MASTERARBEIT | MASTER'S THESIS

Titel | Title

Urban Last-Mile Delivery Optimization in Vienna: Combining Mobile Parcel Lockers and Cargo Bike Deliveries - A Quantitative Analysis

verfasst von | submitted by

Lachezar Georgiev B.A.

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2024

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 915
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Betriebswirtschaft
Betreut von | Supervisor:	Univ.-Prof. Dr. Jan Fabian Ehmke
Mitbetreut von | Co-Supervisor:	Dr. Yannick Scherr B.Sc. M.Sc.

Zusammenfassung

Diese Arbeit stellt einen innovativen Ansatz für die Zustellung auf der letzten Meile vor, bei dem mobile Packstationen und Lastenfahrräder kombiniert werden. Dafür werden zwei Gruppen von Kunden definiert: Selbstabholer, die ihre Pakete von der mobilen Packstation selbst abholen, sowie Kunden, die von den Lastenfahrrädern direkt nach Hause beliefert werden. Die Studie untersucht, ob die Kombination aus mobilen Packstationen und Lastenfahrrädern in der Stadt zu niedrigeren Zustellkosten führen kann. Für das Tourenplanungsproblem in der Arbeit wird ein zweistufiges Transportsystem mit einer zeitlich synchronisierten heterogenen Fahrzeugflotte berücksichtigt. In der ersten Stufe des Transportsystems fungieren Transporter sowohl als Satelliten (Ort, wo die Fahrzeuge aus der zweiten Stufe beladen werden) als auch als mobile Packstationen und bedienen Selbstabholer innerhalb derer Zeitfenster. Lastenfahrräder operieren in der zweiten Stufe und bedienen diejenigen Kunden, die direkt nach Hause beliefert werden möchten. Für die Lösung des Problems wurden eine Konstruktionsheuristik und eine Metaheuristik *(Adaptive Large Neighbourhood Search)* implementiert und an Instanzen mit realen Daten aus Wien getestet. Die Ergebnisse zeigen, dass die Kombination von mobilen Packstationen und Lastenfahrrädern zu signifikanten Kostensenkungen führt, wenn die Wartezeit der Transporter, die für Selbstabholer gewährt wird, minimiert und die Anzahl der Selbstabholer maximiert wird. Kürzere Wartezeiten bieten mehr Flexibilität und ein höherer Anteil an Selbstabholer sorgt für Synergieeffekte und Kosteneffizienzen. Praktische Einschränkungen können jedoch das Ausmaß begrenzen, in dem Logistikunternehmen die Anzahl der Selbstabholer erhöhen können. Die Gewährung von Anreizen für Selbstabholer kann dazu beitragen, den Anteil an Selbstabholern zu erhöhen, so dass Lieferunternehmen weiterhin von der Kombination aus mobilen Packstationen und Lastenfahrrädern profitieren können.

Abstract

This thesis presents an innovative approach to last-mile delivery, where self-pick-up customers (SPUs) and at-home-delivered customers (AHDs) are serviced by mobile parcel lockers (MPLs) and cargo bikes, respectively. The study investigates whether a combination of MPLs and cargo bikes can lead to lower operating costs in urban areas. It introduces the Two-Echelon Vehicle Routing Problem with Mobile Parcel Lockers, Time Windows and Time Synchronization (2E-VRP-MPL-TW-TS), where heterogeneous fleet operates within a two-echelon distribution network with strict synchronization. In the first echelon, vans serve as both satellites and MPLs, meeting SPUs within their time windows, while cargo bikes operate in the second echelon, satisfying the AHDs' demand. A construction heuristic and an Adaptive Large Neighbourhood Search (ALNS) algorithm are implemented to solve the problem, tested on instances with real-life data of Vienna. The results reveal that combining MPLs and cargo bikes leads to significant cost reductions, when minimizing the vans' 'standing time' (the time a van stays to cover a SPU) and maximizing the number of SPUs. Shorter 'standing times' offer greater flexibility, and a higher share of SPUs provides synergy effects and cost efficiency. However, practical limitations may restrict the extent to which logistics companies can increase the number of SPUs. Offering incentives to SPUs can help boost the share of SPUs, enabling companies to still benefit from the combination of MPLs and cargo bikes.

Table of Content

List of Figures

List of Tables

List of Algorithms

List of Appendices

1. Introduction

According to statistics, eCommerce has grown by 8.9% in 2023 compared to prior year (eMarketer, 2023) and this trend is expected to continue as forecasts for the next years till 2028 expect an average annual growth rate of 9.83% globally (Statista, 2024). This rapid expansion of eCommerce introduces a new way of shopping containing convenient online ordering and delivery. As a logical consequence the number of customers – especially in urban areas – and their demand for deliveries behave proportionally. Therefore, delivery companies have to cope with continuously rising number of parcel deliveries that need to be handled daily for customers in densely populated regions. This is a huge challenge as last-mile delivery costs have with 41% the highest share of total supply costs worldwide (Capgemini, 2019). Thus, logistics companies have the challenging task to develop last-mile delivery concepts that minimize their total costs, while covering their customers' demand and expectations by providing a good service level[1].

A typical mode of a last-mile delivery service is the attended home delivery (AHD). This is very convenient for customers as they receive their parcels directly at home. Thus, the service level of delivery companies can be perceived as very high. On the other hand, if a customer is absent during the delivery attempt, this type of delivery can be costly for logistics companies due to failed deliveries and further delivery attempts (Kötschau et al., 2023). Surveys show that on average German courier service providers face 14.69 € of additional costs per failed delivery (Loqate, 2021). This is in conflict with companies' desire of cost efficiency. But this is not the only problem. The increased parcel deliveries lead to increased traffic volumes of delivery vans on the last mile (Schwerdfeger and Boysen, 2022). This is challenging for city logistics. Having many (big) trucks or vans in urban areas is inconvenient, as these produce high amounts of carbon emissions, pollute the air, generate noise, and cause traffic congestion and jams (Hemmelmayr et al., 2012; Ramirez-Villamil et al., 2022). Moreover, it can be even impossible for such vehicles to enter many city spaces due to legal restrictions like parking prohibitions or access restrictions for certain vehicles exceeding weight or size limitations or delivery restrictions during certain hours (Ramirez-Villamil et al., 2022; Sluijk et al., 2023). Consequently, the need of alternatives to the AHDs is evident.

Fixed parcel lockers (FPLs) are an alternative delivery service. Once the transport companies have delivered customers' parcels to an FPL, they are not required to visit each customer at

[1] Service level means all necessary activities to satisfy the customer's demand in a convenient way (Querin and Göbl, 2017).

his/her home. This might deteriorate companies' service level because customers have to become active and pick up their parcels on their own. On the other hand, the FPL concept is believed to promise lower costs for companies and better impact on environment and traffic congestion (Kötschau et al., 2023; Mancini and Gansterer, 2021), because they can consolidate the delivery of multiple customers at one location, and they avoid the time synchronization between courier and customer (Orenstein et al., 2019). However, Wang et al. (2020) argue that FPLs require very high initial investments and incur additional fixed costs for land use at the sites where they are placed, making them not beneficial for courier providers in terms of total operating costs.

A novel last-mile concept is the use of mobile parcel lockers (MPLs). These are "electrified vehicles with a number of lockers installed" (Wang et al., 2020, p. 876). In comparison to the FPLs, MPLs are always filled at the depot, which does not require additional filling tours, and these adapt quickly to changing customer demand as they can be always placed at different places due to their mobility (Kötschau et al., 2023). However, the customers are more restricted because they have to become active and pick up their parcels within a time window, which might deteriorate companies' service level.

Having a look at the alternatives mentioned above, the courier service providers face a trade-off between costs and service level when performing last-mile deliveries. Home delivery is still one of the customers' most important wishes when it comes to last-mile delivery (eft, 2020). Therefore, there might be some customers not willing to do a self-pick-up of their parcels. However, low shipment costs or free delivery are even more important wishes customers have (eft, 2020). That's why it can be possible to motivate some customers to accept the use of parcel lockers if transport companies could pass on potential cost savings to customers.

For that reason, this master's thesis deals with a novel model of combining AHDs with MPLs. The choice of MPLs as parcel lockers is based on the fact that MPLs are considered to require less investment and maintenance costs than FPLs, but the research into MPLs is rather limited (Wand et al., 2020), which makes them an interesting concept for examination. The model is based on a distribution network of two levels (echelons), called also two-echelon distribution network, where between the origin (depot) and the destination (customer) there is one or there are many intermediate facilities called *satellites* (Hemmelmayr et al., 2012). Using satellites as intermediate facilities for deliveries in urban areas benefits city logistics as big trucks are kept out of the city (centre), making it possible to use smaller and environment-friendly vehicles for the last-mile distribution (Crainic et al., 2008). From this distribution system two problems arise: the two-echelon location routing problem (2E-LRP) and the two-echelon vehicle routing problem (2E-VRP). In the 2E-LRP it has to be decided, which depots and satellites out of a given set of possible depots and satellites to open and use, and the first and second echelon routing has to be solved, while in the 2E-VRP the depots and satellites are predefined and only the routing for both echelons has to be solved (Cuda et al., 2015).

In this master's thesis I introduce the Two-Echelon Vehicle Routing Problem with Mobile Parcel Lockers, Time Windows and Time Synchronization (2E-VRP-MPL-TW-TS), considering real-life instances of Vienna. In the first echelon, vans route among predefined satellite spots, which are assumed to be convenient free parking spots on the street. On the vans there are parcel lockers installed, from which some predefined customers with individual time windows pick up their parcels (self-pick-up customers or SPUs). Therefore, the vans act as MPLs and stay at a satellite spot for a given period of time that matches SPUs' time windows. Moreover, the first echelon vehicles also have capacity to fill cargo bikes, which are the second echelon vehicles, performing the deliveries to the customers who are defined to be delivered directly at home (attended-home-delivery customers or AHDs). Thus, the vans are acting as mobile satellites within the two-echelon distribution network, which eases the network's setting because no distribution centres or storage rooms are needed for satellites. On the other hand, this forces the problem to consider a strict time synchronization because cargo bikes can be loaded to perform the second echelon tours only when both vans and cargo bikes are at the same spot at the same time. The problems' main goal is to minimize the total operating costs. Therefore, the following research question can be derived:

"Can mobile parcel lockers with cargo bike deliveries lead to lower total operating costs?"

The combination of AHDs and MPL is rarely examined. To the best of my knowledge, only Kötschau et al. (2023) deal with a model combining these two delivery services. However, they assume a given fleet size of homogeneous vehicles and examine the customer coverage with it in a one-level distribution system. The model in this master's thesis examines the total operating costs by performing both MPLs and AHDs in a two-echelon distribution network, considering heterogeneous fleet, mobile parcel lockers as satellites and strict time synchronization among the vehicles. This is a novel contribution to the literature as this setting has not been examined yet.

This thesis is structured as follows. Chapter 2 provides a review of relevant literature dealing with the 2E-VRP and (mobile) parcel locker concepts. Chapter 3 describes the 2E-VRP-MPL-TW-TS and the model's assumptions in detail. The solution method of the problem is presented in Chapter 4, while Chapter 5 shows experimental results and analyses on adapted toy instances. Chapter 6 is dedicated to a case study of Vienna, analysing results on self-generated real-life instances. Chapter 7 concludes this thesis with a summary of the findings.

2. Literature Review

2.1. The Two-Echelon Vehicle Routing Problem (2E-VRP)

The basic variant of the 2E-VRP is the 2E-CVRP (C=capacitated), where an uncapacitated depot and capacitated satellites are given and the first and second echelon routings have to be solved, considering the fleet and satellite capacity constraints (Cuda et al., 2015). Cuda et al. (2015) performed an extensive survey on 2E-VRP papers, claiming that as of the time of their survey the research of this problem was limited to few papers. Eight years later, Sluijk et al. (2023) conducted a literature review on 2E-VRP, showing that since the survey by Cuda et al. in 2015, the number of papers, dealing with the 2E-VRP and its variants, has notably increased.

The 2E-CVRP was first introduced by Perboli et al. (2008). In their paper the researchers propose a Mixed Integer Linear Programming (MILP) formulation, considering a primary echelon of distribution (goods from depot to satellites) and a secondary one (goods from satellites to customers) as well as three groups of variables (arc usage, assignment of customer to a satellite, freight flow through each arc for both distribution levels). Moreover, the authors introduce valid inequalities for a relaxation of the problem by using two families of cuts – to "forbid the presence in the solution of subtours not containing the depot" and to "reduce the splitting of the [flow] values of the binary variables" (Perboli et al., 2008, pp. 23-24). Perboli et al. (2008) use their MILP formulation to introduce two matheuristics as well. Matheuristics are solution methods that combine mathematical formulations and (meta)heuristics (Cuda et al., 2015). In their work, the authors implement once a "diving heuristic" and once a "semi-continuous" heuristic – both of them aiming to explore the search or solution space (Perboli et al., 2008). Jepsen et al. (2013) challenge the mathematical formulation by Perboli et al. (2008) by arguing that it "may not provide correct upper bounds when more than two satellites are selected in the solution" (Cuda et al., 2015, p. 192) and apply their specified branch-and-cut algorithm based on edge flow relaxations (Jepsen et al., 2013).

The 2E-CVPR (and all its variants) is a version of the VRP, and it is a NP-hard problem (Lenstra and Kan, 1981; Cuda et al., 2015). Thus, this problem is practically impossible to be solved to optimality, especially when the instances are very large (Lenstra and Kan, 1981). Therefore, considering the computational time for obtaining optimal or at least good solutions, exact methods are not appropriate for solving large NP-hard problems (Gendreau and Potvin, 2009). On the other hand, (meta)heuristics can deliver quite good solutions in acceptable amount of time (Gendreau and Potvin, 2009). That's why, there are many papers studying and implementing different (meta)heuristics for the 2E-VRP. It is noticeable that the majority of these papers follow the same pattern of splitting the problem into two subproblems. In the first subproblem, an initial solution for the second echelon is generated using methods such as nearest neighbour, cheapest/random insertion, or savings algorithm. This determines the satellite demands. Afterward, with the satellite demands known, the first-echelon routing is solved, mostly using exact methods as there are not many satellites (Sluijk et al., 2023).

In terms of improving initial solutions by (meta)heuristics, Crainic et al. (2008) first introduced a clustering-based heuristics, where large second-echelon routes are split, or customers are moved to other tours or exchanged with customers from other tours. This heuristic is improved by Crainic et al. (2011a) by a multi-start heuristic, where many new random customer-to-satellite assignments are generated, considering weighted probabilities. Crainic et al. (2011b) improve this approach by Greedy Randomized Adaptive Search Procedure (GRASP), which generates many random perturbed solutions, with Path Relinking, where in each itineration some parts of the current solution that differ to the best-found solution so far are adjusted to be the same as in the best-found solution in order to try finding better solutions.

Hemmelmayr et al. (2012) introduce an Adaptive Large Neighbourhood Search (ALNS), where destroy and repair operators manipulate solutions. They use two sets of destroy operators. One of the sets has a large impact and manipulates the satellites in the solution by removing existing ones, opening new ones, or swapping two existing ones. All of the assigned customers of the destroyed satellites are put in a pool for a reallocation by the repair operators. The other set of destroy operators has a small impact and concentrates on destroying customers or routes. As repair operators, Hemmelmayr et al. (2012) use greedy insertion (considering insertion costs), perturbed greedy insertion (weighting costs), forbidden greedy insertion (forbidding customers to be reassigned to a satellite to which they have belonged before the destruction), and regret insertion (calculating regret values [difference between the best and the second-best insertion position of the customer in a route] for each customer and starting to reassign the customers with the highest regret value so that it is avoided to assign them to their second best option which would be very costly). The ALNS by Hemmelmayr et al. (2012) is the best performing metaheuristic as of its time (Cuda et al., 2015). After that, many researchers have implemented ALNS for their 2E-VRP in the last years, using the same destroy and repair operators as Hemmelmayr et al. (2012) have implemented (Sluijk et al., 2023). They have recognized that the operators by Hemmelmayr et al. (2012) – especially the ones dealing with the satellite eliminations and choices – are crucial for obtaining diversified solutions and, thus, for being able to reach new search spaces leading to better results (Breunig et al., 2019). Recently, Variable Neighbourhood Search (VNS) and Genetic Algorithm (GA) implementations for 2E-VRP have also been elaborated, but ALNS still continues being the metaheuristic that outperforms the others (Sluijk et al., 2023).

In the recent years, some variants of the 2E-VRP including stochastic demands and travel times, different vehicle types and number of commodities as well as time windows, synchronizations and pick-ups and deliveries have been examined in order to consider some real-life aspects as well (Sluijk et al., 2023). For the purposes of the problem introduced in this master's thesis, research on 2E-VRPs with time windows, temporal synchronization and heterogeneous fleet is of interest. The following paragraphs give short overview of some of them.

Grangier et al. (2016) introduce a 2E-VRP with time windows, satellite synchronization and multiple trips for the second-echelon vehicles and solve it by ALNS. The researchers assume no storage possibility for the satellites, and, thus, first- and second-echelon vehicles have to be synchronized, so that the second-echelon vehicles can deliver to the customers, for whom hard time windows are assumed. Therefore, they initially construct the second-echelon tours and after that they coordinate and schedule the first-echelon tours to guarantee synchronization. For the ALNS, Grangier et al. (2016) implement own destroy operators (removing vehicles or routes and covered customers there, just customers, or customers in trips with "bad" synchronization where the waiting time of one vehicle for the other one is too long) and repair operators (inserting destroyed customers in existing trips with or without split, or in newly created tours).

Li et al. (2020a) introduce a 2E-VRP with time windows and mobile satellites, where vans act as first-echelon vehicles and have drones on them, which perform second-echelon tours. In this problem, the vans deliver to customers in the first echelon, but some customers' locations are also satellite spots, where the van waits and a drone starts delivering to customers in the second echelon. The van has to wait until the drone is back so that the first-echelon trip can continue. Li et al (2020a) solve their problem by ALNS, implementing random, worst or Shaw (destroying customers who are similar to each other) removals as destroy operators, and greedy or regret insertions as repair operators.

Anderluh et al. (2017) solve a 2E-VRP with temporal and spatial synchronization and heterogeneous fleet with a real case in Vienna. For this, vans are first-echelon vehicles, which are allowed to deliver to customers. However, there are some spots, where vans and cargo bikes (second-echelon vehicles) have to meet at the same time, so that the cargo bikes can perform second-echelon tours to customers in the centre of Vienna. The authors implement GRASP with Path Relinking.

Li et al. (2020b) present a Variable Neighbourhood Search (VNS) algorithm for their Two-Echelon City Logistics System with On-street Satellites (2E-CLS-OS) with heterogeneous fleet and temporal synchronization, as trucks (first-echelon vehicles) and electric tricycles (second-echelon vehicles) meet at a spot on the street that is used as a satellite. As there is no possibility to let parcels on the street and go, neither transport vehicles can stay at the meeting spot for long and congest the traffic, the transshipment process from truck to tricycle has to be synchronized.

2.2. Problems Dealing with (Mobile) Parcel Lockers

Fixed parcel lockers (FPLs) exist already in many countries for many years and there are plenty of research papers dealing with them (Schwerdfeger and Boysen, 2020). On the other hand, mobile parcel lockers (MPLs) are a novel concept and there are few papers incorporating them (Kötschau et al., 2023). This makes them interesting for further investigation. *Table 1* shows an overview of relevant papers which introduce problems with parcel lockers.

Orenstein et al. (2019) introduce the Flexible Parcel Delivery Problem (FPDP), where FPLs are used as service points from where customers have to pick up their parcels. In this problem, vans route from a depot to each service point in order to fill the parcel lockers. Thus, the basic problem the authors deal with is the capacitated VRP, as only one echelon is considered. Orenstein et al. (2019) aim to minimize the total travel and vehicle costs while also considering penalties for uncovered customers. A specific assumption to the FPDP is that customers reveal information on multiple locations where they are willing to pick-up their parcel and the delivery company decides to which FPL to deliver. The researchers found out that the more possible locations a customer revealed, the better this was for transport companies to lower their costs and travel times.

Veenstra et al. (2018) deal with a real case of medication logistics in the Netherlands, in which the transportation company has the freedom to decide which customers will be delivered directly at home or which ones will be serviced via FPLs lockers. The model in this paper is based on the capacitated LRP, because besides the routing, the choice, which FPLs to open from a set of potential locations, is made as well.

Enthoven et al. (2020) introduce the 2E-VRP with covering options, aiming to minimize the total operating costs. In the first echelon trucks deliver to FPLs and to satellites. From the satellites, cargo bikes perform second-echelon tours directly to customers defined for attended home deliveries (AHDs). Thus, there are some customers who perform a self-pick-up from an assigned FPL and some customers who are delivered directly at home. The authors point out some interesting insights. Customers in the same area should not be serviced by both cargo bike and FPL but only by one of the services. The introduction of FPLs can reduce costs up to 35.4%. Thus, the distance driven by cargo bikes can be reduced up to 60.4%, but the routes driven by trucks increases up to 37.8%. FPLs are more beneficial when the covering range is larger. Overall, the researchers "conclude that a good routing solution aims to serve customers near each other in a similar way" (Enthoven et al, 2020, p. 11). Yu et al. (2021) extends the research by Enthoven et al. (2020) by adding occasional drivers as an additional option for the second-echelon tours. Occasional drivers are "on-road crowds who are willing to detour to serve customers in exchange for monetary or other compensation" (Yu et al., 2021, p. 2).

Mancini and Gansterer (2021) investigate an extended VRP-TW (TW=time windows), where the company itself decides if a customer will be covered at home, considering his/her time windows, or via FPL (also taking penalties into account if FPL is not within customer' radius). This study aims to show if the customer flexibility leads to lower operating costs and, thus, benefits for transportation companies. As a result, this flexibility for companies reduces delivery costs by 9.5%. Interesting to mention is also that for large instances, the number of customers covered by FPL is lower than for small and medium instances. The authors argue that this is due to the fact that if many customers are very close to each other (high density), it is not very costly to perform additional AHDs and to cover additional customers, while ensuring high service quality and avoiding penalty costs.

Schwerdfeger and Boysen (2020) are the first to present MPL in their research, aiming to minimize the locker fleet size needed to satisfy all customers' demand, using the benefit of MPLs to change their locations. Their model is based on the following deterministic data: each MPL is filled only at the beginning of the day; customers have many different locations with different time being (time window) during the day; customers are willing to pick-up their parcels only if the MPL is within their accepted walking range. In order to measure the location flexibility of MPLs, Schwerdfeger and Boysen (2020) solve their problem also by using only FPLs. The authors find out that MPLs help to reduce the fleet size by 70% to 400% compared to if only FPLs would be used instead. The intensity of success to minimize the fleet size depends on customers' accepted walking range: customers willing to walk longer distances to a locker leads to decreasing the advantage of the MPL concept. Moreover, when the number of customers increases, the ability of MPLs to change locations becomes less beneficial. In addition, Schwerdfeger and Boysen (2020) report that customers have to walk approximately the same distance to reach the parcel locker, regardless of whether it is mobile or fixed. However, the researchers point out that MPLs traverse longer distances through the city compared to the trucks that have to refill the FPLs. Two years later, the same authors change the research focus to minimizing total operating costs and extend their study on MPL by adding and investigating different concepts such as mounted vs. loaded lockers on the van, autonomous driving as well as swapping drivers (Schwerdfeger and Boysen, 2022). The different concepts examined in Schwerdfeger and Boysen (2022) confirm the findings in Schwerdfeger and Boysen (2020). In terms of costs, three findings in Schwerdfeger and Boysen (2022) are worth mentioning: 1) the variant with swapping drivers (e.g., one driver moves MPL1 to a new location and travels via public transport [ticket costs considered] to the location of MPL2 to move it to its new location, meaning that for two MPLs only one driver is needed) reduces the personnel costs, but on the other hand, more MPLs are needed; 2) customers with more than one known location during the day are more beneficial for the MPL concept, meaning that companies should give customers incentives to motivate them to reveal more of this private data; 3) providing customers with more flexibility to pick up their parcels (e.g., MPLs wait longer at pick-up locations) can be costly.

Wang et al. (2020) introduce a problem with a combined delivery service, where customers with individual time windows are covered by either FPLs or MPLs. The researchers focus on two customer assignment strategies. The results in Wang et al. (2020) show that aggregating a number of customers into different clusters leads to a very high decrease of delivery time and a very slight increase of operating costs.

Kötschau et al. (2023) introduce the Heterogeneous Locker Location Problem (HLLP) with the objective to maximize the customers serviced while considering a given fleet size as well as customers' preferences such as walking willingness and time windows. The motivation of this paper is to examine the service quality and efficiency of the delivery services AHDs, FPLs and MPLs on different spatial instances, applying each of the delivery services individually or in a combination with another one out of the three services. The authors conclude that the combination of AHDs with MPLs leads to the highest number of served customers, and thus, it dominates all other options. Moreover, they find out that AHDs with MPLs can adapt best to different spatial characteristics.

