A Bi-Objective Heterogeneous Open Vehicle Routing Problem

Laurens Papadopoulos

(2640057)

August 23, 2023

Thesis MSc Business Analytics - Computational Intelligence

Thesis committee:

Prof. dr. Rob van der Mei (supervisor)

Dr. Joost Berkhout (co-reader)

Prof. dr. Elenna Dugundji (CWI supervisor)

Abstract

This thesis delves into the reverse logistics strategies of the CIRCULAR FOAM project, targeting the sustainable circularity of high-performance plastics, rigid polyurethane foam. A unique bi-objective model is introduced, emphasizing the integration of environmental benefits from utilizing an electrical fleet and its subsequent cost implications. Our methodology consists of two main pillars: (i) A bi-objective optimization model, offering a balance between transportation costs and emission, with the Pareto efficient set derived using the ε-constraint method, and (ii) The Adaptive Large Neighbourhood Search (ALNS) that, enhanced by novel operators tailored for a heterogeneous fleet, aims for efficient problem-solving and minimizes solution biases. The model uniquely allows emissions to depend on dynamic loads, a feature not evident in prior studies. The model indicates that with the utilization of an electrical fleet, emission can be significantly reduced within the reverse logistics of the CIRCULAR FOAM project. It has been seen that the ALNS can swiftly find near-optimal solutions, addressing the longstanding challenge of scalability. Lastly, the obtained Pareto front shows the different trade-offs between transportation expenses and emissions, offering invaluable insights for CIRCULAR FOAM stakeholders.

Keywords: OR, OVRP, VRP, Reverse logistics, Electric vehicles/trucks, ALNS, MILP, Pareto-Efficient, ε-constraints, CIRCULAR FOAM, Recycling, Circularity, Sustainability.

Contents

1 Introduction

Europe aiming to become the world's first climate-neutral continent by 2050. Achieving this ambitious goal required a rapid and comprehensive transformation of sociotechnical systems, integrating various elements such as technologies, infrastructures, organizations, markets, regulations, and user practices. One promising approach to this transformation was the promotion of well-functioning, sustainable Industrial-Urban Symbiosis of circular value chains. This concept involved engaging all relevant stakeholders in a collaborative effort to create a climate-neutral, resource-efficient, and competitive economy. To make this vision a reality, there were significant coordination challenges that needed to be addressed among different actors. Institutions also required restructuring to break free from harmful path dependencies and align human institutions with the Earth system.

In the heart of this endeavor, the European chemical industry faced multiple pressures. It struggled with feedstock and energy costs compared to other regions, while simultaneously facing mounting expectations to reduce emissions and fossil-based resource usage even further. Innovations were sought to push for further reductions in CO2 emissions and to transform waste and emissions into valuable feedstocks. Chemical recycling emerged as a promising opportunity to reduce raw material imports, mitigate CO2 emissions, and decrease waste. However, cost competitiveness and the need for regulatory adjustments posed significant challenges to the practical implementation of circular value chains and new technologies throughout Europe.

One specific area that demanded attention was the management of plastic waste, which had been steadily increasing with global plastic production. Polyurethane, as a high-performance polymer, presented both challenges and opportunities. Its widespread use in refrigeration appliances and construction meant that significant waste streams needed to be managed responsibly and sustainably. The introduction of circular economy principles was underway, but it required holistic solutions, interdisciplinary collaboration, and changes in economic models and social structures to achieve widespread practical implementation.

The CIRCULAR FOAM project is aimed to develop a comprehensive blueprint for circular regions. Its primary objective is to establish a large-scale, and sustainable systemic solution for achieving circularity in high performance plastics from different applications. The project is specifically interested in waste streams coming from rigid polyurethane (PU) foams, which have not been adequately recycled yet (Demharter, 1998). Rigid polyurethane is used as insulation material in construction elements and refrigerators. Through chemical upcycling, the waste stream will be valorized, generating new virgin-equivalent feedstocks for the chemical industry. This process will enable to generate new high-performance plastics and hence will reduce the dependency on finite fossil-based sources while simultaneously boosting the use of renewable-based alternatives. Therefore the project not only addresses the issue of collecting all different waste streams, but mainly contributes to chemical processes that are able to produce new sustainble materials. At the end, CIRCULAR FOAM aims to integerate the different element of the circular value chain into one cohesive and optimized system. To assure optimal provision of waste-based resources, a robust reverse logistic framework will be developed, that will delegate the accumulation of waste-based resources from several sources. Consequently, this system is efficiently making use of resources and minimizing the environmental impact.

The resulting systematic approach are to be implemented in thee pilot regions: North Rhine-Westphalia (Germany), Silesia (Poland) and Greater Amsterdam region (Netherlands). The results in these pilot regions will be used as a blueprint for future replication of the circular framework across Europe. Within the scope of this research, the domain of reverse logistics will be investigated. Reverse logistics is "the process whereby companies can become more environmentally efficient through recycling, reusing, and reducing the amount of materials used" (Carter and Ellram, 1998). As defined by the Council of Logistics Management (1999), reverse logistics is the movement of material from the point of consumption toward the point of origin. This research aims to investigate and propose effective strategies for the reverse logistics of the CIRCULAR FOAM project, thereby contributing to the project's goals of creating a more sustainable solution for achieving circularity in high-performance plastics, such as rigid polyurethane foam.

The remainder of in this thesis is organised as follows: In Section 2 the problem of this thesis is described in detail and the corresponding research questions are stated. Section 3 provides a review of the literature related to OVRP. In Section 4 we formulate the mathematical optimisation model and give our solution approach of the ALNS algorithm. Section 5 presents the computational experiments and results. Lastly, conclusions are made and possible future research is discussed in Section 6.

2 Problem Description

The Circular-Foam project has a large distribution network. Generalized it starts with a supply of PU-foam from different type of sites in the region of Amsterdam. Then there exist intermediate facilities, where processes as sorting take place. Lastly, there is the chemical recycling plant, Covestro, located in the North Rhine-Westphalia (Germany).

2.1 Supply of PU Foam

The main supply streams we consider in this research originate from the following 3 type of sites, namely the factory UNILIN and large and residential construction sites. For each of these sites the specifics of the supply stream are explained below.

2.1.1 Factory

UNILIN is Europe's second largest manufacturer of PU insulation boards. PU insulation boards with recycled content will replace the existing boards from virgin raw materials. The size of supply from UNILIN can be assumed to be fixed for a re-occurring period as a factory has a production capacity and it can be assumed that a factory utilizes its full capacity.

In the process of manufacturing PU insulation boards, large pieces of PU foam have to be cut down into smaller parts and this often leads to residual waste. Furthermore, to produce neat PU insulation boards, edges also have to be cut very precisely, which leads to small pieces of extra residual waste. At the end of the process, all collected residual waste is compressed into highly-dense briquettes by special grinding machinery, which UNILIN has access to at their facility. Since, UNILIN can produce these briquettes themselves, they can directly transport their PU waste to the chemical recycling plant and are in no need of intermediate processing facilities.

2.1.2 Large Construction Sites

The second stream of supply comes from large construction sites, think of industrial areas or infrastructure. From these construction sites PU foam is collected from their waste of raw materials. An assumption can be made that there are special grindings machine on site at the constructions sites. Thus the PU foam can be compressed into highly-dense briquettes. Since we assume that briquettes can be made at the construction sites, the supply coming from these construction sites can also be transported directly to the chemical recycling plant, Covestro, and are also not in need of intermediate processing facilities.

2.1.3 Residential Construction Sites

The third and last stream of supply comes from residential construction sites. These can be seen as relatively small construction sites. Therefore we can assume that these sites do not contain special grinding machinery at plain site. Additionally, we can assume that the waste coming from these sites are not from raw material, thus the PU foam is mixed together with other types of material, such as wood or plastics. This stream of supply is therefore in need of intermediate processing facilities to first sort and compress the PU foam into highly-dense briquettes, before transporting the waste to the chemical recycling plant, unlike the 2 previously mentioned type of sites. In the Greater Amsterdam region, the government regulated waste stations serve as the intermediate processing facilities.

2.2 Problem Statement

Supply is generated by the three parties explained above. Each of these parties have their own routing options as discussed. At the end of the supply chain lies Covestro. Covestro is supplied by the direct supply streams coming from UNILIN and large construction sites plus the supply streams from the intermediate sorting center.

In this case an assumption can be made that for both UNILIN and large construction sites, separately, multiple large trucks can be loaded to leave directly for Covestro. It is assumed that supply streams coming from UNILIN and large construction sites are sufficiently large enough to fill up truck at its capacity. Furthermore, do not need any kind of sorting/processing of its respective waste. Resulting in a route where only a single stop has to be taken, namely the final destination, the chemical power plant, Covestro. Thereby, the supply streams respectively from UNILIN and large construction sites will not be included in this research's logistic network. For the supply stream coming from the small residential construction sites it can be assumed that the volume of waste from a single site is insufficient to fully load a truck, and in addition is in need of a processing facility to sort out the rigid polyurethane. It needs to be sorted, as it is assumed that all waste coming from construction is mixed.

In this research the goal is to provide a routing plan for the accumulation of the waste coming from the residential construction sites, which then need to be transported to intermediate processing facilities. The supply chain is organised as follows: Each residential construction site produces a supply of construction waste. Every route starts at a depot, where all trucks are stored. A truck leaves from a depot to one of the residential construction sites. Thus, a truck will never directly travel from a depot to a waste station. Once a truck has visited a construction site it can only visit other construction sites or a waste station. The route of a truck ends when it has reached a waste station, where the carried supply will be processed. Hence, it cannot return to the depot. Each construction site is only visited once, therefore trucks who do not have enough space left to collect all waste from a construction site, are unable to visit that specific construction site. An example of such a supply chain is shown in Figure 2.1.

This problem can be formulated as an Open Vehicle Routing Problem (OVRP), a variant of the "classical" Vehicle Routing Problem (VRP). In the OVRP, the problem is "open" because the the vehicles are not required to return to the depot after completing their service (Salari et al., 2010). The OVRP is a combinatorial optimization problem, where the optimal set of routes for a fleet of vehicles is to be determined, while satisfying various constraints and considering objectives. The goal is to minimize a designed objective function, considering factors as vehicle capacities, travel distances and other operational constraints.

The OVRP is a complex problem due to its combinatorial nature, as the number of possible routes grows exponentially with the number of locations that have to be visited and the problem becomes increasingly difficult as the constraints increases. The OVRP is known to be NP-hard in the strong sense, as it generalizes the Bin Packing Problem and the Hamiltonian Path Problem Salari et al., 2010. Determining the exact optimal solution for the OVRP is a time-consuming task and is inherently unscalable, due to its NP-hard nature. Thus, making it infeasible to exactly solve larger problem instances. In this research a heterogeneous fleet is introduced consisting of

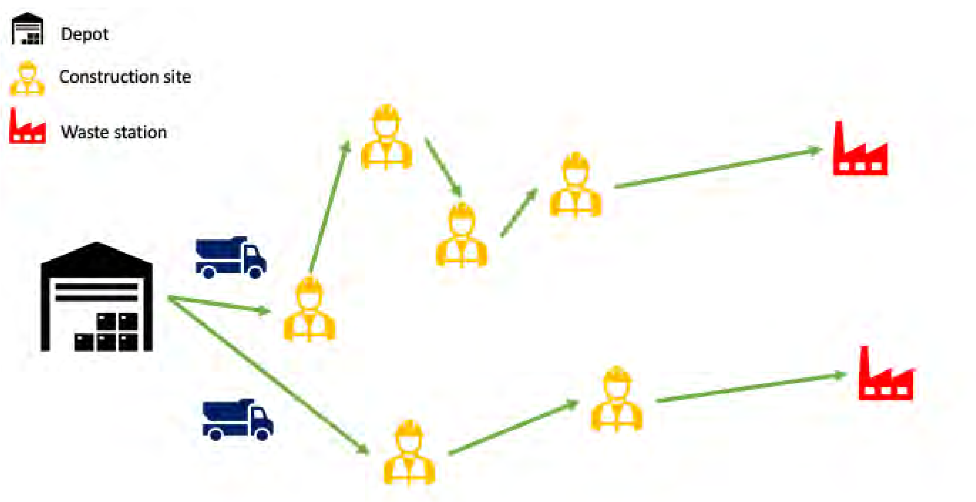


Figure 2.1: Example of supply chain

diesel and electric trucks. Electric trucks introduce a new constraint to the problem, namely a limited travel distance, thereby it becomes an even more highly constraint problem. Contrary to electric trucks, diesel trucks are assumed to have no limit. However, it does not make the problem unsolvable. Using approximation algorithms and heuristics near-optimal solutions can be found within a reasonable amount of time (Hromkovič, 2013). Pisinger and Ropke (2007) presented a unified heuristic which is able to solve five different variants of the vehicle routing problem: the vehicle routing problem with time windows (VRPTW), the capacitated vehicle routing problem (CVRP), the multi-depot vehicle routing problem (MDVRP), the site-dependent vehicle routing problem (SDVRP) and the open vehicle routing problem (OVRP).

In this thesis, we propose a bi-objective model to integrate the impact of using an greener, electrical fleet on the costs and emissions of the routing problem of the CIRCULAR FOAM project. Since the goal of CIRCULAR FOAM is to create a more sustainable infrastructure. The two primary contributions of this thesis are as follows: (i) A bi-objective optimisation model is proposed to provide a trade-off between costs and environmental impacts; (ii) Adaptive Large Neighbourhood Search (ALNS) is proposed to solve each of the two objectives within a reasonable amount of time. New operators are used that differ from the ones in literature as they are based on the exploration of the heterogeneous fleet, to prevent bias in the solutions.

3 Related Literature

The problem in this thesis is a Bi-Objective Integer Linear Program (BOILP). Therefore, Subsection 3.1 starts of with a discussion on literature on multi-objective optimization problems. Then, related problems on VRP are discussed in Subsection 3.2 and subsequently literature on heuristic solution algorithms in Subsection 3.3.

3.1 Multi-Objective Optimization

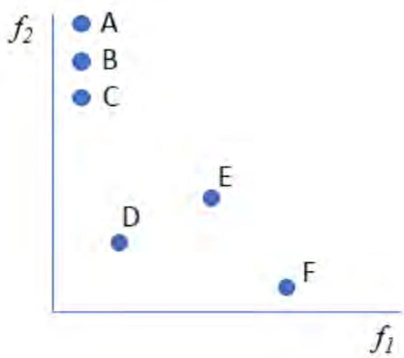
In the majority of optimization problems, the primary objective is often to determine the optimal solution. Nevertheless, certain contexts present ambiguity regarding the definition of the 'best solution'. This is where the research area of Multi-Objective Optimization falls comes into place. Rather than seeking a singular best solution, this area of research aims to generate a set of optimal solutions, allowing the decision-maker to manually select the preferred solution. The basic notation of of a multi-objective-program is generally formulated in the following way:

$$\min f(\mathbf{x}) = \begin{pmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{pmatrix} \quad \text{s.t.} \quad \mathbf{x} \in \Omega \subseteq \mathbb{R}^n \tag{3.1}$$

Considering a function $f : \mathbb{R}^n \to \mathbb{R}^m$, where $m \in \mathbb{N}$, $m \geq 2$, and Ω represents the solution space.

