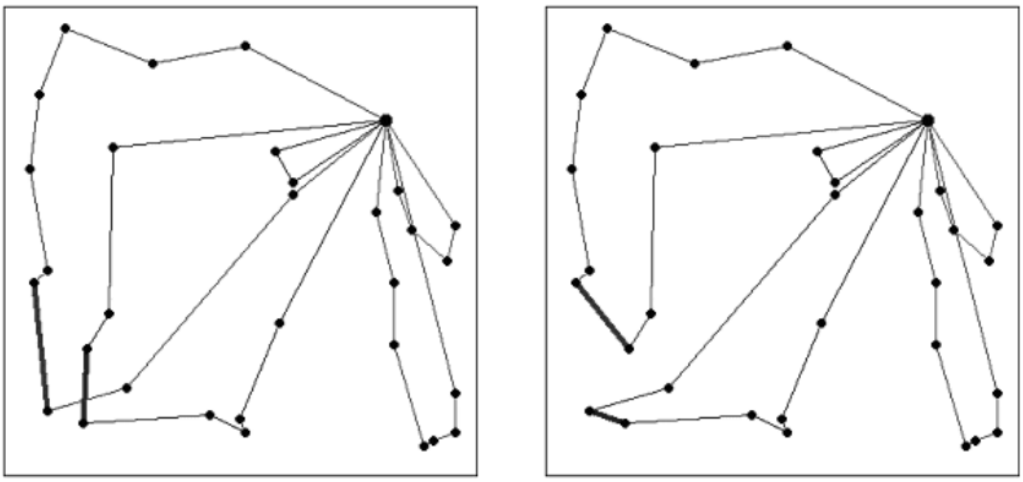

Figure 3.4 Movement of the 2-opt inter-route heuristic

The *1-point* move (Groër et al., 2010) in Figure 3.5 modifies a route in its cost, by moving one point from one route to another. Hence it can be used to obtain good solutions concerning any of the objectives, given that the appropriate direction is adopted. So, when trying to improve the unbalance of the routes, the search direction of the moves carried out by the 1-point move has to be in this direction, and not, for example, toward decreasing the costs.

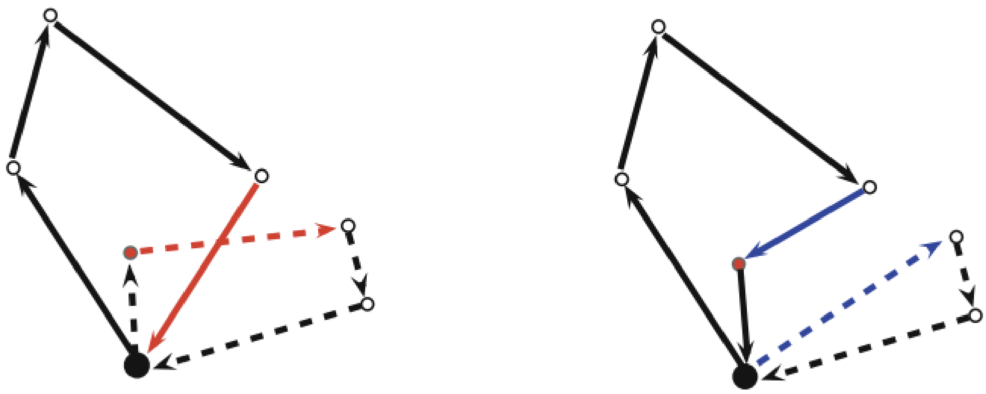

Figure 3.5 1-point move

The *2-point* move is an extension of the intra-route exchange move, but it operates in two routes. This operation is sometimes referred to as “exchange” (e.g., in, Bräysy & Gendreau, 2005; Savelsbergh, 1992) but the term “2-point” was adapted to differentiate it from the intra-route which operates only on one route.

The inter-route *insertion* (Figure 3.6) removes a vertex from its position in a route and reinserts it in a different route (Lei et al., 2012). If the insertion of the vertex in a newly created route is better than an insertion in any of the existing routes, a new route is then initiated with that vertex. This move may delete or create a route of the current solution.

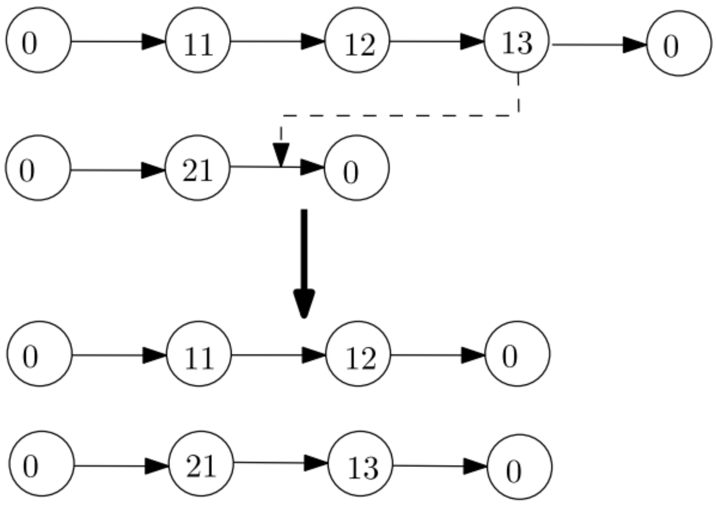

Figure 3.6 Insertion move

With the inter-route *redistribute* move, we try to insert the nodes of the first route into the other routes if possible (Rasku, Kärkkälnen and Musliu, 2019). The insertion order matters, so it is possible to try all permutations of nodes to be inserted and routes to insert to (and return the best).

The *chain exchange* move is found in literature as the "pair" operation (Wren & Holliday, 1972). It involves moving and replacing a node on route 2 with a node on route 1. The replaced node is then inserted on route 3 (if able).

Recharging station operators are critical elements of the methodology. As a result, deleting them or shifting their placements in a route's visit sequence may enhance the solution. As a result, after a set number of repetitions, a Station Removal (SR), followed by a Station Insertion (SI), technique is used. The operators used in this thesis are *Worst-Charge Station Removal* and *Greedy Station Insertion* (Keskin & Çatay, 2016b).

The *Worst-Charge Station Removal* operator utilizes the battery as much as possible before requiring recharging and increases the effective stations' utilization. We propose the elimination of stations that are visited by EVs with significantly high charge levels. The stations are sorted in the non-increasing order of the battery level of the EVs that visit them for recharging, and the stations are deleted starting with the first station in the list.

When the vehicle arrives with a low battery level, the *Greedy Station Insertion* operator detects the first customer along the route and inserts the "cheapest" (shortest distance) station on the arc between that customer and the previous customer. If this insertion fails, the previous arcs are attempted in the same manner.

Chapter 4. Experimental Methodology

In this chapter, we present the setting on which the algorithm was tested, the benchmarks it competed against, as well as the design of the experimental process.

4.1 Experimental Setting

The main objective of the experiment is to evaluate our method's performance against the results of common EVRP benchmarks. The experiment is executed in a set environment with the following characteristics presented below.

Software Environment: Python (Python 3.9.0) on the PyCharm 2022.1.1 IDE. We modified the VeRyPy library to our unique problem specifications (EVRP-PR-MT) and used matplotlib to plot the graph of the routes. Python was selected because of its high readability and the wide range of open-source libraries available.

Hardware Environment: The hardware specifications on which we run the experiment are found in Table 4. Hardware Environment below.

System	MacBook Pro 2019
GPU	Intel Iris Plus Graphics 655 1536 MB
CPU	Intel Core i5 Quad-Core 2.4 GHz
RAM	8 GB
Operating System (OS)	macOS Monterey 12.0.1

Table 4. Hardware Environment

Benchmark Instances: As the EVRP-PR-MT has not been modelled in previous research, there is no set of instances to test our algorithm on. Therefore, we compare our results against the E-CVRP benchmarks created by Mavrovouniotis et al., (2020) for the 2020 IEEE Congress on Evolutionary Computation (CEC). These E-CVRP benchmarks were generated based on the C-VRP benchmarks of Christofides N. et al., (1981); Christofides & Eilon, (1969); Fisher, (1994) and Uchoa et al. (2017). The E-CVRP benchmark set consists of two groups of problems: a) six small-scale instances, b) eighteen large-scale instances, which are presented below in

Table 5 and Table 6. In the aforementioned tables, column “C” refers to each vehicle’s total capacity in transported goods and column “D” represents the vehicle's battery capacity, in terms of driving range. These datasets contain the following basic information which is used as input in our solution method: a) Problem dimension, b) Nodes coordinates, c) Number of stations, d) Maximum vehicle load capacity, e) Maximum driving range (Energy capacity).