Table 1: Comparison of relevant literature

<table>
<tr><th rowspan="2"></th><th colspan="3">Single Delivery Service[1)]</th><th colspan="3">Combined Delivery Service</th><th rowspan="2">(Closest) Basic Routing Problem[2)]</th><th rowspan="2">Hetero-geneous Fleet</th><th rowspan="2">Objective Function</th><th rowspan="2">Solution Method[1)]</th></tr>
<tr><th>AHD</th><th>FPL</th><th>MPL</th><th>FPL & MPL</th><th>AHD & FPL</th><th>AHD & MPL</th></tr>
<tr><td>Orenstein et al. (2019)</td><td></td><td>✓</td><td></td><td></td><td></td><td></td><td>VRP</td><td></td><td rowspan="6">min total operating costs</td><td>MH</td></tr>
<tr><td>Veenstra et al. (2018)</td><td></td><td></td><td></td><td></td><td>✓</td><td></td><td>LRP</td><td></td><td>E/MH</td></tr>
<tr><td>Enthoven et al. (2020)</td><td>✓</td><td></td><td></td><td></td><td>✓</td><td></td><td>2E-VRP</td><td>✓</td><td>E/MH</td></tr>
<tr><td>Yu et al. (2021)</td><td>✓</td><td></td><td></td><td></td><td>✓</td><td></td><td>2E-VRP</td><td>✓</td><td>E/MH</td></tr>
<tr><td>Mancini and Gansterer (2019)</td><td></td><td></td><td></td><td></td><td>✓</td><td></td><td>VRP-TW</td><td></td><td>MH</td></tr>
<tr><td>Wang et al. (2020)</td><td></td><td></td><td></td><td>✓</td><td></td><td></td><td>LRP-TW</td><td></td><td>MH</td></tr>
<tr><td>Schwerdfeger and Boysen (2020)</td><td></td><td>✓</td><td>✓</td><td></td><td></td><td></td><td>LRP-TW</td><td></td><td>min locker fleet</td><td>E</td></tr>
<tr><td>Schwerdfeger and Boysen (2022)</td><td></td><td>✓</td><td>✓</td><td></td><td></td><td></td><td>LRP-TW</td><td></td><td>min total operating costs</td><td>MH</td></tr>
<tr><td>Kötschau et al. (2023)</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>LRP-TW</td><td></td><td>max cus-tomers serviced</td><td>E</td></tr>
<tr><td>This master's thesis</td><td>✓</td><td></td><td>✓</td><td></td><td></td><td>✓</td><td>2E-VRP-TW</td><td>✓</td><td>min total operating costs</td><td>MH</td></tr>
</table>

1) AHD = attended home delivery FPL = fixed parcel locker MPL = mobile parcel locker E = exact MH = (meta)heuristic
2) In all of the papers vehicle (and facility) capacities are considered, so all of the closest basic routing problems are capacitated.

From all topic-relevant papers, only Kötschau et al. (2023) deal with the combination of both delivery services AHDs and MPLs. However, the work in this master's thesis differs in several ways. First, this thesis' model is built on a two-echelon distribution network, and a heterogeneous fleet of vehicles is introduced. Second, the model's objective is to minimize the total operating costs of the 2E-VRP with both AHDs and MPLs combined. Third, results are assessed on real-life instances. Fourth, a metaheuristic is introduced for solving the problem. Moreover, the setting of this thesis' problem requires a time synchronization between the first- and second-echelon vehicle fleets. All in all, the literature review shows that there is no research analysing the cost effects of combining MPLs with cargo bikes for AHDs to tackle the challenges of delivering in densely populated urban spaces. This master's thesis aims to cover this research gap.

3. The Two-Echelon Vehicle Routing Problem with Mobile Parcel Lockers, Time Windows and Time Synchronization (2E-VRP-MPL-TW-TS)

3.1. Problem Description

This master's thesis addresses a Two-Echelon Vehicle Routing Problem including mobile parcel lockers, customers with time windows and time synchronization between two types of vehicles. This problem requires considering two groups of customers: customers who are delivered directly at home (AHDs) and customers who pick-up their parcels on their own from the mobile parcel locker within their individual time window (SPUs). Moreover, a heterogeneous fleet is considered: vans (first-echelon vehicles) as MPLs for covering the SPUs, and cargo bikes (second-echelon vehicles) for covering the AHDs. The vans route among predefined satellite spots and meet with cargo bikes there. Subsequently, the cargo bikes are loaded, and they perform the second-echelon tours, delivering to the AHDs. Thus, the vans act as satellites as well. This requires a time synchronization with the cargo bikes because both vehicle types have to be at the same time at the same satellite spot, so that the cargo bikes can be loaded and thus, the AHDs covered. Moreover, the vans stay at the satellite spots for a required amount of time so that the SPUs can pick-up their parcels. Once the vans have no parcels for distribution, they return to the depot for a refill. *Figure 1* presents an example of a feasible solution for the 2E-VRP-MPL-TW-TS.

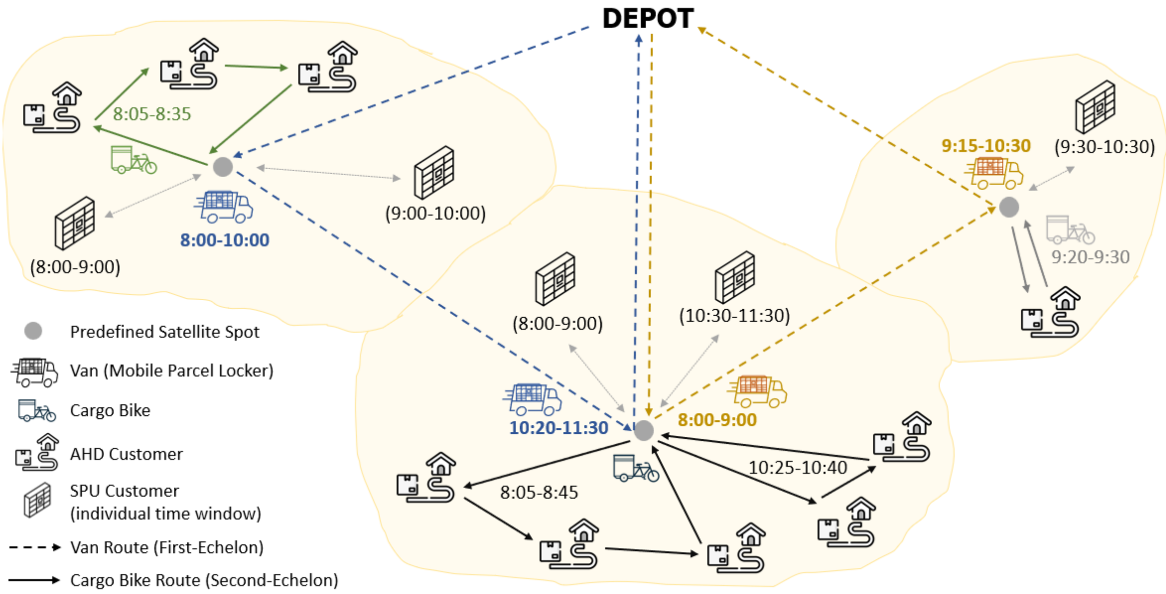


Figure 1: Example of a feasible solution for the 2E-VRP-MPL-TW-TS

In the example in *Figure 1* there are three clusters with AHDs and SPUs. In each cluster there is a predefined satellite spot and one cargo bike waiting at it for a van to come. Two vans start from the depot. The first van (van #1 in blue) goes to the satellite spot in the left cluster and arrives there at 8 a.m. The cargo bike is loaded within five minutes with parcels from van #1 and performs a tour from 8:05-8:35 a.m. delivering to the AHDs in the cluster. The bike comes back to the satellite spot after covering all AHDs (or if it has no capacity left). Van #1 stays at the satellite spot for two hours till 10 a.m., so that both SPUs in the cluster can pick-up their parcels considering their individual time windows. The second van (van #2 in orange) goes to the satellite spot of the middle cluster. It arrives at 8 a.m., the cargo bike for the middle cluster gets loaded and performs a tour covering three AHDs (let's assume due to capacity reasons). Van #2 stays till 9 a.m. and its time at the satellite spot allows only one SPU (with time window 8:00-9:00 a.m.) to pick up his parcel. Thus, at this time there are two AHDs and one SPU in the middle cluster that are not covered. After that, van #2 goes to the satellite spot of the right cluster, arrives at 9:15 a.m. and stays till 10:30 a.m. there, so that the SPU there can be covered. The bike for this cluster gets loaded and covers the one AHD as well. At 10:30 a.m., van #2 starts heading back to the depot (where it can refill if it is needed). On the other hand, van #1 goes from the satellite spot of the left cluster to the satellite spot of the middle cluster and arrives there at 10:20 a.m. It stays till 11:30 a.m. so that the remaining SPU (with time window 10:30-11:30 a.m.) can be covered by servicing during his individual time window. As the bike is back to the satellite spot from its first route at 8:45 a.m., it is available, so it can be loaded after the arrival of van #1 at 10:20 a.m. (loading time is five minutes). The bike performs a tour to cover the remaining

AHDs. After staying at the satellite spot, van #1 goes back to the depot. All customers are covered.

As the 2E-VRP-MPL-TW-TS is an extension of the VRP, and therefore it is NP-hard (Lenstra and Kan, 1981), the applied solution method for solving it is of a heuristic nature. Chapter 4 describes the solution approach considering all model's specifications. Hence, a mathematical formulation is left out. For a mathematical formulation of the basic 2E-VRP refer to Cuda et al. (2015).

3.2. Model Assumptions and Parameters

Assumptions are needed to simplify and solve complex problems and models by specifying missing information, making it possible to set a framework and concentrate on factors that influence the result. The following paragraphs present and explain the assumptions used for solving the 2E-VRP-MPL-TW-TS. *Table 3* gives an overview of the assumed parameters, used for each test run and result analysis.

3.2.1. General Assumptions

First of all, the customers are clustered (refer to Chapters 5 and 6 for more information on the cluster methods used for each instance). For each cluster, a satellite spot is predefined where vans and cargo bikes meet and where the cluster's SPUs pick-up their parcels. These spots are assumed to be convenient free parking spots on a street or a parking lot with no need to pay a parking fee. Only one cargo bike waits at each satellite spot, and it covers the cluster's AHDs after it gets loaded by a van. On the other hand, more than one van is allowed to be at the same satellite spot at the same time.

The assumed general working hours for a van are between 8 a.m. to be at the first satellite spot of the tour till 5 p.m. However, if the van is at a satellite spot at 5 p.m. or later, but it has remaining capacity, it continues routing among the satellite spots till the remaining capacity is fully used or if there are no customers left anymore and only then it finishes its tour.

The travelling times, which are needed for determining the arrival and departure times at satellite spots and customers, are calculated by the following formula:

$$travelling\ time = \frac{distance}{velocity}$$

The distances in the van and bike tours are defined by the Euclidean metric. The velocity is assumed to be 25 km/h for the vans and 12 km/h for the cargo bikes.

3.2.2. Vehicles and Capacities

All parcels for delivering are assumed to be homogeneous with the size of 40x40x40 (length, width, height) in cm, which according to the courier, express and delivery industries report for 2022 is the average parcel size delivered in Vienna (Wirtschaftskammer Wien, 2022).

In terms of vehicles, this master's thesis doesn't have the purpose to simulate diverse vehicle capacities, but to depict the vehicles with their needed requirements as realistic as possible. After some research, I opted for Sevic V500e for the vans' fleet and Yokler for the cargo bikes' fleet.

Sevic V500e is suitable to be a van and at the same time a mobile parcel locker due to its manually configurable cargo box (Sevic, no date - a). Considering its measurements and configuration possibilities, it allows to have 12 compartments of 45x50x50 in cm on the left and right sides each, making in total 24 compartments. In the middle, it provides capacity for 20 parcels.[2] *Figure 2* illustrates this configuration. It is important to note that the compartments are primarily intended for covering SPUs. However, when there are few or no SPUs left, the remaining capacities of the compartments are also used for loading the cargo bike and thus, covering AHDs as well. On the other hand, the parcel capacity in the middle is assumed to be exclusively designated for loading cargo bikes and covering AHDs.

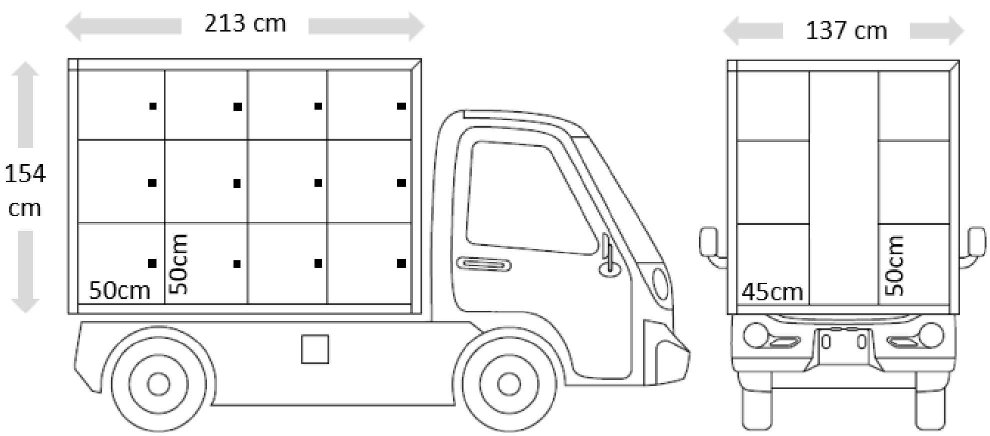


Figure 2: Sevic V500e with configuration as a mobile parcel locker (Adapted from Sevic, no date – b, pp. 3, 5. Modifications: compartments visualisation and slight cargo box's height increase)

[2] For a parcel with size 40x40x40 cm: using van's length, width (excl. left and right compartments) and height: (213 cm / 40 cm) x ((137 cm – 2x45 cm) / 40 cm) x (154 cm / 40 cm) ≈ 5 x 1 x 4 = 20 parcels.

The cargo bike by Yokler allows storing 12 parcels.[3] *Figure 3* illustrates this vehicle with its measurements.

Figure 3: Yokler cargo bike with measurements (Adapted from Yokler, no date. Modifications: measurements added)

3.2.3. Customers and Parameters for Delivery Times

The AHD customers are assumed to have no time windows and to be available the whole day. Thus, once the cargo bike arrives at an AHD, the delivery is successful. On the other hand, each SPU customer is assumed to have individual time window and only within this time window, the SPU can pick-up his parcel from the van. Three time windows are assumed: one in the morning (8-11 a.m.), one at midday (11 a.m. – 2 p.m.) and one at afternoon (2-5 p.m.). One of these three time windows is randomly assigned to each SPU customer considering a deterministic probability for selection as presented in *Table 2*. These probabilities assume that the first and last time windows are more probable because people should have more free time before and after work or school.

Table 2: Overview of SPUs' time windows' selection probabilities

Time Windows For SPU	Probability for Selection
8 a.m. – 11 a.m.	40%
11 a.m. – 2 p.m.	20%
2 p.m. – 5 p.m.	40%

[3] For a parcel with size 40x40x40 cm: using bike's storage room's length, width, and height: (80 cm / 40 cm) x (88 cm / 40 cm) x (143 cm / 40 cm) ≈ 2 x 2 x 3 = 12 parcels.

As SPUs pick-up their parcels only within their time windows, the van should be at a satellite spot during a sufficient amount of time so that SPUs can have a realistic opportunity for a self-pick-up. For this, the parameters '**standing time'** and '**pick-up time'** are very important. The 'standing time' represents the maximal amount of time a van should stay at a satellite spot when there is a SPU to be serviced, while the 'pick-up time' is the minimum amount of time the SPU should have to pick-up his parcel. It is important to note that these parameters are not fixed values in the model. They are instead variable and a part of the model's scenario analysis. Normally, both parameters are set to be the same value, e.g., "*standing time = pick-up time = 60 minutes"*. This example would mean that the van stays at a satellite spot for 60 minutes and a SPU is covered only if these 60 minutes are at some point within his time windows. *Figure 4 a)* depicts this example considering a SPU customer with time window from 11 a.m. to 2 p.m. In order to cover this customer, by setting both parameters to be equal 60 minutes, it means for the van that only if it arrives at the SPU's satellite spot in the interval from 11 a.m. to 1 p.m., the SPU will be covered, because only then after staying there for 60 minutes, the SPU will have 60 minutes 'pick-up time' within his time window in order to pick up his parcel. By distinguishing between these two parameters, however, the model allows a relaxation for delivery companies to cover more SPUs if the 'pick-up time' is set to be lower than the 'standing time'. *Figure 4 b)* depicts this idea as the 'standing time' is set to be 60 minutes and the 'pick-up time' only 15 minutes. Thus, the interval for van's possible arrival is extended compared to the scenario in *Figure 4 a)* when having both parameters equal. In scenario *b)*, the van can arrive at the spot at 10:15 a.m. the earliest and at 1:45 p.m. the latest, and the SPU would be still covered. This is due to the fact that 15 minutes are enough for the SPU for a self-pick-up. Therefore, if the van arrives at 10:15 a.m. and stays for 60 minutes ('standing time') till 11:15 a.m., the SPU would have exactly 15 minutes for a pick-up ('pick-up time') from 11:00 a.m. till 11:15 a.m. The same applies for arrival time at 1:45 p.m. the latest – it is sufficient to offer 15 minutes of pick-up time to the SPU from 1:45 p.m. to 2 p.m. in order to cover him.

In addition to 'standing time' and 'pick-up time', there is another parameter for the customer coverage: 'service time'. This parameter accounts for the time required to refill vans at depots, to load cargo bikes from the vans, and to cover an AHD. Unlike 'standing time' and 'pick-up time', which are variable, the 'service time' parameter is fixed and assumed to have the predetermined value of 5 minutes.

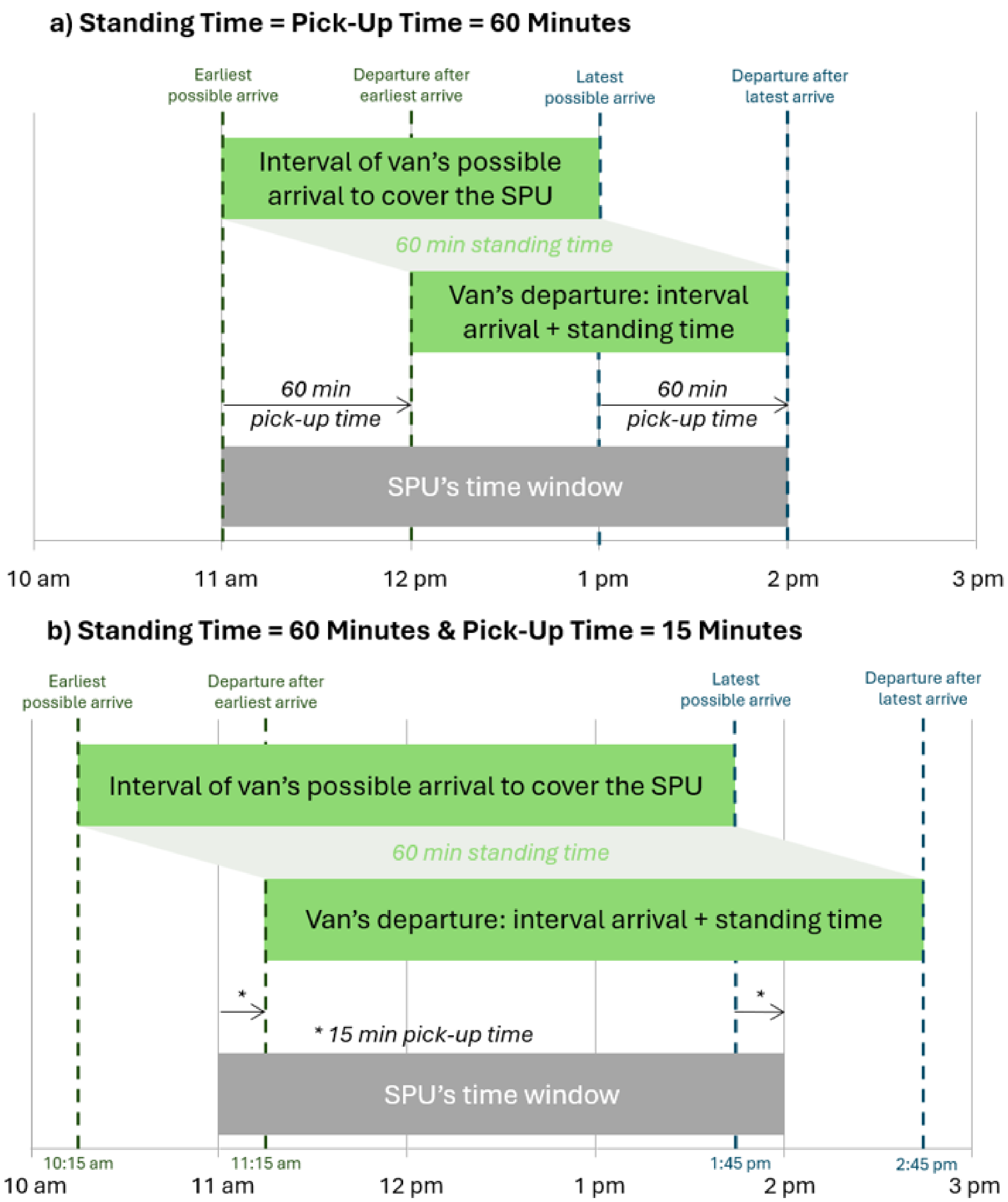


Figure 4: Example of covering a SPU customer with time windows from 11 a.m. to 2 p.m. by using different 'standing time' and 'pick-up time' settings

3.2.4. Objective Function

The objective of this master's thesis problem is to minimize the total operating costs. The following formula represents the objective function, which consists of three components.

$$\text{min total costs} = \textit{travelling costs} + \textit{personnel costs} + \textit{vehicle usage costs}$$

The ***travelling costs*** in the model are derived using the following formula:

$$travelling\ costs = (\sum_{v=1}^{n} d_v) \times c_{van} + (\sum_{b=1}^{m} d_b) \times c_{bike}$$

Where $v \in \{1, \dots, n\} \rightarrow number\ of\ van; b \in \{1, \dots, m\} \rightarrow number\ of\ cargo\ bike;$

$d_v \rightarrow distance\ travelled\ by\ van$ "v"; $d_b \rightarrow distance\ travelled\ by\ cargo\ bike$ "b";

$c_{van} \rightarrow costs\ per\ km\ per\ van\ (0.5€/km);\ c_{bike} \rightarrow costs\ per\ km\ per\ cargo\ bike\ (0.01€/km)$

The total distance travelled by all vans and the total distance travelled by all cargo bikes are respectively multiplied by the costs per km for a van 'c_{van}' and for a bike 'c_{bike}'. The costs per km for a van 'c_{van}' are assumed to be 0.5 €/km, while the costs per km per cargo bike 'c_{bike}' are assumed to be 0.01 €/km. These costs primarily represent fuel and energy expenses but also include eventual costs for maintenance, repairs, licenses, and insurance for simplicity.

The ***personnel costs*** represent the salary expenses for all drivers. They can be expressed by the following formula:

$$personnel\ costs = (\sum_{w=1}^{k} h_w) \times s_{driver}$$

Where $w \in \{1, \dots, k\} \rightarrow number\ of\ driver;\ h_w \rightarrow working\ hours\ of\ driver$ "w";

$s_{driver} \rightarrow salary\ per\ hour\ for\ each\ driver\ (12€/hour/driver)$

The sum of the total working hours of all drivers is multiplied by the driver's salary per hour 's_{driver}' which is assumed to be 12 €/h per driver. There is no differentiation between a van and a cargo bike driver. All drivers are assumed to have the same salary per hour.

The ***vehicle usage costs*** are calculated based on the daily depreciation of each vehicle. To determine this, the vehicle's purchase price is divided by its useful life to find the yearly depreciation. After that, the yearly depreciation is divided by 360 days to get the daily depreciation. For the van, a purchase price of 24.000 € (Sevic, no date – a) with useful life of 12 years is assumed. This results in the following van's usage costs per day: $u_{van} = 24.000 / 12 / 360 \approx 5.56$ *€/day*. For the bike, a purchase price of 8.000 € with a useful life of 6 years is assumed. This leads to the following bike's usage costs per day: $u_{bike} = 8.000 / 6 / 360 \approx 3.71$ *€/day*. The sum of each vehicle's usage costs represents the total vehicle usage costs.

$$vehicle\ usage\ costs = n \times u_{van} + m \times u_{bike}$$

Where $n \rightarrow number\ of\ vans\ used;\ m \rightarrow number\ of\ cargo\ bikes\ used$

3.2.5. Summary

The previous subchapters provide a detailed explanation of all assumptions made as a simplification complying with the framework of the problem addressed in this master's thesis. *Table 3* summarizes the key value assumptions used to run the algorithm and solve the problem.

Table 3: Overview of all assumed value parameters

Deterministic Parameters	Value for Van	Value for Cargo Bike
Capacity for SPUs	24 compartments	n/a
Capacity for AHDs	20 parcels (to load the bike)	12 parcels (to deliver to AHDs)
Velocity	25 km/h	12 km/h
Costs per km travelled	0.5 €/km	0.01 €/km
Purchase price	24 000 €	8 000 €
Useful life	12 years	6 years
Driver's salary per hour	12 €/h	12 €/h
Service time to refill van / load bike / cover an AHD	5 minutes	5 minutes
Variable Parameters	**Value for Van**	**Value for Cargo Bike**
Standing time	diverse, see Chapters 5 and 6	n/a
Pick-up time (for a SPU)	diverse, see Chapters 5 and 6	n/a
Customer's Parameters (Deterministic)	**AHDs**	**SPUs**
Time window	no	yes: 8 a.m. – 11 a.m. (40% probability) OR 11 a.m. – 2 p.m. (20% probability) OR 2 p.m. – 5 p.m. (40% probability)

4. Solution Method

For solving the 2E-VRP-MPL-TW-TS, this thesis introduces an Adaptive Large Neighbourhood Search (ALNS), coded in C# in Visual Studio. The initial solution proposed is an extensive construction heuristic based on slightly adjusted Nearest Neighbour Algorithm for the van and bike tours, and many capacity and time synchronization checks. The ALNS consists of two destroy and three repair operators. The following subchapters describe in detail how an initial solution is obtained and how ALNS is implemented.

