TECHNISCHE UNIVERSITÄT MÜNCHEN
Integrative Research Center Campus Straubing für Biotechnologie und Nachhaltigkeit

Operations Research Methods for the Sustainable Use of Autonomous Delivery Robots in Retail

Andreas Heimfarth

Vollständiger Abdruck der von der promotionsführenden Einrichtung Campus Straubing für Biotechnologie und Nachhaltigkeit der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Wirtschaftswissenschaften (Dr. rer. pol.)

genehmigten Dissertation.

Voritzender:	Prof. Dr.-Ing. Jakob Burger
Prüfer der Dissertation:	1. Prof. Dr. Alexander Hübner
	2. Prof. Dr. Heinrich Kuhn
	3. Prof. Dr. Stefan Minner

Die Dissertation wurde am 14.04.2021 bei der Technischen Universität München eingereicht und von der promotionsführenden Einrichtung Campus Straubing für Biotechnologie und Nachhaltigkeit am 14.07.2021 angenommen.

Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

ALNS	Adaptive Large Neighborhood Search
DC	Distribution center (warehouse)
GVNS	General Variable Neighborhood Search
LB	Lower bound for objective value obtained from MIP solver
LS	Local Search
MIP	Mixed Integer Program
MIP&GVNS	Solution approach based on an MIP for clustering and GVNS for routing of each cluster
MTR	Mixed Truck and Robot Delivery
MTR-RP	Mixed Truck and Robot Routing Problem
MVTR-RP	Multi-Vehicle Truck-and-Robot Routing Problem
OR	Operations Research
PR	Priority rule

SINS	Set Improvement Neighborhood Search
STR	Separate Truck and Robot Tours
SVTR-RP	Single-Vehicle Truck-and-Robot Routing Problem
TD	Truck-only delivery
TnR	Truck-and-Robot
TRC	Truck-and-Robot Cost-optimal Routing approach
TRC-RSH	Truck-and-Robot Cost-optimal Routing approach based on robot scheduling heuristic
TRCR	Truck-and-Robot Clustering and Routing approach
TRL	Truck-and-Robot Lateness-optimal Routing approach
TSP	Traveling Salesman Problem
VND	Variable Neighborhood Descent
VNS	Variable Neighborhood Search
VRP	Vehicle Routing Problem

1 Introduction

This thesis deals with sustainable last-mile delivery using truck-and-robot systems. It presents approaches to planning delivery tours, e.g. for parcels shipped to consumers' homes. This chapter provides an introduction to the methodology applied, namely Operations Research (Section 1.1), and the application field of last-mile delivery (Section 1.2), focusing on the truck-and-robot concept.

The remainder of this thesis is structured as follows. Chapter 2 describes the specific scope and key findings of the three contributions presented. For each of them, it summarizes the authors, journal and current status of its publication. Chapters 3 to 5 consist of one of these articles each. The final Chapter 6 summarizes the findings and highlights opportunities for future research.

1.1 Methodology

How can business decisions be made efficiently? This is the fundamental question Operations Research (OR) tries to answer by modelling, analyzing and solving practical problems. It can be applied to a wide variety of decisions such as what to produce when on a machine or (in the case considered here) in which order to visit given customers to deliver parcels. What these problems have in common is an objective that should be minimized or maximized, such as costs, process time or profit. Further, there are dependencies that describe the underlying physical process. As an example, a truck requires a minimum time to travel between visiting two customers. The decision to be made must comply with all these constraints and should lead to a minimum (or maximum) objective value.

Mathematical models A typical problem for retail and logistics companies is the distribution and transportation of goods from a warehouse to its customers. As an example, consider the task of delivering parcels to a set of n customers by truck on the shortest-possible route. The decision to be made is the order in which the customers are visited. This is a classical OR problem in last-mile delivery, called the Travelling Salesman Problem (TSP). For five customers, there are already $5! = 120$ possible tours. For ten customers, this number becomes 3.6 million. It is therefore impossible to assess all individual possibilities and pick the shortest route. We can model the problem mathematically and apply algorithms to solve the problem, which exploit certain problem characteristics to reduce the search space. To model the TSP, we define the set L of all customer locations and the set L^0, which additionally contains the warehouse location (i.e., the start and end point of the truck tour). We then use the distance $d_{i,j}$ as input parameter and define a decision variable $s_{i,j}$ for every pair of locations i and j (with $i, j \in L^0$). The variable $s_{i,j}$ is defined as 1, if the truck travels from location i to location j, and 0 otherwise. This means our objective, the total distance to be minimized, is the sum of the distances $d_{i,j}$ travelled. This is expressed by the objective function (1.1). We must choose all $s_{i,j}$ as 1 or 0 to define a tour. This is defined in Constraints (1.2). Since every customer must be reached and the truck must return to the starting point, we demand that the truck arrive at each point and leave each point i exactly once, as expressed in Constraints (1.3) and (1.4). Next, we must enforce flow constraints, that is, ensure that the truck can only start from locations after it arrived there. This eliminates so-called sub-tours, for which an example is presented in Figure 1.1. First, we define an auxiliary decision variable z_i for every location in Equations (1.5). This variable indicates the order of the stops such that a higher z_i corresponds to a later position on the tour. It is not needed to define a tour (the variables $s_{i,j}$ are sufficient for this), but it allows enforcing practical constraints. Constraints (1.6), enforce that if the truck travels from i to j, then $z_j \geq z_i + 1$. This leaves only the warehouse location as the starting point for a circle tour and prevents sub-tours as shown in Figure 1.1, since a feasible set of values for z_i does not exist for the customers on the sub-tour. Toth and Vigo (2001) provide a more detailed review of vehicle routing problems.