Instance name	#Customers	#Stations	#Vehicles	C	D
E-n29-k4-s7	21	7	4	6000	99
E-n30-k3-s7	22	7	3	4500	162
E-n35-k3-s5	29	5	3	4500	138
E-n37-k4-s4	32	4	4	8000	238
E-n60-k5-s9	50	9	5	160	99
F-n49-k4-s4	44	4	4	2010	260

Table 5. Small-scale benchmark set and its characteristics

Instance name	#Customers	#Stations	#Vehicles	C	D
E-n89-k7-s13	75	13	7	220	87
E-n112-k8-s11	100	11	8	200	100
M-n110-k10-s9	100	9	10	200	118
M-n126-k7-s5	120	5	7	200	199
M-n163-k12-s12	150	12	12	200	100
M-n212-k16-s12	199	12	16	200	100
F-n80-k4-s8	71	8	4	30000	53
F-n140-k7-s5	134	5	5	2210	307
X-n147-k7-s4	142	4	7	1190	2762
X-n221-k11-s9	213	7	11	944	1204
X-n360-k40-s9	351	9	40	436	1236
X-n469-k26-s10	458	10	26	1106	1230
X-n577-k30-s4	572	4	30	210	2191
X-n698-k75-s13	684	13	75	408	1336
X-n759-k98-s10	748	10	98	396	1367
X-n830-k171-s11	818	11	171	358	1385
X-n920-k207-s4	915	4	207	33	2773
X-n1006-k43-s5	1000	5	43	131	2536

Table 6. Large-scale benchmark set and its characteristics

4.2 Experimental Design

In the experimental process, we have designed a set of procedures, which will allow us to reach the best possible global solution of our local search framework while minimising the computational time of each instance. This has been done in accordance with the nature of the framework utilised.

In this notion, we noticed that the algorithm sometimes gets stuck on local optima, resulting in a worse global solution. To find a solution around that, we followed the steps shown in Figure 4.1, until there is no new local search strategy to implement. The term "reasonable time" below is a comparison to the computational time required to reach the optimal solution of each instance, according to the benchmarks of Mavrovouniotis et al., (2020). The term "local search strategy" refers to the sequence in which the intra-route, inter-route, RS insertion and RS removal moves are going to be iteratively applied, as seen in Figure 4.2. The total number of strategies is equal to the number of possible combinations between all moves, which is equal to $m!$, in our case 479001600 different strategies. Of course, as seen in Figure 4.1 and Figure 4.2, we don't test all possible scenarios as it would violate the computational time constraint.

Through testing, we reached the outcome of the best working strategy on the basis of solution optimality and computational time, being an initial implementation of the intra- and inter-route moves in alternation with each other, followed by a refinement process, on the produced solution, with a few iterations of separately applying intra- and inter-route moves. The results are shown in the next chapter (Chapter 5).

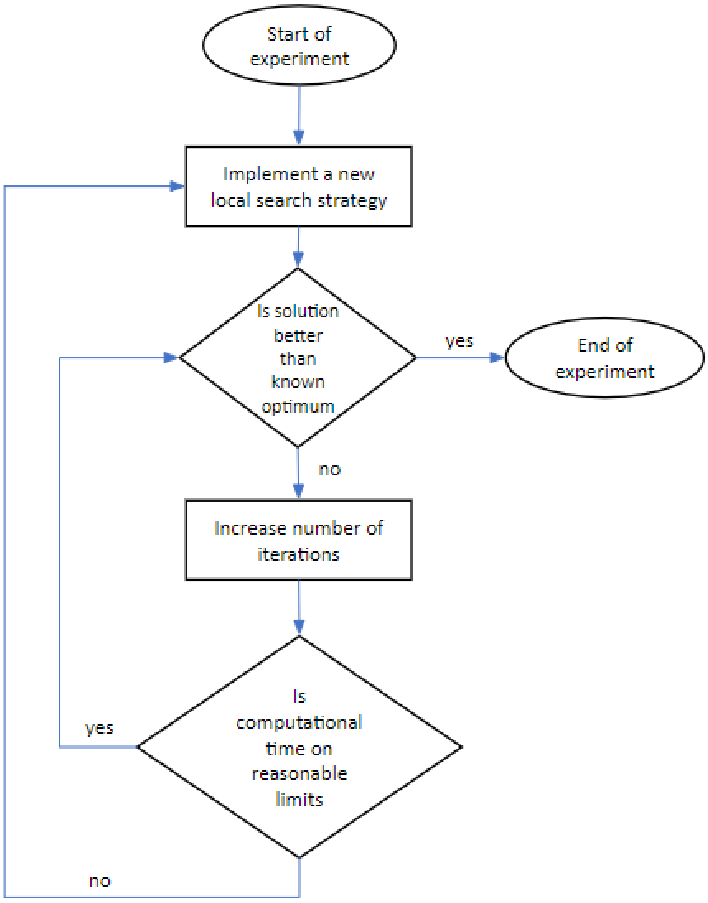

Figure 4.1 Experimental procedure

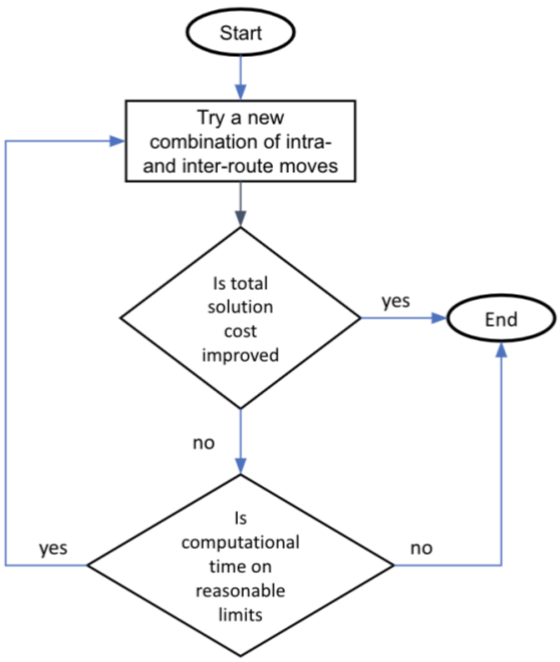

Figure 4.2 New local search strategy search

Chapter 5. Results

In this chapter, we present the results of the experiments executed, while also comparing the latter with the benchmark instances mentioned in Section 4.1.

5.1 Experimental Results

Below in Figure 5.1 to Figure 5.9, we showcase some of the lower dimension instances that were solved. The colour of a route represents a single vehicle. The nodes with a black star act as recharging stations; the depot, node 0, acts as a reload point and a recharging station as well. The problems of smaller dimensions were chosen based on the visual clarity of routes and trips presented, making it easier to comment on and assess them.

On this notion, we can observe a visual attractiveness on the majority of routes, which is considered an important element of route planning when facilitating practical issues and collaborating with transportation industry stakeholders (Rossit et al., 2018). In the majority of instances, we have an enhanced visual attractiveness with cyclical and clearly separated routes. Specifically, the instances E-n37-k4-s4 and F-n49-k4-s4 represented in Figure 5.4 and Figure 5.6 possess both the elements of optimality, in terms of solution quality, and visual attractiveness.

In Figure 5.1, Figure 5.3 and Figure 5.5 we can observe clearly the multi-trip function of our method. Specifically, in Figure 5.1, the dark gold vehicle traverses through nodes 12-15-18-20-17-14 and then returns to the depot to reload and partially recharge to continue on a second trip to node 16. The instance E-n60-k5-s9 which is represented in Figure 5.5, is performing lower than the optimum; the nodes that visually seem like bottlenecks to the improvement phase are 1, 32 and 46. However, integrating them either in the dark green or orange route would lead to a capacity constraint violation.

In Figure 5.7 we can observe an overall cyclicity, but an optimal value is not reached. A few improving points can be seen within each route in the areas where congestions of nodes exist, i.e., in the pink route nodes 12, 13, 16, 17. In these areas, more intra-route iterations are often needed in order to find the optimum sequence of visits. Nevertheless, these improvements lead to minimal enhancement in the solution quality.

On the same note as above, Figure 5.8 showcases a perfect example of intra-route refinement, which is reaffirmed both visually and in regards to solution quality, achieving around 9% better results in total solution cost and subsequently an optimum result.

Last but not least, Figure 5.9 presents an interesting feature of the method developed in this thesis, which is fleet reduction. While the instance X-n147-k7-s4 is generated with 7 vehicles serving 142 customers with a total solution cost of 17704, the LS meta-heuristic proposes a solution which utilizes 6 vehicles and reaches a solution cost with more than 5% better result.