4.1. Initial Solution

First of all, in a nutshell, the way the initial solution is obtained for this thesis' problem is worth mentioning as it differs from the way the initial solution for the 2E-VRP and its variants are generated in literature. Here, I start with the first-echelon tour (van tour) and after that I create simultaneously the second-echelon tours (bike tours) at each satellite visit of the van tour in case the bike is available, and the van has parcels left to load the bike. Thus, it is ensured that both van and bike tours are synchronized. On the other hand, as already mentioned in Chapter 2, in the literature, the second-echelon tours are generated first, giving the demand of each satellite, and afterwards the first-echelon routing is solved. The paper by Grangier et al. (2016) is a good example to explain why I choose a different approach as it contains time synchronization as well. In the model of Grangier et al. (2016), the customers who are serviced by the second-echelon vehicles are the ones with time windows. Therefore, it is reasonable to start with constructing the second-echelon tours, considering the customers' time windows. From there on, the first-echelon tours can be scheduled based on the second-echelon tours, so that both first- and second-echelon vehicles meet at the same satellite at the same time. For this thesis' problem, the SPU customers are the ones with time windows. The SPUs are serviced by the vans which are the first-echelon vehicles, while the second-echelon vehicles deliver to customers without any time windows restrictions. Moreover, the vans act as MPLs and have to stay at the satellite spot in order to be able to cover SPUs. Therefore, it is more important to timely consider the SPUs and, thus, the first-echelon tours, while the second-echelon tours tie in with the first-echelon tours. The following paragraphs explain the approach for the initial solution for this thesis' problem in detail. *Figure 5* and *Figure 6* visualize this approach.

The initial solution algorithm is based on conducting van and bike tours until we have non-covered customers left. Each van tour starts from the depot. After that the next satellite visit for the van is determined based on a Nearest Neighbour Algorithm, whose pseudocode is shown in *Algorithm 1* and it works as follows. Variables for storing the next satellite visit, the last satellite in the current van tour so far, and the minimal distance found so far, which is set to a very high number at the beginning, are initialized (lines 1-3 in *Algorithm 1).* After that, we itinerate over each satellite to get the still not covered AHD customers in the satellite's cluster from the current itineration (line 5). The distance from the current itineration's satellite to the last satellite in the current van tour is calculated (line 6). The potential arrival time for the current itineration's satellite (if it would be the next visit for the tour), is calculated based on the distance to the last satellite visit in the tour and on the van's velocity (line 7). All SPUs that could be covered in the cluster of the current itineration's satellite are determined and stored in a list, considering van's potential arrival time and the time the it has to stay at the satellite spot *('standingTime')* and the time the SPUs should at least have for a pick-up *('pickupTime')* (line 8). After having all these components, I check if there are any SPUs or AHDs to cover. If there are no SPUs for covering at the potential time the van could be at the satellite spot, and if there are no AHDs for covering neither or the van has no parcels left to load the cargo bike, the satellite from the current itineration is not considered (lines 9-10). Otherwise – if there are SPUs or AHDs for covering and the van has parcels for distribution – the current itineration's satellite is considered and its distance to the last satellite visit in the current tour is compared to the minimal distance found so far (line 11). If this distance is not equal to zero (to not have the same satellite and thus, stay at the same satellite spot) and if it is lower than the minimal distance found so far, then it is stored to be the next satellite, and the minimal distance so far is updated (lines 12-13). The algorithm returns the next satellite visit for the van tour and a list of satellite's SPUs who have to be covered at the time when the van would visit this satellite.

Algorithm 1: Get next nearest satellite visit

```
1:  nextNearestSatellite = null
2:  lastSatelliteInTour = currentVanTour.Last
3:  minimalDistance = max.Double
4:  FOREACH satellite DO
5:      nonVisitedAHDs = GetNonVisitedAHDs(satellite)
6:      distanceToLastVisit = satellite.DistanceToNode(lastSatelliteInTour)
7:      arrivalTime = CalculateArrivalTime(distanceToLastVisit, vanVelocity)
8:      spusToCover = SPUsToBeCoveredList(satellite, arrivalTime, standingTime, pickupTime)
9:      IF (spusToCover = {} AND (vanRemainingCapacityForAHDs = 0 OR nonVisitedAHDs = {})) THEN
10:         skip satellite
11:     ELSE IF (distanceToLastVisit > 0 AND distanceToLastVisit < minimalDistance) THEN
12:         nextNearestSatellite = satellite
13:         minimalDistance = distanceToLastVisit
14:     END IF
15: END FOREACH
16: return nextNearestSatellite, spusToCover
```

Algorithm 1 for defining the next satellite visit makes sure to visit satellites where customers can be covered. However, it can happen that all of the satellites should not be considered if for the time being there are no SPUs or AHDs for covering. In this case, the algorithm will return an empty item (nextNearestSatellite = 'null'). Therefore, after running *Algorithm 1,* I check in the van tour algorithm if the next satellite is 'null' or not. If it is not 'null' (means: the van could visit a satellite), and the van's capacity check is positive (means: the van has parcels left to cover SPUs and/or AHDs) the following happens:

- I add the determined next satellite visit by *Algorithm 1* to the van tour and determine van's arrival time at this satellite considering the distance to the predecessor in the tour and van's velocity.
- Next, I consider the list of SPUs who can be covered in the current satellite's cluster that I have obtained as an output from *Algorithm 1*, and I cover all the SPUs who can be covered considering van's remaining capacity for SPUs (remaining compartments with parcels for self-pick-up). Moreover, I determine van's departure time according to the rules of 'standing time' and 'pick-up time' described in Chapter 3. Thus, the van has done its work as a mobile parcel locker.
- In the next step, I check if the cargo bike operating in the satellite's cluster is available at the satellite spot when the van arrives. If the bike is gone, this means that it still performs a previously scheduled tour. In this case, I retrieve the time the bike would be back from its tour and calculate the time the van has to wait for the bike. Based on this waiting time, I adjust van's departure time (if needed) so that van and bike are timely synchronized. After that I perform a bike tour to cover AHDs in the satellite's cluster, considering van's remaining capacity for loading a bike. The time between departure and arrival of each scheduled bike tour for each cargo bike is stored in a list ('bike not available'), which is used for checking the bike availability.[4] For the start of the bike tour, I take the arrival time of the van plus its waiting time for the bike and the 'service time' for loading. Note that if the bike is available when the van arrives, the waiting time is zero. The following examples clarify the van and bike time synchronization after the bike availability check.
  <u>*Example 1:*</u> *The van arrives at 9:00 and stays till 10:00 in order to cover the SPUs. The bike is available to cover AHDs. Therefore, it starts its tour at 9:05 (+5 min 'service time' for loading) and no adjustment of van's departure time is to be made because there is no shift of time.*
  <u>*Example 2:*</u> *The van arrives at 9:00 and stays till 10:00 in order to cover the SPUs. The bike is not available, when the van arrives, as it performs a previously scheduled*

[4] For example, if cargo bike #2 (BIKE2) has three scheduled tours from 9:00-10:00, 10:00-10:50 and 12:30-13:10, in the 'bike not available' list for BIKE2 the following items will be stored: {9:00-10:50}, {12:30-13:10}. Thus, during this times BIKE2 is marked as not available while performing bike availability checks.

tour, from which it comes back at 9:15. Thus, the calculated waiting time for the van is 15 minutes, but van's departure time is not to be adjusted as the van will be still there when the bike comes. However, the calculation of waiting time is important for the bike, so that it can start its next tour when it comes back, which is 9:20 (van's arrival at 9:00 + waiting time 15 minutes + 5 minutes 'service time' for loading).

<u>*Example 3:*</u> *The van arrives at 9:00 and stays till 09:05 (5 minutes 'service time') because there are no SPUs for covering, but there are AHDs for covering. However, the bike is performing a tour already and it comes back at 9:15. Thus, the van has to wait for 15 minutes. Therefore, the van's departure time is adjusted from 9:05 to 09:20 (15 minutes waiting time on top). After coming back at 9:15 (equals van's arrival at 9:00 + 15 minutes waiting time), the bike is loaded within 5 minutes ('service time') and starts its next tour at 9:20.*

If the next satellite from *Algorithm 1* is 'null' (means: the van could not visit a satellite), I check if all customers have been covered. If this is the case, I add the depot to the tour, meaning that the van finishes its tour by coming back to the depot. With this the whole algorithm for van and bike tours finishes as well, as there are no more non-covered customers. However, if not all customers have been covered, this means that the van should go back to the depot to refill. Thus, I add the depot to the tour and refill the van with parcels. This last step is performed as well when the van could visit next satellite, but it has no parcels left for distribution. After going back to the depot and refilling, I check if the van would departure from the depot after 17:00 o'clock. If this is the case, the working time is exceeded and this van tours finishes. The initial solution algorithm would start another van tour (additional van for the routing). However, if the departure time is before 17:00 o'clock, the same van continues its tour, performing another subtour, following the same procedure as described above. *Figure 5* visualizes the described procedure for generating van tours.

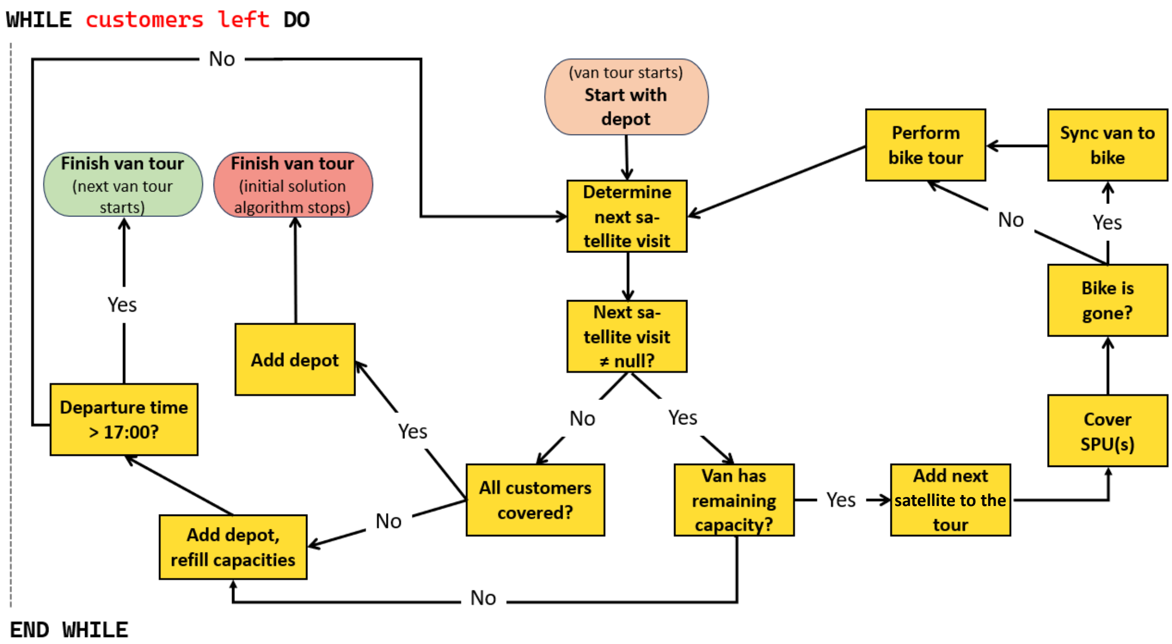


Figure 5: Flowchart of van tour algorithm

While implementing the construction heuristic, one not desirable case – depending on the customers left – was identified. This not desirable case happens when for example in all other satellites there are no more SPUs and AHDs left, but only in the current satellite (e.g., 6 AHDs left in satellite S2), and the van has many parcels (e.g. 6 units left) for loading the bike. In this case, the van loads the bike fully (let's assume a bike has a capacity of 3 parcels) and it stays for a short amount of time at the satellite spot because there are no SPUs. Following the main procedure, *Algorithm 1* would set the next satellite to be 'null' because it does not consider the same satellite when looking for the next nearest satellite and because in all other satellites there are no customers left. Thus, the van would go back to the depot. After that, *Algorithm 1* would give S2 to be the next satellite, as this is the only satellite with customers left. Therefore, the van's would perform the tour {S2 -> depot -> S2 -> depot}, although if it would stay longer at S2, the tour would be {S2 -> depot}, because the van has enough not distributed parcels to load the bike twice and cover all AHDs during one satellite visit instead of two. This way, distance and, thus, costs can be saved. Therefore, in order to avoid this unnecessary cycling, a special feature is included at the process step "Cover SPU(s)", namely: *Algorithm 1* is performed again to check if the next satellite would be 'null'. If this is the case, the van has still free parcels for loading a bike and there are AHDs waiting to be covered at the same satellite spot, the van is set to stay longer at this satellite so that the bike can perform multiple tours one after another, and cover all/more AHDs.

As already mentioned, bike tours are performed at each satellite visit, when the van has free parcels for loading the bike. *Figure 6* visualizes the algorithm for generating bike tours.

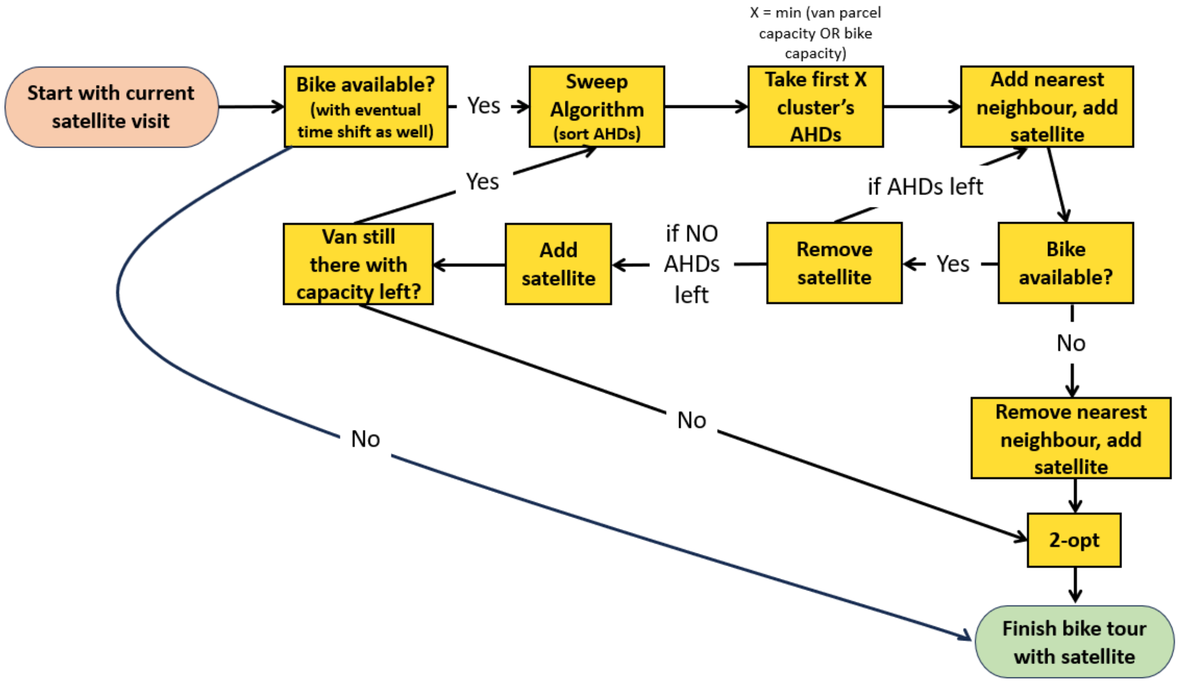


Figure 6: Flowchart of bike tour algorithm

Each bike tour starts from the current satellite visit and takes the arrival time of the van plus van's waiting time for the bike and the 'service time' for loading as a tour start. Thus, van and bike are timely synchronized as described above. However, if the bike availability check is negative, meaning that the bike cannot perform a tour because it is gone and/or directly after coming back a new tour has already been scheduled, no bike tour is generated, otherwise a bike tour is generated as follows:

- All AHD customers within the satellite's cluster, where the cargo bike operates, are retrieved, and sorted by using the Sweep Algorithm. This algorithm is a simple method for clustering, and it works as follows. A 'ray' starts from the satellite and moves clockwise, grouping all customers that intersect with this 'ray' into one cluster until a predetermined capacity is reached (Cavaliere et al., 2022). The AHDs' sorting here is based on the logic of the Sweep Algorithm, utilizing a 'ray' originating from the satellite. For each customer, this 'ray' builds an angle with the straight-line connection 'satellite to AHD'. The AHDs are sorted in a list by their angles in a descending order.
- The first AHDs from the sorted list are selected for delivering as long as the capacity allows it. It is important to note that the capacity is determined by the lower number of either the remaining van's free parcels for bike loading or bike's capacity for taking over parcels (e.g., if the bike can be fully loaded with three parcels, but the van has only two parcels left, the bike can deliver to only two AHDs).

- After selecting the AHDs for covering, they are successively added to the bike tour by applying the Nearest Neighbour Algorithm. When all AHDs are covered, the bike goes back to the satellite and finishes its tour. However, after adding one AHD to the tour, the bike tour algorithm always checks if the bike would be available (means: no previously scheduled tours that could collide with this tour) if it would go back to the satellite to finish its tour. If this is not feasible, this means that the added AHD cannot be covered because the bike could not finish its tour by going back to the satellite. Thus, the lastly added AHD is removed from the tour and the remaining AHDs are not covered, but the bike goes back to the satellite to finish the tour.
- After having finished one bike tour, I check if the van is still at the satellite spot. If this is the case and the van still has free parcels for AHDs, the bike gets refilled immediately after coming back from its previous tour and it performs another bike tour, which is generated following the same steps described above. The one exception is that the start of the tour equals the time the bike is back from its recently generated previous tour plus 'service time' for loading. Note that all bike tours that are generated at one satellite spot are merged to represent one 'big' bike tour (e.g., the bike performs the tour "satellite-1-2-3-satellite" [numbers represent AHDs], and after coming back, if the van is still there and has two free parcels, the bike starts a new tour "satellite-4-5-satellite". Both tours are merged to "satellite-1-2-3-satellite-4-5-satellite").
- After the (merged) bike tour is finished, a 2-opt optimization is performed by exchanging the positions of two customers along the tour route. As the bike routes are not very long, all exchange possibilities are computed and the best one (shortest distance) is taken.

Appendix A depicts an example of a feasible initial solution. The initial solution is optimized by the ALNS algorithm, which is described in the next subchapter.

4.2. Adaptive Large Neighbourhood Search (ALNS)

The ALNS heuristic is an advanced version of the Large Neighbourhood Search (LNS) heuristic. The idea behind the Large Neighbourhood Search (LNS) heuristic is that new solutions are obtained by destroying parts of already existing solution and after that repairing this solution in a different way, which makes it possible to explore large neighbourhoods in the search of optimal solutions (Pisinger and Ropke, 2009). The ALNS extends the LNS by including multiple different destroy and repair methods, which are weighted according to their success of finding better solutions in the course of the optimization process and are chosen randomly within each search based on their weighting (Pisinger and Ropke, 2009).

Algorithm 2 shows a pseudocode of the implemented ALNS in this master's thesis. The following Subchapters 4.2.1 and 4.2.2. describe its destroy and repair operators in detail.

Algorithm 2: Implemented ALNS

```
1:  s, s'' = InitialSolution()
2:  scores = InitializeOperatorsScores()
3:  i = 0
4:  WHILE (i < w) DO
5:      IF δ = 0 THEN
6:          s' = s
7:      ELSE
8:          s' = s''
9:      END IF
10:     D = ChooseDestroyOperator(D_set, scores)
11:     s' = DestroySolution(s', D)
12:     R = ChooseRepairOperator(R_set, scores)
13:     s' = RepairSolution(s', R)
14:     feasibleSolution = FeasibilitiyCheck(s')
15:     IF (feasibleSolution = TRUE AND f(s') < f(s)(1+δ)) THEN
16:         IF δ = 0 OR (δ > 0 AND f(s') < f(s)) THEN
17:             s, s'' = s'
18:             i = 0
19:         ELSE IF (f(s') >= f(s) AND f(s') < f(s)(1+δ)) THEN
20:             s'' = s'
21:             i = i + 1
22:         END IF
23:         scores = UpdateOperatorsScores(scores)
24:     ELSE
25:         i = i + 1
26:     END IF
27: END WHILE
28: return s
```

<u>*Parameters' description:*</u>

s → best found solution

s'' → incumbent solution to compare for next itineration (worse than *s*, better than *allowed deterioration of s*)

s' → incumbent (to be) destroyed and repaired solution

i → itinerations without improvement

w → maximum number of itinerations without improvement allowed

δ → allowed deterioration rate

D → chosen destroy operator

D_{set} → set of destroy operators

R → chosen repair operator

R_{set} → set of repair operators

f(x) → objective function value of *x*

The introduced ALNS algorithm in this thesis works as follows. Starting from the initial solution (line 1 in *Algorithm 2*), the incumbent solution destroyed and repaired by choosing randomly a destroy and a repair operator based on their success rate of finding better solutions (lines 10-13). The destroy operators select some satellite visits of a van tour and remove the covered customers from these visits. The repair operators try to reinsert the removed customers somewhere else. After having repaired the incumbent solution, a feasibility check is done (line 14). This is necessary as van and bike tours have a lot of interdependencies due to the time synchronization, and each change in a van tour requires time resynchronizations, which can lead to infeasible solutions. The following feasibility checks are performed:

- all customers are covered;
- for each cargo bike no bike tours collide;
- vans and bike tours are timely synchronized;
- all SPUs are covered within their individual time windows;
- all the van and bike capacities are not violated.

After that, if the repaired solution is feasible, the algorithm checks if the repaired solution's objective function value is within δ percent of the best found solution '*s*' (line 15), and if yes, the chosen destroy and repair operators each get a point and the success rates are updated (line 23). If the repaired feasible solution is better than the best found solution, an improvement is found, and it is accepted as new best found solution (lines 16-18) and in the next itineration this solution will be set to be the incumbent solution (line 6). If the repaired solution is not better than the best found solution, but it is within δ percent of the best found solution (line 19), in the next itineration it will be accepted to be incumbent solution (and, thus, to be destroyed and repaired) (lines 8 and 20), but the current itineration gets marked to be 'without improvement' (line 21). The idea with δ as allowed deterioration rate aims to favour diversification: different neighbourhoods might be explored by accepting worse solutions for manipulation, which might help escaping local optima. If the repaired incumbent solution is not within δ percent of the best found solution '*s*', the solution is marked to be 'without improvement' and thus, the counter for 'itinerations without improvement' is increased (line 25). The ALNS algorithm destroys and repairs incumbent solutions until the stopping criterion 'maximum number of itinerations without improvement' is found (line 4). The algorithm returns the best found solution (line 28).

4.2.1. Destroy Operators

The ALNS algorithm chooses between two destroy operators: the Random Removal and the Worst Removal. These operators are adapted to the 2E-VRP-MPL-TW-TS.

The Random Removal destroy operator was firstly introduced by Ropke and Pisinger (2006) and successfully taken over by Hemmelmayr et al. (2012) for the 2E-VRP. This operator selects a certain number of customers randomly, removes them from the solution and puts them in a customer pool of uncovered customers. The Random Removal method in this master's thesis is slightly modified. Instead of directly selecting customers, it now randomly picks two to four (number is randomly generated as well) satellite visits from the van tours. Each satellite visit has an equal probability of being chosen. If there are multiple vans operating, the van tour is randomly chosen before selecting a satellite visit to ensure that visits from different van tours are included. This adjustment aims to have a more substantial impact on the solution. All customers covered in the selected satellite visits are eliminated from the solution and placed in a customer pool.

The 'original' Worst Removal by Ropke and Pisinger (2006) and later by Hemmelmayr et al. (2012) for the 2E-VRP removes a certain number of customers, whose elimination is beneficial, also called a high removal gain with gain "defined as the difference between the cost when the customer is in the solution and the cost when it is removed" (Hemmelmayr et al., 2012, p. 3219). The Worst Removal operator in this master's thesis is modified and works similarly to the Random Removal operator in this thesis with the following difference. Each satellite visit gets an individual score and probability for selection. Thus, satellite visits, whose removal is assumed to be more beneficial, are prioritized for selection and destruction. It is assumed that satellite visits, where only AHDs are serviced, are the most beneficial for removing, followed by satellite visits with only SPUs covered. The logic behind this is that usually it is better for a van to visit one satellite and to cover both SPUs and AHDs at the same time instead of visiting the same satellite twice to cover first the one type of customers and then the other one. Consolidating deliveries to cover multiple types of customers in a single visit can lead to significant time and cost savings, as it minimizes the need for multiple visits to the same location. The rationale of prioritizing satellite visits with 'only AHDs' over those with 'only SPUs' is that AHDs are more flexible for reallocation. Reallocating SPUs often requires adjusting times due to the 'standing time' the van as MPL is required to satisfy while covering SPUs.

Table 4 shows how the probability for selection is calculated by each destroy operator. In this small example, eight satellite visits are considered. The Random Removal operator takes each of them equally into account for selection as the choice is completely random. The Worst Removal operator ranks the satellite visits. Satellite visits #2 and #4 cover only AHDs and thus, they have the highest probability for selection, followed by satellite visits #6 and #8, where only SPUs are covered. The other satellite visits covering both types of customers have the lowest probability for selection.

Table 4: Selection probabilities by Random and Worst Removal destroy operators on a small example

Tour Data											
Satellite Visit		#1	#2	#3	#4	#5		#6	#7	#8	
Van Tour 1)	**0**	**S1**	**S2**	**S3**	**S4**	**S1**	**0**	**S2**	**S3**	**S4**	**0**
Covered SPUs		{1,2}	/	{6,7}	/	{16}		{12}	{13,15}	{8}	
Covered AHDs		{4,5}	{3,8}	{10}	{9,11}	{17}		/	{14}	/	
Random Removal											
Total Satellite Visits: 8											
Selection Probability 2)		12.5%	12.5%	12.5%	12.5%	12.5%		12.5%	12.5%	12.5%	
Worst Removal											
Individual Score		1	5	1	5	1		3	1	3	
Total Score: 20											
Selection Probability 3)		5%	25%	5%	25%	5%		15%	5%	15%	

1) 0 = Depot S = Satellite

2) Calculation for each satellite visit: 1 / Total Satellite Visits

3) Calculation for each satellite visit: Individual Score / Total Score

As already mentioned, both destroy operators select randomly two, three or four satellite visits. The customers covered during these satellite visits are eliminated from the solution and put in a customer pool. After this elimination, the subtour capacity, where customer(s) have been removed, is updated. If we consider the small example from *Table 4*, this means the following. Let’s assume that satellite visits #2, #5 and #6 are randomly selected by the Worst Operator. Therefore, customers 3, 8 and 12 (for satellite S2) as well as 16 and 17 (for S1) are eliminated. This elimination requires a capacity update for the van. In the first subtour containing satellite visits #1 to #5, one SPU (16) and three AHDs (3, 8, 17) are eliminated from the selected satellite visits #2 and #5. Thus, after the elimination, in this subtour the van has one parcel more in the compartments for covering an additional SPU (or AHD) and three parcels more for covering additional AHDs. For the second subtour containing satellite visits #6 to #8, a capacity for one SPU is released, because one SPU (12) is removed from selected satellite visit #6. The removed customers are put in a customer pool and they have to be reinserted into the solution by the repair operators.