Multiple optimal solutions may exist in the problem, and each of these optimal solutions is referred to as an Edgeworth-Pareto (EP) optimal point. In Marler and Arora (2004) the following definitions are stated: A point $\bar{\mathbf{x}}$ is considered strongly EP optimal if there exists no other point $\mathbf{x} \in \Omega$ that satisfies $f_i(\mathbf{x}) \leq f_i(\bar{\mathbf{x}})$ for all $i = 1, \ldots, m$, and if $f_j(\mathbf{x}) < f_j(\bar{\mathbf{x}})$ for at least one $j \in 1, \ldots, m$. In other words, no improvement can be made in any objective function without compromising the performance of another objective function. On the other hand, a point $\bar{x}$ is considered weakly EP optimal if there is no point $\mathbf{x} \in \Omega$ such that $f_i(\mathbf{x}) < f_i(\bar{\mathbf{x}})$ holds for all $i = 1, \ldots, m$. Thereby, no point exists which strictly dominates $\bar{\mathbf{x}}$ in every direction.

Each strongly EP optimal point is therefore also weakly optimal. However, it should be noted that not every weakly EP optimal point is necessarily strongly EP optimal. In the case of bi-objective optimization, the two types of EP optimality are visualized in Figure 3.1. The set of solutions that are non-dominated is commonly referred to as the efficient set.



Note: points C,D,F are strongly EP. Points A,B,C,D,F are weakly EP. Point E is not EP, since E is strictly dominated by D in every direction.

Figure 3.1: Strongly and weakly EP for the bi-objective case

3.1.1 Solution Methods

For multi-objective optimization problems, a variety of solution methods exist. These methods can be categorized into three classes: (1) no decision maker, (2) a priori methods and (3) a posteriori methods (Hwang et al., 1979). In the methods without a decision maker, a neutral solution is the sense that is compromises between objectives. Such a solution is located in the middle of the efficient set. Thereby, no preferences are known. In the a priori methods it involves the decision maker setting desired expectations, and the closest solution to these expectations is found. However, it can be very challenging for the decision maker to set a transparent expectation. In the a posteriori methods a representation of the efficient set is obtained, such as Figure 3.1, allowing the decision maker to make a decision after examining all EP optimal solutions. This allows the decision maker to make a trade-off between different objectives and make a critically informed choice. In this research a posteriori methods are used.

3.1.2 Efficient Set

To find the efficient set for a multi-objective optimization problem, several methods have been established. One approach, the most common one, is the weighted sum method. This method specifies a weight vector w (Zadeh, 1963). Subsequently, all objective functions are combined to form a single function. Which gives the following optimization problem to solve:

$$\min f(\mathbf{x}) = \begin{pmatrix} w_1 \dots w_m \end{pmatrix} \begin{pmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{pmatrix} \quad \text{s.t. } \mathbf{x} \in \Omega \subseteq \mathbb{R}^n \tag{3.2}$$

where $\sum_{i=1}^{m} wi = 1$, thus the weights are normalized. By using different weight factors, different solutions can be found. All found solutions combined form the efficient set. A benefit of the weighted sum method is it's simplicity. It only needs to assign weights to the objectives. However, it assumes a linear trade-off between objectives, which most often is not the case. This can lead to an incorrect representation of the true trade-offs in the problem space. Furthermore, the quality of the solutions depend heavily on the chosen weights, and it can be very time consuming to find appropriate weights.

Another approach is the Lexicographic method, where the objective functions are arranged in order of importance. Then, the following optimization problem are solved iterative:

$$\begin{aligned} &\min f(\mathbf{x}) = f_i(\mathbf{x}) \\ &\text{s.t. } f_j(\mathbf{x}) \leq f_j(\mathbf{x_j^*}), \quad j = 1, 2, \dots, i-1, i > 1, \\ &i = 1, 2, \dots, m, \\ &\mathbf{x} \in \Omega \subseteq \mathbb{R}^n \end{aligned} \tag{3.3}$$

In Zadeh (1963), i represents the ranking of the function in the preferred sequence, and $f_j(\mathbf{x_j^*})$ represents the optimal solution value of the j^{th} objective function, found in the j^{th} iteration.

In Osyczka (1984) a variation is discussed, using the following constraints:

$$f_j(\mathbf{x}) \leq \left(1 + \frac{\delta_j}{100}\right) f_j(\mathbf{x_j^*}), \quad j = 1, 2, \dots, i, i > 1, \tag{3.4}$$

, differentiating the hierarchical method from the lexicographic approach. When comparing (3.4) with (3.3), (3.4) represents a constraint relaxation, caused by increasing the right hand side of the equation, thereby increasing δ_j. Note that δ_j ranges between 0 and 100. Using different values for δ_j result in different optimal solutions. Again creating the efficient set.

Lastly, the ε-constraint method was proposed by He and Zhang (2022). In contrast to alternative methods, this approach has the advantage of efficiently acquiring the Pareto solution set without a uniform measure. Subsequently, it is able to produce non-extreme efficient solutions as opposed to for example the weighted sum method as indicated by Mavrotas (2009). The generalized ε-constrained problem is formulated as follows:

$$\begin{aligned} &\min f(\mathbf{x}) = f_i(\mathbf{x}) \\ &\text{s.t. } f_j(\mathbf{x}) \leq \varepsilon_j, \quad (1 \leq j \leq m, i \neq j), \\ &i = 1, 2, \ldots, m, \\ &\mathbf{x} \in \Omega \subseteq \mathbb{R}^n \end{aligned} \tag{3.5}$$

where ε_j is the value of the j^{th} ε-constraint and $f_i(\mathbf{x})$ denotes that the i^{th} objective function is chosen to be optimized.

For the bi-objective optimization problem, a simple solution is to solve the ε-constrained problem by optimizing on one of the objective and transforming the other into a constraint. Then (3.5) takes the following form:

$$\begin{aligned} &\min f(\mathbf{x}) = f_1(\mathbf{x}) \\ &\text{s.t. } f_2(\mathbf{x}) \leq \varepsilon_2, \end{aligned} \tag{3.6}$$

By gradually de- or increasing the ε values, the Pareto optimal solutions can be generated. The benefits of the ε-constraint in case of the bi-objective problem in this thesis is that Pareto optimal solutions can be generated relatively simple with respect to the other mentioned methods. Furthermore, it is far less time consuming. Therefore, the ε-constraint method will be used in this research.

3.2 Related problems

In the most common VRP, vehicles leave from a depot, where they are loaded, visiting one or more customers on a single route, and then return to the starting depot, which generalizes the well-known Traveling Salesman Problem (TSP) (Cordeau et al., 2007), and thus a NP-hard problem. In a simple world this would be a simple problem to solve. However, in current times there exist a lot of restrictions for vehicles in urban areas, thus near customer locations. For example, in certain areas of Amsterdam there exist low emission zones for diesel vehicles, where they are restricted from entering. Therefore, distribution networks are split into two echelons, in order to adhere all regulations. With a two-echelon distribution network, urban vehicles and city freighters (Crainic, Ricciardi, et al., 2004) are used for picking up and delivering goods. Intermediate facilities (satelites) consolidate the process of loading goods from the vehicles used in the first echelon to the vehicles used in the second echelon (Guastaroba et al., 2016), or simply first echelon (FE) vehicles and second echelon (SE) vehicles. In Perboli et al. (2011), FE vehicles and SE vehicles differed in size, large and smaller vehicles, respectively. The authors introduce a single depot, where large vehicles could transport large amount of freight to distribution stations (satelites), where smaller vehicles are then loaded to distribute all deliveries to the customers. The routing problem of a two-echelon distribution network is known as the two-echelon vehicle routing problem (2E- VRP), where its first formulation is given in Crainic, Ricciardi, et al. (2009), where they study multiple variants of the 2E-VRP problem. Two-echelon networks are often found in practical examples. One of such is in the field of city logistics, where freight deliveries in inner-city centres have to be done efficiently, while simultaneously considering the negative impact on the environment (Crainic, Gendreau, et al., 2021; Dolati Neghabadi et al., 2019; Savelsbergh and Van Woensel, 2016). In a traditional 2E-VRP it is often assumed that satellites have no storage capacity, such that minimal changes have to be made to the supply chain. Li, Liu, Jian, et al. (2018) studied a variant of a 2E-VRP with limited storage capacity at satellite locations. Additionally new constraints are introduced to the problem to simulate the real-time transshipment capacity. For further research done on 2E-VRPs this thesis refers to an extensive literature review done at the Technical University of Eindhoven by Sluijk et al. (2022). In the most common VRP, vehicles leave from a depot, where they are loaded, visiting one or more customers on a single route, and then return to the starting depot, which generalizes the well-known Traveling Salesman Problem (TSP) (Cordeau et al., 2007), and thus a NP-hard problem. In a simple world this would be a simple problem to solve. However, in current times there exist a lot of restrictions for vehicles in urban areas, thus near customer locations. For example, in certain areas of Amsterdam there exist low emission zones for diesel vehicles, where they are restricted from entering. Therefore, distribution networks are split into two echelons, in order to adhere all regulations. With a two-echelon distribution network, urban vehicles and city freighters (Crainic, Ricciardi, et al., 2004) are used for picking up and delivering goods. Intermediate facilities (satelites) consolidate the process of loading goods from the vehicles used in the first echelon to the vehicles used in the second echelon (Guastaroba et al., 2016), or simply first echelon (FE) vehicles and second echelon (SE) vehicles. In Perboli et al. (2011), FE vehicles and SE vehicles differed in size, large and smaller vehicles, respectively. The authors introduce a single depot, where large vehicles could transport large amount of freight to distribution stations (satelites), where smaller vehicles are then loaded to distribute all deliveries to the customers. The routing problem of a two-echelon distribution network is known as the two-echelon vehicle routing problem (2E- VRP), where its first formulation is given in Crainic, Ricciardi, et al. (2009), where they study multiple variants of the 2E-VRP problem. Two-echelon networks are often found in practical examples. One of such is in the field of city logistics, where freight deliveries in inner-city centres have to be done efficiently, while simultaneously considering the negative impact on the environment (Crainic, Gendreau, et al., 2021; Dolati Neghabadi et al., 2019; Savelsbergh and Van Woensel, 2016). In a traditional 2E-VRP it is often assumed that satellites have no storage capacity, such that minimal changes have to be made to the supply chain. Li, Liu, Jian, et al. (2018) studied a variant of a 2E-VRP with limited storage capacity at satellite locations. Additionally new constraints are introduced to the problem to simulate the real-time transshipment capacity. For further research done on 2E-VRPs this thesis refers to an extensive literature review done at the Technical University of Eindhoven by Sluijk et al. (2022). In the most common VRP, vehicles leave from a depot, where they are loaded, visiting one or more customers on a single route, and then return to the starting depot, which generalizes the well-known Traveling Salesman Problem (TSP) (Cordeau et al., 2007), and thus a NP-hard problem. In a simple world this would be a simple problem to solve. However, in current times there exist a lot of restrictions for vehicles in urban areas, thus near customer locations. For example, in certain areas of Amsterdam there exist low emission zones for diesel vehicles, where they are restricted from entering. Therefore, distribution networks are split into two echelons, in order to adhere all regulations. With a two-echelon distribution network, urban vehicles and city freighters (Crainic, Ricciardi, et al., 2004) are used for picking up and delivering goods. Intermediate facilities (satelites) consolidate the process of loading goods from the vehicles used in the first echelon to the vehicles used in the second echelon (Guastaroba et al., 2016), or simply first echelon (FE) vehicles and second echelon (SE) vehicles. In Perboli et al. (2011), FE vehicles and SE vehicles differed in size, large and smaller vehicles, respectively. The authors introduce a single depot, where large vehicles could transport large amount of freight to distribution stations (satelites), where smaller vehicles are then loaded to distribute all deliveries to the customers. The routing problem of a two-echelon distribution network is known as the two-echelon vehicle routing problem (2E- VRP), where its first formulation is given in Crainic, Ricciardi, et al. (2009), where they study multiple variants of the 2E-VRP problem. Two-echelon networks are often found in practical examples. One of such is in the field of city logistics, where freight deliveries in inner-city centres have to be done efficiently, while simultaneously considering the negative impact on the environment (Crainic, Gendreau, et al., 2021; Dolati Neghabadi et al., 2019; Savelsbergh and Van Woensel, 2016). In a traditional 2E-VRP it is often assumed that satellites have no storage capacity, such that minimal changes have to be made to the supply chain. Li, Liu, Jian, et al. (2018) studied a variant of a 2E-VRP with limited storage capacity at satellite locations. Additionally new constraints are introduced to the problem to simulate the real-time transshipment capacity. For further research done on 2E-VRPs refer to an extensive literature review done at the Technical University of Eindhoven by Sluijk et al. (2022) on unique 2E-VRPs variants.

Most literature on VRPs assume homogeneous vehicles. Recently, more researches have been studied where heterogeneous vehicles are assumed (Bevilaqua et al., 2019; Crainic, Errico, et al., 2016; Kancharla and Ramadurai, 2019; Kergosien et al., 2013; Yong Wang et al., 2018), e.g. vehicles with different capacities. Nowadays, electrical vehicles can be included in a fleet of vehicles. In practice, multiple companies have already transitioned to put electrical vehicles in use (Foltyński, 2014). Electric vehicles do come with complications, e.g. range limitations or relatively smaller capacity.

Conrad and Figliozzi (2011) were one of the first to consider a recharging VRP to solve the limitations of electical vehicles. The authors attempted to optimize a VRP including electrical vehicles and recharging stops. In this proposed model, a vehicle could recharge the battery at customer locations, but at additional costs. Another VRP that introduces the possibility of recharging is the green vehicle routing problem (GVRP), proposed in Erdoğan and Miller-Hooks (2012). In their GVRP vehicles can stop at a recharging station en-route. This way a vehicle can visit customers, where the traveled distance is longer than its range, contrary to the recharging VRP mentioned above. Alternatively, a constraint could be imposed that sets the maximum length of a route to the maximum range of an electrical vehicle (Li, Liu, Chen, et al., 2020). In later literature, additional real-life problems such as time-windows were introduced. Schneider et al. (2014) were the first to incorporate delays in their model, caused by recharging time, as to subject to time-window constraints. More recently, companies adhere smaller range limits in real life than the theoretical range limits, e.g. factory range limit (up to half of the range) (Yusheng Wang et al., 2017).

Most papers study an objective function based on costs per traveled arc. The underlying computation of these costs are often not specified in researches. However, some papers provide a derivation of the costs, e.g. the cost per traveled arc is based on fuel consumption (Kancharla and Ramadurai, 2019; Paul et al., 2021; Soysal et al., 2015; K. Wang et al., 2017). The fuel consumption is then computed as a function of dependent parameters, such as fuel cost and distance.