$$\min \sum_{i,j \in L^0} d_{i,j} s_{i,j} \tag{1.1}$$

subject to

$$s_{i,j} \in \{0,1\} \qquad \forall i,j \in L^0 \tag{1.2}$$
$$\sum_{j \in L^0} s_{j,i} = 1 \qquad \forall i \in L^0 \tag{1.3}$$
$$\sum_{j \in L^0} s_{i,j} = 1 \qquad \forall i \in L^0 \tag{1.4}$$
$$z_i \geq 0 \qquad \forall i \in L^0 \tag{1.5}$$
$$z_i - z_j + n s_{i,j} \leq n - 1 \qquad \forall i \in L^0, j \in L \tag{1.6}$$

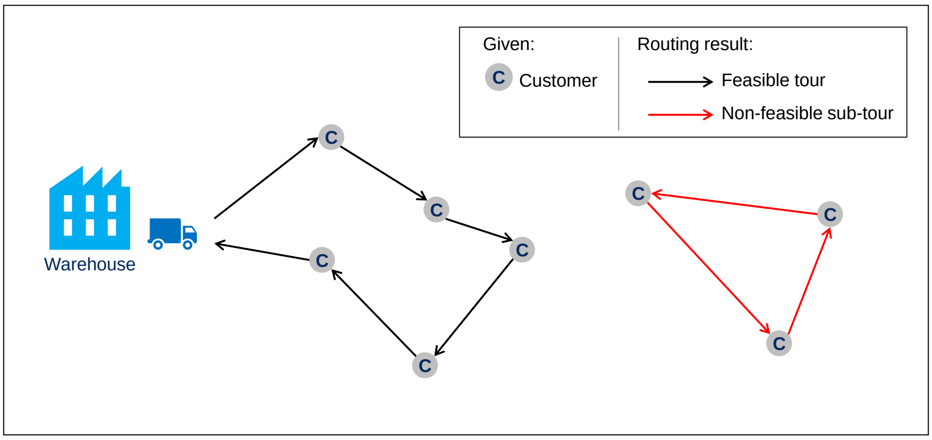

Figure 1.1: Example of a non-feasible TSP solution with a sub-tour

For such models, there are exact methods to find the optimal solution, i.e. the decision that leads to the minimum (or maximum) objective value. There are commercial solvers available (e.g., CPLEX, Gurobi or LINDO) that rely on the mathematical formulation and some advanced exact procedures (e.g., branch&bound, branch&price). However, for large problems these exact methods can take a long time to find the optimal solution or to confirm that a given solution is optimal. The presented TSP model has already 110 decision variables in the case of nine customers and one warehouse (10×10 $s_{i,j}$ and 10 z_i). In practical settings, problems often involve more locations and additional constraints and variables. As an example, arrival at each customer can be required within a specific time window or goods could first be picked up at one location before being delivered to another. As often with operations problems, the TSP model is a Mixed Integer Program (MIP). This means its solution is given by a tuple of numbers, of which some (in our case $s_{i,j}$) must be integers. This type of problem creates particularly high computational effort, since a solver must often try many combinations of such integer values.

Heuristics For this reason, alternative methods are developed, which make use of problem-specific knowledge to generate a good (not always optimal) solution. Such a method is called heuristic. An example for problem-specific knowledge of the TSP is that the shortest tour does not cross itself, as the removal of a crossing reduces the total distance. A heuristic builds and tests solutions based on a certain strategy. For instance, it could sequentially append the closest next location to the truck tour until all locations have been visited. A more sophisticated heuristic could then try to improve the obtained tour by swapping two random locations and testing if this reduces the distance. Since heuristics cannot prove an optimum, they typically abort the search after a certain number of iterations of when a certain solution quality is reached. The result is then the best-known solution. Our contributions in Chapters 3 to 5 develop such tailored heuristics for problems a standard MIP solver could not solve in reasonable time. As an example, when a parcel delivery tour to different customers is planned in practice, the solver must find a solution before the vehicle can start the tour.

1.2 Last-mile delivery

The application considered in this thesis is in the field of last-mile delivery, i.e., the task of bringing orders from a distribution center to a customer‘s door. This section provides background information on last-mile delivery in general and on the concept studied in this thesis, which is based on robot delivery.