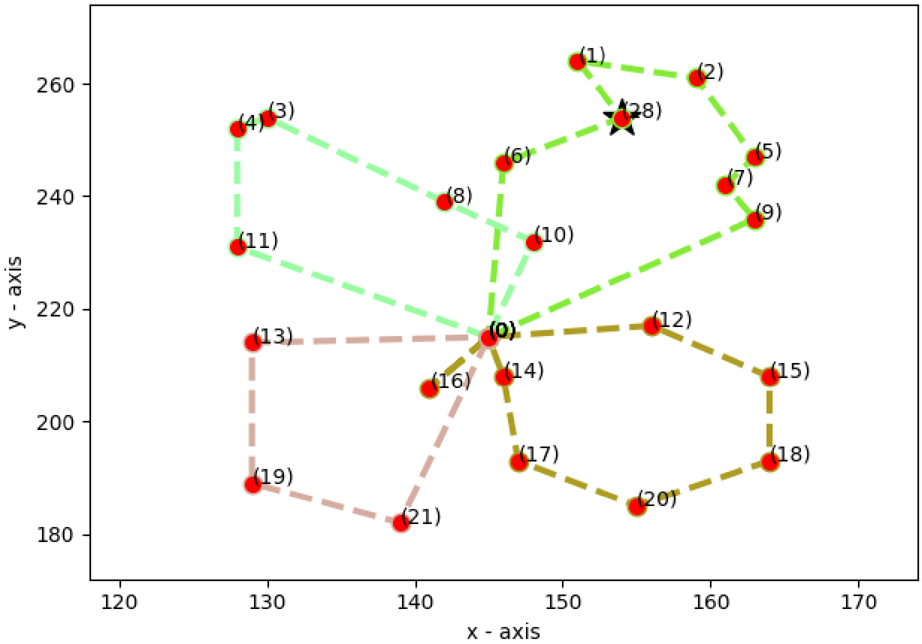

Figure 5.1 Instance E-n29-k4-s7

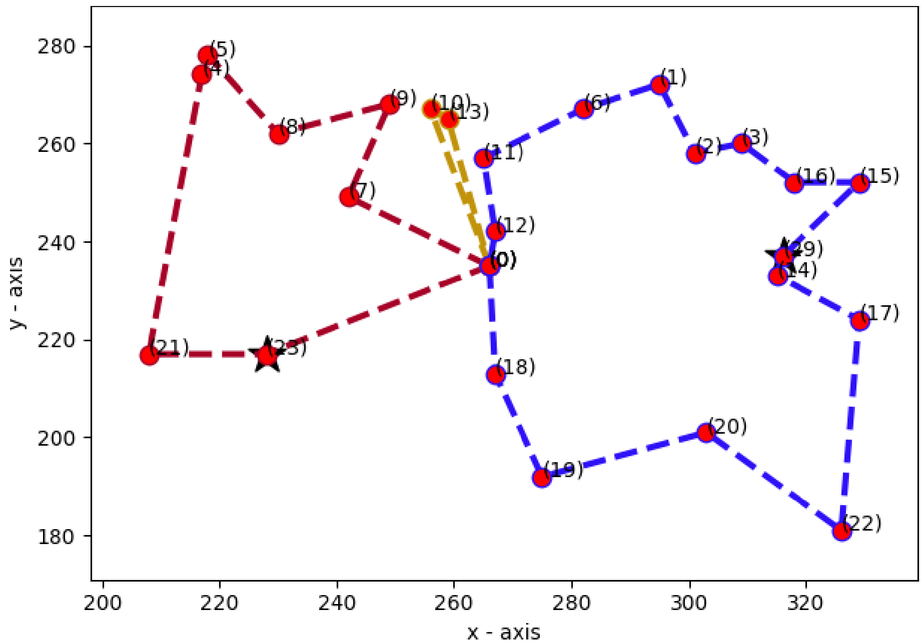

Figure 5.2 Instance E-n30-k3-s7

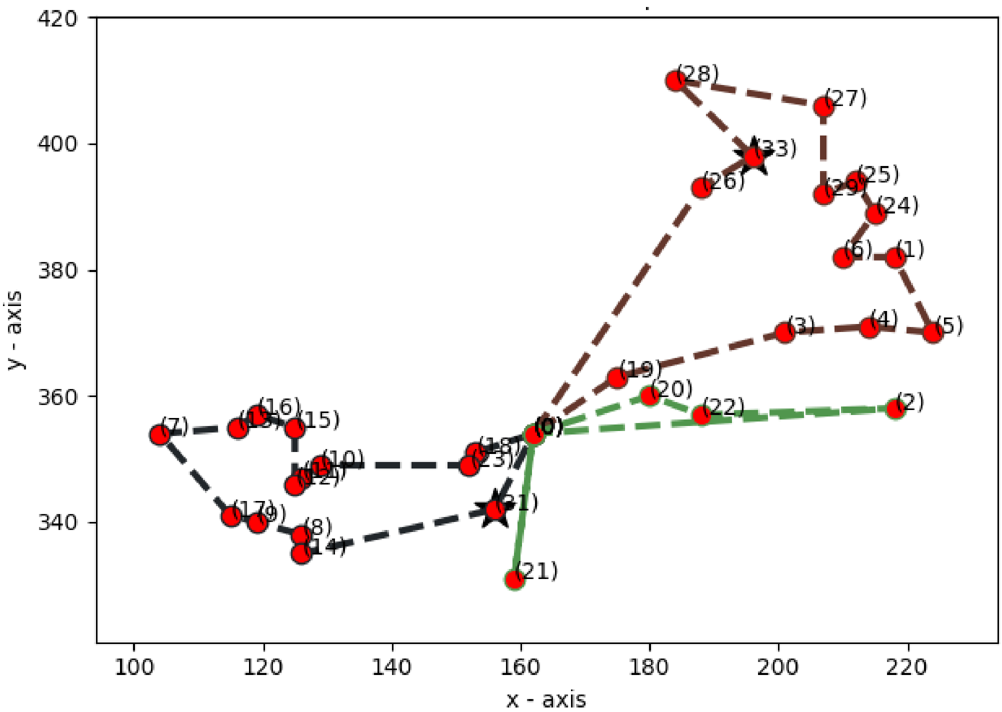

Figure 5.3 Instance E-n35-k3-s5

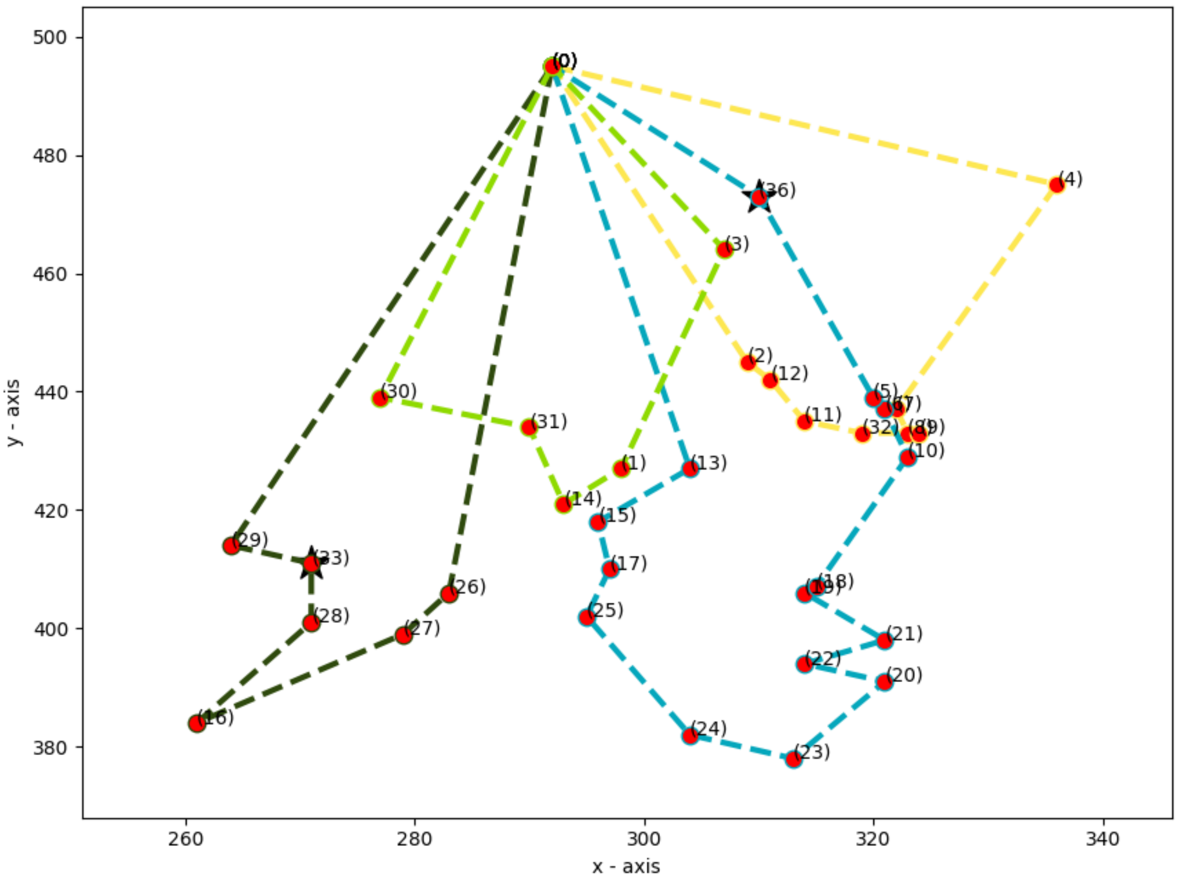

Figure 5.4 Instance E-n37-k4-s4

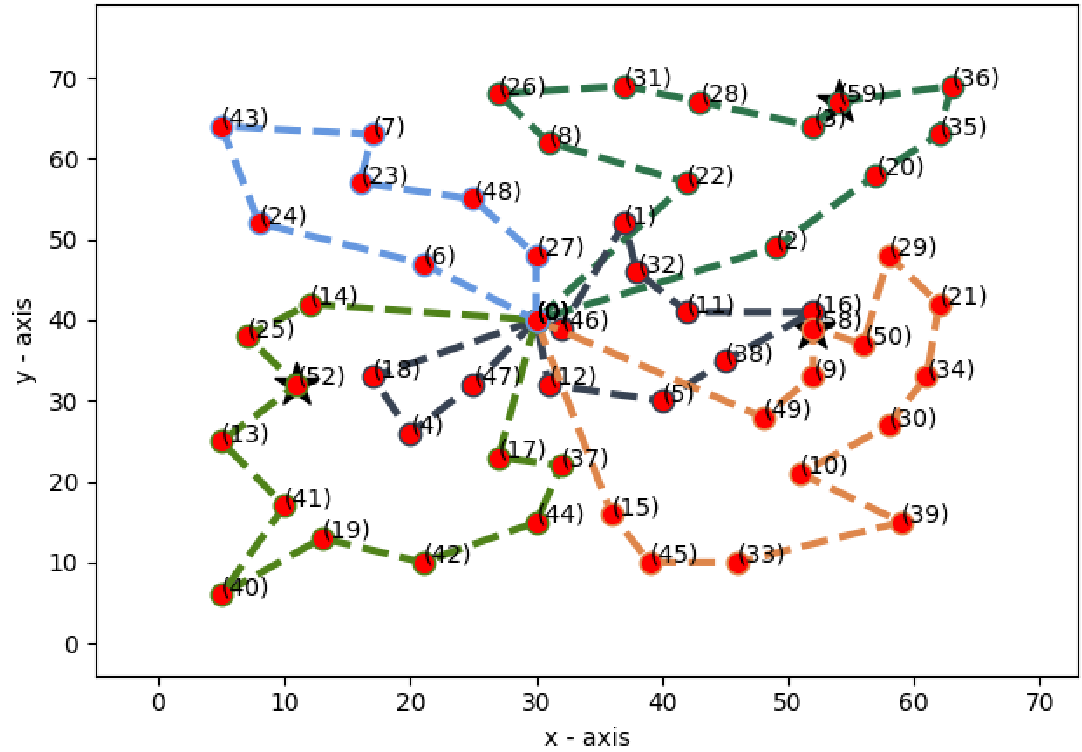

Figure 5.5 Instance E-n60-k5-s9

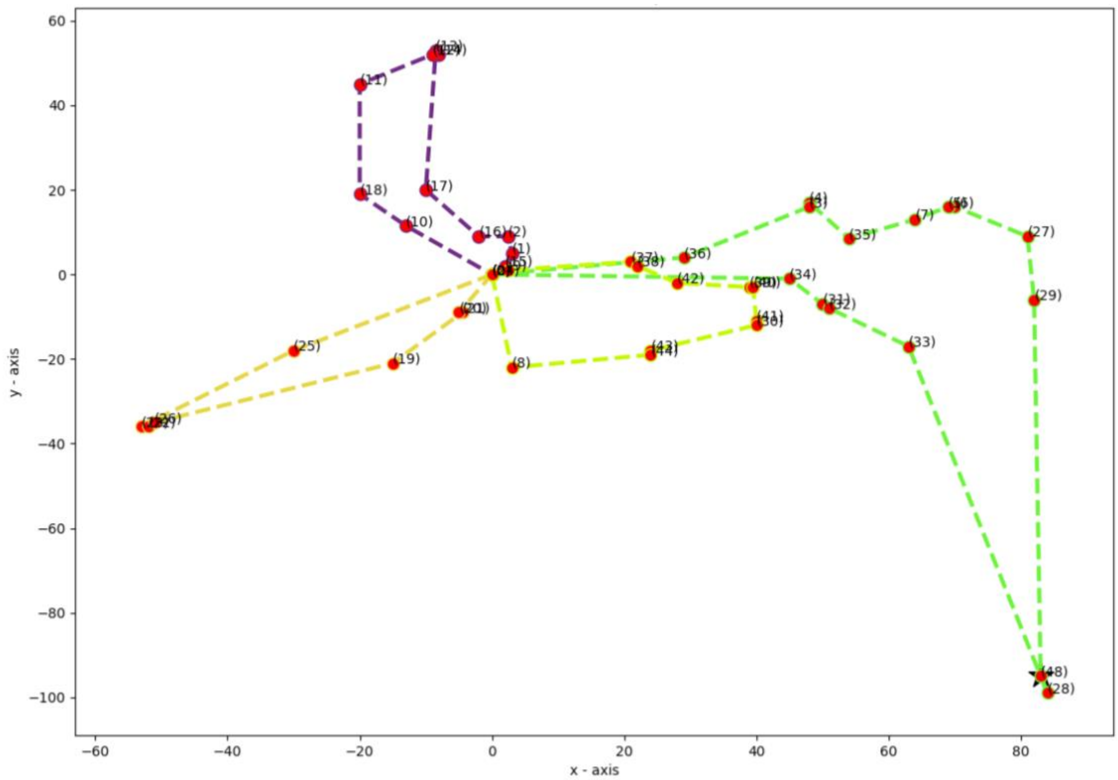

Figure 5.6 Instance F-n49-k4-s4

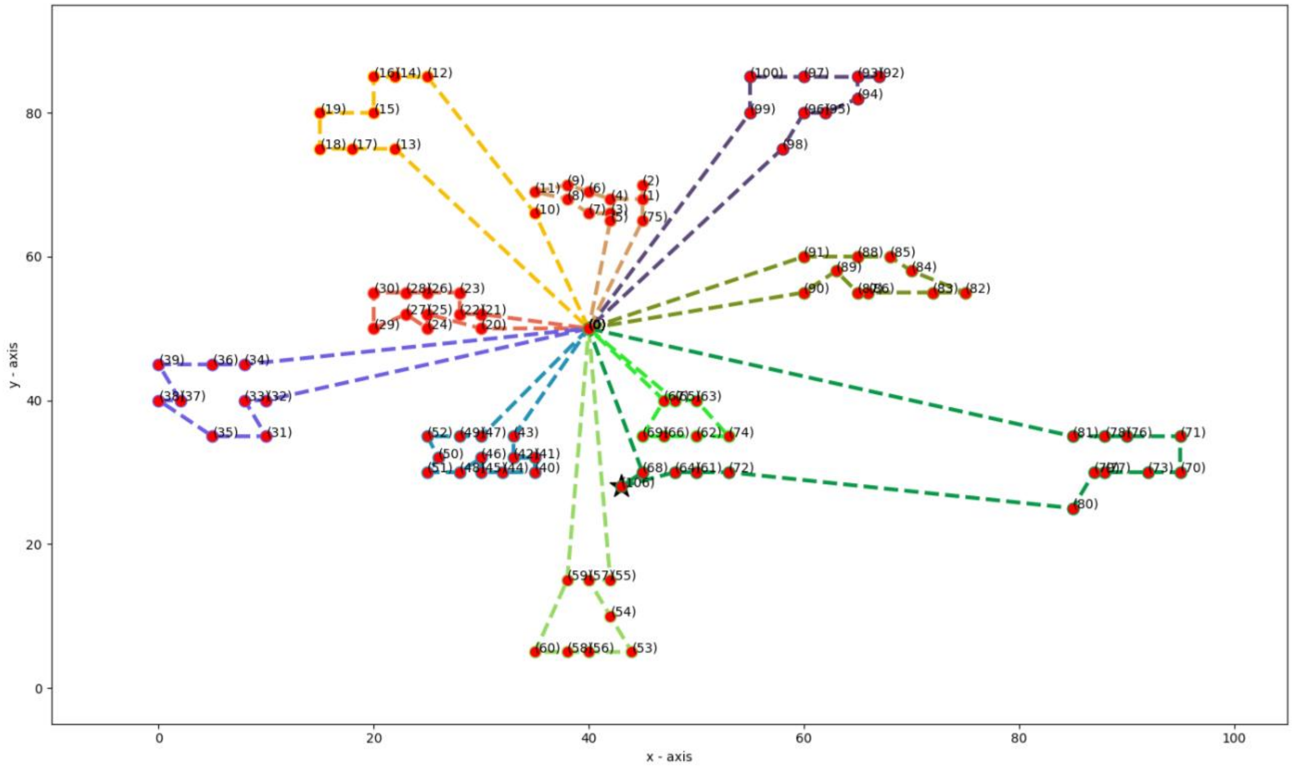

Figure 5.7 Instance F-n80-k4-s8