In more recent papers, environmental aspects are included in the objective function, e.g. carbon emission. Z. Wang and Wen (2020) express carbon emission as a cost dependent on the vehicle's travel distance. Demir et al. (2014) propose to optimize speed as to minimize emission costs. Ade Irawan et al. (2022) developed an optimization model where the minimization of the total logistic costs and the total amount of carbon emissions were conflicting objectives, thus introducing a bi-objective function. This paper considers simultaneously both forward and reverse supply streams, which is a challenging problem as mentioned in Pishvaee and Razmi (2012). I.e. traditional approaches, where forward and reverse supply are solved separately, resulted in sub-optimal solutions. However, in our paper only reverse supply is considered, thus avoiding this problem. Furthermore, Ade Irawan et al. (2022) try to solve two problems at one, namely to determine the locations of distribution centers and recycling plants, and the corresponding supply chain of their whole logistics network. As mentioned, we will only focus on the second part, i.e. the logistics.

3.3 Heuristic Solution Algorithms

To find a good solution for a bi-objective problem, one has to find a point that lies on the Pareto efficient set. There are multiple methods that are able to find these points, such as Pareto efficient set generation and Compromise Programming (CP). CP has proven to be successful in many applications, under which the environmental are of wind farming scheduling by Irawan et al. (2017). Therefore an matheuristic was designed based on an aggregation technique, a reduced EM, an interchange-based heuristc and variable neighbouring search (VNS). This matheurstic proved to be an efficient methods, as it was able to compromise solutions of higher quality than the EM with reduced computational time.

K. Wang et al. (2017) approached optimizing a two-echelon capacitated vehicle routing problem (2E-CVRP), with environmental considerations (2E-CVRP-E), using a VNS with integer programming. The use of integer programming is to find better solutions that the VNS algorithm had failed to find. In this problem, the second-echelon is replaced by a route-based model instead of arc-based. The idea behind this is that using VNS second-level routes can be found in local optima and then stored in the set of feasible routes. And a final solution can then be found with solving the model with the routes obtained by VNS as input. Imran et al. (2009) proposed to use the VNS algorithm for solving the heterogeneous fleet vehicle routing problem (HFVRP), similar to the problem discussed in this research. Their method was able to outperform existing methods using the same data sets from literature.

For the bi-objective Pollution-Routing Problem (PRP), where the objective is to minimize fuel consumption, emissions costs and driving time. This can be considered similarly to our problem since we aim to optimize on emission and routing costs. In the paper of Demir et al. (2014) an ALNS is presented to solve the bi-objective PRP. ALNS has proven to be effective in even more complex problems, such as the real-life Multi Depot Multi Period Vehicle Routing Problem with a Heterogeneous Fleet (MDMPVRPHF) shown in Mancini (2016). In this research the ALNS will be implemented as heuristic.

In recent years, new heuristic based methods have proven to be promising, such as evolutionary algorithms (EA) and reinforcement learning (RL). Uneversiti Tenaga Nasional (UNITEN) has a VRP, where a route has to be found such that each bus can include all students waiting at the visited locations (CVRP). To achieve an solution that reduces the time and distance for the CVRP of UNITEN, a genetic algorithm was used in Mohammed et al. (2017). Genetic algorithms are good for solving complex problems, it is able to find solutions in a very large solution space, thus highly scalable to the problem size contrary to linear programming. It can also deal with multi-constraints problems, such as VRP, and has proven to find approximate solutions. Lu et al. (2019) propose a RL-based heuristic. They use a Generative adversarial network (GAN) as learning agent. The agent learns from the state, which includes the capacities and distances between customers. Based on the learning process, the agent chooses from several possible actions. The agent keeps repeating the process until convergence. It has to be noted that there still has to be done more research on RL-based heuristics. As most studies use identical sizes for the training and test instances, i.e. the same number of nodes. There is still a gap in the performance analysis for solving larger problems than the training instances (Yan et al., 2022).

4 Methodology

4.1 Mathematical Optimisation Model

The mathematical notation of our model is provided below. Meaning, the sets, parameters, objective function and constraints are defined.

1. **Sets:**

   D : set of truck depots, $D = \{0\}$
   C : set of construction sites
   W : set of waste stations (Processing facilities)
   L set of all locations, $L = D \cup C \cup W$
   T : set of types of trucks, $T = \{\text{Diesel, Electric}\}$

2. **Parameters:**

   s_c : expected supply of construction site c.
   d_{ij} : distance in km from location i to j.
   f_t : transportation costs of truck type t per kilometre (€$/km$).
   e_t : carbon emission of truck type t per tonne-kilometre ($kgCO2_e/tkm$).
   Q : capacity of a truck in tonnes.
   A_t : action radius of truck type t.
   w : weight of an empty truck in tonnes.
   ε : small neglectable value.

3. **Decision Variables:**

   $x_{ij}^t \in \{0, 1\}$: Binary variable equal to 1 if a truck of type t leaves from node i to j, and 0 otherwise
   $y_{ij}^t \in \mathbb{R}^+$: Load on truck type t when leaving from i to j

4. **Auxiliary Variables:**

   $u_i^t \in \mathbb{R}^+$: Accumulated distance traveled of truck type t when arriving at i.

5. **Objective functions:**

$$\begin{aligned} \min \sum_{i \in D \cup C} \sum_{j \in C} \sum_{t \in T} x_{ij}^t d_{ij} f_t + \sum_{i \in C} \sum_{j \in W} \sum_{t \in T} x_{ij}^t d_{ij} f_t \\ + \sum_{i \in D \cup C} \sum_{j \in C} \sum_{t \in T} y_{ij}^t d_{ij} \varepsilon + \sum_{i \in C} \sum_{j \in W} \sum_{t \in T} y_{ij}^t d_{ij} \varepsilon \end{aligned} \tag{4.1}$$

$$\begin{aligned} \min \sum_{i \in D \cup C} \sum_{j \in C} \sum_{t \in T} x_{ij}^t d_{ij} e_t w + \sum_{i \in C} \sum_{j \in W} \sum_{t \in T} x_{ij}^t d_{ij} e_t w \\ + \sum_{i \in C} \sum_{j \in C \cup W} \sum_{t \in T} y_{ij}^t d_{ij} e_t \end{aligned} \tag{4.2}$$

6. **Flow constraints:**

$$\sum_{i \in D \cup C} x_{ij}^t = \sum_{k \in C \cup W} x_{jk}^t, \quad \forall j \in C, t \in T \tag{4.3}$$

$$\sum_{i \in D \cup C} \sum_{t \in T} x_{ij}^t = 1, \quad \forall j \in C \tag{4.4}$$

$$x_{ii}^t = 0, \quad \forall i \in L, t \in T \tag{4.5}$$

7. **Load constraints:**

$$\sum_{t \in T} y_{ij}^t + s_j \leq \sum_{k \in C \cup W} \sum_{t \in T} y_{jk}^t, \quad \forall i \in D \cup C, j \in C \tag{4.6}$$

$$x_{ij}^t s_i \leq y_{ij}^t \leq x_{ij}^t Q_t, \quad \forall i \in D \cup C, j \in C, t \in T \tag{4.7}$$

$$x_{ij}^t s_i \leq y_{ij}^t \leq x_{ij}^t Q_t, \quad \forall i \in C, j \in W, t \in T \tag{4.8}$$

$$y_{0j}^t = 0, \quad \forall j \in C, t \in T \tag{4.9}$$

8. **Distance constraints:**

$$u_i^t + x_{ij}^t d_{ij} \leq u_j^t + A_t(1 - x_{ij}^t), \quad \forall i \in D \cup C, j \in C, t = \text{Electric} \tag{4.10}$$

$$u_i^t + x_{ij}^t d_{ij} \leq u_j^t + A_t(1 - x_{ij}^t), \quad \forall i \in C, j \in W, t = \text{Electric} \tag{4.11}$$

$$u_j^t \leq A_t, \quad \forall i \in D \cup C, j \in C, t = \text{Electric} \tag{4.12}$$

$$u_j^t \leq A_t, \quad \forall i \in C, j \in W, t = \text{Electric} \tag{4.13}$$

$$u_0^t = 0, \quad \forall t = \text{Electric} \tag{4.14}$$

9. **Domain constraints:**

$$x_{ij}^t \in \{0, 1\}, \quad \forall i, j \in L, t \in T \tag{4.15}$$

$$y_{ij}^t \in \mathbb{R}^+, \quad \forall i, j \in L, t \in T \tag{4.16}$$

$$u_i^t \in \mathbb{R}^+, \quad \forall i \in L, t \in T \tag{4.17}$$

4.1.1 Objective functions

The objective function 4.1 consist of two parts. The first two summations minimize the total transportation costs for the traveled distance. The second part, the last two summations, minimize the load decision variable. This is needed, because we want to track the exact load at any point along the route. If we would not include y_{ij}^t in the objective function, y_{ij}^t could take any arbitrary value between the lower bound determined by constraint 4.6 and the upper bound determined by constraints 4.7-4.8. However, we do not want the load to impact the function. Therefore, the load variable is multiplied by a small neglectable value denoted by ε. The second objective function 4.2 minimizes the total emission. Note that emission is based on both the distance and the load in terms of per tonne-kilometre. This function also consist of two parts. An empty truck has already a significant weight and thus this needs to be held into account. The first two summations determine the total emission by an empty truck. Then the last summation determines the extra emission caused by the extra load that all trucks carry along the route. Note that we only have to minimize the load leaving from a construction site, since the first visit to a construction site has not load yet and the emission for the first stop is thus covered in the first summation.

Note that for every part we sum over two different (i, j) combinations, since we cannot travel from the depot directly to processing facilities. Therefore it is split in two parts, the first part is for trucks arriving at a construction site from either the depot or an other construction site, and the second part is for the arrivals at the processing facilities from construction sites.

4.1.2 Constraints

Constraint 4.3 ensures that the flow of trucks coming in at a construction site equals the flow coming out. Furthermore, it controls where the incoming flow comes from and where the outgoing flow goes to, which is needed to uphold the supply chain as was illustrated in Figure 2.1. Constraint 4.4 ensures that each construction site must be visited once. We sum over the union of set D and C and over T, which gives us the number of times construction site j is visited in total. Constraint 4.5 prevents loops.

Constraint 4.6 ensures that the load of a truck increases along the route. Thus meaning if a truck goes to location i then j then K, then the load at i must be smaller than the load at j henceforth the load at j is smaller than at k. This is a rewritten Miller–Tucker–Zemlin subtour elimination constraint (Miller et al., 1960). Since a location can only be visited once, we know that there is only one combination that has a nonzero value. Therefore we can sum over t, to obtain the load that leaves from i to j. Then if we add the supply of location j, this must be smaller or equal than the sum of load leaving location j, therefore in the right part we sum over k and t. Constraint 4.7-4.8 forces that the load of a truck of type t when leaving from i to j is set to 0 if that arc is not traveled or it ensured that the load is equal or larger than the supply at location i and it must be smaller or equal than the capacity of a truck type t. Constraint 4.9 sets the load at the beginning of a route at 0. To track the load dynamically per segment of the routes was quite challenging as it appeared to be a non-linear problem at first. However, because of the unique nature of our problem, we have been able to rewrite load constraints of existing studies and formulate it as a linear problem.

Constraints 4.10-4.11 keep track of the distance traveled by a truck of type t along their route. These are also rewritten Miller–Tucker–Zemlin subtour elimination constraints. The right hand side consists of two elements. First is the accumulated distance traveled of truck type t when arriving at node j, and the second element is to ensure feasability of the solution. Note that if $x_{ij}^t = 0$, it is possible that u_j^t is smaller than u_i^t. $A_t(1 - x_{ij}^t)$ ensures that the inequality still holds in these cases, since u_i^t can never exceed A_t, see the right-hand side of constraints 4.12-4.13. However, if $x_{ij}^t = 1$, then u_j^t must be equal to u_i^t plus the distance from node i to j. Thereby, the second element equals to 0 when $x_{ij}^t = 1$, thus nullified. Constraints 4.12-4.13 ensures that if a truck of type t travels from i to j, the accumalated distance traveled by a truck of type t at node j is at least the distance from i to j and at most the action radius of a truck type t. Lastly, Constraint 4.14 sets the accumalated distance at 0 for the start of every route.

Note that constraints 4.10-4.14 are only for electric trucks, since diesel trucks know no limitations on route distances.

4.2 Adaptive Large Neighbourhood Search

To solve the bi-objective Heterogeneous Open Vehicle Routing Problem, an implementation of the Adaptive Large Neighbourhood Search (ALNS) metaheuristic algorithm is used (Ropke and

Pisinger, 2006b). Fundamentally, ALNS operates as an iterative ruin-and-recreate algorithm, persistently executing until a predefined stopping criterion is attained. In contrast to the search within one large neighbourhood, as employed in the Local Neighborhood Search (LNS) algorithm, it adopts multiple destroy and repair operators. For further information on LNS the reader may refer to Pisinger and Ropke (2019). Repair operators are used to repair a partially destroyed solution by re-inserting the nodes removed by a destroy operator. When re-inserting the nodes in existing routes, the repair operator checks for feasibility with regard to the capacity and distance limit. If it cannot uphold feasibility, new routes are created. A destroy and repair operator are chosen each iteration. The dynamic selection of destroy and repair operators is based on their past performance. The new solution is evaluated by an acceptance criteria, defined by the record-to-record travel (RRT) (Dueck, 1993) as local search framework employed at the outer level, and possibly replaces the current solution. Drawing upon the results of that evaluation, the operator selection scheme updates the likelihood of selecting the respective operators for subsequent iterations. A general framework of the ALNS algorithm is shown in Algorithm 1.

The primary features of the ALNS algorithm will be thoroughly described in sections 4.2.1-4.2.3, and the used destroy and repair operators are explained in sections 4.2.5 and 4.2.6.

Algorithm 1: General Adaptive Large Neighbourhood Search framework

Input: Initial solution: x_0
Input: Initial acceptance parameters
Input: Initial destroy/repair scores

$x = x_0$ `/* Initialise current solution */`
$x^* = x_0$ `/* Initialise best solution */`
$i = 0$ `/* Time count */`
repeat
    Choose a destroy operator d
    Choose a repair operator r
    $x' \leftarrow r(d(x))$
    if *Accept new solution x'* **then**
        $x = x'$
    end
    if $f(x) < f(x^*)$ **then**
        $x^* = x$
    end
    Update(Destroy/Repair scores)
    Update(Acceptance parameters)
until *the maximum run-time has expired*
return x^*

4.2.1 Initialization

We need an initial solution that is going to be destroyed and repaired by the ALNS heuristic. To this end, we use a simple nearest neighbor (NN) heuristic (Nilsson, 2003). The quality of the initial solution does not significantly impact the performance of the ALNS, since the algorithm is designed to recover from a suboptimal starting point. The NN preserves the feasibility of capacity and distance constraints, ensuring that the initial solution adheres to the problem's constraints. Additional constraints are incorporated into the NN, to adjust for the fact that we are dealing with an OVRP. The NN is shown in Algorithm 2. NN starts with an empty solution and iteratively adds the nearest customer to the routes, starting from the depot. If there are no routes available, then a new route is created. A route is created with equal chance for a diesel or electric truck, to ensure diversification, if a truck type is not given as input argument.