1.2.1 Relevance and trends

Last-mile delivery is becoming increasingly important due to growing volumes of online orders. At the same time it creates problems in cities, as the classical delivery by truck causes emissions and traffic congestion. Together with a global trend towards urbanization, this puts cities under pressure to prevent traffic systems from collapsing and avoid harmful levels of pollution (World Economic Forum, 2020). For retailers, last-mile delivery is an important cost driver and a way to differentiate their offering (e.g., same-day delivery). Therefore, both cities and retailers (with their logistics providers) have a strong motivation to find new delivery concepts that reduce costs and environmental impact. (Hübner et al., 2016b; McKinsey & Company, 2016)

1.2.2 Related last-mile delivery concepts

Truck delivery Traditionally, last-mile deliveries are performed by a truck and a driver, who visits every customer individually to hand over the order. When operating such a delivery truck, the sequence in which customers are visited must be defined (TSP). When several trucks are available, additional decisions on which customer to serve by which vehicle must be made. This problem is called the Vehicle Routing Problem (VRP). An exemplary routing result is shown in Figure 1.2.

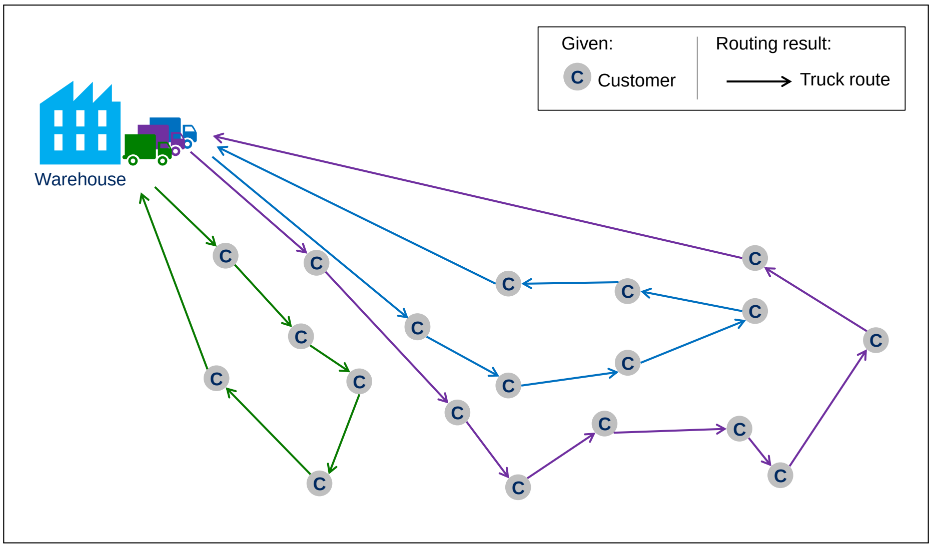

Figure 1.2: Exemplary VRP result

Possible objectives for minimization are the total distance travelled or the time until the last vehicle returns. The basic version of the VRP can be solved in an exact manner for relevant numbers of customers. However, there are many variants characterized by additional constraints or alternative objectives which make the problem very complex. E.g., customers could require their delivery within an individual time window, vehicles could have limited capacities or require different travel and processing times. Other variants consider energy consumption and charging of electric vehicles. There exists a large body of literature on such variants of the VRP and tailored solution approaches typically relying on heuristics (Konstantakopoulos et al., 2020).

2-Echelon networks In recent years, several concepts have been proposed to reduce the environmental impact of last-mile delivery, particularly in large cities. Many of them rely on small zero-emission vehicles such as cargo bikes, aerial drones or electric vans for the delivery to customers. Therefore, several locations are needed, in which the orders arriving from outside the city by truck are transshipped to the smaller vehicles, since these have a limited range and speed. For a given set of customers and potential transshipment points, the operator must find (i) truck routes from the warehouse to transshipment points and (ii) routes for the smaller vehicles from these points to the customers (see Figure 1.3). Decision problems of this kind are called 2-Echelon Vehicle Routing Problems. Based on the vehicle types, customer requirements and type of transshipment facilities, different variants of the problem exist (Mühlbauer and Fontaine, 2021).