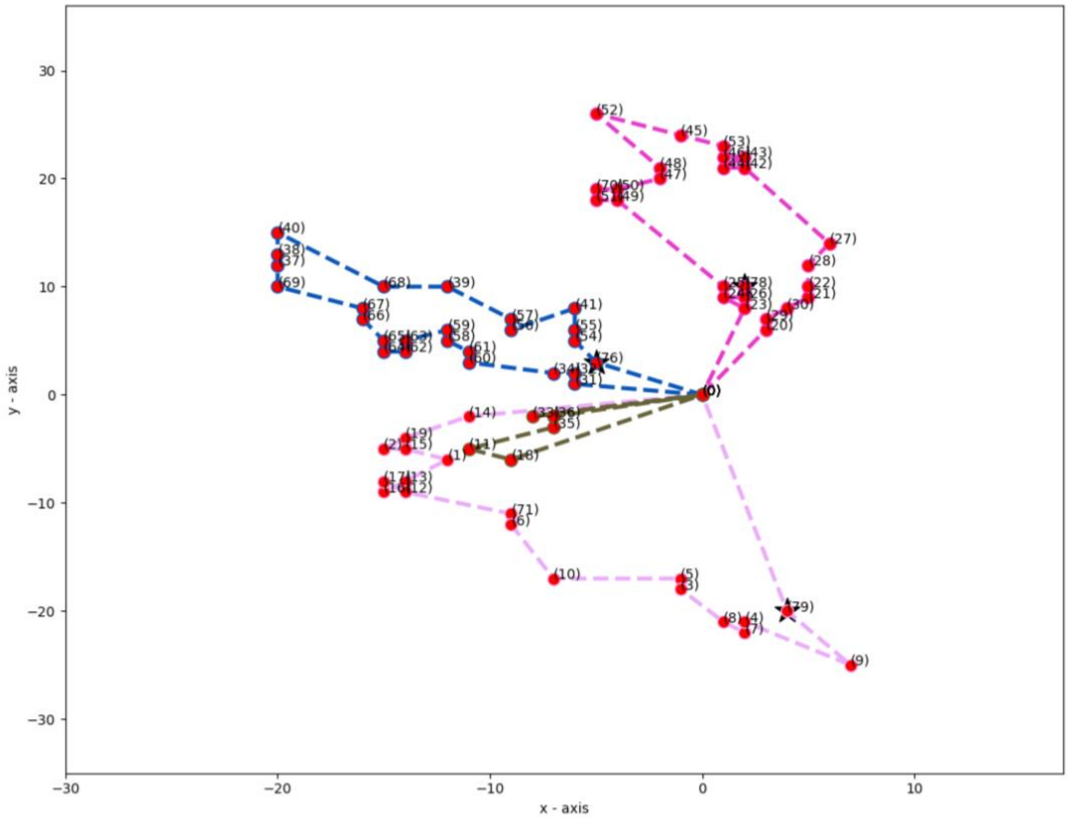

Figure 5.8 Instance M-n110-k10-s9

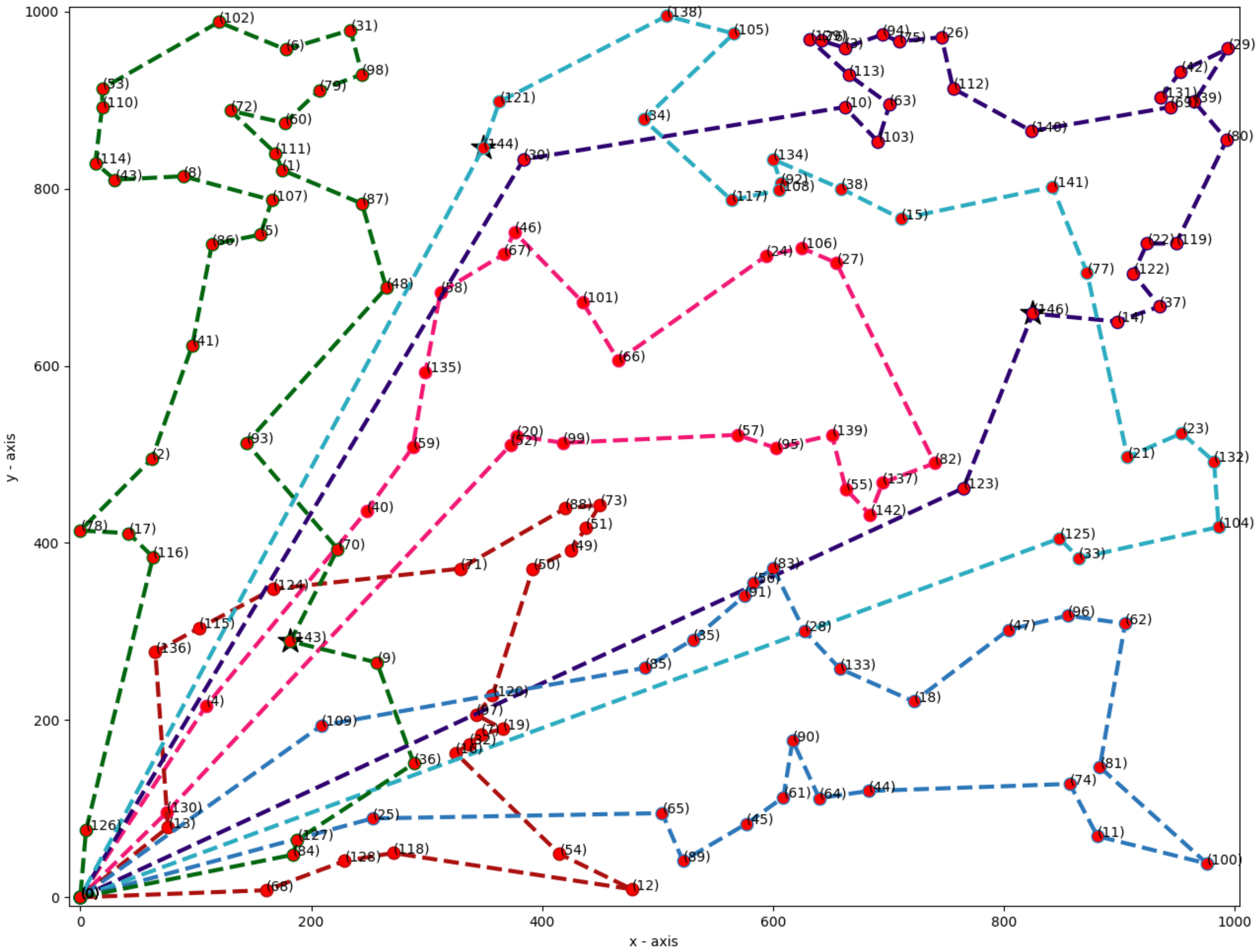

Figure 5.9 Instance X-n147-k7-s4

5.2 Performance Analysis and Discussion

The performance of the local search meta-heuristic against the benchmarks mentioned in Section 4.1 is presented in Table 7 and Table 8, for small- and large-scale problems. In both tables, the “Optimal Cost” and “Optimal Time” columns refer to the performance of the algorithm developed by Mavrovouniotis et al., (2020). The “Optimal Cost” values provided on the benchmark sets in Table 7 and Table 8 are the best-known results generated yet. Mavrovouniotis et al., (2020) proposed two solution methods for the instances; a mixed linear programming approach for the small-scale problem set and an ACO meta-heuristic for both problem sets. The “Optimal Time” values refer to the CPU time (in seconds) required for each instance, on the following experimental setup: Linux System with an Intel Core i7-3930K 3.20GHz processor with 12MB cache and 16GB RAM (Mavrovouniotis et al., 2020).