Algorithm 2: Modified Nearest Neighbour

```
Input: Capacity: Q
Input: Action radius: A
Input: Supply: s_i
x = []                                          /* Initialise empty solution */
L ← C                                      /* Initialise set of unvisited nodes */
while L ≠ ∅ do
    route = Route([0])                   /* Initialise route starting at depot */
    current = 0                                 /* Set current node at depot */
    y = 0                                                    /* Set load to 0 */
    u = 0                                       /* Set distance traveled to 0 */
    while L ≠ ∅ do
        Pick nearest neighbour nearest ∈ NN(current) ∩ L
        if y + s_nearest > Q then
            break
        end
        if route.type = Electric then
            Pick nearest processing facility w ∈ W
            if u + Distance(current, nearest) + Distance(nearest, w) > A then
                break
            end
        end
        current = nearest
        Add current to route and remove from L
        Update(y)
        Update(u)
    end
    Pick nearest processing facility w ∈ W
    Add w to route
    Add route to x
end
return x
```

4.2.2 Adaptive score adjustment procedure

Within each iteration of the ALNS, a destroy and repair operator are selected through a roulette-wheel mechanism. Each operator has an associated weight that reflects its proportional representation on the wheel. The weights are updated based on their past performance in producing new solutions. Note that this process is done separately for the group of destroy and repair operators. Denote ω_i as the weight assigned to operator i, then the operator is selected with a probability $\omega_i / \sum \omega_j$, where the sum goes over all operators. This probabilistic approach ensures that operators with higher scores possess a greater likelihood of being selected by the roulette-wheel mechanism, while still allowing for a degree of diversity and exploration of alternative operators. At the end of each iteration, the weights are updated as shown in Algorithm 3. All weights are set to equal values, namely 1, at the start of the algorithm to ensure a fair starting point for the exploration process. In each iteration, the selected operator is applied to the current solution, resulting in a new candidate solution. This candidate solution is evaluated by the ALNS algorithm, which leads to one of four rewards: $\sigma_1, \sigma_2, \sigma_3$, σ_4, where $\sigma_1 > \sigma_2 > \sigma_3$ > σ_4. The rewards are then assigned to score π as follows:

$$\pi = \begin{cases} \sigma_1, & \text{if the candidate solution is a new global best.} \\ \sigma_2, & \text{if the candidate solution is better than the current solution, but not a new global best.} \\ \sigma_3, & \text{if the candidate solution is accepted} \\ \sigma_4, & \text{if the candidate solution is rejected} \end{cases} \tag{4.18}$$

Then the weights are updated as shown in line 19 using the score π and decay rate $\eta \in [0, 1]$. The parameter η controls how much the past score affects the current weight.

Note that a selection of destroy and repair operators define the neighbourhood of a solution x, and is denoted by $N_i(x)$.

4.2.3 Acceptance and stopping criteria

The record-to-record travel method was used as a local search framework for our ALNS algorithm. The RRT criterion (Santini et al., 2018) is presented in Algorithm 4. The acceptance of the new solution transpired under the condition that the deviation between said solution and the best solution achieved thus far is less than a threshold T. The initial and end value of the threshold are pre-specified. The threshold then linearly decreases, at each iteration, towards the end value. Note that the time of each iteration cannot be determined at forehand. Therefore, T is updated over a number of iterations I. After I iterations, T is not updated anymore. The algorithm returns the best found solution after the maximum run-time has expired.

4.2.4 Operator coupling

In ALNS a destroy and repair operator are normally chosen independently of each other. In our implementation we allow for operator coupling, which refers to the combination of different operations during the search process. Using operator coupling, operators can work jointly, rather than in isolation, thereby enhancing each other's strengths. This allows for better exploration capabilities and efficient search through the solution space. This process works as follows: We specify an

Algorithm 3: Roulette-Wheel

Input: Initial solution: x_0
Input: Initial scores: $\sigma_1, \sigma_2, \sigma_3, \sigma_4$
Input: Initial decay rate: η

$x = x_0$ `/* Initialise current solution */`
$x^* = x_0$ `/* Initialise best solution */`
$\omega_i = 1$ `/* Set weight to 1 for all operators */`
repeat
  Select an operator i with probability $\omega_i / \sum \omega_j$
  Pick $x' \in N_i(x)$
  if $f(x') < f(x^*)$ **then**
    $\pi = \sigma_1$
    $x^* = x'$
  else if $f(x') < f(x)$ **then**
    $\pi = \sigma_2$
    $x = x'$
  else if x' *Accept new soluton x'* **then**
    $\pi = \sigma_3$
    $x = x'$
  else
    $\pi = \sigma_4$
  end
  $\omega_i = \eta\omega_i + (1 - \eta)\pi$
  return x^*
until *the maximum run-time has expired*

Algorithm 4: Record-to-Record Travel

Input: Initial solution: x_0
Input: Initial treshold: T_0
Input: Final treshold: T

$x = x_0$ /* Initialise current solution */
$x^* = x_0$ /* Initialise best solution */
$T = T_0$ /* Initialise current threshold */
repeat
  Pick $x' \in N(x)$
  if $\frac{f(x') - f(x^*)}{f(x')} < T$ **then**
    $x = x'$
  end
  if $f(x) < f(x^*)$ **then**
    $x^* = x$
  end
  Update(T)
  return x^*
until *the maximum run-time has expired*

boolean matrix that indicates coupling between destroy and repair operators. $Entry(i, j)$ is True if destroy operator i can be used together with repair operator i, and False otherwise. Suppose that we want repair operator j to only be applied if destroy operator j has partially destroyed the solution x. Then we specify this in a boolean matrix by setting $Entry(i, j)$ to True and every other $Entry(d, j)$ with $d \in \Delta \backslash i$ to False, where Δ is the set of destroy operators.

4.2.5 Destroy Operators

In this section, the seven destroy operators used in our ALNS are described. The first four operators have been derived or influenced by works of Ropke (2005), Ropke and Pisinger (2006b) and Ropke and Pisinger (2006a), Pisinger and Ropke (2007), and Shaw (1998), while the latter is a relative new contribution in literature by Christiaens and Vanden Berghe (2020), obtaining state-of-the-art results. The destroy operators are described below:

1. **Random removal (RR):** This operator randomly removes n nodes from the solution, where n is determined by a separate parameter Ω, the degree of destruction, to control the extent of the damage done to a solution in each step. It is calculated as follows: $n = |C|\Omega$, where $|C|$ is the size of set C. Randomly selecting nodes helps for diversification in the search.

2. **Worst-distance removal (WDR):** This operator targets high-cost customers. The cost is determined as the combined sum of distances between the preceding and following customers along the route. More formally, the operator removes node $j^* = \text{argmax}_{j \in C}\{d_{ij} + d_{jk}\}$.

3. **Route removal (RoR):** This operator randomly removes a complete route from the solution. Thereby, all nodes of that route are removed and become unassigned.

4. **Historical knowledge node removal (HR):** This operator maintains a comprehensive record of the position cost of every node j. The position cost is determined by summing the distances between the preceding and following nodes, calculated as $d_j = d_{ij} + d_{jk}$ during each iteration. Subsequently, at any given point in the algorithm, the best position cost d_j^* for node j is updated to represent the minimum among all d_j values calculated up to that point. Then a node j is selected based on its maximum deviation from its best position cost and removes it from the solution. More formally the operator picks node $j^* = \text{argmax}_{j \in C}\{d_j - d_j^*\}$.

5. **Slack-induced string removal (SISR):** This operator resembles the Node neighbourhood removal (NNR) proposed in Demir et al. (2012). The NNR operator selects a random node and removes $n - 1$ nodes that lie within a rectangular area centered around that node. The NNR process is illustrated in Figure 4.1.

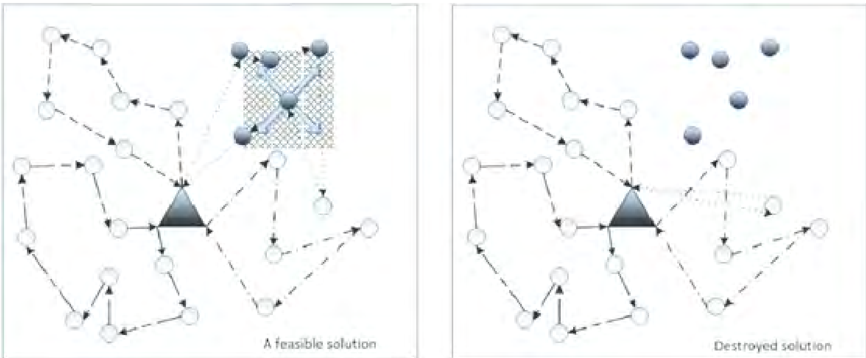


Figure 4.1: An example of node neighbourhood removal

SISR has a different mechanism of removing nodes. It removes partial routes (called strings) around a randomly chosen node. The number strings to remove k_x and cardinality l_r for every string is determined as follows. First let us set the initial maximum number of removed strings K^{max} and maximum cardinality of removed strings L^{max}. Then the number of strings to be removed is the minimum of K^{max} and the cardinality of the set of routes $|R|$ as shown in (4.20). To determine the maximum string cardinality l_x^{max}, we take the minimum of the initial maximum cardinality L^{max} and the mean route cardinality $\overline{|r \in R|}$ of the solution and round it down to the nearest integer, see (4.19). Then the maximum cardinality l_r^{max} for each removed string is the minimum of l_x^{max} and the route cardinality $|r|$ ((4.21)). Ultimately, in (4.22) we obtain the cardinality of the string that needs to be removed from route r by drawing a random sample from a uniform distribution ranging from 1 and $l_r^{max} + 1$.

$$l_x^{max} = \lfloor min\{L^{max}, \overline{|r \in R|}\} \rfloor \tag{4.19}$$

$$k_x = min\{K^{max}, |R|\} \tag{4.20}$$

$$l_r^{max} = min\{l_x^{max}, |r|\} \tag{4.21}$$

$$l_r = \lfloor U(1, l_r^{max} + 1) \rfloor \tag{4.22}$$

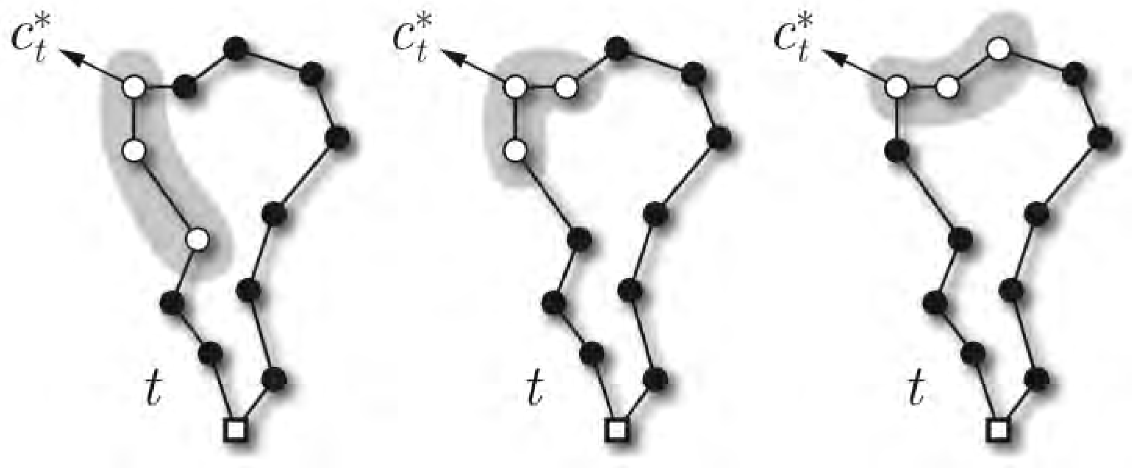
Then let us define solution $x = \{R, \mathcal{L}\}$, where $\mathcal{L}$ the set of unvisited customers, and make the ground rule that a route may be destroyed one time at most. Then the procedure of SISR is shown in Algorithm 5. First, we calculate l_x^{max} and k_x as described in (4.19)-(4.20). A random customer is selected to act as the seed customer, denoted as c^{seed}. An empty set S is initialized to serve as the set for storing destroyed routes. We draw a list from the nearest neighbourhood method using the seed customer as center. Then an iterative process is started, iterating over the neighbourhood, where the customers are ordered by increasing distance. This process continues until set S contains k_x destroyed routes. In each iteration, we check if customer c is not assigned to the set of unvisited customers and route r is not in set S, meaning it has not been destroyed. If these statements hold, c_r^* is initialized and l_r^{max} and l_r are calculated sequentially. Finally, a randomly selected string of cardinality l_r, containing customer c_r^*, is removed from route r. An example of a string removal procedure for $l_r = 3$ is shown in Figure 4.2, note that subscript t is used to denote the routes as well as in Figure 4.3, whereas r is used in Algorithm 5. This figure shows the three possible stri ngs to be removed from route r, where eventually one is randomly selected. A complete removal phase for $k_x = 2$ is illustrated in Figure 4.3, note that subscript s is used in (a) to denote the solution, whereas x is used in this thesis. In Figure 4.3(a), a seed customer is randomly selected. Then the nearest neighbourhood list is iterated over, where the seed customer is the first element and a string with cardinality l_x is removed (see (b) and (c)). It can be seen in (d) that the next customer in the nearest neighbourhood list was part of the destroyed route and is disregarded and we iterate further until a customer is found that is not part of a destroyed route. Then in (e) and (f) a string is again removed and the final destroyed state is shown in (f).

Algorithm 5: Slack-induced string removal

Input: Initial maximum cardinality of removed strings: L^{max}
Input: Initial maximum number of removed strings: K^{max}
Input: Solution: $x = \{R, \mathcal{L}\}$

calculate l_x^{max}, k_x
$c_x^{seed} \leftarrow$ randomCustomer(x)
$S \leftarrow \emptyset$
for $c \in NN(c_x^{seed})$ ***and*** $|S| \leq k_x$ **do**
    Find route $r \in R$ that contains c
    if $c \notin \mathcal{L}$ ***and*** $r \notin S$ **then**
        $c_r^* \leftarrow c$
        calculate l_r^{max}, l_r
        $\mathcal{L} \leftarrow \mathcal{L} \cup removeString(r, l_r, c_r^*)$
        $R \leftarrow R \cup \{r\}$
    end
end

Note that for every destroy operator the following rule applies: if the last node (construction site) from a route is removed then the nearest waste station is automatically updated with respect to the new final node.



Note: There are three possible string selections, which include c_t^* from route t when $l_t = 3$.