4.2.2. Repair Operators

The three repair operators in this master’s thesis are: “Swap Satellite Visits & Reallocate Customers”, “Reinsert Satellite Visits & Reallocate Customers” as well as “Just Reallocate Customers”. These are newly introduced for the 2E-VRP-MPL-TW-TS. The first two repair operators aim to manipulate the van tours by altering the order of satellite visits, providing opportunities to explore various areas of the solution space. However, given the numerous interdependencies among vans and bikes in the problem, overly large manipulations of van tours may have a negative impact on finding feasible solutions. In contrast, the last operator exclusively searches for available capacity to reinsert removed customers and thereby aims to ensure greater solution feasibility.

The "Swap Satellite Visits & Reallocate Customers" operator exchanges the positions among the selected satellite visits by the destroy operators. After that, the customers from the customer pool are reallocated within the van tours.

The "Reinsert Satellite Visits & Reallocate Customers" operator determines randomly for each destroyed satellite visit the van tour and the place within the determined van tour, where the satellite visit should be reinserted. After that, the reallocation of customers begins.

The "Just Reallocate Customers" starts directly with the customer reallocation. As this operator does not manipulate the van tours, here for each customer in the customer pool it is prohibited to reallocate him/her to the satellite visit, from where it was originally removed. Other than that, the repair mechanism for customers reallocation is the same for all three operators and it works as follows:

1) Check each van's subtour for free capacity to cover SPUs and/or AHDs.
2) Itinerate over each satellite visit in each van's subtour with free capacity.
3) Itinerate over each SPU in the customer pool and reallocate each of them to its first adequate satellite visit from the itineration described in 2), considering that van's arrival time and SPU's time window should match (e.g., SPU in cluster #2 with time window 8-11 a.m. can be assigned to a visit of satellite S2 at 10 a.m., but not to a visit of S2 at 4 p.m.).
4) Itinerate over each AHD in the customer pool and reallocate each of them to its first satellite visit from the itineration described in 2). Determine the corresponding bike tour of the satellite visit, to which the AHD(s) is(are) reallocated and build a new bike tour by the cheapest insertion heuristic, considering the new reallocated AHD(s) and the existing ones in the bike tour. Assure that the bike is available to perform the tour by not colliding with other scheduled bike tours.
5) In case after 3) and 4) not all customers are reallocated, create 'dummy' subtour(s) with full capacity, containing all satellites. Reallocate remaining customers from the customer pool to the adequate satellite visits from the 'dummy' subtour.
6) Check for satellite visits with no customers assigned and eliminate them.

7) Recalculate arrival and departure times of each satellite visit, considering the distances between the visits (as van tours might have been changed), the rules for 'standing time' and 'pick-up time' at satellite visits with SPUs covered (as SPUs were reallocated). Timely resynchronize the bike tours with the vans due to the changed times in the vans' tours.

After these steps, the destroyed solution is repaired. However, due to the interdependencies in the problem setting, the repaired solution might be infeasible on some occasions. The integrated feasibility check into the ALNS algorithm, outlined in Subchapter 4.2., aims to detect and identify infeasible solutions, so that these can be discarded.

5. Computational Experiments

In order to analyse the performance of the ALNS algorithm and its operators, many computational experiments with different parameter values for 'number of SPUs' as well as 'standing time' and 'pick-up time' were conducted. Due to the novelty of the 2E-VRP-MPL-TW-TS, there are no instances in the literature, which could be used directly for the computational experiments. Therefore, I adapted one toy instance for the travelling salesman problem (TSP) and one toy instance for the vehicle routing problem (VRP) to meet the specific requirements of the 2E-VRP-MPL-TW-TS. The ALNS algorithm was run on these adapted instances on a computer with a 2.40 GHz Intel Core i5 processor and 8 GB RAM. The following subchapters provide details on the instance adaptation and the results obtained from the computational experiments.

5.1. Instances Description

Two instances were created for testing the ALNS algorithm for the 2E-VRP-MPL-TW-TS. The customers' locations are taken from two existing toy instances. The first one is "berlin52.tsp" which is a travelling salesman problem instance, containing 52 real customer locations in Berlin (Ruprecht-Karls-Universität Heidelberg, no date). This data set considers customers in a densely populated urban area. This makes it appropriate for the 2E-VRP-MPL-TW-TS. The second toy instance is the VRP instance "E-n51-k5.vrp", created by Christofides and Eilon in 1969 (VRP-REP, no date). It contains 50 randomly generated and uniformly distributed customers on a Cartesian coordinate system and one depot location.

After taking the customers' coordinates from each of the toy instances, I did the following steps to create instances that comply with the requirements of the 2E-VRP-MPL-TW-TS:

1) I divided the area with clients into four clusters by one vertical and one horizontal line so that each cluster contains approximately the same number of customers.
2) Using Solver in Microsoft Excel, I determined a satellite location for each cluster by applying the Steiner-Weber Problem. This problem represents a basic location model that calculates the optimal location for a facility or a satellite so that the total distance from this location to the customers is minimized (Dörner, 2022). If needed, the model allows for weighting, so that the location can be nearer to more important customers. Here, all customers in the cluster were considered equally.
3) I set the depot location to be at the origin of the Cartesian coordinate system with x and y axes values being 0.
4) I randomly selected a given number of customers to be SPUs. I set the remaining customers to be AHDs.
5) I randomly assigned an individual time window to each SPU customer considering the probabilities presented in Chapter 3.

Figure 7: Adapted "berlin52.tsp" instance with 15 SPU customers

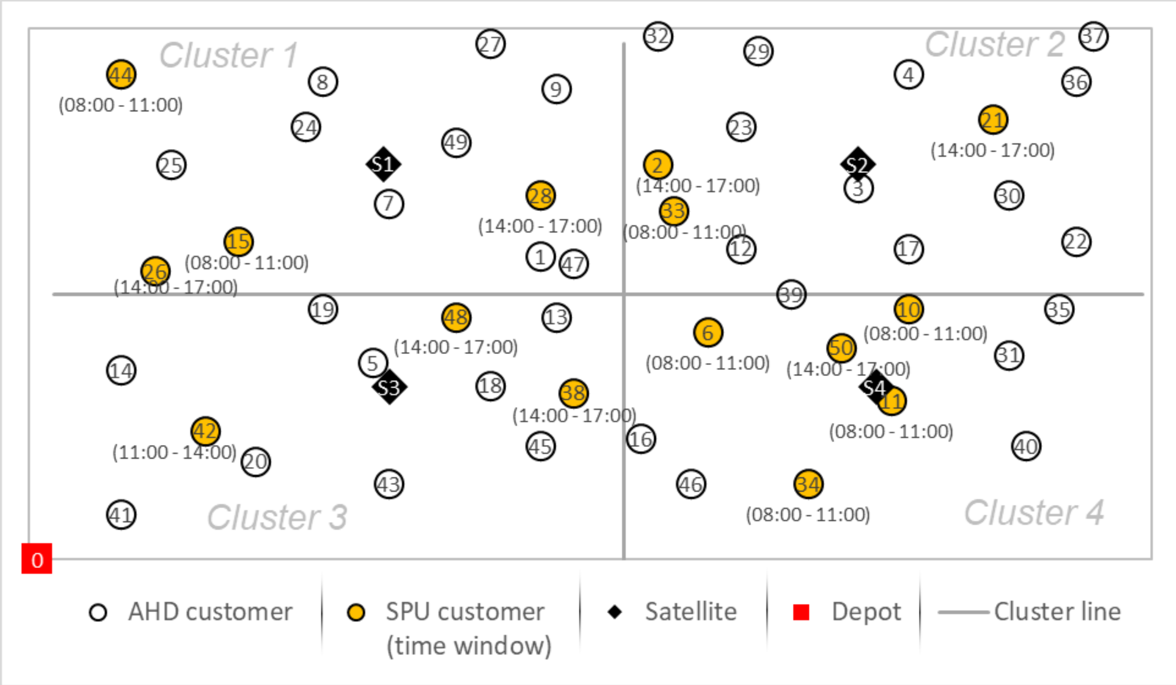


Figure 8: Adapted "E-n51-k5.vrp" instance with 15 SPU customers

The obtained results on the adapted instances are presented in the following subchapters. There are optimal results published for the toy instances, on which the adapted instances are based. However, no comparison can be made as the original toy instances and the adapted instances here differ in many aspects like setting (one echelon vs. two echelon), number of vehicles, type of customers, capacities and demands.

5.2. ALNS Performance Analysis

In the testing process, different scenarios considering all possible combinations among "0, 15, 25, 30 and 52 SPU customers[5]", "standing time of 60 and 30 minutes" and "pick-up time of 15 minutes or equal standing time" as well as "deterioration rate for accepting next incumbent solution of 0% and 5%" were considered for both instances. The ALNS stopping criterion for all scenarios tested was set to "1.000 itinerations without improvement". The ALNS algorithm was run 10 times and both the average result from these runs and the best result among them were stored.

[5] The variant with 52 SPUs holds for the berlin52 instance, while for the E-n51-k5 instance 50 SPUs were tested instead (representing 100% of the customers to be SPUs).

5.2.1. Initial Solution Improvement and Solution Quality

Table 5 presents the performance results of the ALNS algorithm on the adapted "berlin52" instance, when no deterioration is accepted within the itinerations. The best and the average results from all 10 runs are represented respectively in columns "ALNS best" and "ALNS average".

Table 5: Computational results with 0% deterioration rate (adapted "berlin52" instance)

	Objective Function Value				**Solution Quality**			
Scenario [1)]	**Initial Solution**	**ALNS average**	**ALNS best**	**% GAP [2)]**	**Coeffi-cient of Variation**	**Total Iti-nerations**	**Feasible Solutions**	**% Feasi-ble Sol.**
'Standing Time' = 'Pick-up Time'								
0SPUs	191.05	181.00	176.92	-7.4%	1%	4.708	3.068	65%
15SPUs_60ST	275.50	242.46	233.98	-15.1%	2%	1.635	212	13%
15SPUs_30ST	212.82	181.68	171.04	-19.6%	4%	3.056	172	6%
25SPUs_60ST	260.41	235.49	227.86	-12.5%	3%	1.081	364	34%
25SPUs_30ST	199.82	161.60	156.78	-21.5%	2%	1.360	133	10%
30SPUs_60ST	263.34	223.94	222.68	-15.4%	1%	1.430	602	42%
30SPUs_30ST	198.12	150.83	145.15	-26.7%	2%	1.588	773	49%
52SPUs_60ST	284.66	253.34	228.63	-19.7%	3%	1.011	434	43%
52SPUs_30ST	281.58	231.93	133.60	-52.6%	15%	1.135	449	40%
Diverse 'Standing Times' and 'Pick-up Time' = 15 Minutes								
0SPUs	192.05	177.87	175.98	-8.4%	1%	3.963	2.223	56%
15SPUs_60ST	270.10	215.33	199.38	-26.2%	5%	2.696	150	6%
15SPUs_30ST	213.24	178.59	170.21	-20.2%	4%	1.140	244	21%
25SPUs_60ST	271.39	222.16	205.52	-24.3%	7%	1.265	98	8%
25SPUs_30ST	197.29	154.31	148.65	-24.7%	2%	2.678	858	32%
30SPUs_60ST	268.28	205.89	192.02	-28.4%	6%	1.067	46	4%
30SPUs_30ST	214.64	146.17	134.39	-37.4%	4%	1.915	460	24%
52SPUs_60ST	582.06	404.72	203.18	-65.1%	26%	1.537	235	15%
52SPUs_30ST	476.72	307.11	142.21	-70.2%	28%	1.611	109	7%
Total [3)]						**34 876**	**10 630**	**30%**

1) "ST" in the scenario names stands for "standing time". For case '0SPUs', the standing time is irrelevant as all customers are AHDs.
2) The gap represents the percentage improvement of the initial solution relative to the best ALNS solution. It is calculated as "ALNS best / initial solution – 1", where a negative value indicates a reduction (improvement) of the initial solution's objective function value.
3) The total numbers for the (feasible) solutions consider both scenario types ("standing time = pick-up time" and "pick-up time = 15 minutes").

The results show that the ALNS algorithm improves the initial solution across all scenarios. It can be noted that when there are no SPUs included in the instance, the initial solution does not improve as significantly as when SPUs are included. One reason for this might be that including SPUs introduces many interdependencies in time and place, adding additional complexity to the problem, which the initial solution algorithm cannot fully handle. Therefore, there is a huge potential for optimizing routes, times and costs. On the other hand, in the case of delivering only to AHDs, the time windows of SPUs and the longer standing times of the van are omitted. Since the van's tour consists of routing among only four satellites, the nearest neighbour algorithm in the initial solution algorithm can handle this quite good, resulting in a lower potential for improvement. In terms of 'standing time', the results show that lower 'standing time' leads to higher the potential for improvements. Moreover, when lowering the 'pick-up time' to 15 minutes, the improvements are higher as the best results by ALNS are slightly lower than in the case when the 'pick-up time' equals the 'standing time'. Thus, it appears that this relaxation is beneficial for logistic companies. The highest improvements can be observed when all customers are SPUs. However, in these cases we can notice that the initial solutions are relatively high, and they seem to be outliers, while the corresponding ALNS best solutions are in line with the ALNS solutions of the other cases. This explains the increased improvement of the initial solution when having all customers as SPUs. At the same time, for these particular cases the algorithm seems to perform with some variability as shown by the Coefficient of Variation.

The Coefficient of Variation (CV) is a metric that provides insights into the relative variability of an algorithm's performance across multiple runs (Everitt and Skrondal, 2010). In this thesis, the CV is calculated by dividing the standard deviation by the average of the results from the 10 algorithm runs. It gives a sense of how much the results from the ALNS algorithm vary relative to the average performance. The purpose to include the CV in the analysis is to assess the solution quality of the implemented ALNS as we don't have a benchmark to compare our problem with. Guidelines indicate a CV in the range of 0% to 10% as low, meaning that the algorithm's performance is consistent and reliable across different runs (Gomez and Gomez, 1984). A moderate range (10% to 20%) implies some variability, but it may still be considered acceptable depending on the complexity of the problem (Gomez and Gomez, 1984). A CV above 20% suggests significant variability in the result, indicating that the algorithm's performance is less consistent (Gomez and Gomez, 1984).

The computational results show low CVs (1% to 7%) for all scenario cases involving 0, 15, 25 and 30 SPUs. However, in the four tested cases containing 52 SPUs, we have one case with a moderate CV (15%) and two cases with a high CV (26-28%). The relatively high initial solutions in these cases suggest that the construction heuristic delivers a poor initial solution, with which the ALNS algorithm struggles to work well in most runs. The reason for this might be the problem complexity, being challenging for the algorithm to consider all customers as SPUs while considering different 'standing times', which leads to greater fluctuations in construction heuristic's and ALNS algorithm's performance.

The same insights and trends as described above can be observed as well when we accept a deterioration rate of 5% within the itinerations of the ALNS algorithm, as shown in *Table 6*. However, this way the ALNS has found a lower number of feasible solutions.

Table 6: Computational results with 5% deterioration rate (adapted "berlin52" instance)

	Objective Function Value				Solution Quality			
Scenario 1)	**Initial Solution**	**ALNS average**	**ALNS best**	**% GAP 2)**	**Coeffi-cient of Varia-tion**	**Total Iti-nerations**	**Feasible Solutions**	**% Feasi-ble Sol.**
'Standing Time' = 'Pick-up Time'								
0SPUs	191.05	189.70	185.37	-3.0%	1%	1.008	324	32%
15SPUs_60ST	275.50	243.55	234.58	-14.9%	5%	1.083	11	1%
15SPUs_30ST	212.82	187.92	177.32	-16.7%	5%	1.088	13	1%
25SPUs_60ST	260.41	236.59	228.46	-12.3%	3%	1.038	19	2%
25SPUs_30ST	199.82	164.47	158.21	-20.8%	3%	1.124	87	8%
30SPUs_60ST	263.34	232.85	225.00	-14.6%	3%	1.029	19	2%
30SPUs_30ST	198.12	156.18	146.49	-26.1%	5%	1.131	17	2%
52SPUs_60ST	284.66	251.73	228.23	-19.8%	4%	1.028	18	2%
52SPUs_30ST	281.58	222.98	142.41	-49.4%	19%	1.039	25	2%
Diverse 'Standing Times' and 'Pick-up Time' = 15 Minutes								
0SPUs	192.05	189.68	186.53	-2.9%	1%	1.002	186	19%
15SPUs_60ST	270.10	219.12	204.45	-24.3%	4%	2.320	41	2%
15SPUs_30ST	213.24	177.89	171.60	-19.5%	3%	1.778	212	12%
25SPUs_60ST	271.39	223.58	214.06	-21.1%	5%	1.536	28	2%
25SPUs_30ST	197.29	160.60	151.81	-23.1%	5%	1.694	246	15%
30SPUs_60ST	268.28	205.71	197.54	-26.4%	2%	1.051	32	3%
30SPUs_30ST	214.64	147.19	137.89	-35.8%	4%	1.991	948	48%
52SPUs_60ST	582.06	406.82	206.86	-64.5%	25%	1.035	88	9%
52SPUs_30ST	476.72	308.91	149.88	-68.6%	27%	1.491	40	3%
Total 3)						**23 466**	**2 354**	**10%**

1) "ST" in the scenario names stands for "standing time". For case '0SPUs', the standing time is irrelevant as all customers are AHDs.
2) The gap represents the percentage improvement of the initial solution relative to the best ALNS solution. It is calculated as "ALNS best / initial solution – 1", where a negative value indicates a reduction (improvement) of the initial solution's objective function value.
3) The total numbers for the (feasible) solutions consider both scenario types ("standing time = pick-up time" and "pick-up time = 15 minutes").

Besides the CV, the number of feasible solutions found by the ALNS is another indicator for the solution quality. As already mentioned previously, the problem contains many interdependencies, so destroying and repairing a feasible initial solution might often lead to infeasible solutions. The results confirm this conjecture. In the scenarios where 'standing time' equals 'pick-up time' and no deteriorations are accepted within the ALNS itinerations, 30% of the solutions are feasible on average. However, when accepting a deterioration rate, the average number of feasible solutions decreases to 10%, as noted in *Table 6*. This implies that accepting worse solutions is not beneficial for the problem. The highest percentage of feasible solutions is found in the scenarios with 0 SPUs. This corresponds to the reflections in the previous paragraphs, addressing this scenario's lower complexity.

The findings for the adapted "berlin52" instance hold also for the adapted "E-n51-k5" instance, except the CV results. For the "E-n51-k5" instance, only results with no deterioration rate allowed were recorded, because preliminary results on this instance with an accepted deterioration rate greater than 0% indicated a low percentage of feasible solutions, similar to the results on the adapted "berlin52" instance in *Table 6*.

Table 7: Computational results with 0% deterioration rate (adapted "E-n51-k5" instance)

	Objective Function Value				**Solution Quality**			
Scenario 1)	**Initial Solution**	**ALNS average**	**ALNS best**	**% GAP 2)**	**Coeffi-cient of Varia-tion**	**Total Iti-nerations**	**Feasible Solutions**	**% Feasi-ble Sol.**
'Standing Time' = 'Pick-up Time'								
0SPUs	220.46	213.33	210.03	-4.7%	1%	5.237	3.535	68%
15SPUs_60ST	303.65	270.34	245.02	-19.3%	5%	1.520	9	1%
15SPUs_30ST	250.62	214.51	207.28	-17.3%	2%	1.835	83	5%
25SPUs_60ST	321.12	281.86	278.68	-13.2%	2%	1.029	274	27%
25SPUs_30ST	239.01	210.74	205.23	-14.1%	2%	1.441	351	24%
30SPUs_60ST	316.38	278.48	273.15	-13.7%	2%	1.036	518	50%
30SPUs_30ST	219.76	196.86	192.42	-12.4%	2%	1.554	738	47%
50SPUs_60ST	322.64	256.43	251.51	-22.0%	4%	1.051	755	72%
50SPUs_30ST	333.15	172.31	166.55	-50.0%	5%	1.033	787	76%
Diverse 'Standing Times' and 'Pick-up Time' = 15 Minutes								
0SPUs	220.46	213.33	210.03	-4.7%	1%	5.237	3.535	68%
15SPUs_60ST	305.93	235.72	228.34	-25.4%	2%	3.297	74	2%
15SPUs_30ST	249.36	212.01	202.61	-18.7%	5%	3.998	65	2%
25SPUs_60ST	288.29	258.84	244.29	-15.3%	3%	1.022	38	4%
25SPUs_30ST	221.08	197.97	190.66	-13.8%	2%	1.102	320	29%
30SPUs_60ST	284.92	255.64	239.82	-15.8%	4%	1.084	61	6%
30SPUs_30ST	207.19	190.28	185.60	-10.4%	1%	2.271	637	28%
50SPUs_60ST	661.26	281.45	252.65	-61.8%	8%	2.689	32	1%
50SPUs_30ST	623.00	189.44	178.62	-71.3%	3%	1.865	157	8%
Total 3)						**38 301**	**11 969**	**31%**

1) "ST" in the scenario names stands for "standing time". For case '0SPUs', the standing time is irrelevant as all customers are AHDs.
2) The gap represents the percentage improvement of the initial solution relative to the best ALNS solution. It is calculated as "ALNS best / initial solution – 1", where a negative value indicates a reduction (improvement) of the initial solution's objective function value.
3) The total numbers for the (feasible) solutions consider both scenario types ("standing time = pick-up time" and "pick-up time = 15 minutes").

For the adapted "E-n51-k52" instance, the ALNS algorithm improves the initial solution significantly in scenarios with SPUs included and slightly in cases with no SPUs. On average, 31% of the solutions are feasible when no deterioration rate is accepted, as we can see in *Table 7*. The only difference to the findings for the “berlin52” instance is that in all of the cases the CV is low. Thus, for the “E-n51-k5” instance the ALNS algorithm seems to be more consistent. This result might be influenced by many aspects such as customers’ locations and time windows, which determine the routes, making this instance more “friendly” for the ALNS algorithm.

5.2.2. Operators Performance

Table 8 shows how often (frequency) each of the operators has been selected and how successful (final probability) it has been for different scenarios of both adapted instances. The results are obtained by considering a deterioration rate of 0% and the best ALNS result out of the 10 runs. The operators’ performance can be derived by the operators’ final probability presented in *Table 8*: the higher the final probability (presented in %), the more often the operator has led to improvement. Regarding the destroy operators’ performance, the test results indicate that the ‘Worst Removal’ destroy operator outperforms the ‘Random Removal’ destroy operator. The ‘Worst Removal’ operator performed better in 10 of the 18 scenarios for instance “berlin52” and in 11 of the 18 scenarios for the instance “E-n51-k5”, whereas the ‘Random Removal’ operator was better in 6 scenarios for each of the instances tested. Both destroy operators were equally successful in two scenarios when considering the instance “berlin52” and in one scenario for the “E-n51-k5”. This finding underlines the idea that it is more desirable to destroy satellite visits covering only one type of customers, as covering both types of customers in a single satellite visit is more cost-efficient than doing so in two separate visits. In terms of repair operators, the ‘Just Reallocate’ repair operator is clearly the best performing, as it is the most frequently used repair operator with the highest success rate in 14 scenarios for “berlin52” and in 15 scenarios for “E-n51-k5”. The ‘Swap & Reallocate’ repair operator is the best in 6 scenarios for “berlin52” and in only 2 scenarios for the “E-n51-k5”. The ‘Reinsert & Reallocate’ could outperform the other two only once when considering the “E-n51-k5” instance, but other than that it could be only the second-best operator in some cases. Still, it has value for diversification purposes, and it can help explore new solution spaces. Both the ‘Swap & Reallocate’ and the ‘Reinsert & Reallocate’ repair operators manipulate the van tour. Their poorer performance compared to the ‘Just Reallocate’ operator might indicate that changes to the van’s route do not often lead to better solutions. This could be because the van travels among only four satellite spots, which the nearest neighbour algorithm in the initial solution algorithm handles well, or because many solutions become infeasible due to the temporal and spatial interdependencies in the problem.