The “LS Cost” and “CPU Time” columns are referring to the performance of the model developed in this thesis. With the “Gap” column, we present the deviation between the solution obtained by implementing our algorithm and the solution proposed in the work of Mavrovouniotis, in terms of cost and computational time.

In Table 7 we present the experimental results of the runs on the small-scale problem set. In four out of six instances, we have achieved better than the known-optimal solution with an average of 1.7% decrease in the total solution cost. At the same time, the CPU time of the LS method, in all instances, is smaller by 93% on average. Given that our setup is less efficient in terms of computational power, this percentage highlights the efficiency of the method developed in this thesis. On the other hand, in two out of six instances, specifically in E-n29-k4-s7 and E-n60-k5-s9, we haven’t reached the known optimal value, falling behind by 5% approximately. In those instances, the algorithm is not able to provide an optimum solution within the acceptable time frame (see Section 4.3), thus only the solutions obtained within that range are shown.

Instances	Optimal Cost	LS cost	Gap	Optimal Time	CPU Time	Gap
E-n29-k4-s7	383	397	-3.66%	0.1	0.1	0.00%
E-n30-k3-s7	577	570	1.21%	3.1	0.1	96.77%
E-n35-k3-s5	527	520	1.33%	2.2	0.2	90.91%
E-n37-k4-s4	865	845	2.31%	3.4	0.3	91.18%
E-n60-k5-s9	544	579	-6.41%	20.7	1	95.17%
F-n49-k4-s4	740	726	1.89%	8.9	0.7	92.13%

Table 7. Solution Quality comparison for small-scale E-CVRP instances

In Table 8 we present the results of the large-scale instances. Here we can notice an improved efficiency, in terms of solution cost reduction, of the local search meta-heuristic on the larger-dimension problems, achieving higher than optimum results on fourteen out of eighteen instances by an average of 4.5%. In a few instances, the LS method does not reach optimal values, specifically in E-n89-k7-s13, E-n112-k8-s11, M-n163-k12-s12 and F-n80-k4-s8, being on average 2.5% below the optimal values. As with the small-scale problems, here we notice a clear advance in CPU time by an average of 95% lower. On this benchmark set, we decided to keep the iterations at a minimal size, around 1÷10, in order to avoid high computational times. Keeping that in mind, we chose to include worse results in terms of solution cost in Table 8; presenting a significant deviation in computational times, while at the same time achieving optimal values in the majority of instances. Specifically, on instances with over 500 nodes, we ran the local search algorithm for only one iteration; the obtained result highlights the performance of our algorithm on large-scale problems.

Instances	Optimal Cost	LS cost	Gap	Optimal Time	CPU Time	Gap
E-n89-k7-s13	724	743	-2.62%	31.8	1	96.86%
E-n112-k8-s11	860	890	-3.49%	71.8	4.7	93.45%
M-n110-k10-s9	914	832	8.97%	57.6	10.7	81.42%
M-n126-k7-s5	1099	1045	4.91%	63.7	12.5	80.38%
M-n163-k12-s12	1109	1111	-0.18%	158.4	11.8	92.55%
M-n212-k16-s12	1398	1350	3.43%	266.1	17.9	93.27%
F-n80-k4-s8	240	250	-4.17%	23.7	4.3	81.86%
F-n140-k7-s5	1229	1175	4.39%	92.5	21.6	76.65%
X-n147-k7-s4	17704	16745	5.42%	104.5	9.6	90.81%
X-n221-k11-s9	12235	11814	3.44%	161	24.2	84.97%
X-n360-k40-s9	27701	27095	2.19%	1119.4	115	89.73%
X-n469-k26-s10	26881	25988	3.32%	1905.2	84	95.59%
X-n577-k30-s4	55266	52201	5.55%	3182	43	98.65%
X-n698-k75-s13	75048	70899	5.53%	5511.7	84	98.48%
X-n759-k98-s10	84996	79307	6.69%	7258.8	136	98.13%
X-n830-k171-s11	167575	164601	1.77%	6612.6	342	94.83%
X-n920-k207-s4	345214	344246	0.28%	6774.9	378	94.42%
X-n1006-k43-s5	80765	76873	4.82%	8380.6	607	92.76%

Table 8. Solution Quality comparison for large-scale E-CVRP instances

On a general note, we can notice that on both tables we have achieved very low computational times. In order to maximise the ability of our algorithm of achieving better results within this time scale, we investigated different strategies on the instances that perform poorly. During this process, we noticed two possible backlogs.

First, it came to light that due to the nature of the multi-trip modelling, our method often utilises one less route than the currently known optimal, which integrates within a vehicle trip. That happens at the initial construction phase of the solution, which later adheres a computational burden on the optimisation phase to reach a global optimum. Specifically, the efficacy of inter-route heuristics is lower when there are fewer routes to apply them on, and the inter-route moves are responsible for larger changes in the total solution cost.

The second thing we observed was that the computational difficulty of each instance was not increasing exponentially due to an increase in the problem's dimension but due to the increase in the total number of vehicles utilised. Specifically, in instances with a large fleet of vehicles, we had to minimise the iterations in order to stay within a reasonable time frame. Nevertheless, this didn't lead to lower-than-optimum results in most cases.

Chapter 6. Conclusions and Recommendations

In this thesis, we present a new flavour of the vehicle routing problem which calculates minimum distance routes for electric vehicles. The EVRP-PR-MT considers a limited vehicle driving range, cargo load and multiple trips per vehicle while enabling vehicles to partially charge in order to decrease the total route duration. The problem is formulated with linear programming and solved using a local search framework along with custom neighbourhood search heuristics regarding the recharging stations. We tested our results against E-CVRP benchmarks, and the results (Chapter 5) showcase promise in terms of solution approach and quality. In the majority of instances, we managed to obtain better than the known optimum values in proportionally minimum time compared to the ones obtained in the instances run by Mavrovouniotis et al., (2020).

Through many runs on large-scale instances, we noticed that enabling the vehicles to perform multiple trips often minimised the need for at least one vehicle per routing problem, which reaffirms our initial motive of developing a method that could adapt to real-life industry constraints. On the same note, the reason we modified our algorithm to charge partially was to minimise the total route time as much as possible.

Based on the results shown in Chapter 5, we can confidently say that EVRP-PR-MT can be adapted to industry-specific problems, such as last-mile delivery for two important reasons. First, it minimises the fleet of vehicles, which in the context of a third-party logistics (3PL) company transitioning into EVs, would mean less initial capital investment and later, less operating costs. In parallel, keeping in mind the available recharging station grid in most city centres and its growing demand, enabling partial recharging translates to a priori booking of narrow charging slots, which would improve the credibility of EVs as inner city transport vehicles. Of course, it also reduces the total route duration, which gives way to possibly more customer visits in a single shift.

In parallel, during the modelling process of our algorithm, we also developed an additional constraint which limits the route's duration, enabling a possible time-scheduling feature for the drivers of the fleet. However, during the improvement phase of the algorithm, this constraint is often violated to a minimum extent, thus we didn’t include it in the final solution formulation. Nevertheless, it is an important addition which would improve the EVRP-PR-MT competitiveness in the transportation industry applications.

As Goel A. & Gruhn V. (2006) stress, not much attention has been drawn into including the driver’s working hours in the vehicle routing phase. Integrating the driving time, breaks and rest periods, would not only reduce the uncertainties of the routing plan execution, but it will also aid the company in preemptively aligning with driver regulations. Looking at the horizon of Industry 5.0, which puts humans at the centre of the value chain and where driver’s and worker regulations are respected and implemented, the research in the transportation industry should find the aforementioned issue fruitful, and necessary in order to reach a holistically sustainable operational approach.