Figure 4.2: An example of string removal (Christiaens and Vanden Berghe, 2020)

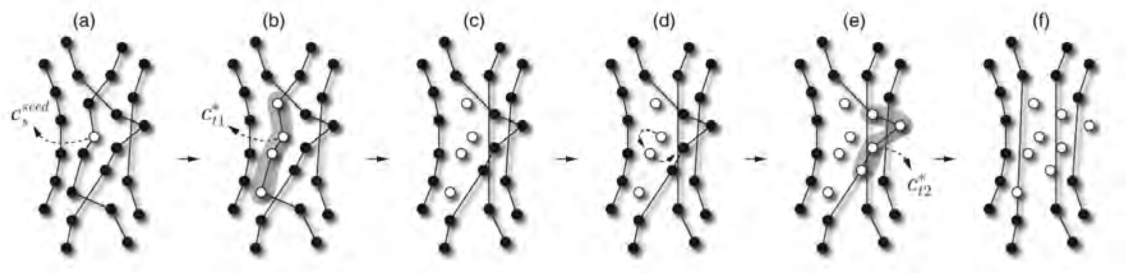


Figure 4.3: An Example of slack induced string removal (Christiaens and Vanden Berghe, 2020)

4.2.6 Repair Operators

In this section the four repair operators used in our ALNS are described:

1. **Greedy repair (GR):** This operator evaluates the insertion cost of each node $j \in \mathcal{L}$ into each existing route. The insertion cost represents the increase in total cost if the node is inserted at a specific position in the route, and is calculated as $d_j = d_{ij} + d_{jk} - d_{ik}$, where i and k were consecutive nodes in the route before inserting j. Then select the node j^* and the position in the route that yields the lowest insertion cost and insert the node there. This process is repeated until all nodes are assigned to a route. If there are no feasible insertions, then a new route is created.

2. **New route repair (NR):** This operator starts a new route and assigns all nodes $i \in \mathcal{L}$ using the NN heuristic. This operator is purely used for exploration of the search space, as during this research it was noticed that if the optimal solution contained more routes than the initial solution, it would not be found. This operator prevents that. Furthermore, this operator is used for diversification of the truck types, to prevent bias, and more importantly to see if the same route that was done by a diesel trucks can be done by an electric truck, resulting in the same cost, but lesser emission. On the other hand, if there were nodes nearby the same route that could not be visited by the electric truck due to its nature with respect to their action radius, these nodes could possibly be inserted into the new diesel route later along the search.

Note that all repair operators maintain feasibility of capacity and distance constraints, ensuring that a new solution adheres to the problem's constraints. Furthermore, if a node (construction site) is inserted at the end of a route then the nearest waste station is automatically updated with respect to that node.

4.3 ε-constraint method

This section describes the implementation of the ε-constraint method as shown in Figure 4.4. The different steps are explained below:

1. **Construct a single-objective optimization model.** First, a single-objective model is constructed, meaning we optimize one of the two objective functions, 4.1-4.2, and disregard the other. The cost objective function 4.1 is referred to as $f_c(x)$ and the emission objective function 4.2 as $f_e(x)$. Then the optimal values $f_c^*(x)$ and f_e^* can be obtained. Let us specify the single objective models S_{cost} and $S_{emission}$, where the cost and emission functions are optimized respectively.

2. **Define the ideal point and nadir point.** The set of ideal points is referred to as $f^{IDEAL} = (f_c^{IDEAL}, f_e^{IDEAL})$, where $f_c^{IDEAL} = f_c^*$ and $f_e^{IDEAL} = f_e^*$). The the nadir point represents to the lowest achievable value for each objective within the Pareto front. It is the extreme point where one objective is minimized, while still being constraint to the other objective. The set of nadir points is referred to as $f^{NADIR} = (f_c^N, f_e^N)$, where $f_c^N = min\{f_c(x)|f_e(x) = f_e^*(x)\}$ and $f_e^N = min\{f_e(x)|f_c(x) = f_c^*(x)\}$.

3. **Obtain the boundary points.** First we solve model S_{cost}, while adding the constraint $f_e(x) \leq f_e^*(x)$. Thereby, obtaining the objective values f_e^N and f_c^{IDEAL}. Then model $S_{emission}$ is solved the other way around to obtain f_c^N and f_e^{IDEAL}. The boundary points of the Pareto front are then the solutions (f_e^N, f_c^{IDEAL}) and (f_e^{IDEAL}, f_c^N).

4. **Construct the single-objective optimization model based on the ε-constraint methods.** First we choose model $f_c(x)$ as the objective function to be optimized based on the ε-constraint methods as shown in (3.6), thereby $f_e(x) \leq \varepsilon_e$ as the ε-constraint. ε_e is calculated as follows: $\varepsilon_e = f_e^N - \omega \times \vartheta$, $\omega = 1, 2, \ldots, \rho - 2$, where ρ is the number of Pareto solutions we want to obtain and ϑ is the step size of ε_e. Then ϑ is calculated as $\vartheta = (f_e^N - f_e^{IDEAL})/(\rho - 1)$.

5. **Solve the single-objective optimization model based on the ε-constraint methods.** Solve the model explained in the previous step for $\omega = 1, 2, \ldots, \rho - 2$ and add all solutions to the Pareto set and output the Pareto frontier.

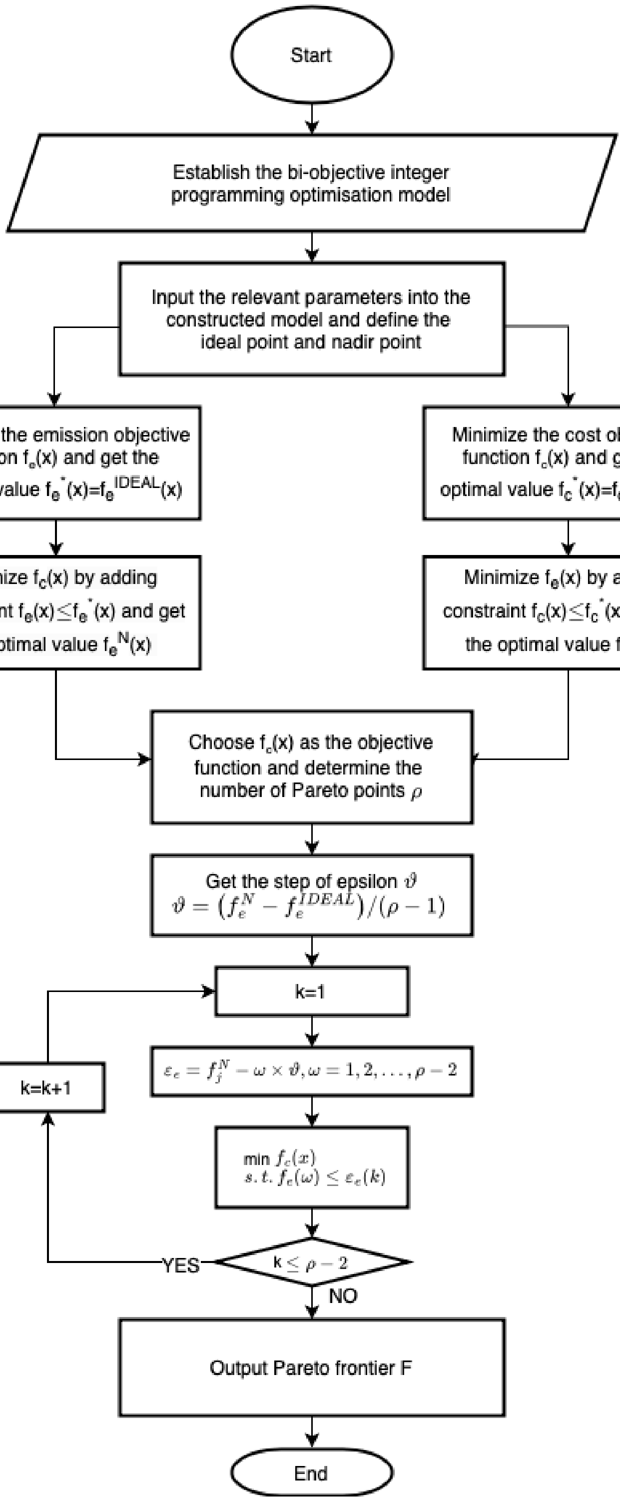


Figure 4.4: Procedure of the ε-constraint method

5 Computational Experiments

This section presents results of computational experiments performed to assess the impact of using electric trucks and the performance of our ALNS heuristic with respect to the designed MILP described in Section 4.1, i.e. running time and the quality of the solution. We first describe the generation of parameters and the test instances. We then present the results.

5.1 Parameters

The values of parameters that are dependent on the characteristics of the truck type are are shown in Table 5.1. The values used are based on sources of several institutes. Emission rates of electric and diesel trucks are reported in the Global Logistics Emissions Council (GLEC) Framework by the Smart Freight Centre. The transportation costs per km are determined as follows: For diesel trucks, the average liter diesel per km is calculated and for electric trucks the average kWh per km. Then the data of liter diesel and kWh prices are obtained from the Dutch Central Bureau of Statistics (CBS). Finally, the average liter diesel and kWh usage per km is multiplied with the corresponding prices. All parameters used for the ALNS mechanisms are copied from Wouda and Lan (2023) and shown in Table 5.2. The parameters used in the ALNS algorithm are grouped into three categories. The first group define all parameters used for the selection procedure incorporating the roulette wheel mechanism. The second group of parameters is used to configure the record-to-record travel method. The third group of parameters are operator specific parameters, calibrating their impact on the solution. The total run times for small and large instances have been chosen based on convergence times.

Table 5.1: Truck related parameters.

Parameters	Description	Values	
		Diesel	Electric
e_t	CO2 emission per tonne-kilometre	0.94	0.10
f_t	transportation costs per kilometre	0.45	1.11
A_t	Action radius in kilometres	-	300
w	Empty weight in tonnes	26	26
Q	Capacity in tonnes	12	12

5.2 Supply

The supply is generated using a clipped normal distribution with mean = 3 and standard deviation = 1 with a lower bound of 1, see Figure 5.1. A mean of 3 is chosen, as the assumption is made that the average waste of a construction site is a quarter of the capacity of a truck. Then the supplies are randomly assigned to all construction sites.

5.3 Experimental regions and results

In this section, we will describe how we generated the test instances, the design choices made according to the regions characteristics, and the corresponding results and analysis. As the designed

Table 5.2: Parameters used in the ALNS heuristic.

Group	Parameters	Description	Values
	R_t	Total run-time small instances	$5s$
	R_t	Total run-time large instances	$60s$
I	σ_1	New global solution	25
	σ_2	Better solution reward	5
	σ_3	Accepted solution	1
	σ_4	Rejected solution	0
	η	Decay rate	0.8
II	T_0	Initial threshold	0.02%
	T	Final threshold	0%
	N_i	Number of iterations to update T small instances	5000
	N_i	Number of iterations to update T large instances	15000
III	Ω	degree of destruction	0.05
	L^{max}	Maximum string cardinality	10
	K^{max}	Maximum string removals	2

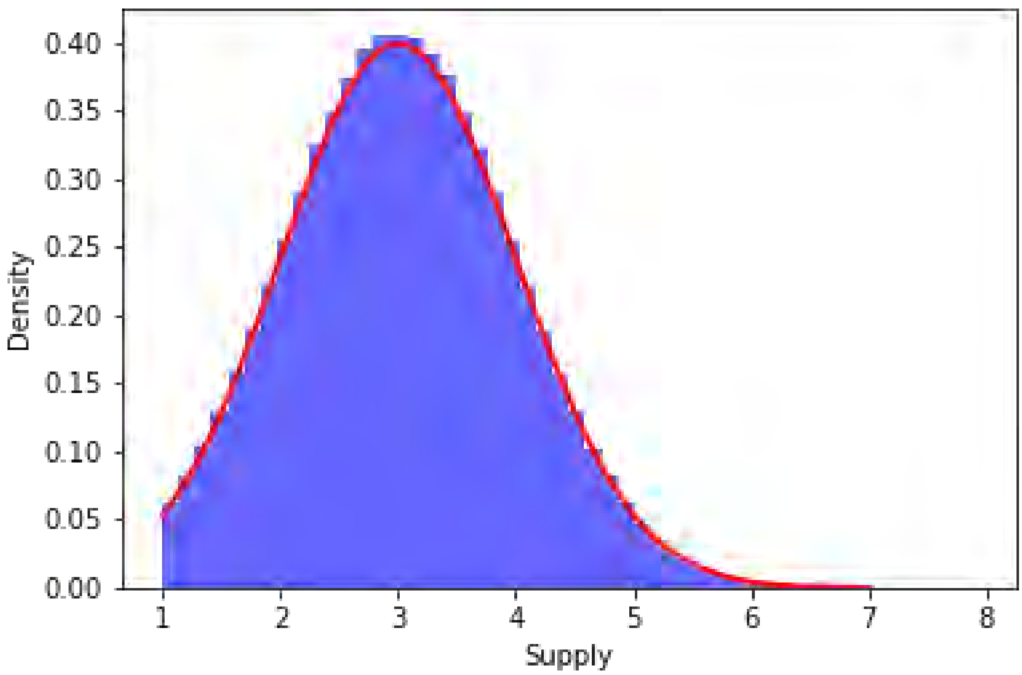


Figure 5.1: Normal distribution with mean = 3, standard deviation = 1 and lower bound = 1

MILP is not scalable, we first experiment on a small instance to validate the performance of the proposed heuristic against the exact method. Gurobi is used to find an exact optimal solution for the MILP. Then the heuristic is experimented on a large instance.

The test instances are generated to represent the three pilot regions: Greater Amsterdam Region (Netherlands), Germany and Poland. Since Poland and Germany have approximately the same geographical size and characteristics, only a test instance of Germany is generated, due to better available geographical data. Note that distance matrices are obtained by requesting the shortest path between two locations for the real road network, using Rapid API. The specifics are given in Sections 5.3.1 and 5.3.2.

5.3.1 Greater Amsterdam Region

For the Greater Amsterdam Region, 'Sloterdijk' has been selected as the depot, due to its logistics importance. For the waste stations, five locations are selected of one the largest recycling company (Renewi) of the Netherlands. The five selected locations are located in industrial areas of large cities and are listed in Table A.1. The construction sites are randomly sampled within the Greater Amsterdam Region, approximately within a radius of $300km$. To ensure all points are within the Greater Amsterdam Region, we verify if the points lie in the set of multi-polygons, obtained from the open source database of the CBS. An example of an instance with 25 sampled construction sites is illustrated in Figure A.1, where the blue point represents the depot, green the construction sites and red the waste stations.

To determine the instance size used for this region, we solved the MILP for several instance sizes, with a time-limit of $3,600$ seconds. The CPU time in seconds are shown in Table 5.3. As can be seen in Table 5.3, the running times explode exponentially and cannot obtain the optimal solution within an hour for an instance of size 20. Another reason for this, besides the non-scalable nature of an OVRP, is the small radius of this region. The Greater Amsterdam region is approximately $300km$, this means that with relative small instance sizes, all routes can be done by either diesel or electric trucks. Since, we assume that electric trucks have an action radius of $300km$. This increases the number of feasible solutions and thus the number of nodes our model has to explore. Furthermore, this also means that the extreme solutions (optimal cost and optimal emission) are simply the complete opposites of each other. Where, all routes will be done by diesel trucks when optimizing on costs, and by electric trucks when optimizing on emission.