Table 8: ALNS operators' performance results

		Destroy Operators (frequency, final probability)		Repair Operators (frequency, final probability)		
Instance	**Scenario [1)]**	**Random Removal**	**Worst Removal**	**Swap & Reallocate**	**Reinsert & Reallocate**	**Just Reallocate**
'Standing Time' = 'Pick-up Time'						
berlin52	0SPUs	423 (10.0%)	4285 (90.0%)	2761 (61.9%)	723 (9.5%)	1224 (28.6%)
	15SPUs_60ST	308 (21.4%)	1327 (78.6%)	204 (13.3%)	476 (26.7%)	955 (60.0%)
	15SPUs_30ST	529 (19.0%)	2527 (81.0%)	1200 (40.9%)	573 (18.2%)	1283 (40.9%)
	25SPUs_60ST	680 (63.6%)	401 (36.4%)	333 (33.3%)	124 (8.3%)	624 (58.3%)
	25SPUs_30ST	906 (66.7%)	454 (33.3%)	368 (30.8%)	109 (7.7%)	883 (61.5%)
	30SPUs_60ST	906 (64.3%)	524 (35.7%)	410 (33.3%)	134 (6.7%)	886 (60.0%)
	30SPUs_30ST	110 (6.3%)	1478 (93.8%)	215 (17.6%)	274 (11.8%)	1099 (70.6%)
	52SPUs_60ST	514 (50.0%)	497 (50.0%)	405 (40.0%)	221 (20.0%)	385 (40.0%)
	52SPUs_30ST	756 (66.7%)	379 (33.3%)	150 (14.3%)	401 (28.6%)	584 (57.1%)
E-n51-k5	0SPUs	1363 (26.7%)	3874 (73.3%)	2079 (43.8%)	2126 (31.3%)	1032 (25.0%)
	15SPUs_60ST	853 (50.0%)	667 (50.0%)	322 (22.2%)	314 (22.2%)	884 (55.6%)
	15SPUs_30ST	449 (27.3%)	1386 (72.7%)	329 (16.7%)	149 (8.3%)	1357 (75.0%)
	25SPUs_60ST	239 (25.0%)	790 (75.0%)	88 (11.1%)	135 (11.1%)	806 (77.8%)
	25SPUs_30ST	342 (18.2%)	1099 (81.8%)	300 (25.0%)	742 (41.7%)	399 (33.3%)
	30SPUs_60ST	146 (14.3%)	890 (85.7%)	108 (12.5%)	157 (12.5%)	771 (75.0%)
	30SPUs_30ST	1037 (63.6%)	517 (36.4%)	309 (25.0%)	180 (8.3%)	1065 (66.7%)
	50SPUs_60ST	719 (66.7%)	332 (33.3%)	131 (14.3%)	180 (14.3%)	740 (71.4%)
	50SPUs_30ST	787 (75.0%)	246 (25.0%)	128 (11.1%)	138 (11.1%)	767 (77.8%)
Diverse 'Standing Times' and 'Pick-up Time' = 15 Minutes						
berlin52	0SPUs	2249 (58.3%)	1714 (41.7%)	1606 (48.0%)	812 (16.0%)	1545 (36.0%)
	15SPUs_60ST	2358 (85.0%)	338 (15.0%)	1147 (47.6%)	1151 (33.3%)	398 (19.0%)
	15SPUs_30ST	382 (33.3%)	758 (66.7%)	175 (15.4%)	84 (7.7%)	881 (76.9%)
	25SPUs_60ST	660 (50.0%)	605 (50.0%)	498 (34.8%)	76 (4.3%)	691 (60.9%)
	25SPUs_30ST	430 (16.7%)	2248 (83.3%)	647 (26.5%)	61 (2.0%)	1970 (71.4%)
	30SPUs_60ST	151 (14.8%)	916 (85.2%)	97 (7.1%)	45 (3.6%)	925 (89.3%)
	30SPUs_30ST	854 (41.7%)	1061 (58.3%)	290 (13.5%)	360 (13.5%)	1265 (73.0%)
	52SPUs_60ST	514 (28.6%)	1023 (71.4%)	400 (20.0%)	203 (20.0%)	934 (60.0%)
	52SPUs_30ST	577 (38.1%)	1034 (61.9%)	427 (40.9%)	694 (27.3%)	490 (31.8%)
E-n51-k5	0SPUs	1363 (26.7%)	3874 (73.3%)	2079 (43.8%)	2126 (31.3%)	1032 (25.0%)
	15SPUs_60ST	613 (17.4%)	2684 (82.6%)	267 (8.3%)	1024 (33.3%)	2006 (58.3%)
	15SPUs_30ST	2355 (61.1%)	1643 (38.9%)	493 (10.5%)	999 (21.1%)	2506 (68.4%)
	25SPUs_60ST	274 (25.0%)	748 (75.0%)	272 (33.3%)	307 (22.2%)	443 (44.4%)
	25SPUs_30ST	836 (75.0%)	266 (25.0%)	244 (22.2%)	140 (11.1%)	718 (66.7%)
	30SPUs_60ST	885 (81.8%)	199 (18.2%)	257 (25.0%)	94 (8.3%)	733 (66.7%)
	30SPUs_30ST	930 (46.7%)	1341 (53.3%)	236 (12.5%)	165 (6.3%)	1870 (81.3%)
	50SPUs_60ST	820 (33.3%)	1869 (66.7%)	93 (4.0%)	371 (12.0%)	2225 (84.0%)
	50SPUs_30ST	175 (7.7%)	1690 (92.3%)	52 (7.1%)	865 (35.7%)	948 (57.1%)

Best performing operator for the particular scenario

1) "ST" in the scenario names stands for "standing time". For case '0SPUs', the standing time is irrelevant as all customers are AHDs.

6. Case Study: Mobile Parcel Lockers with Cargo Bike Deliveries in Vienna

Vienna is a typical big city, having small streets and urban areas in the centre and many people living in there. This makes the city a perfect representant for the 2E-VRP-MPL-TW-TS. The following subchapters describe the instance generation and analyse the obtained results.

6.1. Instance Generation

All requirements and assumptions described in Chapter 3 are considered and applied for the instance generation. In order to generate an instance for the 2E-VRP-MPL-TW-TS for Vienna, it is important to define the delivery area, to predefine the satellite spots and a depot location, to generate customers and divide them into AHDs and SPUs, and to assign each of them to a satellite. The following subchapters give details on each of the crucial steps. In order to consider one week (Monday to Friday), I created five instances for Vienna considering 200 randomly chosen customers in each instance, using the same approach. All instances consider the same delivery area as well as the same satellite spots and depot, but they differ in the customers' locations within the delivery area. For each instance generated, I created 11 different variants, each considering a different percentage of total customers being SPUs with randomly assigned time windows as described in Chapter 3. These percentages range from 0% to 100% SPUs out of the total number of customers with each variant increasing in 10% steps (i.e., 0%, 10%, 20%, ..., 100%). Each variant builds upon its previous one. This means that in each next instance variant all the SPUs and their individual time windows from the previous instance variant stay unchanged and only additional 10% SPUs and their individual time windows are newly determined by randomly choosing 10% of the remaining AHDs to become SPUs.

6.1.1. Determination of Delivery Area

Before starting to generate the customers' and satellites' locations, it is important to derive those areas in Vienna that are best suitable for the purpose of this thesis' problem. Vienna has in total 23 districts and an area of approximately 415 km². Its districts can be divided into three groups: "Inner City" – including districts 1 to 9 and 20, "Outer City" – including districts 10 to 19 and 23, as well as “Northen from Danube” – consisting of districts 21 and 22 (Wirtschaftskammer Wien, 2022). For this master’s thesis problem, the “Inner City” of Vienna is of interest because it represents the most densely populated area in the city as shown in *Table 9*.

Table 9: Vienna's area and population per district group (Wirtschaftskammer Wien, 2022; Stadt Wien, 2023)

Vienna District Group	District	Area in km^2	Population	Population per km^2 (population / area)
	1., Innere Stadt	2.9	16 623	5 794
	2., Leopoldstadt	19.2	108 306	5 629
	3., Landstraße	7.4	96 776	13 073
	4., Wieden	1.8	33 633	18 938
	5., Margareten	2.0	55 016	27 344
	6., Mariahilf	1.5	31 433	21 603
	7., Neubau	1.6	31 597	19 650
	8., Josefstadt	1.1	24 765	22 720
	9., Alsergrund	3.0	42 229	14 233
	20., Brigittenau	5.7	85 713	15 011
Inner City		**46.1**	**526 091**	**11 404**
Outer City		**234.4**	**1 125 743**	**4 803**
Northern from Danube		**134.4**	**330 608**	**2 461**

Inner City’s top 5 districts according to population per km^2

For the 2E-VRP-MPL-TW-TS we want to examine the combination of mobile parcel lockers and cargo bikes in urban areas, which are known for being relatively small and having many habitants. For this reason, for the instance generation, the top five districts in the inner city according to their population density are considered to be the delivery area. Therefore, the choice falls on districts 4 to 8 as it can be seen in *Table 9*. This is very convenient as well because these districts are adjacent to each other, forming one continuous delivery area as shown in *Figure 9*.

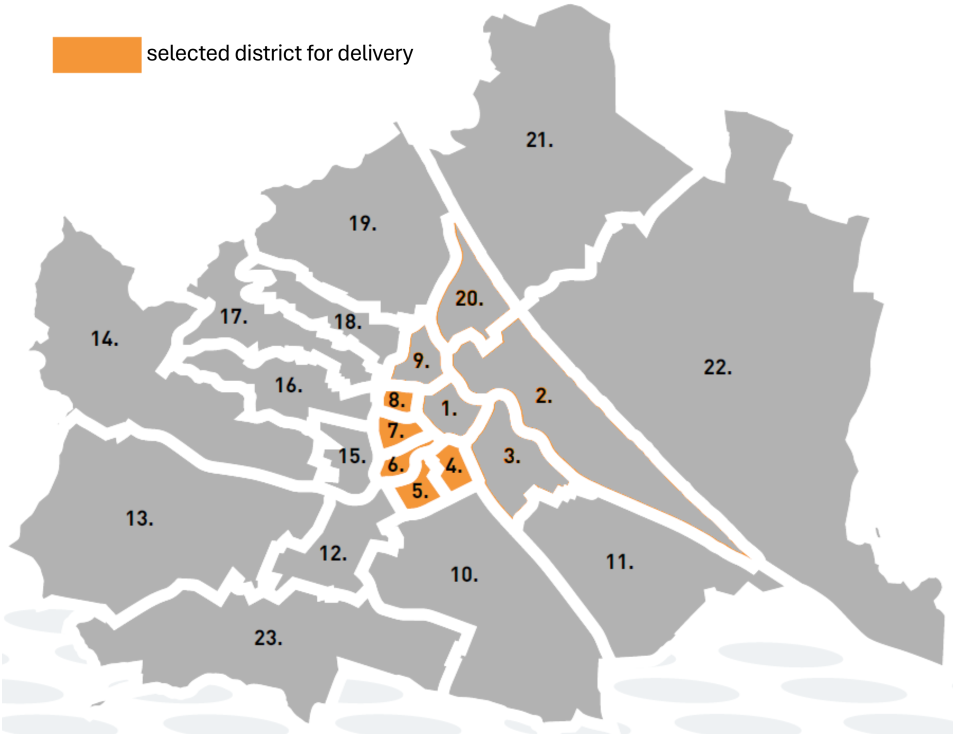


Figure 9: Vienna's districts (Wirtschaftskammer Wien, 2022)

6.1.2. Determination of Satellite Spots

The 2E-VRP, which is a part of this master's thesis problem, requires predefined fixed satellite locations. After an analysis of the determined delivery area, I noted that this area is quite huge to have only one satellite spot per district (means: five satellite spots in total). Therefore, I assumed that we would need six satellite spots in order to cover the delivery area properly. To identify possible satellite spot locations, I used Google Maps and its satellite and street view features to discover various spots and visually assess their convenience and accessibility. This way, six satellite spots could be predefined under the assumption that these can be used without any problems for free whenever needed. I also considered to have satellite spots that do not disturb cars and pedestrians, that are easily accessible and whose locations cover a good radius of the delivery area. *Figure 10* shows all of the predefined satellite spots.

Figure 10: Overview of chosen fixed satellite spot locations (images' data source: Google Maps)

6.1.3. Generation of Customers and Clustering

For the generation of customers, all official addresses in the selected Viennese districts 4 to 8 were extracted from the official data catalogue for Austria (Cooperation OGD Austria, no date). From there, 40 addresses per district were selected randomly, which makes in total 200 randomly selected customers. Each selected address represents one customer location.

The idea to cluster the customers per district is not covered in this master's thesis as Ramirez-Villamil et al. (2022) have found for Paris – the capital of France – that solving the 2E-CVRP by grouping multiple districts to serve customers is less costly and, thus, more effective than considering each district as a single cluster. Therefore, for the Vienna instances, I assigned each customer to his nearest satellite spot from the six possible satellite spot locations, resulting in six different clusters, in each of them having one cargo bike that operates only in its cluster.

Figure 11 shows a visualization for the first Vienna instance (representing Monday) without the clustering of customers to their nearest satellite spot. For visualisations of the remaining four instances refer to *Appendix B*.

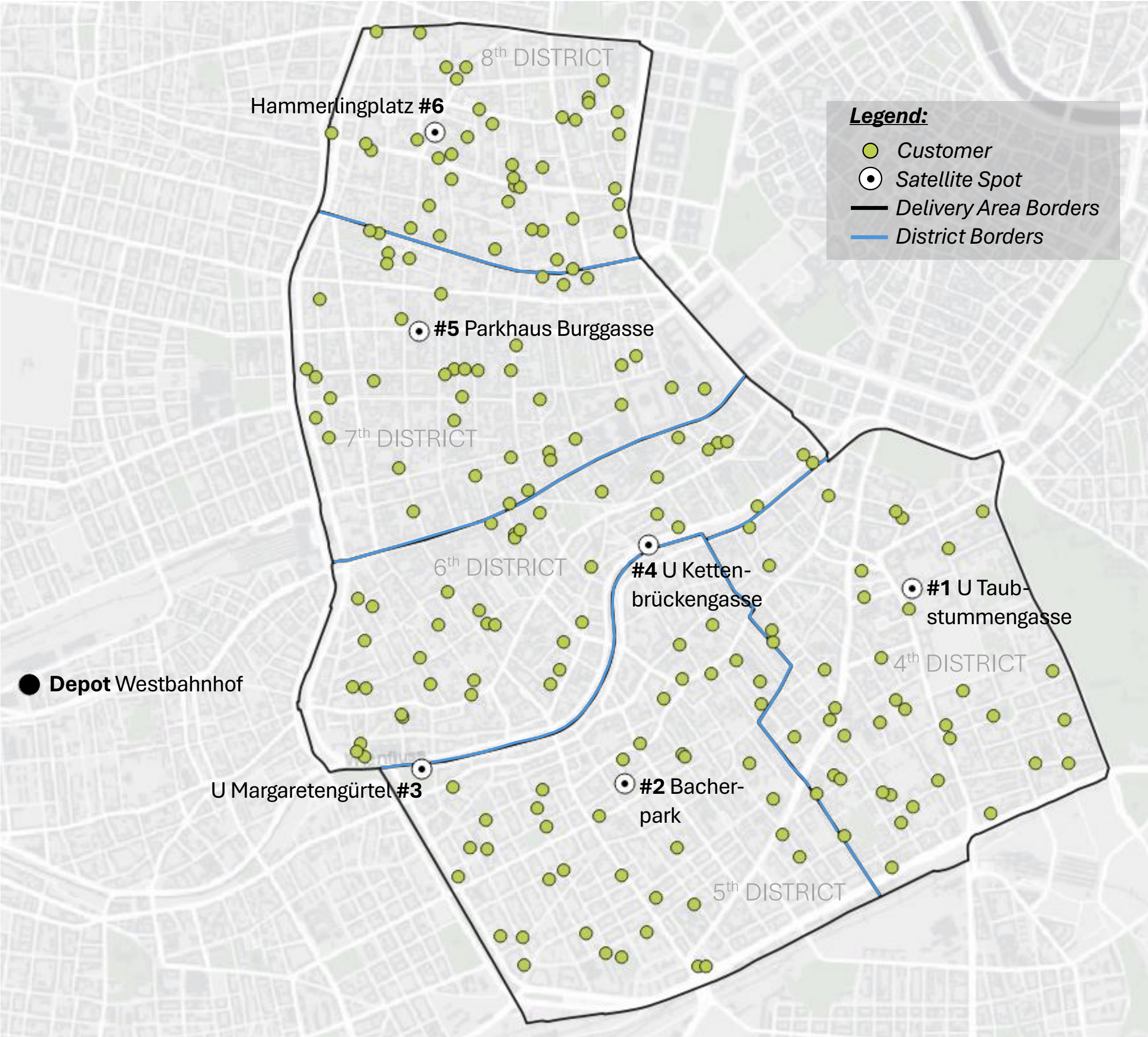


Figure 11: Visualization of Vienna instance #1 (Monday)

6.2. Results and Analysis

The computational results on the Viennese instances aim to give insights on how the model of combining mobile parcel lockers with cargo bikes performs in real-life urban areas. For all five instances and their 11 variants, representing different numbers of SPUs, the main assumptions in Chapter 3 are used and the following additional parameters are set:

- ALNS stopping criterion: 1.000 itinerations without improvement;
- Deterioration rate: 0%, meaning worse solutions are not accepted within the ALNS itinerations as shown to be non-beneficial in Chapter 5;
- Standing and pick-up time: eight combinations of 'standing times' (75, 60, 45 and 30 minutes) and 'pick-up times' (equal to 'standing time' and 15 minutes) are tested for each instance and variant.

For each variant of each of the five instances, I ran the ALNS algorithm 10 times and stored both the best result and the average of all 10 results. After that, for each variant I calculated an average considering the same variant of each instance.[6] The objective of this average calculation is to have average daily results based on one working week, where different customer locations are considered. All results are obtained using a computer with a 2.40 GHz Intel Core i5 processor and 8 GB RAM. The following subchapters provide detailed analysis of the total costs for the average daily results, broken down into travelling costs, vehicle usage costs and personnel costs. All figures and analyses in the following subchapters are based on the best ALNS result from the 10 ALNS runs, if nothing else has been explicitly noted.

6.2.1. "Standing Time = Pick-Up Time"

6.2.1.1. Travelling Costs and Vehicle Usage Costs

Figure 12 and *Figure 13* show the distance travelled by all vans and all bikes for the different "standing/pick-up time" scenarios based on the number of SPUs. The results in *Figure 12* reveal a clear trend, no matter the value of 'standing' or 'pick-up time': as the number of SPUs increases, the distance travelled by all bikes decreases. The reason for this is that the cargo bikes route less as there are less AHDs who should be serviced. The decrease of distance travelled by all bikes is almost linear and it is equal for each of the four scenarios because the number of bikes is always the same: six cargo bikes operate in six clusters if AHDs should be covered (variants with <100% SPUs), and no bike is needed when we have only SPUs (variant =100% SPUs).

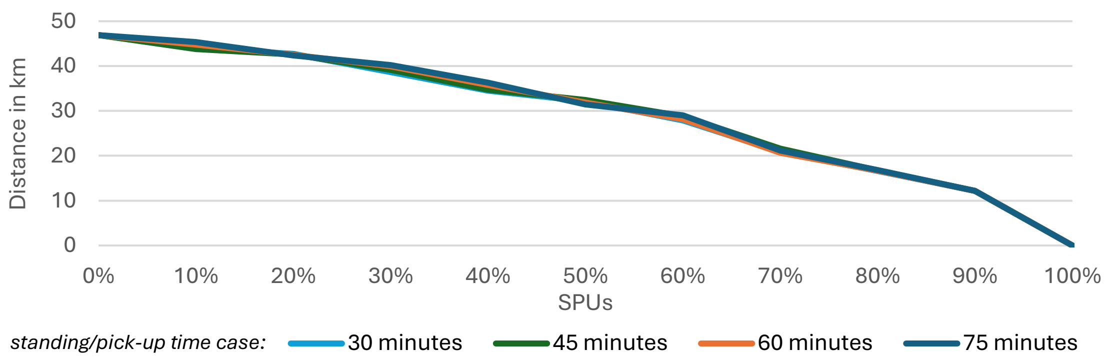


Figure 12: Distance travelled by all bikes used

[6] For example, if for variant "10% SPUs", the total costs for instance #1 to #5 (Monday to Friday) are respectively 100, 140, 120, 110, 123, the average daily costs for "variant 10% SPUs" are (100+140+120+110+123)/5=118.6. The same approach is applied for all variants.

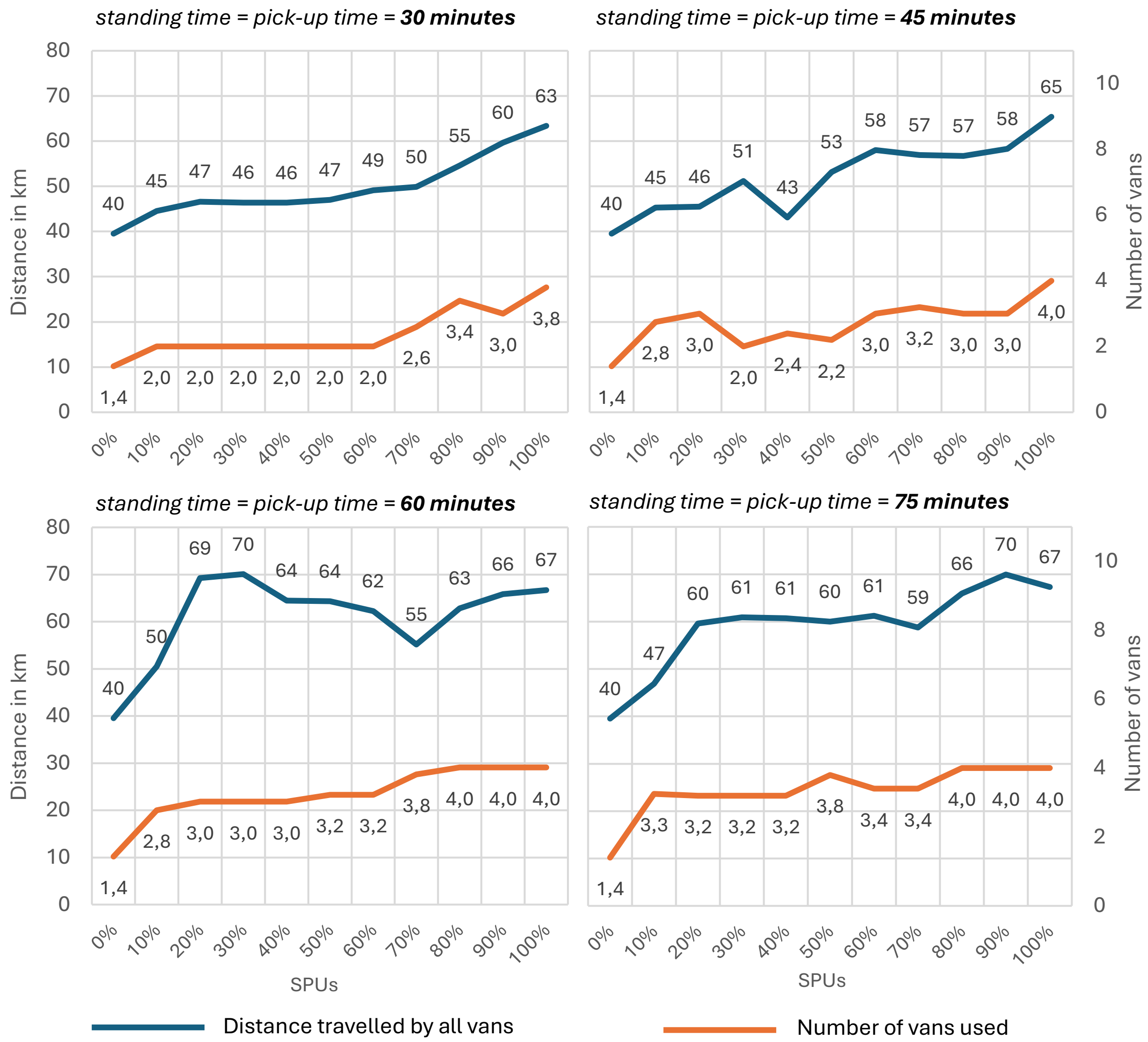


Figure 13: Distance travelled by all vans used

Conversely, increasing the number of SPUs leads to an increase in the distance travelled by vans, as we can see in *Figure 13*. In general, an increased demand of SPUs means a need of more vans. This can be a reason for the increase in distance, but it is not the only aspect. This relationship is complex and influenced by multiple factors such as the SPUs' time windows, the duration a van stays at a satellite spot and the efficiency of its routing.

Increasing the number of SPUs itself might not need always an increase of the number of vans, but it might increase the distance travelled, depending on the SPUs and their time windows.[7] For instance, if two SPUs are assigned to the same satellite but they have two different time windows, a van should visit the satellite twice, resulting in a higher total

[7] E.g., according to *Figure 13* for scenario "30 minutes standing time", for 10% to 60% SPUs only two vans are needed, but the distance increases. Variant 20% SPUs is small exception as in the next scenario a small decrease in distance follows. This emphasizes the complexity of the relation between SPUs and vans' total distance.

distance travelled. On the other hand, if the same two SPUs would have the same time window, the van could cover them with one visit under the condition the capacity is enough.

Moreover, the time the van should stay at a satellite, seems to influence the vans' travelled distance as well. For example, for the 90% SPUs variant for scenario "75 minutes standing time", four vans are needed, and they travel 70 km. For scenario "60 minutes standing time" we still need four vans, but they travel 66 km because possibly the lower standing time allows each van to route a little bit more and thus, to have a higher capacity utilization and more efficient routes. When we decrease the standing time to 30 minutes, we need only three vans and they cover all the SPU demand by travelling 60 km. However, we cannot assume that decreasing the standing time exclusively decreases the distance travelled because there are some outliers. For instance, for the "45 minutes standing time" scenario for the 90% SPUs variant, three vans need only 58 km, which is lower than when having 30 minutes of 'standing time'. In addition, there are cases where more vans are needed, but the total distance travelled is lower – e.g., for variant 20% SPUs for the 75 minutes "standing time" scenario 3.2 vans are needed on average with total distance travelled of 60 km, but for the scenario "60 minutes" 69 km are travelled in total by three vans to cover the total SPU demand. These outliers highlight that many factors influence the vans' total distance travelled. Nevertheless, in general, the insights reveal that lower standing time tends to give more possibilities for more efficient routes than higher standing time, which might be caused by a higher capacity utilization as vans have more time to route among the satellites and thus, to cover more SPUs. Moreover, determining the SPUs and their time windows randomly leads to an alternating behaviour of vans' total distance travelled. Thus, this analysis highlights the importance of synchronizing vans' standing time with SPUs' locations and time windows.