List of Abbreviations

ACO	Ant Colony Optimisation
AI	Artificial Intelligence
BA	Best Accept strategy
CVRP	Capacitated Vehicle Routing Problem
DARP	Dial a Ride Problem
DP	Dynamic Programming
EBs	Electric Buses
E-CVRP	Electric Capacitated Vehicle Routing Problem
EV	Electric Vehicle
EVRP	Electric Vehicle Routing Problem
EVRP-PR-MT	Electric Vehicle Routing Problem with Partial Recharging and Multiple Trips
EVRPTW	Electric Vehicle Routing Problem with Time Windows
EVRP-TW-MT	Electric Vehicle Routing Problem with Time Windows And Multiple Trips
FA	First Accept strategy
GA-IR	Genetic Algorithm with Improved Recombination
GHG	Greenhouse Gas
GVRP	Green Vehicle Routing Problem
Hybrid VNS	Hybrid Variable Neighbourhood Search
LNG	Liquefied Natural Gas
LS	Local Search
MTHF-EVRP	Multi-Trip and Heterogeneous-Fleet Electric Vehicle Routing Problem
PDP	Pick and Deliver Problem
SI	Station Insertion
SoC	State of Charge
SR	Station Removal
TS	Tabu Search
TSP	Travelling Salesman Problem
VRP	Vehicle Routing Problem

List of Figures

List of Tables

References

Adel, A. (2022). Future of industry 5.0 in society: human-centric solutions, challenges and prospective research areas. In *Journal of Cloud Computing* (Vol. 11, Issue 1). Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1186/s13677-022-00314-5

Albuquerque, F. D. B., Maraqa, M. A., Chowdhury, R., Mauga, T., & Alzard, M. (2020). Greenhouse gas emissions associated with road transport projects: Current status, benchmarking, and assessment tools. *Transportation Research Procedia*, *48*, 2018–2030. https://doi.org/10.1016/j.trpro.2020.08.261

Amir, H., Jean-Charles, C., & Abderrafiãa, K. (2013). From the TSP to the Dynamic VRP: An Application of Neural Networks in Population Based Metaheuristic. In *Metaheuristics for Dynamic Optimization* (Vol. 433, pp. 309–339). http://www.springer.com/series/7092

Babaee Tirkolaee, E., Abbasian, P., Soltani, M., & Ghaffarian, S. A. (2019). Developing an applied algorithm for multi-trip vehicle routing problem with time windows in urban waste collection: A case study. *Waste Management and Research*, *37*(1_suppl), 4–13. https://doi.org/10.1177/0734242X18807001

Baldacci, R., Christofides, N., & Mingozzi, A. (2008). An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming*, *115*(2), 351–385. https://doi.org/10.1007/s10107-007-0178-5

Bell, J. E., & McMullen, P. R. (2004). Ant colony optimization techniques for the vehicle routing problem. *Advanced Engineering Informatics*, *18*(1), 41–48. https://doi.org/10.1016/j.aei.2004.07.001

Bertsimas, D. J., Jaillet, P., & Odoni, A. R. (1990). A Priori Optimization. *Operations Research*, *38*(6), 1019–1033.

Brandão, J. C. S., & Mercer, A. (1998). The multi-trip vehicle routing problem. *Journal of the Operational Research Society*, *49*(8), 799–805. https://doi.org/10.1057/palgrave.jors.2600595

Bräysy, O., & Gendreau, M. (2005). Vehicle routing problem with time windows, Part I: Route construction and local search algorithms. *Transportation Science*, *39*(1), 104–118. https://doi.org/10.1287/trsc.1030.0056

Bullnheimer, B., Hartl, R. F., & Strauss, C. (1999). An improved Ant System algorithm for the Vehicle Routing Problem. *Annals of Operations Research*, *89*, 319–328. https://doi.org/10.1023/a:1018940026670

Cattaruzza, D., Absi, N., Feillet, D., & Vidal, T. (2014). A memetic algorithm for the Multi Trip Vehicle Routing Problem. *European Journal of Operational Research*, *236*(3), 833–848. https://doi.org/10.1016/j.ejor.2013.06.012

Chen, M., Gao, L., Chen, Q., & Liu, Z. (2020). *Dynamic Partial Removal: A Neural Network Heuristic for Large Neighborhood Search*. http://arxiv.org/abs/2005.09330

Christofides, N., & Eilon, S. (1969). An Algorithm for the Vehicle-dispatching Problem. In *Source: OR* (Vol. 20, Issue 3).

Christofides N., Mingozzi A., & Toth P. (1981). Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. *Mathematical Programming*, *20*, 255–282.

Clarke, G., & Wright, J. W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research*, *12*(4), 568–581.

Colorni, A., Dorigo, M., & Maniezzo, V. (1991). Distributed Optimisation by Ant Colonies. *European Conference on Artificial Life*, 134–142.

Corne, I. D., Dorigo, M., & Glover, F. (1999). *MACS-VRPTW: A MULTIPLE ANT COLONY SYSTEM FOR VEHICLE ROUTING PROBLEMS WITH TIME WINDOWS*. http://www.idsia.ch

Cortés-Murcia, D. L., Prodhon, C., & Murat Afsar, H. (2019). The electric vehicle routing problem with time windows, partial recharges and satellite customers. *Transportation Research Part E: Logistics and Transportation Review*, *130*, 184–206. https://doi.org/10.1016/j.tre.2019.08.015

Croes, G. A. (1958). A Method for Solving Traveling-Salesman Problems. *Operations Research*, *6*(6), 791–812.

Dantzig, G. B., & Ramser, J. H. (1959). The Truck Dispatching Problem. In *Source: Management Science* (Vol. 6, Issue 1).

Davis, B. A., & Figliozzi, M. A. (2013). A methodology to evaluate the competitiveness of electric delivery trucks. *Transportation Research Part E: Logistics and Transportation Review*, *49*(1), 8–23. https://doi.org/10.1016/j.tre.2012.07.003

Deb, K. (2011). *Multi-Objective Optimization Using Evolutionary Algorithms: An Introduction*. http://www.iitk.ac.in/kangal/deb.htm

Desaulniers, G., Errico, F., Irnich, S., & Schneider, M. (2016). Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, *64*(6), 1388–1405. https://doi.org/10.1287/opre.2016.1535

Dror, M., & Trudeau, P. (1989). Savings by Split Delivery Routing. *Transportation Science*, *23*(2), 141–145. https://doi.org/10.1287/trsc.23.2.141

Dumas, Y., Desrosiers, J., & Soumis, F. (1991). The pickup and delivery problem with time windows. In *European Journal of Operational Research* (Vol. 54).

Erdoĝan, S., & Miller-Hooks, E. (2012). A Green Vehicle Routing Problem. *Transportation Research Part E: Logistics and Transportation Review*, *48*(1), 100–114. https://doi.org/10.1016/j.tre.2011.08.001

Felipe, Á., Ortuño, M. T., Righini, G., & Tirado, G. (2014). A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. *Transportation Research Part E: Logistics and Transportation Review*, *71*, 111–128. https://doi.org/10.1016/j.tre.2014.09.003

Fermín Cueto, P., Gjeroska, I., Solà Vilalta, A., & Anjos, M. F. (2021). A solution approach for multi-trip vehicle routing problems with time windows, fleet sizing, and depot location. *Networks*, *78*(4), 503–522. https://doi.org/10.1002/net.22028

Fisher, M. L. (1994). Optimal Solution of Vehicle Routing Problems Using Minimum K-Trees Author(s). *Operations Research*, *42*(4), 626–642.

GLOVER, F. (1992). NEW EJECTION CHAIN AND ALTERNATING PATH METHODS FOR TRAVELING SALESMAN PROBLEMS. In *Computer Science and Operations Research* (pp. 491–509). Elsevier. https://doi.org/10.1016/b978-0-08-040806-4.50037-x

Goeke, D., & Schneider, M. (2015). Routing a mixed fleet of electric and conventional vehicles. *European Journal of Operational Research*, *245*(1), 81–99. https://doi.org/10.1016/j.ejor.2015.01.049

Goel, A., & Gruhn, V. (2006). Drivers' working hours in vehicle routing and scheduling. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 1280–1285. https://doi.org/10.1109/itsc.2006.1707399

Grangier, P., Gendreau, M., Lehuédé, F., & Rousseau, L. M. (2016). An adaptive large neighbourhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization. *European Journal of Operational Research*, *254*(1), 80–91. https://doi.org/10.1016/j.ejor.2016.03.040

Groër, C., Golden, B., & Wasil, E. (2010). A library of local search heuristics for the vehicle routing problem. *Mathematical Programming Computation*, *2*(2), 79–101. https://doi.org/10.1007/s12532-010-0013-5

Helsgaun, K. (2000). An effective implementation of the Lin–Kernighan traveling salesman heuristic. *European Journal of Operational Research*, *126*(1), 106–130. www.elsevier.com/locate/dsw

International Energy Agency. (2022). *Global EV Outlook 2022 : Securing supplies for an electric future*. www.iea.org/t&c/

Jackson, W. G., Özcan, E., & John, R. I. (2018). Move acceptance in local search metaheuristics for cross-domain search. *Expert Systems with Applications*, *109*, 131–151. https://doi.org/10.1016/j.eswa.2018.05.006

Karakostas, P., Sifaleras, A., & Georgiadis, M. C. (2022). Variable neighbourhood search-based solution methods for the pollution location-inventory-routing problem. *Optimization Letters*, *16*(1), 211–235. https://doi.org/10.1007/s11590-020-01630-y

Keskin, M., & Çatay, B. (2016a). Partial recharge strategies for the electric vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies*, *65*, 111–127. https://doi.org/10.1016/j.trc.2016.01.013

Keskin, M., & Çatay, B. (2016b). Partial recharge strategies for the electric vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies*, *65*, 111–127. https://doi.org/10.1016/j.trc.2016.01.013

Laporte, G. (2007). What you should know about the vehicle routing problem. *Naval Research Logistics*, *54*(8), 811–819. https://doi.org/10.1002/nav.20261

Laporte, G., Desrochers, M., & Nobert, Y. (1984). Two Exact Algorithms for the Distance-Constrained Vehicle Routing Problem. *Networks*, *14*(1), 161–172.