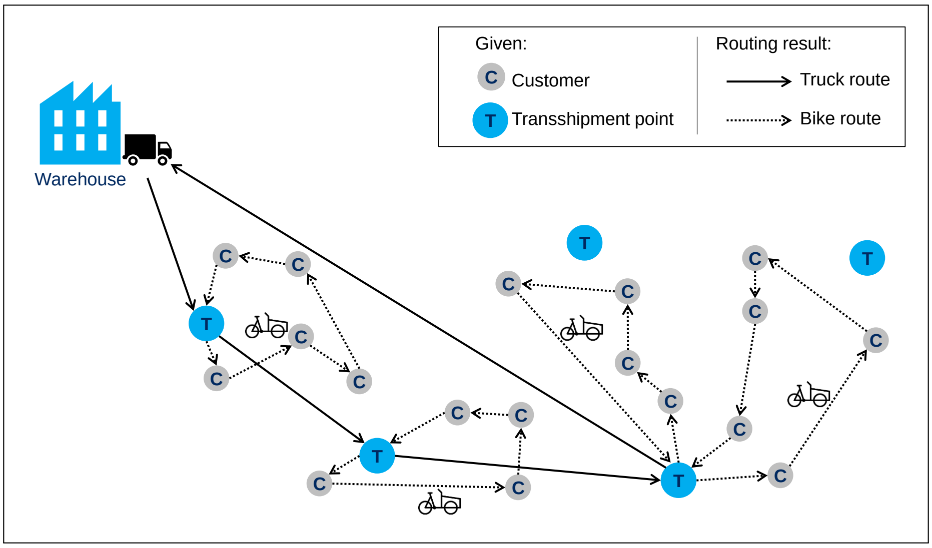

Figure 1.3: Exemplary 2-Echelon VRP result

Truck-and-drone delivery Aerial drones can operate in conjunction with trucks to make deliveries. Besides the classical 2-Echelon approach described above, a drone can be loaded onto a truck and launch while the truck visits a customer. It then makes one delivery (since it cannot carry more parcels) and meets the truck again at one of its later stops. The resulting route is depicted in Figure 1.4. The advantage of this concept is that the truck distance is reduced without the need for dedicated transshipment infrastructure (Murray and Chu, 2015; Otto et al., 2018).

1.2.3 Robot delivery

Several new means of delivery have emerged in recent years, enabled by technological advancements in autonomous driving, electrical propulsion and IT platforms. The

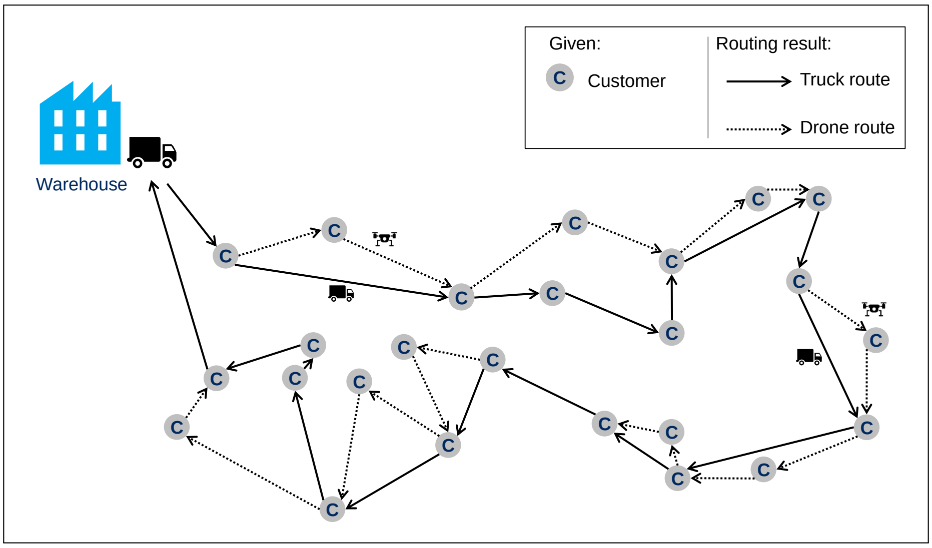

Figure 1.4: Exemplary truck-and-drone route

delivery robots considered in this thesis move autonomously on sidewalks at low speed, transporting a single parcel to a customer, who can retrieve it from the robot‘s freight compartment. Since the robots are battery powered, they have the potential to reduce both emissions and road traffic. Furthermore, several independent robots add flexibility to a logistics system and can enable it to meet delivery time windows or respond to fluctuating demand. Further details on robot technology are provided in chapters 3 to 5.

To ensure fast delivery across larger distances, the robots can be transported by a truck and dropped off close to customers. This compensates for the low speed of the robots, while the truck mileage compared to regular truck deliveries is reduced and the driver‘s productivity increased, because the truck does not visit every customer individually. In the concept considered, the truck does not wait for robots to return, but picks up new robots waiting at charging stations in the customer area. Once a robot has made its delivery, it returns to the closest charging station, also called robot depot, and waits for its next use (Boysen et al., 2018b). The resulting tour is depicted in Figure 1.5 (robot returns are not shown, as they are not part of the decision problem). The key advantages of this concept compared to truck-and-drone are a further reduction of truck distance and the higher robustness (paired with lower costs) of ground vehicles vs. aerial drones.