All in all, increasing the number of SPUs leads to an increasing trend of total travelling costs due to the fact that more vans are needed and/or because vans need to route more among the satellite locations, in order to cover SPUs and satisfy their different time windows, as analysed in previous paragraphs. By increasing the SPUs, the number of AHDs decreases and, thus, the distance travelled by bikes decreases as well. However, this decrease in total distance travelled by bikes cannot compensate the increase of total travelling costs, because the distance travelled by vans is more costly. *Figure 14* shows the total travelling costs for each 'standing time' scenario and "SPU" variant. As discussed before, the random determination of SPUs and their time windows gives varied patterns in vans' total distance and thus, total travelling costs, so that it cannot be exclusively said that decreasing the standing time decreases the travelling costs, as there are some outliers. However, there is a trend that lower 'standing time' gives more opportunities for more efficient routes, and thus, lower travelling costs. This puts emphasis on the complex relationship among all parameters in the model and points out that SPUs and vans should be rather (dynamically) synchronized than randomly and statically determined.

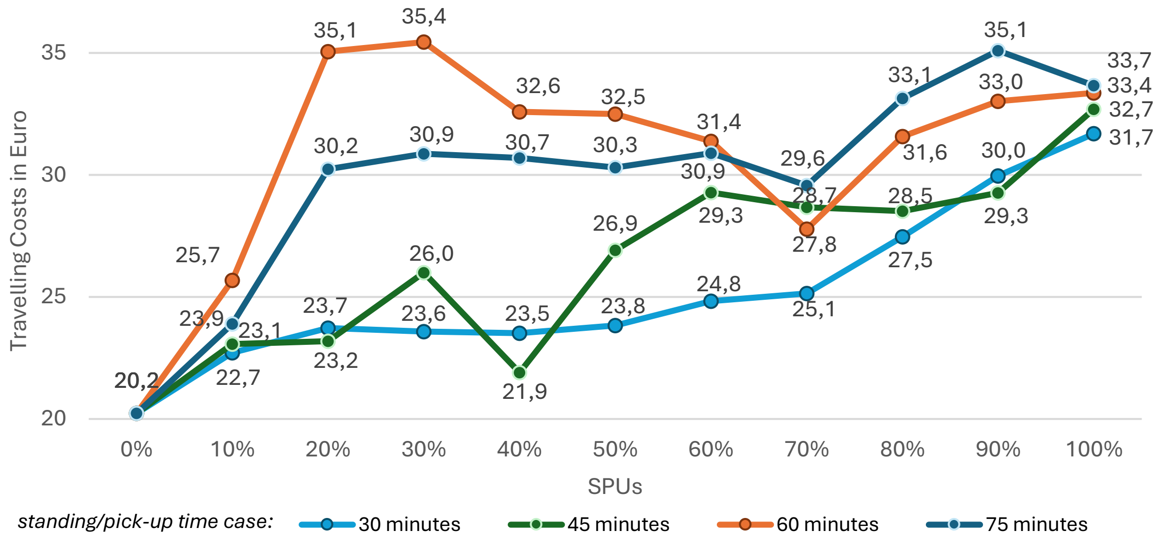


Figure 14: Total travelling costs per scenario

In terms of vehicle usage costs as shown in *Figure 15* and *Figure 16*, we notice that – except for small outliers – increasing the standing time increases the vehicle usage costs as in general more vans are needed. For all scenarios, the variant considering 100% SPUs contains the lowest vehicle usage costs because no bikes are needed, while in all other variants six bikes operate in each of the six clusters. Even if variant "100% SPUs" needs the highest number of vans, this cannot compensate the costs omitted by waiving six bikes.

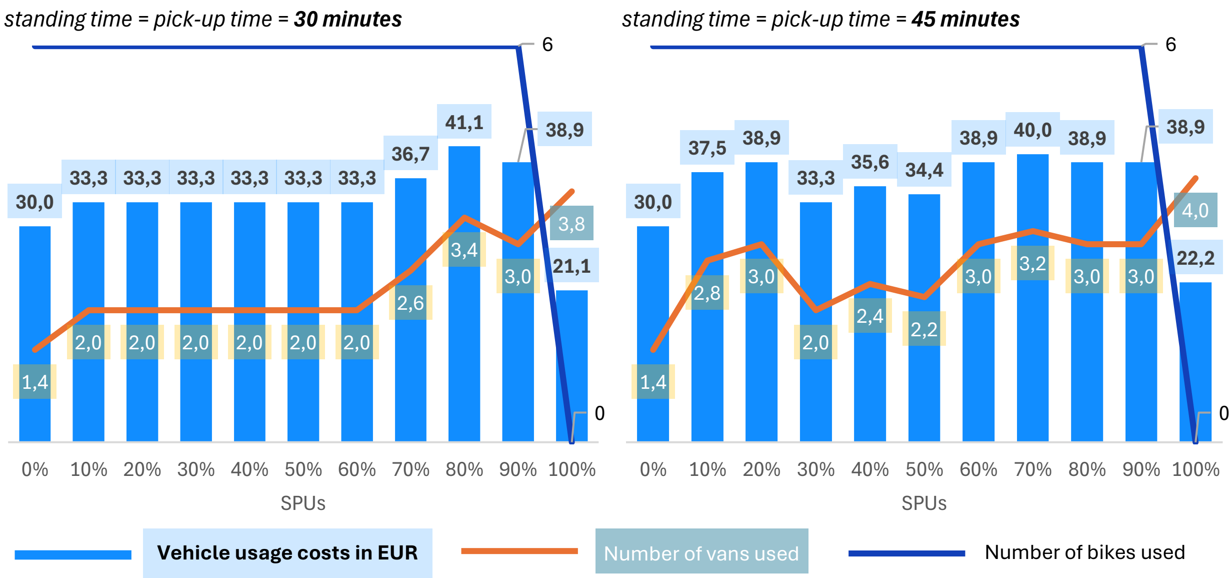


Figure 15: Total vehicle usage costs for low 'standing/pick-up times'

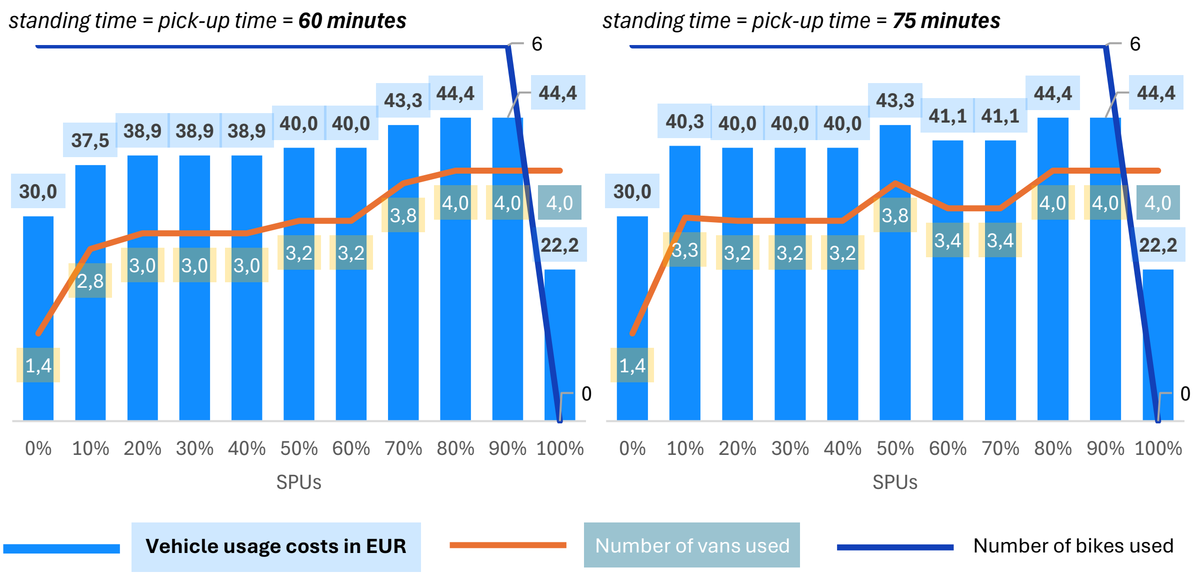


Figure 16: Total vehicle usage costs for high 'standing/pick-up times'

6.2.1.2. Personnel Costs

Figure 17 illustrates a clear trend: as the percentage of SPUs increases, drivers' salaries (personnel costs) generally decrease across all 'standing time' scenario cases. This suggests that a higher proportion of SPUs leads to more efficient operations, where fewer driver hours are required, which lowers overall personnel costs.

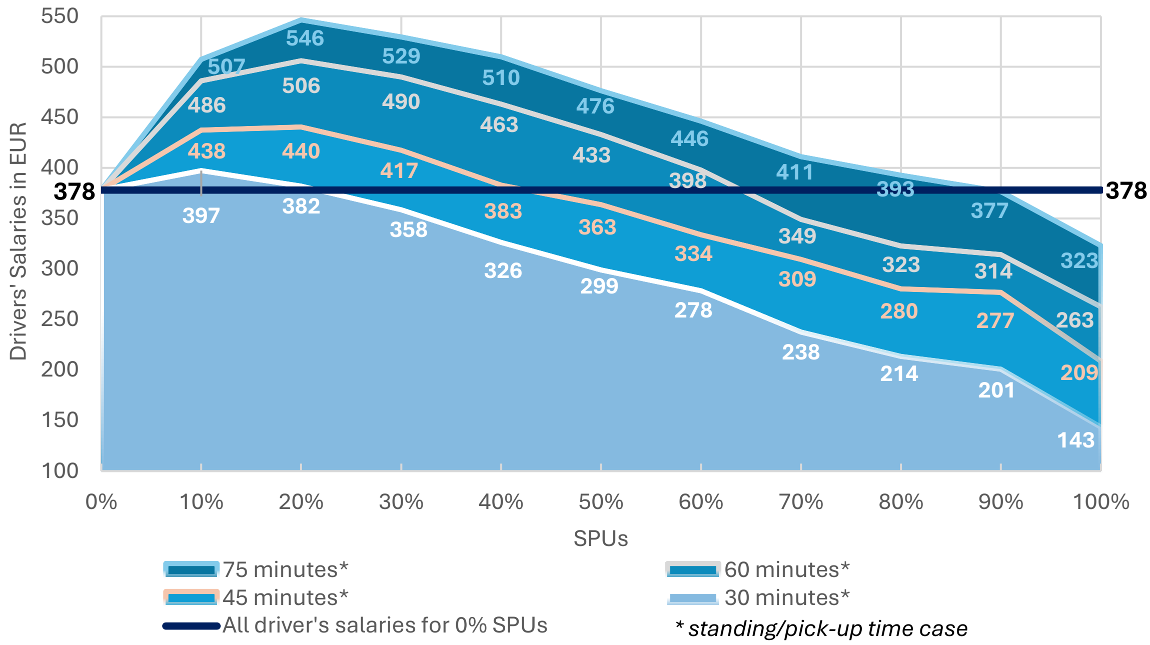


Figure 17: Total personnel costs (all drivers' salaries) per scenario

To go deeper, *Figure 18* breaks down the salaries and working hours of bike and van drivers across different SPU scenarios. The results show that bike drivers' working hours decrease significantly as the percentage of SPUs increases. For instance, with 60 minutes of ‘standing time’, bike drivers' working hours drop from 25.4h for 0% SPUs to 21.2h for 20% SPUs, continuously decreasing to 3.9h for 90% SPUs and to 0h for 100% SPUs. This indicates that bike drivers have significantly less work to do when more customers opt for self-pick-up, leading to corresponding lower personnel costs for logistics companies. In contrast, van drivers' working hours and salaries increase only marginally by adding more SPUs. The most substantial increase is observed till the 20% SPUs mark. Having 20% SPUs is the least efficient variant for logistics companies. At this level, both bike and van drivers' salaries remain relatively high. The bike drivers still have significant work to do, while the van drivers experience a noticeable increase in workload due to the ‘standing time’ required to service SPUs. This results in increased personnel costs without corresponding efficiency gains. When the percentage of SPUs exceeds 20%, the working hours and salaries of bike drivers continue decreasing significantly, but this is the point where the increase in van drivers' working hours and salaries starts being minimal or negligible. This suggests that while vans take on more deliveries with covering more SPUs, efficiency gains are obtained, minimizing the need for additional working hours. All in all, vans can cover multiple SPUs with fewer stops, and the need for bike drivers diminishes.

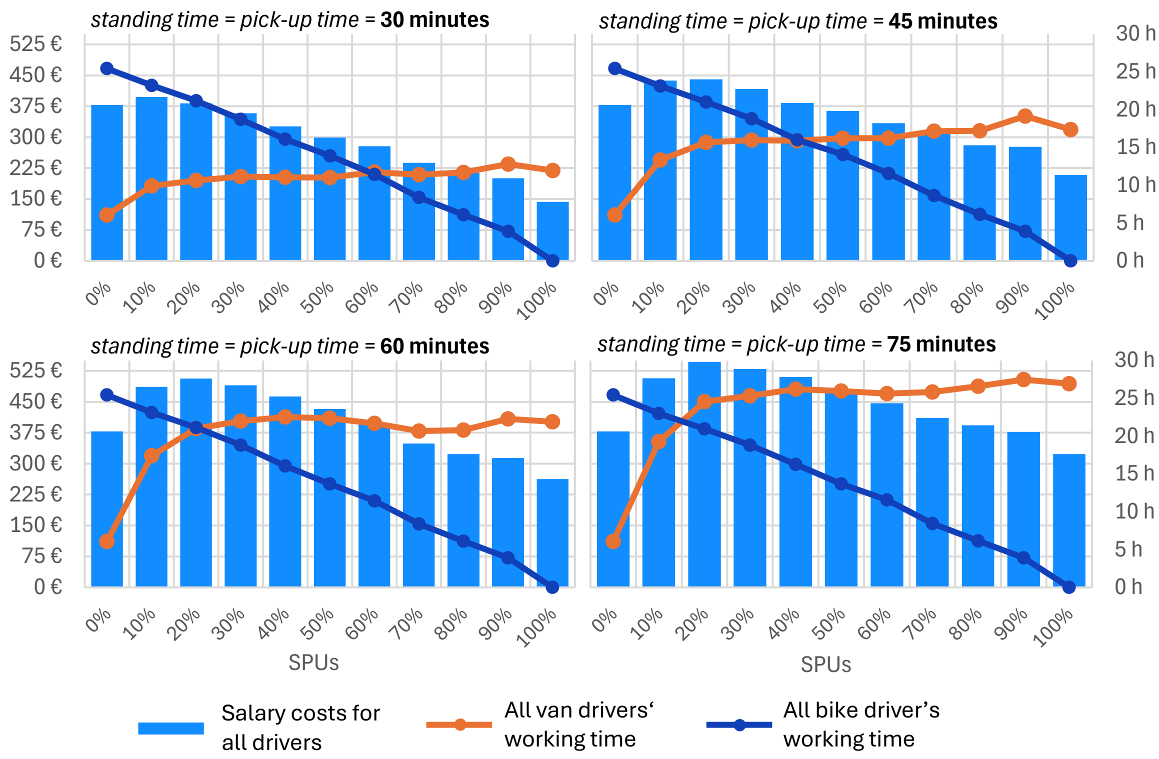


Figure 18: Impact of van and bike drivers' working hours on salary costs

6.2.1.3. Total Costs

The sum of all cost components from the previous subchapters represents the total costs. *Table 10* shows the average daily total costs (averaged from five instances' results) for each SPU variant and 'standing time/pick-up time' scenario from the one week consideration, obtained by the construction heuristic for the initial solution, and the ALNS algorithm. Column "ALNS average" contains the average of the total costs from each of the 10 runs of the algorithm, while column "ALNS best" represents the best objective function value (the lowest total costs) from these 10 runs.

The results show that the ALNS improves the construction heuristic. The extend for the improvement depends on the 'standing time/pick-up time' scenario and on the SPUs variant. For the variant with "0% SPUs", the ALNS algorithm improves the initial solution by 11% on average and 15% for the best case. However, when the number of SPUs increases, the improvement possibilities diminish, especially when the 'standing time' is higher. We can observe that for scenario "standing time = pick-up time = 75 minutes" the ALNS algorithm improves the initial solution from 2% to 10% considering the average ALNS solution and from 4% to 13% considering the best ALNS solution, depending on the different SPU variants. For scenario "standing time = pick-up time = 60 minutes", the improvement is respectively 3% to 9% and 4% to 12%. This improvement increases when setting the standing time lower. So, for scenario with "45 minutes", the ALNS improves the initial solution between 3% to 21% (for ALNS average) or 5% to 24% (for ALNS best) and for scenario "30 minutes" the range is even 6% to 41% considering ALNS average and 10% to 43% considering ALNS best. The highest improvements are observed for the variant "100% SPUs". Thus, we can derive two conclusions. First, setting the 'standing time' lower gives more opportunities for improvement as vans gain more time, which allows them to have more diverse routing options. Second, increasing the number of SPUs adds complexity to the problem, when there is a mix of AHDs and SPUs, among whom the whole routing must be coordinated, limiting the possibilities for higher optimization. This complexity reduces drastically when the mix of both types of customers is eliminated (means: 0% SPUs or 100% SPUs), giving higher potential for improvements.

The Coefficient of Variance is low for all scenarios and variants, ranging between 1% and 3%. It means that the implemented ALNS is consistent and reliable across different runs. Therefore, even if there are no results from similar problems we can use as benchmark, we can assume the ALNS delivers good solution quality for the Viennese instances.

Table 10: Average daily total costs in EUR for the Vienna case study

SPU variant	**Initial solution** (A)	**ALNS average** (B)	**% GAP** (A to B)	**ALNS best** (C)	**% GAP** (A to C)	**Coefficient of Variance**
Standing Time = Pick-Up Time = 75 Minutes						
0%	503.39	449.21	-11%	428.27	-15%	3%
10%	658.62	594.74	-10%	571.38	-13%	3%
20%	651.43	630.74	-3%	616.64	-5%	1%
30%	627.08	612.83	-2%	600.27	-4%	1%
40%	601.83	591.75	-2%	580.46	-4%	1%
50%	574.03	563.87	-2%	549.79	-4%	1%
60%	540.79	530.54	-2%	518.29	-4%	1%
70%	501.85	491.11	-2%	481.76	-4%	1%
80%	497.61	483.06	-3%	470.38	-5%	2%
90%	500.94	470.96	-6%	456.09	-9%	2%
100%	414.02	384.27	-7%	379.04	-8%	2%
Standing Time = Pick-Up Time = 60 Minutes						
0%	503.39	449.21	-11%	428.27	-15%	3%
10%	624.99	570.55	-9%	549.33	-12%	3%
20%	617.14	592.37	-4%	579.95	-6%	1%
30%	589.30	572.37	-3%	564.13	-4%	1%
40%	568.24	547.48	-4%	534.64	-6%	1%
50%	535.88	514.86	-4%	505.33	-6%	1%
60%	493.64	477.83	-3%	469.03	-5%	1%
70%	460.89	433.93	-6%	420.11	-9%	2%
80%	437.73	411.75	-6%	398.90	-9%	2%
90%	425.97	397.08	-7%	391.51	-8%	2%
100%	349.82	322.94	-8%	318.22	-9%	1%
Standing Time = Pick-Up Time = 45 Minutes						
0%	503.39	449.21	-11%	428.27	-15%	3%
10%	576.48	515.54	-11%	498.07	-14%	2%
20%	580.15	516.40	-11%	502.53	-13%	2%
30%	501.83	485.81	-3%	476.64	-5%	1%
40%	479.81	454.59	-5%	440.26	-8%	2%
50%	454.53	438.93	-3%	424.80	-7%	2%
60%	430.33	411.65	-4%	401.85	-7%	2%
70%	412.91	389.34	-6%	378.12	-8%	2%
80%	383.92	356.16	-7%	347.85	-9%	2%
90%	368.48	347.72	-6%	344.83	-6%	1%
100%	346.84	273.70	-21%	263.55	-24%	3%
Standing Time = Pick-Up Time = 30 Minutes						
0%	503.39	449.21	-11%	428.27	-15%	3%
10%	522.86	472.70	-10%	453.35	-13%	3%
20%	519.18	454.80	-12%	438.86	-15%	2%
30%	472.14	433.68	-8%	415.32	-12%	2%
40%	440.57	397.18	-10%	383.00	-13%	3%
50%	396.98	371.12	-7%	356.24	-10%	3%
60%	372.72	349.59	-6%	336.52	-10%	2%
70%	356.94	307.55	-14%	299.33	-16%	2%
80%	379.75	292.56	-23%	282.22	-26%	3%
90%	359.81	279.04	-22%	269.53	-25%	2%
100%	344.85	201.90	-41%	196.20	-43%	2%

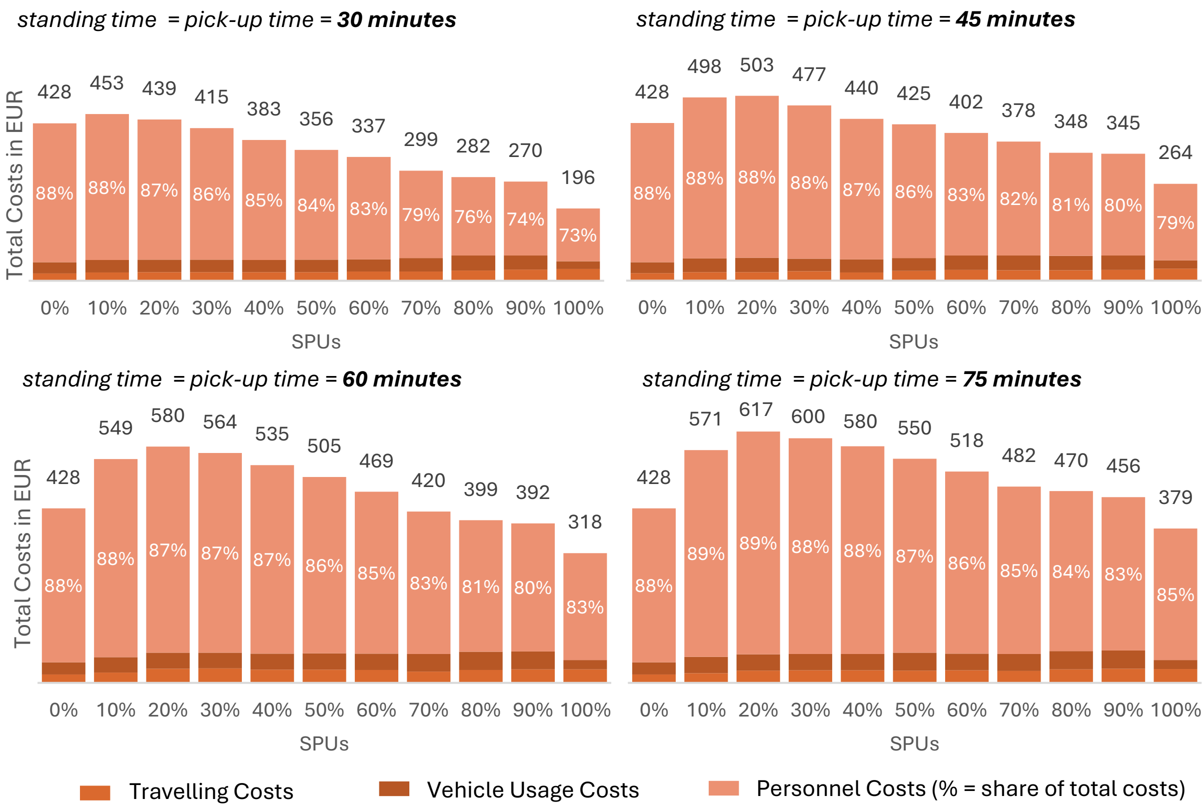


Figure 19: Total costs for each scenario by cost drivers

Figure 19 shows the total costs for each scenario considering all cost drivers, revealing that the personnel costs are the highest cost driver, holding the highest share of total costs. Thus, even if more SPUs mean higher travelling and vehicle usage costs (for the latter it does not hold for 100% SPUs), these represent such a small part of the total costs, so that they are negligible. Therefore, the personnel costs are the ones that influence significantly the total costs and as shown in the previous subchapter, adding more SPUs and reducing the 'standing/pick-up time', reduces the personnel costs significantly and thus, the total costs.

Talking of cost drivers, penalty costs for SPUs who are too far away from their assigned satellite is a practically reasonable cost driver for consideration. In terms of parcel lockers, customers are willing to walk to them if the distance is not exceeding one kilometre (Hovi and Bø, 2024). Buser (2021) has found for Groningen that customers want the parcel lockers to be even within a radius of 800 metres. Our instance analysis has shown that on daily average, from 200 customers in total we have only 16 customers who are more than 800 metres away from their assigned satellites. This is due to the good choice of satellites, covering the delivery area properly. For our problem, including such penalty costs would barely have any effect, making such costs negligible. Therefore, penalty costs for far away SPUs are omitted in this thesis' model.

For all scenarios the total costs could be reduced by introducing SPUs. However, the minimal number of SPUs needed is different for each ‘standing/pick-up time’ scenario. The results show that the shorter the ‘standing/pick-up time’, the less SPUs are needed so that the delivery gets cheaper than when all customers are AHDs. If ‘standing/pick-up time’ equals 75 minutes, all customers should be SPUs so that the model is beneficial. For 60, 45 and 30 minutes of ‘standing/pick-up time’, the minimal number of SPUs should be 70%, 50% and 30%, respectively. Nevertheless, in all scenarios the variant with the lowest total costs is when 100% of the customers are SPUs. In the first place, this finding implies that combining SPUs and AHDs is rather not to be aspired. By having all customers as SPUs, companies can save a lot of effort and money by not needing to coordinate at-home deliveries and by avoiding spending on bikes and bike drivers’ salaries. Moreover, the need for synchronization among vans and bikes can be avoided, allowing van routes to be more flexible. However, these results do not reflect on the service level and customer satisfaction. Delivering should be less costly for companies when all customers pick up their parcels on their own, but in practice many customers still prefer having their items delivered to home (Pharand, 2023).