Laporte, G., & Gendreau, M. (1996). Stochastic vehicle routing. *European Journal of Operational Research*, 3–11.

Lei, H., Laporte, G., & Guo, B. (2012). A generalized variable neighbourhood search heuristic for the capacitated vehicle routing problem with stochastic service times. *TOP*, *20*(1), 99–118. https://doi.org/10.1007/s11750-011-0188-6

Lenstra, J. K., & Kan, A. H. G. R. (1981). Complexity of Vehicle Routing and Scheduling Problems. *Networks*, *11*(2), 221–227.

Lin, J., Zhou, W., & Wolfson, O. (2016). Electric Vehicle Routing Problem. *Transportation Research Procedia*, *12*, 508–521. https://doi.org/10.1016/j.trpro.2016.02.007

Lin, S., & Kernighan, B. W. (1973). An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations Research*, *21*(2), 498–516.

Mahmoud, A., Chouaki, T., Hörl, S., & Puchinger, J. (2022). Extending JSprit to solve electric vehicle routing problems with recharging. *Procedia Computer Science*, *201*(C), 289–295. https://doi.org/10.1016/j.procs.2022.03.039

Mandi, J., Canoy, R., Bucarey, V., & Guns, T. (2021). *Data Driven VRP: A Neural Network Model to Learn Hidden Preferences for VRP*. http://arxiv.org/abs/2108.04578

Mavrovouniotis, M., Menelaou, C., Timotheou, S., Ellinas, G., Panayiotou, C., & Polycarpou, M. (2020). A Benchmark Test Suite for the Electric Capacitated Vehicle Routing Problem. *IEEE Congress on Evolutionary Computation*. https://doi.org/10.1109/CEC48606.2020.9185753

Mester, D., & Bräysy, O. (2007). Active-guided evolution strategies for large-scale capacitated vehicle routing problems. *Computers and Operations Research*, *34*(10), 2964–2975. https://doi.org/10.1016/j.cor.2005.11.006

Nazari, M., Oroojlooy, A., Snyder, L. v., & Takáč, M. (2018). *Reinforcement Learning for Solving the Vehicle Routing Problem*. http://arxiv.org/abs/1802.04240

Or, I. (1976). *Traveling Salesman-Type combinatorial problems and their relation to the logistics of regional blood banking*. Universitäts- und Stadtbibliothek Köln.

Osman, I. H. (1993). Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, *41*, 421–451.

Petch, R. J., & Salhi, S. (2003). A multi-phase constructive heuristic for the vehicle routing problem with multiple trips. *Discrete Applied Mathematics*, *133*(1–3), 69–92. https://doi.org/10.1016/S0166-218X(03)00434-7

Psaraftis, H. N. (1980). DYNAMIC PROGRAMMING SOLUTION TO THE SINGLE VEHICLE MANY-TO-MANY IMMEDIATE REQUEST DIAL-A-RIDE PROBLEM. *Transportation Science*, *14*(2), 130–154. https://doi.org/10.1287/trsc.14.2.130

Rasku, J., Kärkkäinen, T., & Musliu, N. (2019). *Meta-Survey and Implementations of Classical Capacitated Vehicle Routing Heuristics with Reproduced Results*. University of Jyväskylä.

Robert A. Russell. (1977). An Effective Heuristic for the M-Tour Traveling Salesman Problem with Some Side Conditions. *Operations Research*, *25*(3), 517–524.

Rossit, D. G., Vigo, D., Tohmé, F., & Frutos, M. (2018). Visual Attractiveness in Routing Problems: a Review. *Computers & Operations Research*, *103*, 13–34.

Russell, R. A., & Urban, T. L. (2008). Vehicle routing with soft time windows and Erlang travel times. *Journal of the Operational Research Society*, *59*(9), 1220–1228. https://doi.org/10.1057/palgrave.jors.2602465

Sanghavi, S., Hajek, B., & Massoulie, L. (2007). Gossiping with multiple messages. *IEEE Transactions on Information Theory*, *53*(12), 4640–4654. https://doi.org/10.1109/TIT.2007.909171

Savelsbergh, M. (1992). The vehicle routing problem with time windows: minimizing route duration. *ORSA Journal on Computing*, *4*(2), 146–154. https://doi.org/10.1287/ijoc.4.2.146

Schneider, M., Stenger, A., & Goeke, D. (2014). The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*, *48*(4), 500–520. https://doi.org/10.1287/trsc.2013.0490

Scott, C., Lundgren, H., & Thompson, P. (2011). *Management for Professionals Guide to Supply Chain Management*. https://doi.org/https://doi.org/10.1007/978-3-319-77185-4

Stein, D. M. (1978). SCHEDULING DIAL-A-RIDE TRANSPORTATION SYSTEMS. *Transp Sci*, *12*(3), 232–249. https://doi.org/10.1287/trsc.12.3.232

Taillard, É., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J.-Y. (1997). A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science* , *31*(2).

Tavares, L. G., Lopes, H. S., & Lima, C. R. E. (2009). Construction and improvement heuristics applied to the capacitated vehicle routing problem. *2009 World Congress on Nature and Biologically Inspired Computing, NABIC 2009 - Proceedings*, 690–695. https://doi.org/10.1109/NABIC.2009.5393467

Toth, P., & Vigo, D. (2003). The granular tabu search and its application to the vehicle-routing problem. *INFORMS Journal on Computing*, *15*(4), 333–346. https://doi.org/10.1287/ijoc.15.4.333.24890

Touati-Moungla, N., & Jost, V. (2011). Combinatorial optimization for electric vehicles management. *Renewable Energy and Power Quality Journal*, *1*(9), 942–947. https://doi.org/10.24084/repqj09.504

Uchoa, E., Pecin, D., Pessoa, A., Poggi, M., Vidal, T., & Subramanian, A. (2017). New benchmark instances for the Capacitated Vehicle Routing Problem. *European Journal of Operational Research, 257*(3), 845–858. https://doi.org/10.1016/j.ejor.2016.08.012

van Hoorn, J. J. (2016). *Dynamic Programming for Routing and Scheduling: Optimizing Sequences of Decisions.* Vrije Universiteit Amsterdam.

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2014). A unified solution framework for multi-attribute vehicle routing problems. *European Journal of Operational Research*, *234*(3), 658–673. https://doi.org/10.1016/j.ejor.2013.09.045

Wang, W., Ding, D., & Cao, S. (2022). Hybrid Variable Neighborhood Search algorithm for the Multi-trip and Heterogeneous-fleet Electric Vehicle Routing Problem. *Xitong Fangzhen Xuebao / Journal of System Simulation*, *34*(4), 910–919. https://doi.org/10.16182/j.issn1004731x.joss.21-1133

Waraich, R. A. ;, Galus, M. D. ;, Dobler, C. ;, Balmer, M. ;, Andersson, G. ;, & Axhausen, K. W. (2009). *Plug-in Hybrid Electric Vehicles and Smart Grid: Investigations Based on a Micro-Simulation*. https://doi.org/10.3929/ethz-a-005916811

Wren, A., & Holliday, A. (1972). Computer Scheduling of Vehicles from One or More Depots to a Number of Delivery Points. *Operational Research*, *23*(3), 333–344. www.jstor.org

Yellow, P. C. (1970). A Computational Modification to the Savings Method of Vehicle Scheduling. *Journal of the Operational Research Society*, *21*(2), 281–283.

Zhang, J., & Zhang, X. (2022). A multi-trip electric bus routing model considering equity during short-notice evacuations. *Transportation Research Part D: Transport and Environment*, *110*. https://doi.org/10.1016/j.trd.2022.103397

Zhang, S., Chen, M., Zhang, W., & Zhuang, X. (2020). Fuzzy optimization model for electric vehicle routing problem with time windows and recharging stations. *Expert Systems with Applications*, *145*. https://doi.org/10.1016/j.eswa.2019.113123

Zhao, M., & Lu, Y. (2019). A heuristic approach for a real-world electric vehicle routing problem. *Algorithms*, *12*(2). https://doi.org/10.3390/a12020045

Zhao, Z., Li, X., & Zhou, X. (2020). Distribution route optimization for electric vehicles in urban cold chain logistics for fresh products under time-varying traffic conditions. *Mathematical Problems in Engineering, 2020*. https://doi.org/10.1155/2020/9864935