To make the Greater Amsterdam Region scientifically more intriguing, we will reduce the action radius of electric trucks to $100km$. This will positively influence the running time and the trade-off between the cost and emission objectives. Furthermore, to make the instance more realistic with respect of the geographical distribution of the construction sites. Approximately half of the locations are sampled within the Randstad area, and the other half outside of the Randstad. An example of such an instance is shown in Figure 5.2. The new CPU time for solving the model with an instance of size 20 is $206.87s$, also resulting in a 0.0% optimality gap. Hence, this is a justifiable setting of parameters for solving the model with the largest possible instance in terms of an acceptable running time.

Figure 5.2: Realistic instance of Greater Amsterdam Region with 20 sampled construction sites

Table 5.3: Running times of MILP.

Construction site sets	CPU time s	Optimality gap in %
5	0.09	0.0
10	3.22	0.0
15	59.43	0.0
20	3600	9.72

After solving the models S_{cost} and $S_{emission}$ with the exact method, the following boundary points were obtained: $(f_e^N; f_c^{IDEAL}) = (24,217; 388.28)$ and $(f_e^{IDEAL}; f_c^N) = (20,024; 538)$. The running times for models S_{cost} and $S_{emission}$ were $206.87s$ and $179.63s$ respectively. With the introduction of electric trucks, we were able to reduce emission by approximately 17%, which is already a significant amount. However, it comes at an expense of an increase of approximately 39% on transportation costs. The routes obtained for both models are illustrated in Figures 5.3 and 5.4. Note that routes done by diesel trucks are shown by purple routes, and electric routes by yellow routes. From Figure 5.3 it can be observed that diesel routes are favored when optimizing on cost due to their cost efficiency w.r.t. elect ric trucks. Figure 5.4 shows that although optimizing on emission, a significant amount of locations are still visited by diesel trucks. This is a result of the distance limit of electric trucks. Due to this limitation, we observe that only locations in the

Figure 5.3: MILP: optimal routes when solving model S_{cost} for Greater Amsterdam Region with 20 sampled construction sites

Randstad are selected for routes done by electric trucks. Finally, when analyzing the choices within a route, we expected that construction sites with smaller supply would be visited early in a route. This was expected, as emission minimization depends on load optimization. To be efficient, locations with higher loads should be visited later in the routes, since we want to minimize the distance traveled with larger loads. However, after analyzing the routes, we did not observe a clear preference of construction sites with smaller supply to be visited early. Presumably, this is caused by the relatively small distances between locations on a route, which makes load optimization less significant.

In Figure 5.5 the Pareto Front, obtained using the ε-constraint, can be seen. If we follow the non-dominated points in Figure 5.5 from right to left, it can be observed that the decrease in emissions does not impact the costs much at first, almost following a linear trend. However, when the ε-constraint gets tighter towards the ideal point w.r.t. emission, the cost is increasing in a more exponential trend. The sudden exponential increase is caused by inefficient routes. When

Figure 5.4: MILP: optimal routes when solving model $S_{emission}$ for Greater Amsterdam Region with 20 sampled construction sites

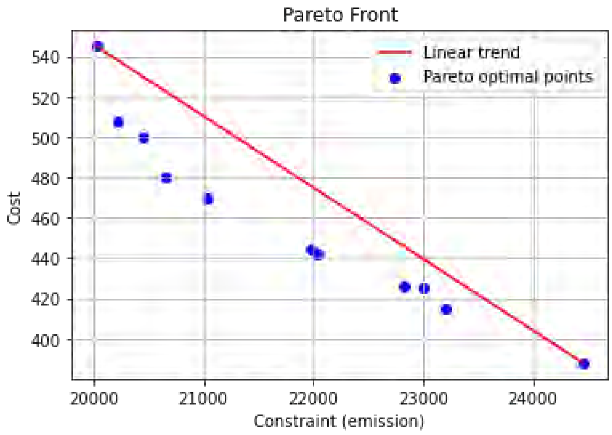


Figure 5.5: MILP: pareto Front for Greater Amsterdam Region

moving to the ideal emission point, construction sites that lie on the path of a route on a diesel truck visiting even further points, but are within the distance limit of an electric truck, will now be visited by an electric truck. However, a diesel truck is still needed to visit the locations that lie further. This results in in two trucks needed to visit these constructions sites, while it could have been done by one diesel truck. When a compromise is desired by the stakeholders between cost efficiency and emission, one should choose a point on the pareto front before the cost starts increasing in a more exponential trend. Then the points to be considered should be all points to the right from the point with an emission of approximately $22,000$ $kgCO2_e$.

Then the ideal points obtained by our proposed ALNS are as follows: After solving model S_{cost} the best obtained solution resulted in a cost of 393.20 and emission of $24,932$. This results in an optimality gap of 1.27% for the primary objective, namely the cost, and an optimality gap of 2.95% for the emission. The ALNS was able to find a near-optimal solution. The optimality gap of the emission was slightly higher w.r.t. to the cost, which is logical since it is not prioritized in this model. However, it still obtained a very reasonable optimality gap. Then, when solving model $S_{emission}$ we obtained an emission of $20,757$ and cost of 558.23, resulting in an optimality gap of 3.66% for the primary objective emission and an optimality gap of 3.72% for the cost. W.r.t model S_{cost}, the optimality gap for the objective that is being optimized, is slightly higher. This is seemingly caused by the higher complexity of model $S_{emission}$. Where for S_{cost}, we know in advance that all routes will be taken by diesel truck, due to their better cost efficiency. Therefore the ALNS is mostly focusing on optimizing the distance lengths of the routes. But for $S_{emission}$ it also needs to find the best distribution between what locations need to be visited by electric or diesel trucks, thus increasing the complexity of finding the best possible solution. However, an optimality gap of 3.66% is still very promising, taking into account that the ALNS only used approximately $1/40$ of the running time of the exact method. The routes obtained by the ALNS are shown in Figures 5.6 and 5.7. When comparing Figures 5.4 and 5.7, we observe that exactly the same locations are visited by electric trucks, however the ALNS was not able to find the exact same routes. To validate that these results are not outliers, the optimality gaps for the primary objective of models S_{cost} and $S_{emission}$ were computed for 25 different instances and the resulting boxplot is shown in Figure A.2.

To analyse if relatively more emission could be reduced on an instance of larger scale, a test

Figure 5.6: ALNS: routes when solving model S_{cost} for Greater Amsterdam Region with 20 sampled construction sites

Figure 5.7: ALNS: routes when solving model $S_{emission}$ for Greater Amsterdam Region with 20 sampled construction sites

instance was generated with 100 construction sites, which is shown in Figure A.3. Then the ideal points obtained by our proposed ALNS are as follows: After solving model S_{cost} the best obtained solution resulted in a cost of $1,584$ and emission of $101,058$. Then, when solving model $S_{emission}$ we obtained an emission of $76,982$ and cost of $3,176$. The ALNS was able to reduce emission by approximately 24%, so relatively 6% more w.r.t. the smaller instance of 20 construction sites. However, it comes at a relatively larger expense, namely an increase of approximately 100% on costs. This is caused by the increase in locations that can be visited by electric trucks, especially on the border of their distance limit. While this is beneficial for the emission reduction, it means even more cost inefficiency as explained before. From an absolute standpoint, the reduction of approximately $24,000$ $kgCO2_e$ emissions was realized with a relatively modest increase of roughly €$1,600$ on transportation costs. The routes obtained for both models are illustrated in Figures A.4 and A.5. Again, it is observed that all locations within the Randstad are visited by electric trucks. So also for larger instances, the ALNS is capable of replicating the patterns of the optimal solution generated by the exact method, as previously observed for the smaller instance.

5.3.2 Germany

For Germany the depot is located in Frankfurt. Frankfurt has an important role in the development of chemical technology and production, as Frankfurt-Höchst Industrial Park in one of Europe's largest chemical sites. The waste stations are located in ten of the most largest and dense cities of Germany, and are listed in Table A.2. The construction sites are randomly sampled within Germany, thus approximately over a width of 625 km and height of 850 km. Again a set of multi-polygons is used to ensure all sampled points lie in the desired area, which in this case is Germany. Then for the size of the small and large test instance, the same values will be used as in Section 5.3.1. Note that for the experiments in this section, the default action radius is used as described in Table 5.1. Then a test instance with 20 constructions sites, randomly sampled throughout Germany, is generated and shown in Figure A.6.

After solving the models S_{cost} and $S_{emission}$ for the instance shown in Figure A.6 with the exact method, the following boundary points were obtained: $(f_e^N; f_c^{IDEAL}) = (81,311; 1,305)$ and $(f_e^{IDEAL}; f_c^N) = (61,839; 2,306)$. The running times for models S_{cost} and $S_{emission}$ were $167.38s$ and $113.95s$ respectively, which is significantly shorter than for the instance of same size of the Greater Amsterdam Region. The routes obtained for both models are illustrated in Figures 5.8 and 5.9. This is most likely caused due to the larger variety of distances between locations, hence it is more evident which pairs to couple sequentially in a route, subsequently the problem becomes less complex to solve. Then electric trucks have enabled us to reduce emission by approximately 24% with an increase of 77% on costs. Relatively, more emission can be reduced w.r.t the Greater Amsterdam Region, but it comes at a higher financial expense. This is caused by the longer distance of the routes, subsequently it causes more cost inefficiency when optimizing on emission. When analyzing the routes shown in Figure 5.9, we did observe some preference of construction sites with smaller supply over construction sites with larger supply to be visited earlier in routes done by diesel trucks when optimizing on emission. Presumably, this is caused by the relative longer distances between locations on a route. Since the longer the distances, the more impact early load has on the emission of a single route.

In Figure 5.10 the Pareto Front, obtained using the ε-constraint, can be seen. If we follow the non-dominated points in Figure 5.10 from right to left, it can be observed that the decrease

Figure 5.8: MILP: optimal routes when solving model S_{cost} for Germany

Figure 5.9: MILP: optimal routes when solving model $S_{emission}$ for Germany

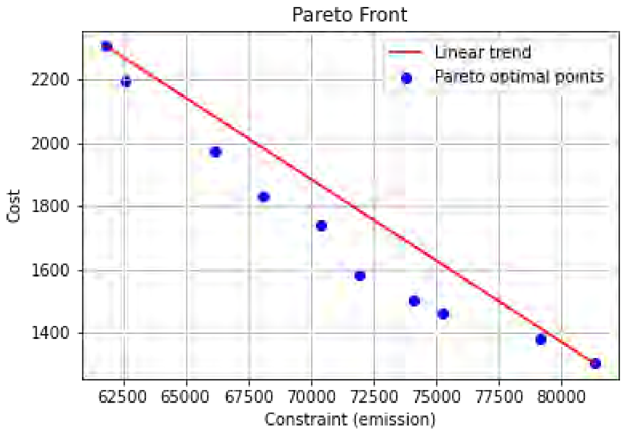


Figure 5.10: MILP: pareto front for Germany

of emission again does not impact the costs significantly at first. The solution that is located at approximately $(72,000; 1,600)$ was able to reduce 50% of the total emission reduction from the nadir to ideal point, with only a 30% increase of the total cost increase from ideal to nadir point. Whereas, all solutions to the left of this point, see a higher expense on costs with relatively even smaller portions of reduction of emission between solutions. However, we do not necessarily see an exponential increase in costs, but a steeper linear increase.

When a compromise is desired by the stakeholders between cost efficiency and emission, one should consider the point referenced above, as it was able to reduce the most emission with the smallest possible increase in costs in terms of percentages.

For model S_{cost} The ALNS was able to find the exact same optimal solution as the exact method had obtained, thus resulting in a 0.0% optimality gap. Then for model $S_{emission}$ the optimal solution obtained by the ALNS resulted in an emission of $62,333$ and cost of $3,252$, thus respectively an optimality gap of 0.8% and 2.0%. The solutions found by the ALNS are very promising as the largest optimality gap for the primary objective that is being optimized is only 0.8%. Again, like the Greater Amsterdam Region, the ALNS performs better on model S_{cost} than on $S_{emission}$. The route for the optimal solution when solving model $S_{emission}$ can be found in Figure 5.11. When comparing Figures 5.9 and 5.11, we observe that again exactly the same locations are visited by electric trucks, but in different route combinations and or orders. The ALNS is thus able to find all points that can be visited by electric trucks and recognizes that these should be visited when optimizing on emission. Then comparing the results of the ALNS on this instance and that of the Greater Amsterdam Region, we observe that the ALNS has a better performance on the instance of Germany. It appears that the ALNS is able to find better near optimal solutions when the distances between points are larger, meaning a larger geographical area. When comparing Figures 5.4 and 5.7, we observe that exactly the same locations are visited by electric trucks, however the ALNS was not able to find the exact same routes. To validate that these results are not outliers, the optimality gaps for the primary objective of models S_{cost} and $S_{emission}$ were computed for 25 different instances and the resulting boxplot is shown in Figure A.7.

Lastly, a test instance was generated with 100 construction sites, which is shown in Figure A.8. The the ideal points yielded by our proposed ALNS are presented below: Upon solving model

Figure 5.11: ALNS: routes when solving model $S_{emission}$ for Germany

S_{cost}, the best solution was achieved with a cost of $5,748$, with an emission value of $349,830$. Subsequently, when solving model $S_{emission}$, the best found solution had an emission of $269,100$, accompanied by a cost amounting to $8,622$. On this instance a reduction in emissions of roughly 23% was yielded, approximately the same relative reduction as on the smaller instance. However, it came at a relatively diminished cost, with only an elevation of 50% in expenses compared to the elevation of 77% on the smaller instance. Whereas for the Greater Amsterdam Region we observed a relatively amplified cost. Presumably, when the instance size grows, the locations that can be serviced by electric trucks increase, resulting in a more dense area which subsequently leads to less cost inefficiency.

From an absolute standpoint, the reduction of approximately $80,000$ $kgCO2_e$ emissions was realized with a relatively modest increase of roughly €$2,000$ on transportation costs.

The routes derived for both models are depicted in Figures 5.12 and 5.13. Upon conducting an analysis of the solution in terms of individual routes, the following is observed: The ALNS algorithm is able to prioritize the sequencing of construction sites characterized by relatively increasing supply towards the later segments of the routes. Additionally, the ALNS was able to allocate construction sites with higher supply to electric trucks and some with really modest supply to diesel trucks. This shows us that the load factor does have a significant impact on the emission.