For our problem it is very unlikely to have only AHDs or only SPUs. Studies for Europe report that 64% of the customers prefer home deliveries, while 24% of the customers opt for deliveries in parcel lockers or shops (DHL Group, 2023). At the same time, the results of our computations show that having 10% to 30% of the customers as SPUs leads to the highest total costs. That’s why, we can try to convince and incentivize more customers to become SPUs by offering a discount in the shipping price for each SPU. These discounts should be considered into the total costs as they are nothing else than penalty costs that the logistic company has to bear. The next subchapter provides an analysis including different penalty costs to the total costs.

6.2.1.4. Consideration of Penalty Costs for SPUs

In our model, we can practically rationalize a higher number of SPUs if we introduce a discount for each SPU. Thus, more customers would be incentivized to pick up their parcels on their own by offering lower shipping prices to them. Studies support this idea. The AlixPartners’ study from 2018 on consumers trends in shipping reveals that 96% of the surveyed customers indicate low delivery costs as the most important factor when it comes to parcel delivery. High shipping costs are a reason for 50% of the surveyed consumers to pick up their parcels on their own (AlixPartners, 2018).

This subchapter tests if by adding penalty costs to the total costs to incentivize SPUs, we can still have a mix of SPUs and AHDs that is better than having only AHDs. *Table 11* to *Table* 14 show the incentive (in EUR) per SPU needed for each scenario and variant, so that the sum of the total costs and penalty costs is equal to the total costs for the "0% SPUs" variant. Those variants marked with "n/a" indicate that the total costs for the particular variant are in any case higher than the total costs for variant "0% SPUs", making additional penalty costs obsolete. The calculations are based on the five instances' average of ALNS best results (average daily results).

Table 11: Break-even incentive per SPU for "standing & pick-up time = 75 minutes"

SPUs Variant	Number of SPUs	Total Costs (ALNS best)	Difference to Variant "0% SPUs"	Break-Even Incentive per SPU
0%	0	428.27	0.00	n/a
10%	20	571.38	-143.11	n/a
20%	40	616.64	-188.37	n/a
30%	60	600.27	-172.00	n/a
40%	80	580.46	-152.19	n/a
50%	100	549.79	-121.52	n/a
60%	120	518.29	-90.02	n/a
70%	140	481.76	-53.49	n/a
80%	160	470.38	-42.11	n/a
90%	180	456.09	-27.82	n/a
100%	200	379.04	49.23	0.24

Table 11 shows that for all variants except the "100% SPUs" variant, the total costs are higher than the costs for the "0% SPUs" variant and adding penalty costs will make the shipment even more costly, when vans' 'standing' and 'pick-up time' is 75 minutes. Only when having 100% SPUs it is reasonable to incentivize customers to become SPUs. The highest possible incentive in this case is 0.24 € per SPU[8]. This will lead to 48 € penalty costs (0.24 €/SPU × 200 SPUs) making the new total costs 427.04 € (48 € + 379.04 €). Every incentive below 0.24 € per SPU will make the costs lower and above 0.24 € - higher. This analysis shows for our model that it is practically rather improbable to convince more customers to be SPUs when our 'standing/pick up time' is 75 minutes due to two reasons. First, we need all customers to be SPUs, but this is practically less probable to reach as justified and supported by studies previously. Second, the low incentive required to keep costs manageable (<= 0.24 € per SPU) is unlikely to be effective in motivating a sufficient number of customers to change

[8] Calculation of the maximal incentive per SPU for the "100% SPUs" variant: (0% SPUs total costs – 100% SPUs total costs) / Number of SPUs for the 100% SPUs variant = (428.27 – 379.04) / 200 ≈ 0.2461 > 0 => 0.24 € per SPU. For all other variants the calculation approach is the same. If the incentive is negative (meaning the total costs of the inspected variant are higher than the total costs for "0% SPUs", this is marked as 'n/a'.

their behaviour. That's why, this scenario is impractical and inappropriate from a business perspective.

Table 12: Break-even incentive per SPU for "standing & pick-up time = 60 minutes"

SPUs Variant	Number of SPUs	Total Costs (ALNS best)	Difference to Variant "0% SPUs"	Break-Even Incentive per SPU
0%	0	428.27	0.00	n/a
10%	20	549.33	-121.06	n/a
20%	40	579.95	-151.68	n/a
30%	60	564.13	-135.86	n/a
40%	80	534.64	-106.37	n/a
50%	100	505.33	-77.06	n/a
60%	120	469.03	-40.76	n/a
70%	140	420.11	8.16	0.05
80%	160	398.90	29.37	0.18
90%	180	391.51	36.76	0.20
100%	200	318.22	110.05	0.55

Table 13: Break-even incentive per SPU for "standing & pick-up time = 45 minutes"

SPUs Variant	Number of SPUs	Total Costs (ALNS best)	Difference to Variant "0% SPUs"	Break-Even Incentive per SPU
0%	0	428.27	0.00	n/a
10%	20	498.07	-69.80	n/a
20%	40	502.53	-74.26	n/a
30%	60	476.64	-48.37	n/a
40%	80	440.26	-11.99	n/a
50%	100	424.80	3.47	0.03
60%	120	401.85	26.42	0.22
70%	140	378.12	50.15	0.35
80%	160	347.85	80.42	0.50
90%	180	344.83	83.44	0.46
100%	200	263.55	164.72	0.82

Table 12 and *Table* 13 show the results for the scenarios with respectively 60 and 45 minutes of 'standing/pick-up time'. Although here we can save costs by incentivizing less than 100% SPUs (for the 60/45 minutes scenario starting from 70%/50% SPUs), the incentive per SPUs is still way too low to convince more customers to change their mind. We need less than 0.05 € per SPU if we want to have 70% SPUs, when vans stay for 60 minutes at a satellite spot, and less than 0.03 € per SPU for 50% SPUs, when vans stay for 45 minutes. Increasing the incentive means increasing the number of SPUs, but these scenarios require a high number of customers as SPUs to make the setting beneficial. At the same time the incentive per SPU is still not convincingly high. Therefore, having a very high number of SPUs while

offering only low incentives is practically improbable as thematized before. Thus, these two scenarios are also inappropriate to be pursued from practical and business perspectives.

Table 14: Break-even incentive per SPU for "standing & pick-up time = 30 minutes"

SPUs Variant	Number of SPUs	Total Costs (ALNS best)	Difference to Variant "0% SPUs"	Break-Even Incentive per SPU
0%	0	428.27	0.00	n/a
10%	20	453.35	-25.08	n/a
20%	40	438.86	-10.59	n/a
30%	60	415.32	12.95	0.21
40%	80	383.00	45.27	0.56
50%	100	356.24	72.03	0.72
60%	120	336.52	91.75	0.76
70%	140	299.33	128.94	0.92
80%	160	282.22	146.05	0.91
90%	180	269.53	158.74	0.88
100%	200	196.20	232.07	1.16

Table 14 shows that for scenario "standing time = pick-up time = 30 minutes", offering for example a 0.50 € incentive per SPU can result in at least 40% of customers becoming SPUs (0.56 € is the incentive needed for a break-even point) and still have lower costs. Even by increasing the incentive up to for example 0.70 €/SPU, it is feasible to achieve a realistic and moderate SPUs rate, with at least 50% of total customers, and still save costs. These results imply that we can reach a practically acceptable number of SPUs by offering more reasonable incentives, making this scenario highly flexible and with potential for practical application. For that reason, *Table 15* offers a simulation of the total costs with different penalty costs added to them.[9] The penalty costs are based on different incentive values per SPU (0.40 €/SPU to 0.90 €/SPU), such that the costs are kept reasonable. Moreover, there is not a huge room for high incentives as the highest incentive a company can allow itself is 1.16 €/SPU and then only 100% SPUs will be acceptable from a cost perspective. The inspected variants start from 40% SPUs and reach to 80% SPUs. The last two variants (90 and 100% SPUs) are not included in the sensitivity analysis evaluation as it is not realistic to have (almost) all customers being SPUs as thematized before. The 0% SPUs variant is used as a benchmark.

[9] Detailed explanation for SPU variant "40%" and incentive of 0.50 €/SPU as an example: for this variant the total costs are 383.00 € (ALNS best), while the penalty costs are: (% of SPUs * total customers) * incentive per SPU = (40% * 200) * 0.50 = 40 €. The sum of total costs and penalty costs is: 383.00 + 40.00 = 423.00 €. The same calculation approach is applied for all other cases.

Table 15: Penalty costs simulation for "standing & pick-up time = 30 minutes"

		Incentive per SPU					
SPUs Variant	**Total Costs (ALNS best)**	**0.40 €/SPU**	**0.50 €/SPU**	**0.60 €/SPU**	**0.70 €/SPU**	**0.80 €/SPU**	**0.90 €/SPU**
		Total Costs + Penalty Costs					
0%	428.27	428.27	428.27	428.27	428.27	428.27	428.27
40%	383.00	415.00	423.00	431.00	439.00	447.00	455.00
50%	356.24	396.24	406.24	416.24	426.24	436.24	446.24
60%	336.52	384.52	396.52	408.52	420.52	432.52	444.52
70%	299.33	355.33	369.33	383.33	397.33	411.33	425.33
80%	282.22	346.22	362.22	378.22	394.22	410.22	426.22
		Total Costs + Penalty Costs: Relative Deviation to Total Costs for Variant "0% SPUs"					
0%		-	-	-	-	-	-
40%		-3.1 %	-1.2 %	0.6 %	2.5 %	4.4 %	6.2 %
50%		-7.5 %	-5.1 %	-2.8 %	-0.5 %	1.9 %	4.2 %
60%		-10.2 %	-7.4 %	-4.6 %	-1.8 %	1.0 %	3.8 %
70%		-17.0 %	-13.8 %	-10.5 %	-7.2 %	-4.0 %	-0.7 %
80%		-19.2 %	-15.4 %	-11.7 %	-8.0 %	-4.2 %	-0.5 %

After the corresponding penalties have been added to the total costs, the second part of *Table 15* contains a sensitive analysis showing the relative deviation in percent to the total costs for variant "0% SPUs". A negative percentage indicates that the corresponding variant is cheaper (including penalty costs) compared to the total costs of the "0% SPUs" variant, while a positive percentage means it is more expensive.[10] In the following paragraph, negative percentages refer to cost improvement (reduction), while positive percentages refer to cost deterioration (increase). The colour scale in *Table 15*, ranging from green to red with orange in between, classifies the extent of change: the greener the value, the greater the improvement (cost reduction), while the redder the value, the greater the deterioration (cost increase). Orange indicates moderate improvement.

The analysis shows that a higher number of SPUs (70% to 80%) and low to moderate incentive values (0.40 to 0.70 €/SPU) are the best ones for having relatively high or good improvement of costs within the range of -7.2% to -19.2%. At the same time, it is questionable if lower incentives could attract a large number of SPUs, where the improvements are the best. High incentives (0.80 and 0.90 €/SPU) deliver only moderate (-4.0% to -4.2%) and low (-0.5% to -0.7%) improvement of costs for large number of SPUs. Moreover, for less SPUs, these incentive values are not beneficial at all, leading to cost increase of 1.0% to 6.2%.

[10] For example, for variant "40% SPUs" and "0.50 €/SPU" incentive, the total costs with the penalty costs included amount 423.00 €, while the total costs for the "0% SPUs" variant are 428.27 €. Thus, we have an improvement, which is calculated as follows: 423.00 / 428.27 – 1 ≈ -1.2%. The same approach is applied for all cases.

Therefore, high incentive values (0.80 and 0.90 €/SPU) seem to be impractical, and it is not recommended for companies to implement them. It is recommendable to follow different strategies with lower to moderate incentives (0.40 to 0.70 €/SPU) as moderate to good improvements can be reached with them, while at the same time the number of SPUs can be kept relatively low.

Table 16 provides a simple scenario simulation. There, it is assumed that an incentive of 0.40 €/SPU could convince 40% to 50% of the customers to become SPUs, for 0.50 €/SPU the assumed SPUs share is 40% to 60%, for 0.60 €/SPU it is 50% to 70% and for 0.70 €/SPU – 60% to 80%. After that, three scenarios are carved out. The first scenario (A) is pessimistic. It considers the worst case for each incentive as the most probable, followed by the basic and the best cases. The probabilities for each case are calculated by geometric progression with a ratio of decrease of 50% (means: each case's probability is 50% lower than its predecessor's probability).[11] In the second scenario (B), equal probabilities are assigned to each case. The last scenario (C) is the optimistic one, considering the basic case as the most probable, followed by the best and the worst cases. There, the probabilities are calculated again by geometric progression with 50% ratio of decrease. As next, the expected improvement for each scenario is calculated for each incentive value by multiplying the corresponding cases' improvement of costs with their corresponding scenario probabilities and summing these results. For each incentive value, the average of the expected improvements of all its scenarios is the incentive values' overall expected improvement.

This simple simulation shows that applying 0.40 €/SPU of incentive could result in an improvement of -5.3%, while 0.50 €/SPU could lead to an improvement of -4.4%. For 0.60 €/SPU and 0.70 €/SPU we could expect -5.5% of improvement. The trend suggested by this particular simulation is that 0.50 €/SPU may not be the optimal choice, while at the same time it can be relatively indifferent from companies' perspective, if we choose 0.40, 0.60 or 0.70 €/SPU as an incentive. From the perspective of customers in Austria, this choice might not be so indifferent. According to Austrian customers, the average optimal price for a shipment is 4.25 € (Freyle, 2024). By reducing this price by 0.40 €, 0.60 € or 0.70 €, the customers can benefit from a discount of 9.4%, 14.1% or 16.5%, respectively.

[11] While modelling the scenario simulation, a sensitivity analysis with different ratios of decrease was performed. The results showed that the simulation is robust and remains largely unaffected by variations in the ratio.

Table 16: Incentive values simulation for "standing & pick-up time = 30 minutes"

	Incentive per SPU			
	0.40 €/SPU	**0.50 €/SPU**	**0.60 €/SPU**	**0.70 €/SPU**
Assumed Feasible Share of Customers Becoming SPUs				
Worst Case	40% SPUs	40% SPUs	50% SPUs	60% SPUs
Basic Case	50% SPUs	50% SPUs	60% SPUs	70% SPUs
Best Case	n/a	60% SPUs	70% SPUs	80% SPUs
Improvement of Costs (Penalty Costs Included)				
Worst Case	-3.1 %	-1.2 %	-2.8 %	-1.8 %
Basic Case	-7.5 %	-5.1 %	-4.6 %	-7.2 %
Best Case	-	-7.4 %	-10.5 %	-8.0 %
Probabilities of Scenario A (Pessimistic)				
Worst Case	67 %	57 %	57 %	57 %
Basic Case	33 %	29 %	29 %	29 %
Best Case	-	14 %	14 %	14 %
Probabilities of Scenario B (Neutral)				
Worst Case	50 %	33 %	33 %	33 %
Basic Case	50 %	33 %	33 %	33 %
Best Case	-	33 %	33 %	33 %
Probabilities of Scenario C (Optimistic)				
Worst Case	33 %	14 %	14 %	14 %
Basic Case	67 %	57 %	57 %	57 %
Best Case	-	29 %	29 %	29 %
Expected Improvement				
Scenario A	-4.6 %	-3.2 %	-4.4 %	-4.2 %
Scenario B	-5.3 %	-4.6 %	-6.0 %	-5.7 %
Scenario C	-6.0 %	-5.2 %	-6.0 %	-6.7 %
Average (equally weighted)	**-5.3 %**	**-4.4 %**	**-5.5 %**	**-5.5 %**

The results from above are based on a simulation, presenting only one possible outcome, making it difficult to give a clear recommendation. The key questions companies should ask themselves are: 1) what share of SPUs could realistically be achieved with which incentive value, and 2) how much risk the company is willing to take. Nevertheless, the analyses and the simulation in this subchapter highlight two findings for this thesis' problem. First, adding incentives for SPUs can practically justify the inclusion of (more) SPUs, making the problem more realistic. Second, even when accounting for penalty costs for SPUs, certain combinations of SPUs and AHDs remain less costly, compared to the same problem where all customers are AHDs. The last finding is especially applicable for the scenario "standing time = pick-up time = 30 minutes" as it gives more realistic opportunities for SPU incentives and total cost reductions.

6.2.2. Diverse 'Standing Times' and "Pick-Up Time = 15 Minutes"

As described in Chapter 3 (refer to *Figure 4* for a recap), by setting the 'pick-up time' lower than the 'standing time', vans are allowed to cover more SPUs, which in turn might result in more cost savings for logistics companies. The results (*Table 17*) support this assumption to a certain level. They show that in general total costs can be reduced by lowering the 'pick-up time' to 15 minutes, but not for all cases. The value of 'standing time' and the number of SPUs seem to have an impact on the improvement success and extend.

Table 17: "Standing time = pick-up time" vs. "pick-up time = 15 minutes" – total costs

ST [1)] Pick-up [2)]	**75 75**	**75 15**	**%**	**60 60**	**60 15**	**%**	**45 45**	**45 15**	**%**	**30 30**	**30 15**	**%**
SPUs	*A*	*B*	*B vs. A*	*C*	*D*	*D vs. C*	*E*	*F*	*F vs. E*	*G*	*H*	*H vs. G*
0%	428.27	428.27	0%	428.27	428.27	0%	428.27	428.27	0%	428.27	428.27	0%
10%	571.38	526.48	-8%	549.33	528.80	-4%	498.07	501.22	1%	453.35	447.78	-1%
20%	616.64	535.58	-13%	579.95	510.80	-12%	502.53	475.48	-5%	438.86	435.31	-1%
30%	600.27	534.33	-11%	564.13	490.71	-13%	476.64	448.61	-6%	415.32	412.75	-1%
40%	580.46	503.47	-13%	534.64	458.85	-14%	440.26	406.66	-8%	383.00	371.24	-3%
50%	549.79	475.82	-13%	505.33	432.69	-14%	424.80	377.99	-11%	356.24	348.24	-2%
60%	518.29	449.20	-13%	469.03	411.60	-12%	401.85	359.78	-10%	336.52	320.49	-5%
70%	481.76	436.89	-9%	420.11	394.18	-6%	378.12	348.17	-8%	299.33	297.66	-1%
80%	470.38	448.04	-5%	398.90	395.47	-1%	347.85	335.88	-3%	282.22	299.89	6%
90%	456.09	447.46	-2%	391.51	387.29	-1%	344.83	346.05	0%	269.53	285.11	6%
100%	379.04	344.22	-9%	318.22	328.61	3%	263.55	276.29	5%	196.20	211.24	8%

Note: Values represent the total costs in EUR (ALNS best) for the corresponding 'standing time' and 'pick-up time' case.
1) Standing time in minutes
2) Pick-up time in minutes

-X%	*Improvement: "Pick-up time = 15 minutes" reduces total costs compared to "standing time = pick-up time"*
X%	*Deterioration: "Pick-up time = 15 minutes" increases total costs compared to "standing time = pick-up time"*

We can observe a trend that the improvement is higher when 'standing time' is set to be high. On the other hand, for lower 'standing time' values, the improvement is lower. One possible explanation for this is that higher 'standing time' values are more restrictive than lower ones. For example, for scenario "standing time = pick-up time = 75 minutes", reducing the 'pick-up time' allows vans to cover additional SPUs at a satellite spot. In contrast, the "standing time = pick-up time = 30 minutes" scenario already allows vans to cover a larger number of SPUs at one satellite spot, because the van has to be only 30 minutes within a SPU's time windows at the satellite spot instead of 75 minutes in order to cover a SPU, which already leads to more optimal routes and less total costs, as shown in the previous subchapter. This is why further reductions in 'pick-up time' have less of an impact for shorter 'standing times'.

The second trend we can observe is that the best improvements are at mid-level number of SPUs (40% to 60%). After that, the improvements diminish as the SPU levels increase. For some cases reducing the ‘pick-up time’ to 15 minutes leads to even higher costs when we have a very high number of SPUs.

All in all, reducing the ‘pick-up time’ to 15 minutes leads to cost improvements if a practically acceptable and reasonable number of SPUs in the range of 30% - 70% (as rationalized in the previous two subchapters) is considered. However, the insights found so far do not change, as still a high number of SPUs is needed so that including SPUs is better than having only AHDs (for ‘standing time’ of 75, 60, 45 & 30 minutes respectively 100%, 60%, 40% & 30% SPUs needed at least when ‘pick-up time = 15 minutes’ vs. 100%, 70%, 50% & 30% SPUs needed at least when “standing time = pick-up time”). Moreover, a lower ‘pick-up time’ means even more inconvenience for SPUs, who – from a practical perspective – should be compensated by even higher incentives. Thus, cost reductions may be vaporized by higher penalty costs, caused by higher incentives. As seen in the previous subchapter, when incentivizing SPUs, the scenario with ‘standing time’ of 30 minutes is the only one that should be pursued, while all scenarios with higher ‘standing time’ are impractical and not beneficial. Here, the scenario with “standing time = 30 minutes and pick-up time = 15 minutes” is the one that delivers very low improvements for 30% to 70% of SPUs (-1% to -3% and -5% for 60 SPUs as a small outlier), and even a cost deterioration for 80% SPUs (6%). Therefore, the low improvements for up to 70% SPUs don’t seem to be enough to cover higher incentives for SPUs. All this being said, practically seen, lowering the ‘pick-up time’ doesn’t seem to be materially beneficial than having it equal to ‘standing time’.

7. Conclusion

This master’s thesis introduces the 2E-VRP-MPL-TW-TS to investigate a novel approach to last-mile delivery using two-echelon distribution network with mobile parcel lockers (MPLs) and cargo bike deliveries. This problem setting defines some customers to be delivered directly at home (AHDs) and some of them to pick up their parcels on their own directly from the MPL they are assigned to within their preferred time windows (SPUs). For the SPUs, three different time windows with a length of three hours are considered. For covering both types of customers, a heterogeneous fleet is taken into account. Vans act as first-echelon vehicles, MPLs and satellites at the same time. Cargo bikes are loaded by the vans at predefined satellite spots and act as second-echelon vehicles, delivering to AHDs. This requires multiple tactical decisions in terms of routing, considering time windows and strict time synchronization requirements, posing the challenge to coordinate deliveries for SPUs and AHDs, while minimizing the total operating costs. The thesis' model allows for a detailed analysis of how combining MPLs and cargo bikes affects total operating costs in an urban context.

To solve the 2E-VRP-MPL-TW-TS, a construction heuristic for obtaining an initial solution and an ALNS algorithm for solution optimization have been implemented. The ALNS algorithm has been tested on adapted toy instances and real-life instances of Vienna using diverse parameters, providing good solution quality and consistently improving upon the construction heuristic solutions, indicating the algorithm's robustness across different instances and configurations.

The experimental tests based on real-life instances of Vienna with 200 randomly selected customers in the centre of the city provide valuable insights into the impact of different 'standing times' for MPLs (the time a van stays at a satellite spot so that SPUs can pick up) and the share of SPU customers on the overall efficiency of the distribution model. The results show that integrating MPLs can reduce total operating costs, but the benefit depends on both the 'standing time' of vans and the share of SPUs. Long 'standing times' are costly, making the inclusion of MPLs not advantageous unless a high share of SPUs is included. Reducing the 'standing time' in turn, amplifies cost reductions, making the mix of SPU and AHD customers beneficial. The lower the 'standing time', the lower the share of SPUs can be to save costs. However, there is limited number of SPUs that has to be included at least, because even with short amount of 'standing time', the inclusion of a small share of SPUs (10% to 20% of the total customers) leads to higher costs, compared to a case where SPUs are omitted. Specifically, for the Vienna instances, having a low 'standing time' of 30 minutes requires at least 30% of the customers as SPUs so that combining MPLs with cargo bikes is advantageous from a cost perspective. In terms of share of SPUs, the results reveal that the greater the number of SPUs, the lower the total costs are. Therefore, when including MPLs to cargo bike deliveries, the results recommend aiming for a lower (but still practically reasonable) 'standing time' and as high as possible share of SPUs in order to minimize total costs.

The most substantial cost improvements are observed in scenarios with 100% SPUs. This indicates that, while a mix of AHDs and SPUs adds complexity and increases costs, focusing on SPUs can be more cost-effective. However, a fully SPU-based system may not be realistic given customers' preference for home deliveries according to studies. At the same time, studies highlight the wish of customers for lower shipment prices. That's why, the share of SPUs can be practically maximized by incentivizing customers to become SPUs with discounts in shipping price. This thesis provides a scenario analysis which incorporates incentives for SPUs, accounted as penalty costs in the objective function. This analysis demonstrates that it is still feasible to save costs by offering incentives to SPUs, even when a moderate share of SPUs (50% to 60%) is reached. However, companies should align the incentive they are willing to give with the potential share of SPUs they can really achieve, as too high incentives diminish the cost savings or even lead to increased costs.

This thesis provides insights into the different cost components. By increasing the number of SPUs, in general – with except of some outliers – more vans are needed, and they travel more. At the same time, the cargo bikes travel less as the number of AHDs decreases. In sum, the travelling costs increase because vans (and their travelled distances) are more costly than bikes. The vehicle usage costs increase as well when having both MPLs and cargo bikes. However, the analysis finds that personnel costs are the most dominant cost driver, when implementing a delivery system of MPLs and cargo bikes, overshadowing the cost factors "travelling costs" and "vehicle usage costs" and influencing the total costs immensely. By increasing the share of SPUs, the personnel costs decrease. More SPUs means synergy gains and more consolidated work for vans and less work for bikes: van drivers' working time increases only marginally, while the bike drivers' working time decreases significantly. This observation holds for each scenario of 'standing time'. The amount of 'standing time' influences the extend of cost decrease – the lower the 'standing time', the lower the personnel costs.