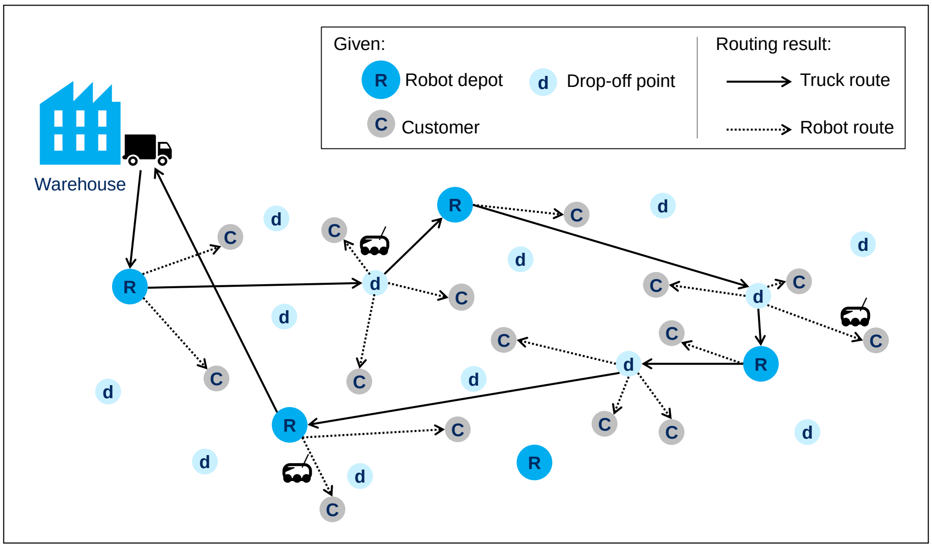

Figure 1.5: Exemplary truck-and-robot route

This leads to the decision problem of where to go by truck to drop off robots and from which of these stops to start each customer‘s robot delivery. Note that this has some similarities with the VRP described in the previous section, since a set of customers must be visited. The challenge in this case is that two different vehicle types must cooperate to serve all customers and to do so, their movements must be synchronized.

This thesis analyzes the truck-and-robot concept as a promising approach to last-mile delivery. It proposes heuristics for the task of finding routes that minimize costs and emissions. The routes resulting from these heuristics in various scenarios are then evaluated to derive practical insights on when a truck-and-robot system is cost-competitive and how it should be operated. This includes questions such as the following:

- **How can we find the best tour for a given set of customers?** Strategies and rules to construct and improve solutions are needed for this.

- **How do the costs of truck-and-robot delivery compare to classical truck delivery?** This requires modelling the delivery process in a way that captures all relevant cost drivers.
- **Which benefits does truck-and-robot delivery bring for the environment and urban traffic?** This requires tracking the truck mileage of the different delivery modes.
- **How do we ensure the truck has sufficient robots on board to supply all customers?** This involves tracking the number of robots aboard the truck as it picks up and releases robots along its route.
- **How many trucks are needed to serve a certain area?** To answer this question, we must decide on how customers are distributed to several available vehicles.
- **How can parcels that do not fit into the robot compartment be handled?** This would require the possibility to include deliveries by the driver in the truck-and-robot tour.

2 Contributions

This chapter introduces three articles (chapters 3-5) written in conjunction with the doctoral thesis. It is intended as a guide to the topics and motivation for the research questions. An overview on the articles and the status of their publication process is given in Table 2.1.

	Contribution	**Co-authors**	**Status**
1	Cost-optimal Truck-and-Robot Routing for Last-Mile Delivery	Ostermeier, Manuel; Hübner, Alexander	Accepted for publication in Networks on January 19, 2021
2	A mixed truck and robot delivery approach for the daily supply of customers	Ostermeier, Manuel; Hübner, Alexander	Submitted to the European Journal of Operational Research (under review)
3	The multi-vehicle truck-and-robot routing problem for last-mile delivery	Ostermeier, Manuel; Hübner, Alexander	Submitted to Transportation Science (under review)

Table 2.1: Status of publication

Methodology Each article presents a state-of-the art routing approach based on OR methodology. The overarching problem considered is how to route trucks and robots for attended home deliveries within time windows at minimal logistics costs. This problem (with its specific extensions and assumptions in each article) is first described qualitatively and formulated as a mathematical model, which represents the necessary aspects of the real problem at hand. Since the models cannot be solved in an exact manner with standard solvers due to their numerical complexity, we then propose a heuristic solution approach tailored to the problem specifics. This involves an innovative solution framework for each problem to consider its variables, parameters, objective function and constraints. The approach is described in detail and the rationale behind each of its components is given. It is then applied to solve test instances generated from real world location coordinates. We compare the heuristic developed to existing approaches to demonstrate its advantages in solution quality and run-time. In the next step, managerial implications are analyzed. This includes quantifying the reductions in costs and emissions achieved by the truck-and-robot concept and assessing the influence of problem parameters such as customer density and fleet size on the results. Each article concludes with a summary of key insights, their practical implications and an outlook on potential future research topics in the field.

Remark The final articles published may differ from the versions presented here due to changes made in the peer review process after this thesis was submitted. The fundamental content and findings remain the same.

2.1 Contribution 1: Cost-optimal truck-and-robot routing for last-mile delivery

In this article, we first provide an overview on existing truck-and-robot literature. The key publications in this field are by Boysen et al. (2018b) and Alfandari et al. (2019), who focus on demonstrating high logistical performance for attended home deliveries with deadlines. They make some simplifying assumptions and do not provide a cost comparison to existing delivery concepts. We therefore introduce a new decision problem, which extends existing work on truck-and-robot routing by several practically relevant aspects.