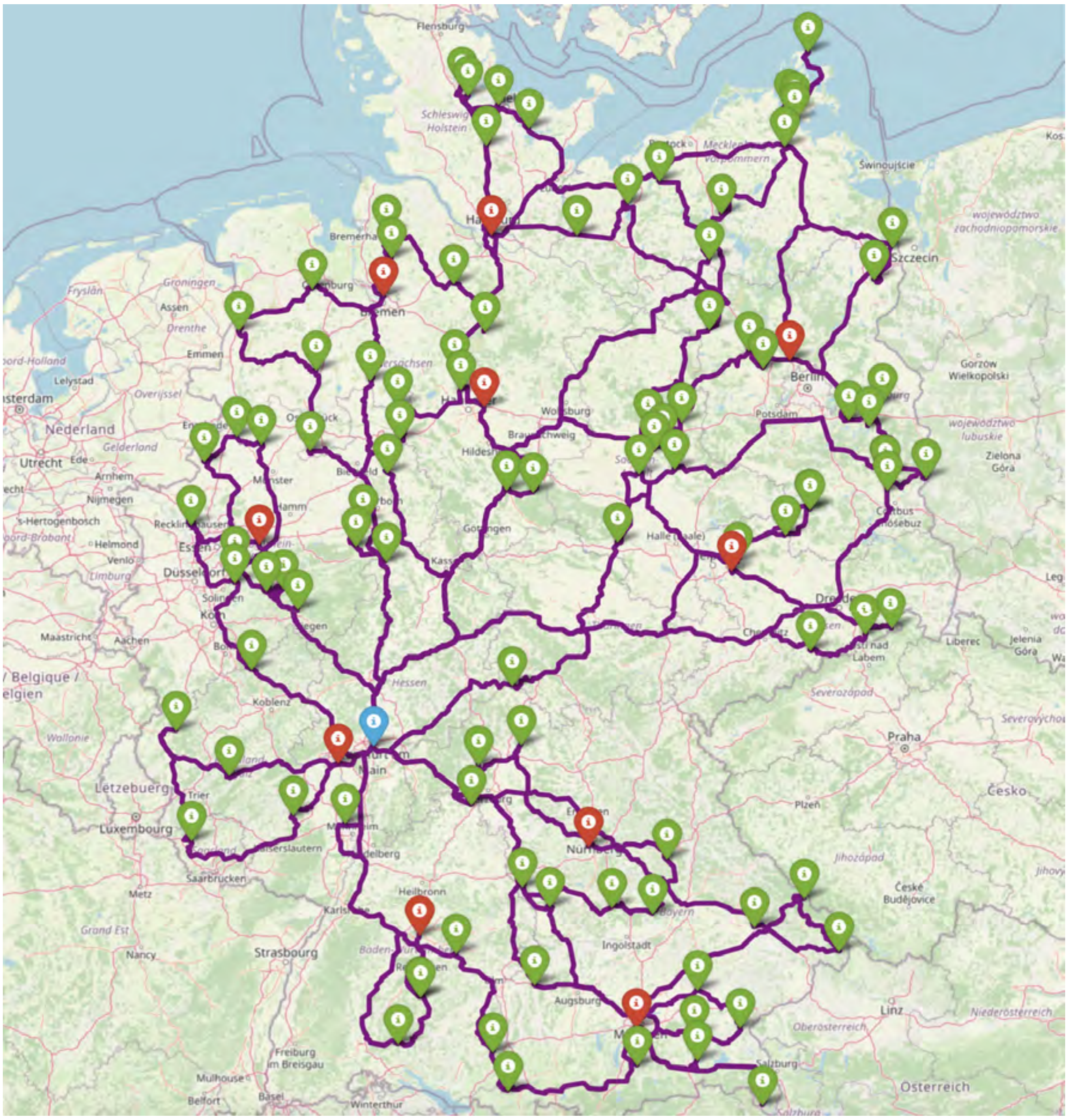


Figure 5.12: ALNS: routes when solving model S_{cost} for Germany

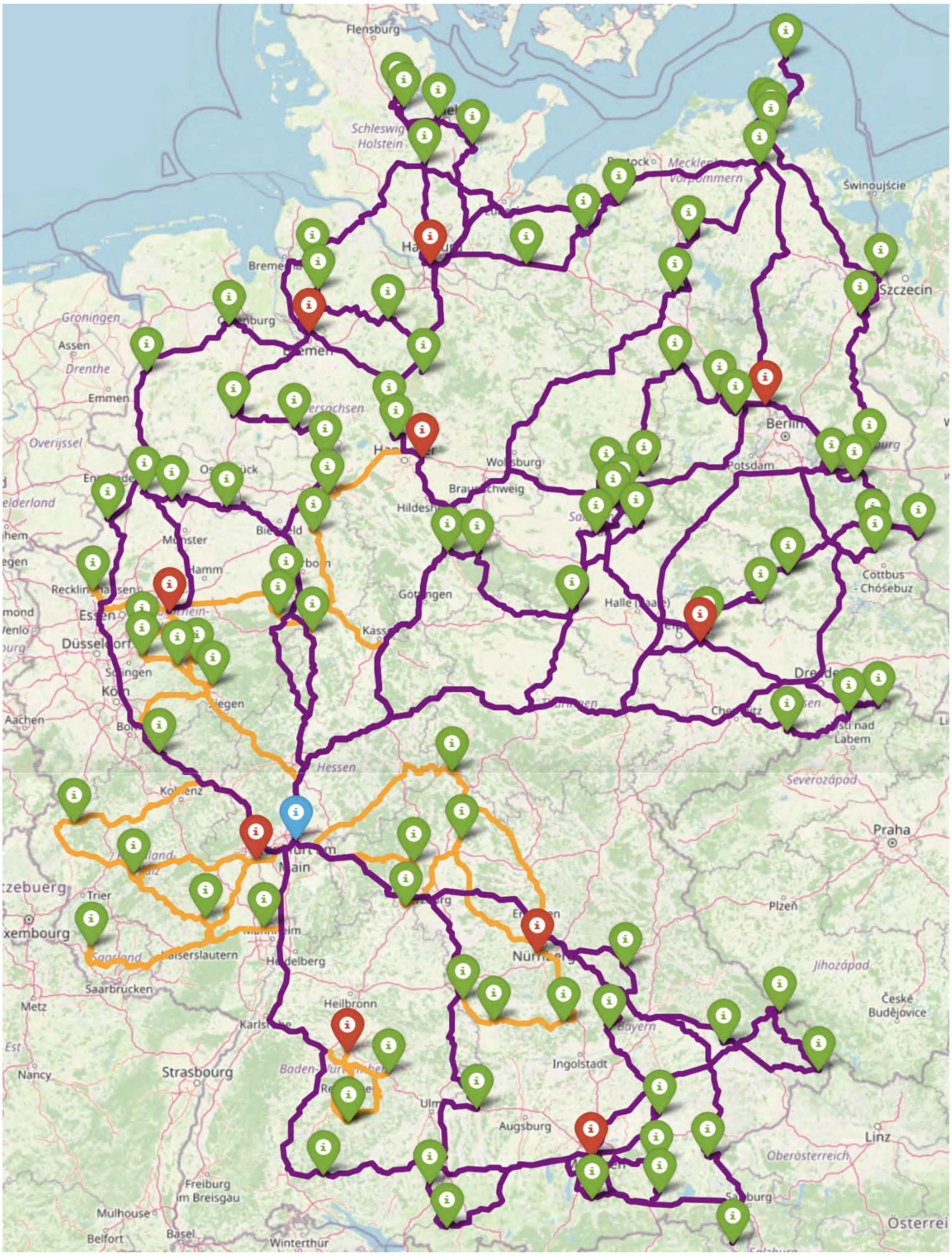


Figure 5.13: ALNS: routes when solving model $S_{emission}$ for Germany

6 Conclusions and future research

In this thesis, it was attempted to make the routing planning for the pilot regions of the CIRCULAR FOAM project more emission efficient, while investigating the effect on the costs. Specifically, it was investigated what the impact of electric trucks would have on the emission reduction and the patterns of the routes. A Bi-Objective Heterogeneous Open Vehicle Routing Problem was developed. The ε-constraint method was used to obtain the Pareto efficient set. Furthermore, an ALNS as heuristic was designed to approximate the ideal points for both objectives. The ALNS uses new, as well as existing destroy and repair operators, to adapt to the unique nature of the problem.

To test the trade-off between the costs and emission, a small instance was generated for the Greater Amsterdam Region and Germany and a Pareto efficient set was obtained. Investigating the ideal points of the Pareto efficient set yielded a reduction of 17% on emission at the expense of around 39% increase on costs, for the Greater Amsterdam Region. For Germany, a reduction of around 24% at the expense of 77%. These results indicate a relatively strong pattern between the size of the area and the potential emission reduction. However, this does come at a far higher expense. To validate the performance of the ALNS, a comparison was made to the ideal points of the exact solutions. It was found that the ALNS had very promising results, even finding the optimal solution of the exact solution for one of the models, and yielded an optimality gap of at most 3.72%. Furthermore, the ALNS showed signs of being able to prioritize the sequencing of constructions sites by visiting sites with relatively increasing supply towards the end of the routes. To investigate the effect of electric trucks on a more real-life scale, a large instance was generated for both regions mentioned. The results of the ALNS yielded a reduction of around 24% at a large expense of 100% increase on costs for the Greater Amsterdam Region and 23% and 50% respectively for Germany.

Interestingly when optimizing on emission, the cost inefficiency seemed to grow as the instance size becomes larger for the Greater Amsterdam Region, but the contrary happened for Germany as it became more cost efficient. Analysis indicated that the smaller instances of Germany were quite inefficient as for some routes, done by electric trucks, only one or two construction sites were served. Due to the increase in locations, the area became more dense and routes could contain more number of visits and were thus more cost efficient. While, for the smaller instances of the Greater Amsterdam Region it was seen that most routes had already maximized their capacity.

Based on the parameters and objectives used in this thesis, it has been shown that electric trucks can reduce the emission of the supply chain significantly, but at a relative high expense. However, with global warming in mind, a price tag cannot be put on the reduction of emission and it is up to the stakeholders of the CIRCULAR FOAM whether the trade-off is acceptable. Furthermore, subsidy is not accounted for in this thesis, which would lower the net cost of electric driving. Lastly, with future possible disincentives for fossil fuel, electric driving could become even more attractive.

Although the ALNS showed to be very promising, it also showed that if the problem to solve became more complex it had a small decline in performance. In future studies this relation should be further investigated, whether it can eventually result in poor quality solutions or that new domain specific operators could better deal with the increasing complexity. In the final course of this research, an unexpected finding was the apparent discord between the ALNS and the ε-constraint method, resulting in the inability of obtaining a Pareto front using the ALNS. While a harmonious synergy was expected between the two, given that the ALNS adaptively explores and exploits different parts of the solution space and the ε-constraint method would ensure a thorough exploration of the Pareto front. After investigating this finding, the following factors have been established: First and foremost, constraint rigidity. The ε-constraints are strict and therefore often clashed with the adaptive, fluid nature of the ALNS. As a result of the imposed constraints, the ALNS often found itself "trapped", unable to effectively adapt its search. Secondly, due to the predefined limits of the ε-constraint method, the ALNS is limited in its explorations of the search and often seemed to be stuck into specific regions of the solution space. Therefore, the following hypotheses have been postulated: There exist mismatched dynamics between the ALNS and ε-constraint method, and the cooperation between the two leads to sub-optimal solutions that do not represent the true Pareto front. The findings from this thesis underscore the importance of compatibility when combining optimization techniques, ensuring that the chosen techniques do not limit each other. If continuing this research, the weighted-sum method should be considered for obtaining the Pareto front, as it does not impose any constraints. However, the weighted-sum method does have its challenges as well, namely finding the correct set of weights. Especially when a generous number of points on the Pareto front is desired, a lot of calibration has to be done around the weights, which could take a large amount of time. For future research, it would be interesting to get a deeper understanding of the incompatibility between the ALNS and ε-constraint method, which could result in effective modifications where these two methods might have a harmonious synergy.

To extend the work of this thesis, modeling green-zones could be interesting to research, which denote low-emission zones that apply to trucks. Inside these zones, heavy duty diesel vehicles may only enter if a certain standard is met. In Germany, these zones exist in almost all city centers, and in the Netherlands in Amsterdam, Utrecht and Arnhem. Therefore, it could be very interesting to research, as electric trucks could even hold a more significant strategic value. Additionally, it would be intriguing to include speed optimization as this could lead to even fewer as emission, as it leads to fewer fuel and electricity usage.

Finally, the aim of this thesis was to model the logistic network of the CIRCULAR FOAM project, and the model was thus designed from scratch. The models developed in this thesis have provided a blueprint for the unique routing problem of the CIRCULAR FOAM project. In this thesis, it has been successfully implemented to let emission depend on dynamic loads, which has not yet been seen in previous studies, and that it should be included as a factor when optimizing on emission as it impacts decision making of the sequencing of routes. Furthermore, it has been confirmed that the ALNS is able to find near optimal solutions within a short amount of time on our OVRP, solving the issue of poor scalability, as shown in existing literature on VRPs. Lastly, the model provides insight between the trade-offs of transportation costs and emission, which could be insightful for the stakeholders of CIRCULAR FOAM.

Acknowledgements

I would like to express my deep and sincere gratitude to prof. dr. van der Mei, for his interest, guidance and valuable suggestions. I would also like to thank Elenna Dugundji for her enthusiasm and ideas to help me keep on track and steer me in the right direction. I also wish to acknowledge Ruurd Jan Buijs, for his daily guidance, and many useful discussions. I am also grateful for the guidance by Leon Lan implementing the ALNS. Then, I would like to thank Dr. Berkhout for being my second reader.

Lastly, special thanks should be given to all AIOs of the Stochastic department of CWI, whom I have had the pleasure of getting to know their enthusiasm, and kind spirits.

References

Ade Irawan, Chandra, Muhammad Dan-Asabe Abdulrahman, Said Salhi, and Martino Luis (2022). "An efficient matheuristic algorithm for bi-objective sustainable closed-loop supply chain networks". In: *IMA Journal of Management Mathematics* 33.4, pp. 603–636.

Bevilaqua, Andre, Diego Bevilaqua, and Keiji Yamanaka (2019). "Parallel island based memetic algorithm with Lin–Kernighan local search for a real-life two-echelon heterogeneous vehicle routing problem based on Brazilian wholesale companies". In: *Applied Soft Computing* 76, pp. 697–711.

Carter, C.R. and L.M. Ellram (1998). "Reverse Logistics: A Review of the Literature and Framework for Future Investigation". In: *Journal of Business Logistics* 19.1, pp. 85–102.

Christiaens, Jan and Greet Vanden Berghe (2020). "Slack induction by string removals for vehicle routing problems". In: *Transportation Science* 54.2, pp. 417–433.

Conrad, Ryan G and Miguel Andres Figliozzi (2011). "The recharging vehicle routing problem". In: *Proceedings of the 2011 industrial engineering research conference*. Vol. 8. IISE Norcross, GA.

Cordeau, J-F., G. Laporte, M.W.P. Savelsbergh, and D. Vigo (2007). "Vehicle routing". In: *Handbooks in operations research and management science* 14, pp. 367–428.

Council of Logistics Management (1999). "1999 Membership Roster". In: *Journal of Business Logistics*, p. 506.

Crainic, Teodor Gabriel, Fausto Errico, Walter Rei, and Nicoletta Ricciardi (2016). "Modeling demand uncertainty in two-tier city logistics tactical planning". In: *Transportation Science* 50.2, pp. 559–578.