Perspectives for future research lie in further refinement of the model. MPLs have the feature to be flexible and able to react on changing demand by changing their locations. In this master's thesis, the satellite spots for the MPL locations are predefined, as the problem is based on the 2E-VRP. It is interesting to examine the impact on costs when MPLs are made more flexible. Therefore, they could be allowed to change their locations based on real-time demand, including the decision-making for satellite locations and shifting the problem toward a 2E-LRP framework, considering the customers left when determining the new MPL location. Moreover, more time windows for SPUs (e.g., eight time windows with a length of one hour) could be included and the impact of different clustering methods investigated. Given the dominance of personnel costs as main cost driver, future research might also explore the implementation of different aspects in the problem that could reduce these costs, such as occasional drivers[12] as well as autonomously driven vehicles or unmanned deliveries. Finally, future studies could evaluate the model's scalability and applicability in other urban environments and implement other metaheuristics for the problem, so that results could be benchmarked.

[12] Occasional drivers are implemented by Schwerdfeger and Boysen (2022) for the use of MPLs only (no combination of MPL with cargo bikes), considering one-echelon distribution network.

References

AlixPartners LLP [Editor] (2018): Top trend in shipping? More consumers want fast and free. URL: https://www.alixpartners.com/media/14552/ap_top_trend_in_shipping_mar_2018.pdf, accessed Aug. 24, 2024.

Anderluh, A.; Hemmelmayr, V. C.; Nolz, P. C. (2017): Synchronizing vans and cargo bikes in a city distribution network. *Central European Journal of Operations Research*, vol. 25, pp. 345-376.

Breunig, U.; Baldacci, R.; Hartl, R. F.; Vidal, T. (2019): The electric two-echelon vehicle routing problem. *Computers & Operations Research*, vol. 103, pp. 198-210.

Buser, M. (2021): The willingness of parcel locker use: A comparative case study of Groningen and Ten Boer. *University of Groningen*, Groningen 2021, URL: https://frw.studenttheses.ub.rug.nl/3637/1/BachelorProject_Final_MaaikeBuser_S3458822.pdf, accessed Aug. 19, 2024.

Capgemini [Editor] (2019): Share of total supply costs worldwide in 2018, by type of cost. In: *Statista*, URL: https://www-statista-com.uaccess.univie.ac.at/statistics/1043253/share-of-total-supply-chain-costs-by-type-worldwide, accessed Apr. 14, 2024.

Cavaliere, F.; Gromicho, J.; Vigo, D. (2022): Heuristics for the Vehicle Routing Problem: Methods and Implementations. *Lecture Notes v. 2.0 (November 25, 2022)*.

Cooperation OGD Austria [Editor] (no date): Adressen Standorte Wien, URL: https://www.data.gv.at/katalog/en/dataset/1d5c2411-9719-4c8f-b99d-57a5f4a4ae41#resources, accessed Jun. 30, 2024.

Crainic, T. G.; Mancini, S.; Perboli, G.; Tadei, R. (2008): Clustering-Based Heuristics for the Two-Echelon Vehicle Routing Problem. *CIRRELT*, vol. 46, pp. 1-28.

Crainic, T. G.; Mancini, S.; Perboli, G.; Tadei, R. (2011a): Multi-Start Heuristics for the Two-Echelon Vehicle Routing Problem. *11th European Conference - EvoCOP 2011 (Torino, Italy, April 27-29)*, vol. 201, pp. 179-190.

Crainic, T. G.; Mancini, S.; Perboli, G.; Tadei, R. (2011b): A GRASP with Path-Relinking Metaheuristic for the Two-Echelon Vehicle Routing Problem. *9th Metaheuristics International Conference - MIC 2011 (Udine, Italy, July 25-28)*, pp. 1-7.

Cuda, R.; Guastaroba, G.; Speranza, M. G. (2015): A survey on two-echelon routing problems. *Computers & Operations Research*, vol. 55, pp. 185-199.

DHL Group [Editor] (2023): Most popular delivery methods for European online shoppers in 2023. In: *Statista*, URL: https://www-statista-com.uaccess.univie.ac.at/statistics/1428400/delivery-preference-europe/, accessed Aug. 24, 2024.

Dörner, K. F. (2022): Basic Location Models: Median/Steiner-Weber; in: *Dörner, K. F.: Network Design in the Supply Chain, Lecture Notes, Summer Term 2022, University of Vienna*, Vienna 2022.

eft [Editor] (2020): What are your customers' most important wishes when it comes to last mile delivery? In: *Statista*, URL: https://www-statista-com.uaccess.univie.ac.at/statistics/817016/last-mile-services-customer-demand-from-retailers/, accessed Apr. 15, 2024.

eMarketer [Editor] (2023): Annual retail e-commerce sales growth worldwide from 2017 to 2027. In: *Statista*, URL: https://www-statista-com.uaccess.univie.ac.at/statistics/288487/forecast-of-global-b2c-e-commerce-growth/, accessed Apr. 13, 2024.

Enthoven, D. L.J.U.; Jargalsaikhan, B.; Roodbergen, K. J.; Broek, M.; Schrotenboer, A. H. (2020): The two-echelon vehicle routing problem with covering options: City logistics with cargo bikes and parcel lockers. *Computers & Operations Research*, vol. 118, 104919.

Everitt, B. S.; Skrondal, A. (2010): The Cambridge Dictionary of Statistics, 4th edition. Cambridge University Press (Publisher), New York.

Freyle, M. J. C. (2024): Versandkosten und deren Zahlungsbereitschaft im österreichischen B2C E-Commerce nach Branchen. *Wissenschaftliches Poster, FH Wiener Neustadt, Campus Wieselburg*, URL: https://www.wko.at/noe/e-commerce/versandkosten-und-deren-zahlungsbereitschaft, accessed Sep. 4, 2024.

Gendreau, M.; Potvin, J.-Y. (2009): Handbook of Metaheuristics, 2nd edition. *International Series in Operations Research & Management Science*, vol. 146, Springer (Publisher), USA.

Gomez, K. A.; Gomez, A. A. (1984): Statistical Procedures for Agricultural Research, 2nd edition. John Wiley & Sons (Publisher), Canada.

Grangier, P.; Gendreau, M.; Lehuédé, F.; Rousseau, L.-M. (2016): An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization. *European Journal of Operational Research*, vol. 254(1), pp. 80-91.

Hemmelmayr, V. C.; Cordeau, J.-F.; Crainic, T. G. (2012): An adaptive large neighborhood search heuristic for Two-Echelon Vehicle Routing Problems arising in city logistics. *Computers & Operations Research*, vol. 39(12), pp. 3215-3228.

Hovi, I. B.; Bø, E. (2024): Unlocking the potential: How can parcel lockers drive efficiency and environmental friendliness in E-commerce? *Sustainable Futures*, vol. 7, 100189.

Jepsen, M.; Spoorendonk, S.; Ropke, S. (2013): A Branch-and-Cut Algorithm for the Symmetric Two-Echelon Capacitated Vehicle Routing Problem. *Transportation Science*, vol. 47(1), pp. 23-37.

Kötschau, R.; Soeffker, N.; Ehmke, J. F. (2023): Mobile parcel lockers with individual customer service. *Networks*, vol. 82(4), pp. 506-526.

Lenstra, J. K.; Kan, A. H. G. R. (1981): Complexity of Vehicle Routing and Scheduling Problems. *Networks*, vol. 11(2), pp. 221-227.

Li, H.; Wang, H.; Chen, J.; Bai, M. (2020a): Two-echelon vehicle routing problem with time windows and mobile satellites. *Transportation Research Part B: Methodological*, vol. 138, pp. 179-201.

Li, H.; Liu, Y.; Chen, K.; Lin, Q. (2020b): The two-echelon city logistics system with on-street satellites. *Computers & Industrial Engineering*, vol. 139, 105577.

Loqate [Editor] (2021): Average cost per failed delivery in 2020, by country. In: *Statista*, URL: https://www-statista-com.uaccess.univie.ac.at/statistics/973182/last-mile-delivery-average-cost/, accessed Apr. 14, 2024.

Mancini, S.; Gansterer, M. (2021): Vehicle routing with private and shared delivery locations. *Computers & Operations Research*, vol. 133, 105361.

Orenstein, I.; Raviv, T.; Sadan, E. (2019): Flexible parcel delivery to automated parcel lockers: models, solution methods and analysis. *EURO Journal on Transportation and Logistics*, vol. 8(5), pp. 683-711.

Perboli, G.; Tadei, R.; Vigo, D. (2008): The Two-Echelon Capacitated Vehicle Routing Problem: Models and Math-Based Heuristics. *CIRRELT*, vol. 55, pp. 1-42.

Pharand, A. (2023): Do we really need parcel lockers for last mile delivery? In: *LinkedIn*, URL: https://www.linkedin.com/pulse/do-we-really-need-parcel-lockers-last-mile-delivery-andre-pharand/, accessed Aug. 19, 2024.

Pisinger, D.; Ropke, S. (2009): Large Neighbourhood Search; in: *Gendreau, M.; Potvin, J.-Y. (ed.): Handbook of Metaheuristics, 2nd edition. International Series in Operations Research & Management Science, vol. 146,* Springer (Publisher), USA, pp. 399-419.

Querin, F.; Göbl, M. (2017): An analysis on the impact of logistics on customer service. *Journal of Applied Leadership and Management*, vol. 5, pp. 90-103.

Ramirez-Villamil, A.; Jaegler, A.; Montoya-Torres, J. R. (2022): Sustainable local pickup and delivery: The case of Paris. *Research in Transportation Business & Management*, vol. 45(A), 100692.

Ropke, S.; Pisinger, D. (2006): An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. *Transportation Science*, vol. 40(4), pp. 455-472.

Ruprecht-Karls-Universität Heidelberg (no date): List of Instances for the Travelling Salesman Problem (TSP). URL: http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/tsp/, accessed Jun. 10, 2024.

Schwerdfeger, S.; Boysen, N. (2020): Optimizing the changing locations of mobile parcel lockers in last-mile distribution. *European Journal of Operational Research*, vol. 285(3), pp. 1077-1094.

Schwerdfeger, S; Boysen, N. (2022): Who moves the locker? A benchmark study of alternative mobile parcel locker concepts. *Transportation Research Part C: Emerging Technologies*, vol. 142, 103780.

Sevic (no date – a): Sevic V500e – Brochure. URL: https://sevic.com/data/fms/products/v500e/new%20payload/en-sevic-v500e-brochure-online_02.pdf, accessed Aug. 4, 2023.

Sevic (no date – b): Sevic V500e – Dimensions. URL: https://sevic.com/data/fms/brochures/English%20Version_V500e_Technical%20Sheet_final.pdf, accessed Aug. 4, 2023.

Sluijk, N.; Florio, A. M.; Kinable, J.; Dellaert, N.; Van Woensel, T. (2023): Two-echelon vehicle routing problems: A literature review. *European Journal of Operational Research*, vol. 304(3), pp. 865-886.

Stadt Wien (2023): Stadtgebiet nach Nutzungsklassen und Gemeindebezirken 2023. URL: https://www.wien.gv.at/statistik/lebensraum/tabellen/nutzungsklassen-bez.html, accessed Jun. 30, 2024.

Statista [Editor] (2024): Retail e-commerce sales compound annual growth rate (CAGR) from 2024 to 2029, by country. In: *Statista*, URL: https://www-statista-com.uaccess.univie.ac.at/forecasts/220177/b2c-e-commerce-sales-cagr-forecast-for-selected-countries, accessed Apr. 13, 2024.

Veenstra, M.; Roodbergen, K. J.; Coelho, L. C.; Zhu, S. X. (2018): A simultaneous facility location and vehicle routing problem arising in health care logistics in the Netherlands. *European Journal of Operational Research*, vol. 268(2), pp. 703-715.

VRP-REP (no date): Dataset "Christofides and Eilon 1969 – Set E" for the Vehicle Routing Problem. URL: http://www.vrp-rep.org/datasets/item/2014-0010.html, accessed Jun. 10, 2024.

Wang, Y.; Bi, M.; Chen, Y. (2020): A Scheduling Strategy of Mobile Parcel Lockers for the Last Mile Delivery Problem. *Promet – Traffic&Transportation*, vol. 32(6), pp. 875-885.

Wirtschaftskammer Wien [Editor] (2022): KEP-Branchenreport 2022, Ausgabe 7/2022, Wien 2022, URL: https://www.econsult.at/images/Bilder/Newsblock/Publikationen/KEP-Branchenreport-2022.pdf, accessed Jul. 30, 2023.

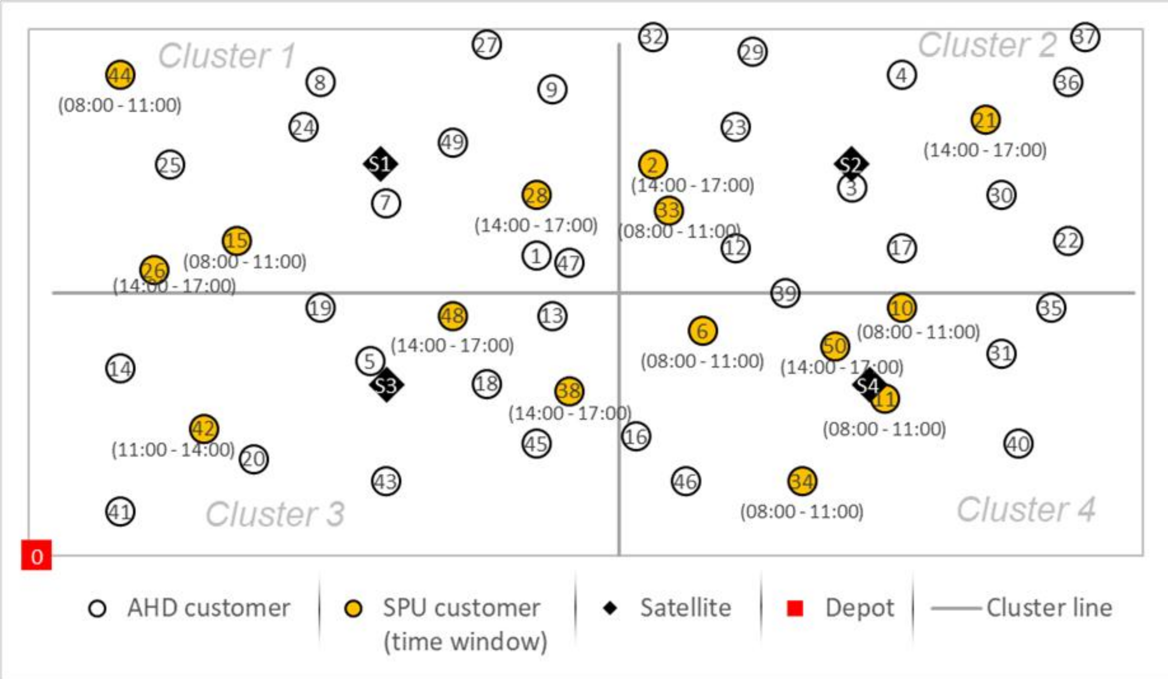
Yokler (no date): Yokler U – Product Features. URL: https://www.yokler.com/accueil-fr/en/buy-electric-cargo-bike-transport-goods-yokler-u/, accessed Aug. 4, 2023.

Yu, V. F.; Jodiawan, P.; Hou, M.-L.; Gunawan, A. (2021): Design of a two-echelon freight distribution system in last-mile logistics considering covering locations and occasional drivers. *Transportation Research Part E: Logistics and Transportation Review*, vol. 154, 102461.

Appendix

Appendix A: Example of a Feasible Initial Solution

<u>*Instance:*</u> *adapted "E-n51-k5.vrp" instance with 15 SPU customers*



<u>*Capacities:*</u> *6 compartments, 5 parcels (van); 3 parcels (bike) [corresponds to 25% of presented capacities in Chapter 3 as the total number of customers in the tested instance here is 25% of the total number of customers for the Viennese instances]*

<u>*Scenario:*</u> *"standing time = pick-up time = 60 minutes"*

<u>*Solution:*</u> *considering the rules in Chapter 4.1, see tables on the next page*

Tour of Van #1 *	0	S3	S1	S2	0	S3	S1	0	S3	S1	0	S4	S2	...
Arrival Time	07:51	08:00	08:09	09:13	10:22	10:31	10:40	10:52	11:01	12:05	12:17	12:30	12:39	...
Departure Time	07:56	08:05	09:09	10:14	10:28	10:37	10:45	10:58	12:01	12:11	12:22	12:34	12:44	...
Serviced SPUs **	/	/	{15, 44}	{33}	/	/	/	/	{42}	/	/	/	/	...
Bike Tour with Serviced AHDs **	/	{S3-5-13-18-S3}	{S1-9-49-S1}	/	/	{S3-41-14-19-S3}	{S1-27-8-S1}	/	{S3-20-43-45-S3}	{S1-25-24-S1}	/	{S4-16-35-31-S4}	{S2-37-36-S2}	...
Bike Tour Time (Departure - Arrival)	/	08:05 - 08:30	08:14 - 08:33	/	/	10:37 - 11:09	10:45 - 11:07	/	11:15 - 11:44	12:11 - 12:30	/	12:34 - 13:06	12:44 - 13:06	...

Tour of Van #1 (continuation) *	...	0	S4	S2	0	S1	S2	S3	S1	0	S2	0
Arrival Time	...	12:52	13:04	13:15	13:30	13:42	13:50	14:00	15:04	16:12	16:25	17:10
Departure Time	...	12:58	13:12	13:20	13:34	13:46	13:56	15:00	16:04	16:17	17:02	17:15
Serviced SPUs **	...	/	/	/	/	/	/	{38, 48}	{26, 28}	/	/	/
Bike Tour with Serviced AHDs **	...	/	{S4-46-40-S4}	{S2-32-29-4-S2}	/	{S1-47-1-7-S1}	{S2-12-23-S2}	/	/	/	{S2-3-39-17-S2-22-30-S2}	/
Bike Tour Time (Departure - Arrival)	...	/	13:12 - 13:36	13:20 - 13:50	/	13:46 - 14:13	13:56 - 14:16	/	/	/	16:30 - 17:21	/

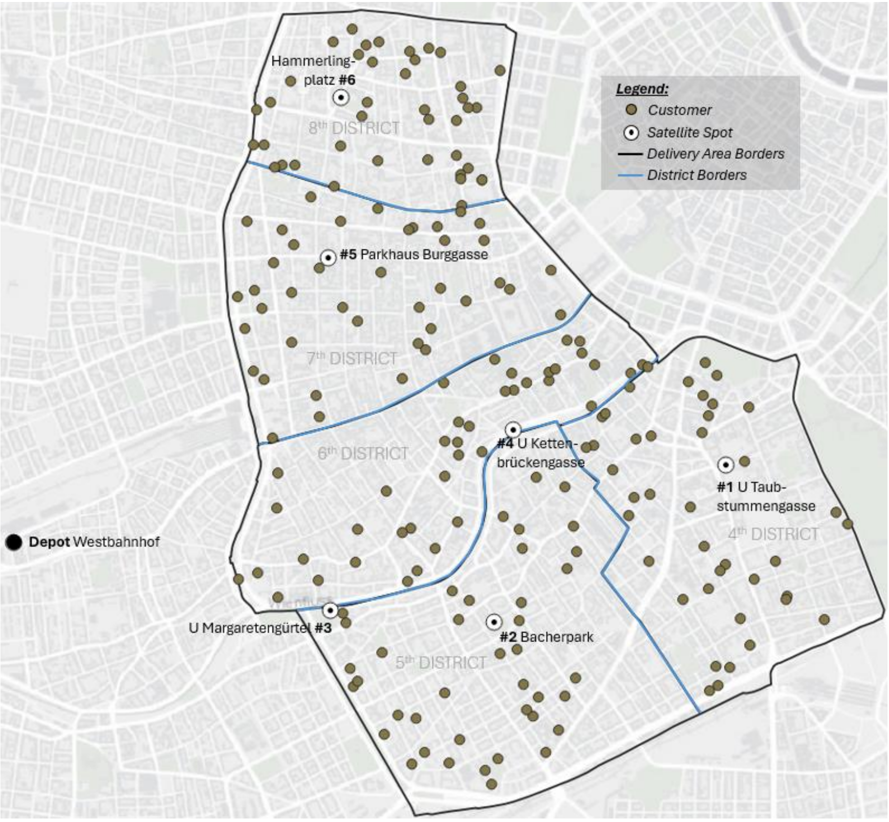
Tour of Van #2 *	0	S4	0	S2	S4	0
Arrival Time	07:48	08:00	09:08	14:01	15:05	16:12
Departure Time	07:53	09:00	13:52	15:01	16:06	16:17
Serviced SPUs **	/	{6, 10, 11, 34}	/	{2, 21}	{50}	/
Bike Tour with Serviced AHDs **	/	/	/	/	/	/
Bike Tour Time (Departure - Arrival)	/	/	/	/	/	/

** 0 means visited depot, Sx visited Satellite*

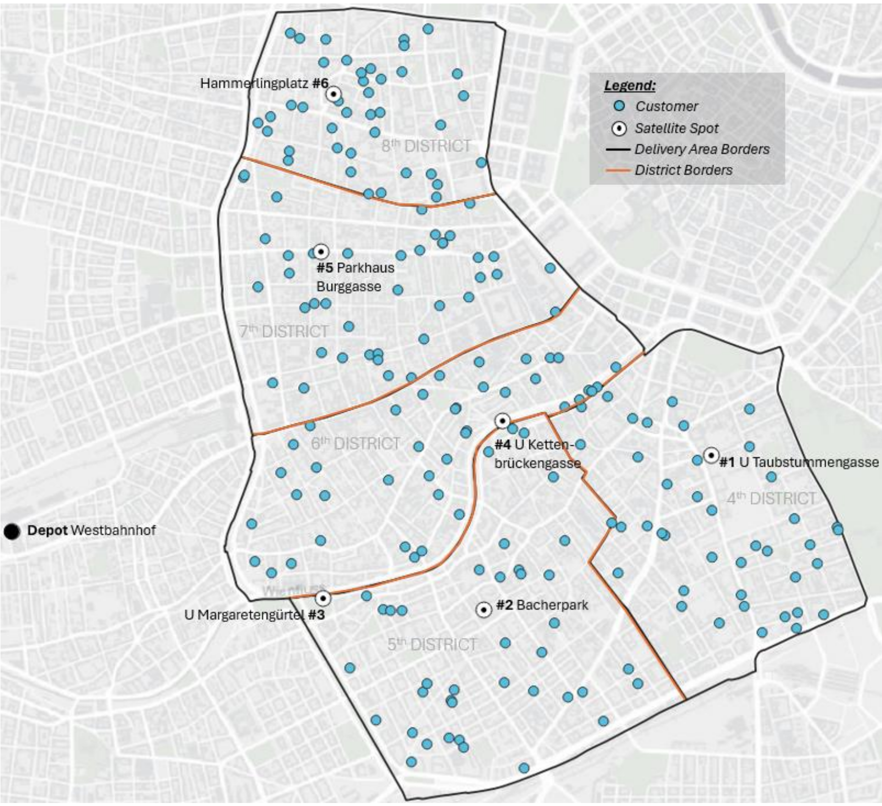
*** numbers 1 to 50 represent customers*

Appendix B: Visualisation of Vienna Instances #2 to #5 (Tuesday to Friday)

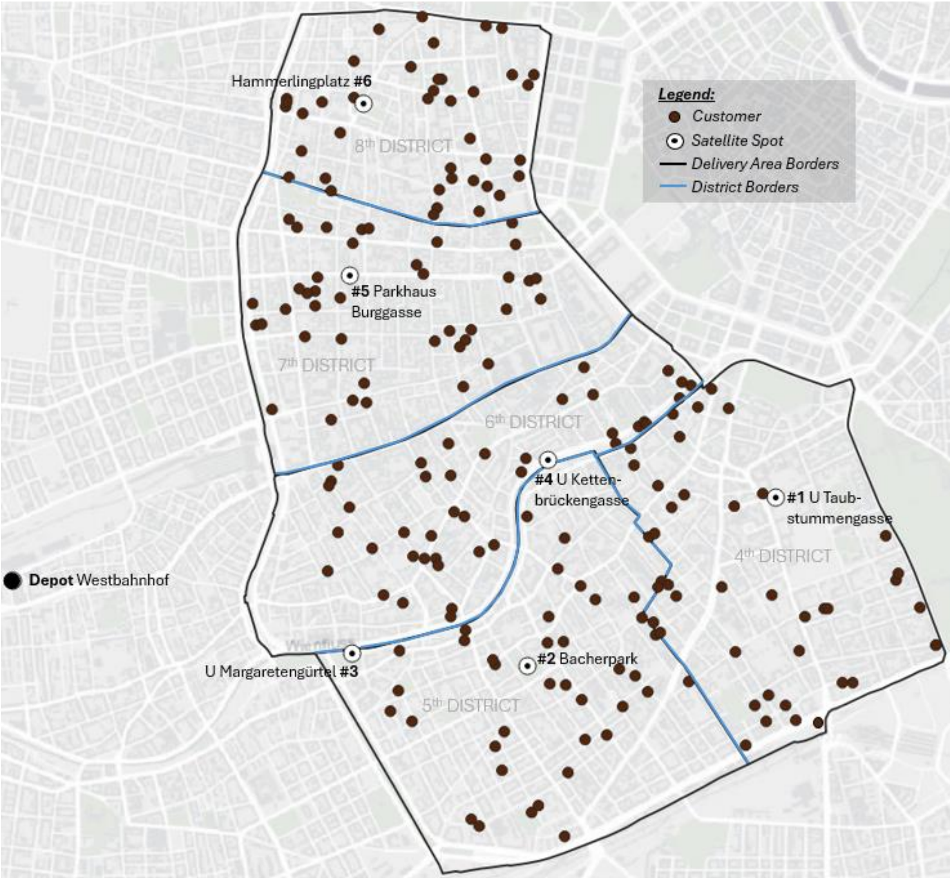
Vienna Instance #2 (Tuesday)



Vienna Instance #3 (Wednesday)



Vienna Instance #4 (Thursday)



Vienna Instance #5 (Friday)