First, delivery time windows are assumed, since the customer must be at home to retrieve the parcel from the robot. Therefore, instead of only considering a deadline for each delivery as in Boysen et al. (2018b), we also impose an earliest delivery time for each customer, assuming they are not at home before that time.

Second, instead of focusing on the number of late deliveries, we formulate an objective function which estimates the total last-mile delivery costs, including labor, vehicle amortization, energy consumption and delay duration. The individual cost factors are obtained empirically. This is crucial for the implementation of truck-and-robot in practice, as companies need to operate the system at minimal costs and they require a cost comparison to their existing delivery concept before investing into robots.

Third, a limited number of available robots is considered. This is a key prerequisite for the practical application of our routing approach, since the resulting tour cannot use more robots than currently at hand.

In total, these enhancements lead to a more complex optimization problem compared to prior research, with additional decision variables, constraints and cost terms. To handle this complexity and find efficient routes in reasonable time, we propose a tailored local search (LS) heuristic. It relies on problem-specific operators, which try to improve a given solution by iteratively making changes that have shown to often lead to improvements. Our numerical experiments show that the LS is competitive in terms of run-time and solution quality. Furthermore, it leads to a cost reduction of 46% compared to to prior routing approaches. Overall, the truck-and-robot concept shows savings of 68% compared to truck delivery in our case study.

2.2 Contribution 2: A mixed truck and robot delivery approach

Based on the promising cost reduction potential identified in Contribution 1, the question arises if a truck-and-robot system can completely replace normal delivery trucks. While we assumed all deliveries are made by robot in our previous work, some goods require personal delivery in practice. This can include bulky items that do not fit into the robots, hazardous substances and valuables. In Contribution 2, we mathematically formulate this problem and propose a solution approach based on General Variable Neighborhood Search (GVNS) which can include these deliveries by the truck driver into the truck-and-robot tour such that delivery time windows are met.

Our performance comparison (on instances with robot deliveries only) shows that the GVNS approach is up to 94% faster than the LS proposed in Contribution 1, at the cost of up to 2% higher objective values. When applied to instances with truck deliveries required (which the LS approach cannot solve), the GVNS run-time increases, but stays at an acceptable level of roughly 30 minutes for 50 customers.

We then analyze the implications of such a "mixed" tour containing deliveries by truck and by robot. In our case study, it still leads to a 43% cost reduction compared to normal truck delivery. The cost advantage of one integrated tour vs. two separate tours for truck and robot delivery is 22%. The following sensitivity analyses identify the length of the time windows for truck delivery as a key cost driver.

2.3 Contribution 3: Vehicle assignment for truck-and-robot deliveries

After proposing solutions for operating a single truck with robots, one important problem remains when the system is scaled up to several trucks. This is necessary as soon as the customers cannot be served by one truck during a single shift, the truck‘s parcel capacity is exceeded or delivery time windows are too tight. In the case of several trucks, the additional decision which customers to serve by which truck must be made. Since this customer assignment is highly interdependent with the routing decisions of each truck, these problems cannot be treated separately.

As already routing one truck required the application of tailored heuristics, so does the more complex, integrated problem. We mathematically formulate the problem and propose an approach based on a heuristic to generate large pools of potential truck tours and an MIP to choose tours from the pool and assign the customers to the stops of these tours.

Within the numerical evaluation, this approach is 23 - 60% faster and yields 18 - 24% better solutions than the benchmark relying on the best customer assignment for a VRP and the GVNS from Contribution 2 for truck-and-robot routing of each individual truck. Further analyses confirm a large cost reduction potential compared to truck delivery (62%) and identify the number of trucks and time window offering as key cost drivers.

3 Cost-optimal Truck-and-Robot Routing for Last-Mile Delivery

Co-autors: Manuel Ostermeier, Alexander Hübner
Accepted in *Networks* on January 19, 2021
(in press, DOI: 10.1002/net.22030)

Abstract During recent years, several companies have introduced small autonomous delivery robots and evidenced their technical applicability in field studies. However, a holistic planning framework for routing and utilizing these robots is still lacking. Current literature focuses mainly on logistical performance of delivery using autonomous robots, ignoring real world limitations, and does not assess the respective impact on total delivery costs. In contrast, this paper presents an approach to cost-optimal routing of a truck-and-robot system for last-mile deliveries with time windows, showing how to minimize the total costs of a delivery tour for a given number of available robots. Our solution algorithm is based on a combination of a neighborhood search with cost-specific priority rules and search operators for the truck routing, while we provide and evaluate two alternatives to solve the robot scheduling subproblem: an exact and a heuristic approach. We show in numerical experiments that our approach is able to reduce last-mile delivery costs significantly. Within a case study, the truck-and-robot concept reduces last-mile costs by up to 68% compared to truck-only delivery. Finally, we apply sensitivity analyses to provide managerial guidance on when truck-and-robot deliveries can efficiently be used in the delivery industry.