Crainic, Teodor Gabriel, Michel Gendreau, and Bernard Gendron (2021). *Network design with applications to transportation and logistics*. Springer Nature.

Crainic, Teodor Gabriel, Nicoletta Ricciardi, and Giovanni Storchi (2004). "Advanced freight transportation systems for congested urban areas". In: *Transportation Research Part C: Emerging Technologies* 12.2, pp. 119–137.

— (2009). "Models for evaluating and planning city logistics systems". In: *Transportation science* 43.4, pp. 432–454.

Demharter, Anton (1998). "Polyurethane rigid foam, a proven thermal insulating material for applications between +130°C and -196°C". In: *Cryogenics* 38.1, pp. 113–117.

Demir, Emrah, Tolga Bektaş, and Gilbert Laporte (2012). "An adaptive large neighborhood search heuristic for the pollution-routing problem". In: *European journal of operational research* 223.2, pp. 346–359.

— (2014). "The bi-objective pollution-routing problem". In: *European journal of operational research* 232.3, pp. 464–478.

Dolati Neghabadi, Parisa, Karine Evrard Samuel, and Marie-Laure Espinouse (2019). "Systematic literature review on city logistics: overview, classification and analysis". In: *International Journal of Production Research* 57.3, pp. 865–887.

Dueck, Gunter (1993). "New optimization heuristics: The great deluge algorithm and the record-to-record travel". In: *Journal of Computational physics* 104.1, pp. 86–92.

Erdoğan, Sevgi and Elise Miller-Hooks (2012). "A green vehicle routing problem". In: *Transportation research part E: logistics and transportation review* 48.1, pp. 100–114.

Foltyński, Marcin (2014). "Electric fleets in urban logistics". In: *Procedia-social and behavioral sciences* 151, pp. 48–59.

Guastaroba, Gianfranco, Maria Grazia Speranza, and Daniele Vigo (2016). "Intermediate facilities in freight transportation planning: a survey". In: *Transportation Science* 50.3, pp. 763–789.

He, Lihua and Lin Zhang (2022). "A bi-objective optimization of energy consumption and investment cost for public building envelope design based on the ε-constraint method". In: *Energy and Buildings* 266, p. 112133.

Hromkovič, Juraj (2013). *Algorithmics for hard problems: introduction to combinatorial optimization, randomization, approximation, and heuristics*. Springer Science & Business Media.

Hwang, Ching-Lai, Abu Syed Md Masud, Ching-Lai Hwang, and Abu Syed Md Masud (1979). "Methods for multiple objective decision making". In: *Multiple Objective Decision Making—Methods and Applications: A State-of-the-Art Survey*, pp. 21–283.

Imran, Arif, Said Salhi, and Niaz A Wassan (2009). "A variable neighborhood-based heuristic for the heterogeneous fleet vehicle routing problem". In: *European Journal of Operational Research* 197.2, pp. 509–518.

Irawan, Chandra Ade, Dylan Jones, and Djamila Ouelhadj (2017). "Bi-objective optimisation model for installation scheduling in offshore wind farms". In: *Computers & Operations Research* 78, pp. 393–407.

Kancharla, Surendra Reddy and Gitakrishnan Ramadurai (2019). "Multi-depot two-echelon fuel minimizing routing problem with heterogeneous fleets: model and heuristic". In: *Networks and Spatial Economics* 19, pp. 969–1005.

Kergosien, Yannick, Ch Lenté, J-C Billaut, and Sandrine Perrin (2013). "Metaheuristic algorithms for solving two interconnected vehicle routing problems in a hospital complex". In: *Computers & Operations Research* 40.10, pp. 2508–2518.

Li, Hongqi, Yinying Liu, Kaihang Chen, and Qingfeng Lin (2020). "The two-echelon city logistics system with on-street satellites". In: *Computers & Industrial Engineering* 139, p. 105577.

Li, Hongqi, Yinying Liu, Xiaorong Jian, and Yingrong Lu (2018). "The two-echelon distribution system considering the real-time transshipment capacity varying". In: *Transportation Research Part B: Methodological* 110, pp. 239–260.

Lu, Hao, Xingwen Zhang, and Shuang Yang (2019). "A learning-based iterative method for solving vehicle routing problems". In: *International conference on learning representations*.

Mancini, Simona (2016). "A real-life multi depot multi period vehicle routing problem with a heterogeneous fleet: Formulation and adaptive large neighborhood search based matheuristic". In: *Transportation Research Part C: Emerging Technologies* 70, pp. 100–112.

Marler, R Timothy and Jasbir S Arora (2004). "Survey of multi-objective optimization methods for engineering". In: *Structural and multidisciplinary optimization* 26, pp. 369–395.

Mavrotas, George (2009). "Effective implementation of the ε-constraint method in multi-objective mathematical programming problems". In: *Applied mathematics and computation* 213.2, pp. 455–465.

Miller, Clair E, Albert W Tucker, and Richard A Zemlin (1960). "Integer programming formulation of traveling salesman problems". In: *Journal of the ACM (JACM)* 7.4, pp. 326–329.

Mohammed, Mazin Abed, Mohd Khanapi Abd Ghani, Raed Ibraheem Hamed, Salama A Mostafa, Mohd Sharifuddin Ahmad, and Dheyaa Ahmed Ibrahim (2017). "Solving vehicle routing problem by using improved genetic algorithm for optimal solution". In: *Journal of computational science* 21, pp. 255–262.

Nilsson, Christian (2003). “Heuristics for the traveling salesman problem”. In: *Linkoping University* 38, pp. 00085–9.

Osyczka, Andrzej (1984). *Multicriterion optimisation in engineering*. Halsted Press.

Paul, Arjun, Ravi Shankar Kumar, Chayanika Rout, and Adrijit Goswami (2021). “A bi-objective two-echelon pollution routing problem with simultaneous pickup and delivery under multiple time windows constraint”. In: *Opsearch*, pp. 1–32.

Perboli, Guido, Roberto Tadei, and Daniele Vigo (2011). “The two-echelon capacitated vehicle routing problem: Models and math-based heuristics”. In: *Transportation Science* 45.3, pp. 364–380.

Pishvaee, Mir Saman and Jafar Razmi (2012). “Environmental supply chain network design using multi-objective fuzzy mathematical programming”. In: *Applied Mathematical Modelling* 36.8, pp. 3433–3446.

Pisinger, David and Stefan Ropke (2007). “A general heuristic for vehicle routing problems”. In: *Computers & operations research* 34.8, pp. 2403–2435.

— (2019). “Large neighborhood search”. In: *Handbook of metaheuristics*, pp. 99–127.

Ropke, Stefan (2005). “Heuristic and exact algorithms for vehicle routing problems”. In: *Unpublished PhD thesis, Computer Science Department, University of Copenhagen*.

Ropke, Stefan and David Pisinger (2006a). “A unified heuristic for a large class of vehicle routing problems with backhauls”. In: *European Journal of Operational Research* 171.3, pp. 750–775.

— (2006b). “An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows”. In: *Transportation science* 40.4, pp. 455–472.

Salari, Majid, Paolo Toth, and Andrea Tramontani (2010). “An ILP improvement procedure for the open vehicle routing problem”. In: *Computers & Operations Research* 37.12, pp. 2106–2120.

Santini, Alberto, Stefan Ropke, and Lars Magnus Hvattum (2018). “A comparison of acceptance criteria for the adaptive large neighbourhood search metaheuristic”. In: *Journal of Heuristics* 24, pp. 783–815.

Savelsbergh, Martin and Tom Van Woensel (2016). “50th anniversary invited article—city logistics: Challenges and opportunities”. In: *Transportation Science* 50.2, pp. 579–590.

Schneider, Michael, Andreas Stenger, and Dominik Goeke (2014). “The electric vehicle-routing problem with time windows and recharging stations”. In: *Transportation science* 48.4, pp. 500–520.

Shaw, Paul (1998). “Using constraint programming and local search methods to solve vehicle routing problems”. In: *International conference on principles and practice of constraint programming*. Springer, pp. 417–431.

Sluijk, Natasja, Alexandre M Florio, Joris Kinable, Nico Dellaert, and Tom Van Woensel (2022). “Two-echelon vehicle routing problems: A literature review”. In: *European Journal of Operational Research*.

Soysal, Mehmet, Jacqueline M Bloemhof-Ruwaard, and Tolga Bektaş (2015). “The time-dependent two-echelon capacitated vehicle routing problem with environmental considerations”. In: *International Journal of Production Economics* 164, pp. 366–378.

Wang, Kangzhou, Yeming Shao, and Weihua Zhou (2017). “Matheuristic for a two-echelon capacitated vehicle routing problem with environmental considerations in city logistics service”. In: *Transportation Research Part D: Transport and Environment* 57, pp. 262–276.

Wang, Yong, Shuanglu Zhang, Kevin Assogba, Jianxin Fan, Maozeng Xu, and Yinhai Wang (2018). "Economic and environmental evaluations in the two-echelon collaborative multiple centers vehicle routing optimization". In: *Journal of Cleaner Production* 197, pp. 443–461.

Wang, Yusheng, Yongxi Huang, Jiuping Xu, and Nicole Barclay (2017). "Optimal recharging scheduling for urban electric buses: A case study in Davis". In: *Transportation Research Part E: Logistics and Transportation Review* 100, pp. 115–132.

Wang, Ziqi and Peihan Wen (2020). "Optimization of a low-carbon two-echelon heterogeneous-fleet vehicle routing for cold chain logistics under mixed time window". In: *Sustainability* 12.5, p. 1967.

Wouda, Niels A and Leon Lan (2023). "ALNS: a Python implementation of the adaptive large neighbourhood search metaheuristic". In: *Journal of Open Source Software* 8.81, p. 5028.

Yan, Yimo, Andy HF Chow, Chin Pang Ho, Yong-Hong Kuo, Qihao Wu, and Chengshuo Ying (2022). "Reinforcement learning for logistics and supply chain management: Methodologies, state of the art, and future opportunities". In: *Transportation Research Part E: Logistics and Transportation Review* 162, p. 102712.

Zadeh, Lofti (1963). "Optimality and non-scalar-valued performance criteria". In: *IEEE transactions on Automatic Control* 8.1, pp. 59–60.

A Appendix

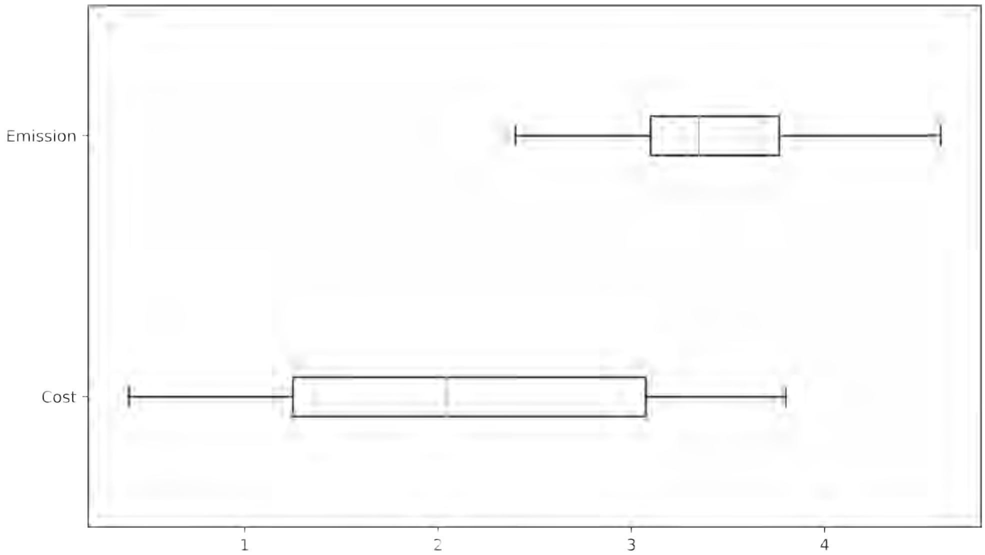
Table A.1: Locations of waste stations for Greater Amsterdam Region.

Cities
Amsterdam
Rotterdam
Eindhoven
Amersfoort
Drachten

Table A.2: Locations of waste stations for Germany.

Cities
Düsseldorf
Frankfurt
Munich
Hamburg
Leipzig
Berlin
Hanover
Stuttgart
Nuremberg
Bremen

Figure A.1: Example instance of Greater Amsterdam Region with 25 sampled construction sites



Note: due to computational tractability 25 instances are used that were solvable using 2000 work units

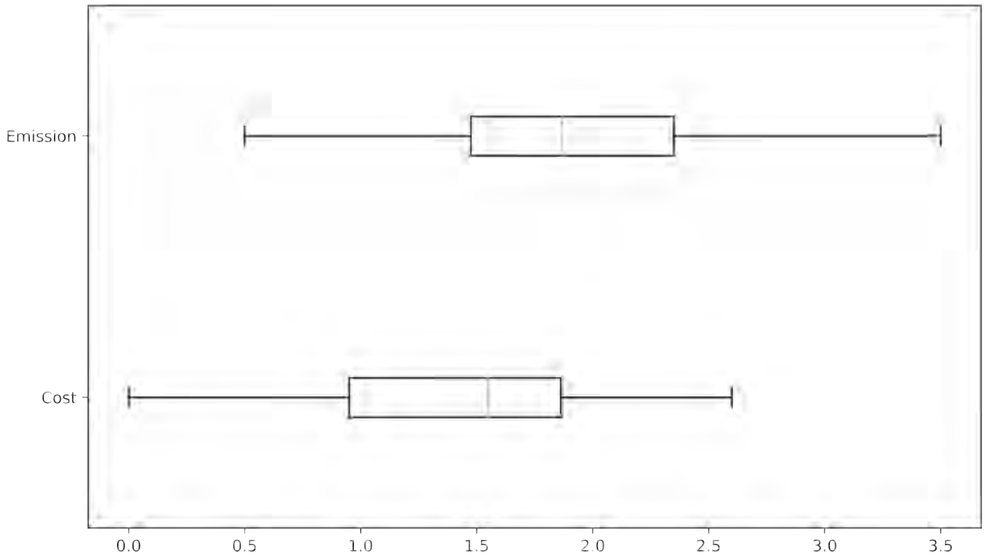
Figure A.2: Boxplot of optimality gaps for the primary objectives for the Greater Amsterdam Region

Figure A.3: Instance of Greater Amsterdam Region with 100 sampled construction sites

Figure A.4: ALNS: routes when solving model S_{cost} for Greater Amsterdam Region with 100 sampled construction sites

Figure A.5: ALNS: routes when solving model $S_{emission}$ for Greater Amsterdam Region with 100 sampled construction sites

Figure A.6: Instance of Germany with 20 sampled construction sites



Note: due to computational tractability 25 instances are used that were solvable for the exact method using 2000 work units

Figure A.7: Boxplot of optimality gaps for the primary objectives for Germany

Figure A.8: Instance of Germany with 100 sampled construction sites