3.1 Introduction

Last-mile delivery describes the final step in the retail supply chain, i.e., actual delivery to the customer. It is a key challenge for retailers and logistics service providers (Hübner et al., 2016b) and is responsible for a large share of logistics costs, often above 50% (see e.g., Glatzel et al. (2012), Kuhn and Sternbeck (2013) or McKinsey & Company (2016)). The last-mile service in urban areas is forecast to grow by 78% by 2030 (World Economic Forum, 2020), particularly driven by the growth of online shopping and home delivery (Wollenburg et al., 2018; Ishfaq et al., 2016; Allen et al., 2018). Furthermore, the share of the world population living in urban areas will grow from 55% in 2018 to 68% by 2030, and with that the volume of freight transportation, emissions and congestion within these areas (United Nations, 2018). Home deliveries are traditionally performed by diesel trucks, which have to visit each customer individually to hand over ordered goods, and thus intensify traffic and pollution problems. Moreover, failed deliveries (i.e., in cases where customers are not at home to receive a delivery) lead to additional customer visits and further increase traffic. Home deliveries usually require customer attendance. To ensure that the customer is at home, time windows are applied in practice (see e.g., Hübner et al. (2016b)). This helps to avoid failed deliveries by arranging fixed service time windows a customer can select when ordering. In this way the delivery is scheduled at a time in which the customer is expected to be at home and can receive the order. The application of scheduled deliveries is steadily increasing for attended home deliveries (see e.g., Agatz et al. (2011), Ulmer (2017), Klein et al. (2019), Köhler et al. (2020) and World Economic Forum (2020)). However, the application of time windows reduces the efficiency of truck tours. Innovative last-mile solutions for retail fulfillment are required to address these challenges (Agatz et al., 2008; Orenstein et al., 2019; Hübner et al., 2019).

In recent years, many new technological concepts have emerged for last-mile deliveries. Delivery with autonomous robots in cities has already been realized (see Starship (2019)). The small robots, e.g., developed by Starship (2019), Marble (2019) and several others, transport an order (e.g., parcel or a grocery basket) to a single customer within an agreed time window. Robots navigate on sidewalks at pedestrian speed and can safely move in autonomous mode most of the time or switch to remote control in the event of problems. Once the robot arrives at the door, the customer is notified and can then unlock the freight compartment to retrieve the order. A further innovation, which received much attention in recent literature, is drone-based delivery (Otto et al., 2018). A well-studied concept in this field is the support of truck deliveries by an aerial drone. A fast-moving drone starts from and returns to the truck to perform individual deliveries on the route. However, robot delivery shows several advantages compared to drones for last-mile delivery in urban areas. First, air traffic is much more restricted and needs to be regulated. While robots can move anywhere on walkways and park safely on sidewalks, drones need to find safe drop-off and landing spots in an area with limited space. Second, robots are less sensitive to weather conditions and weight restrictions. Furthermore, robots move silently on pedestrian walks, and do not interfere with the privacy of residents. Finally, ground vehicles are more energy efficient than drones and also cheaper to produce, since less motor power is required and lightweight solutions are not needed. For large-scale application, the challenge is to benefit from the advantages of robots on the last-mile, but also compensate for their relatively low travel speed. One way to do this would be to create a dense warehouse infrastructure to reduce travel distances, but this involves high investments and is barely feasible in urban areas. The alternative is to combine several robots with faster means of transportation, such as trucks. The truck stops at drop-off points and releases the robots for delivery (see Figure 3.1). This concept is called truck-and-robot delivery.

Figure 3.1: Specialized truck launching robots at the kerbside (Mercedes-Benz Vans, 2016)

Solving the resulting routing problem requires deciding on the truck route (i.e., a sequence of possible robot drop-off locations), and the assignment of each customer order to a location along the truck route, from which a robot is sent out to perform the actual delivery. The robots return to a robot depot after each delivery and from here they will be picked up for later tours. Due to the problem complexity in truck routing and robot scheduling, heuristics are needed to solve instances of relevant size. Current literature on truck-and-robot routing is focused on demonstrating logistical performance (e.g., with respect to late deliveries) for small problem sizes and some stylized assumptions. There is not yet any holistic approach to comprehensively evaluate delivery costs, robot availability, and meet time windows for attended home delivery. We propose an extended truck-and-robot routing problem that (i) is based on decision-relevant costs (for all aspects of truck and robot use), (ii) takes into account restrictions for the use of this system in practice, and (iii) considers trade-off decisions between cost and service quality. We formulate the complete problem as Mixed Integer Program (MIP) and solve it by decomposition into truck routing and robot scheduling. Our solution approach is based on cost-specific priority rules and search operators for the truck routing, while we provide and evaluate two alternatives to solve the robot scheduling subproblem: an exact and a heuristic approach. This allows us to incorporate all problem extensions discussed and keep the computational effort at a minimum.

This paper is organized as follows. Section 3.2 provides an overview of related literature and highlights the differences to existing concepts. Section 3.3 describes the problem of a truck-and-robot system, detailing the concept with its costs and constraints. It then derives the mathematical problem formulation. Section 3.4 details our solution approach. In Section 3.5 we analyze the numerical performance of our approach and provide managerial insights. The final Section 3.6 summarizes our findings.

3.2 Review of related literature

Our problem is related to a growing body of literature on city logistics. This includes not only delivery robots but also drones, autonomous cars and parcel lockers. This section first discusses publications on related delivery concepts that have analogies to robot systems, before detailing the existing literature on robot delivery. It concludes by identifying the gap in research.

3.2.1 Related last-mile delivery concepts

Drone-based delivery Among different autonomous vehicles, aerial drones have received most attention in literature. In a first problem variant, drones start and end their tour at the distribution center (DC). This presumes that the DC lies in customer proximity (e.g., see Murray and Chu (2015), Ulmer and Thomas (2018) and Coelho et al. (2017)). In a second variant a drone starts from a truck visiting customers and picking up the drone at later customer stops during the tour, e.g., Murray and Chu (2015), Yoo and Chankov (2018) or Bouman et al. (2018)). The challenge is to find truck and drone routes that are cost-efficient, guarantee timely delivery and account for the drones' maximum range. Existing approaches rely on a solution of the corresponding travelling salesman problem (TSP, i.e., determining a truck tour to visit all customers) that is then improved iteratively by reassigning customers to the drone. A possible resulting tour is shown in Figure 3.2.

Recent publications allow the use of several drones combined with one truck. Murray and Raj (2020), Phan et al. (2018) and Moshref-Javadi et al. (2020a) propose a concept with several drones started from a truck at a customer location. Again the drones meet the truck at one of its later stops. The authors consider up to four drones, which implies at least 20% truck deliveries (as every drone starts from a customer visited by truck). Murray and Raj (2020) note that more drones per truck would reduce the system‘s efficiency, since drones must keep some distance to other drones and thus take off and land sequentially. The solution approaches therefore rely on solving a TSP for a given set of customers visited by the truck. Chang and

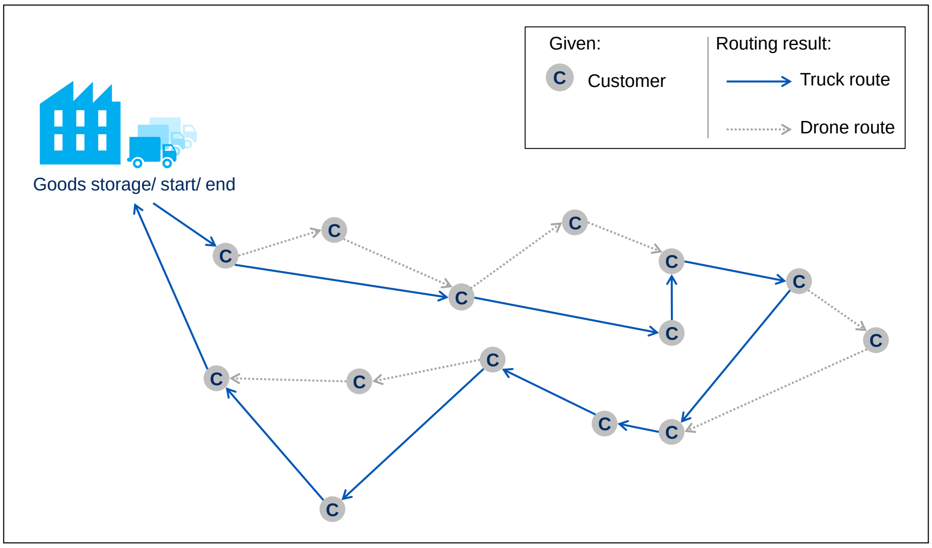

Figure 3.2: Illustrative truck-and-drone tour

Lee (2018), Moshref-Javadi et al. (2020b) and Salama and Srinivas (2020) consider a truck with up to ten drones aboard visiting customer clusters. At the waiting point of each cluster, it launches drones for deliveries that return to the waiting truck. The clusters, their waiting points and the truck tour through these waiting points are subject to optimization. While this concept can reduce truck mileage significantly, it still relies on a TSP solution approach for truck routing, i.e. the visited stops are given, and cannot handle time windows. Kim and Moon (2019) study a hub concept in which a truck leaves parcels at a storage hub, from where drones deliver them. This has the potential to reduce truck mileage and eliminate the need to synchronize the truck and the drone movements, but it requires infrastructure investments into hubs. In summary, a major difference of the truck-and-robot concept is the number of customers served by drone/ robot (see Figures 3.2 and 3.3). The truck routes therefore differ significantly from a TSP for customer visits, as individual robot depots and drop-off points may or may not be visited by the truck. The number of robots launched is higher than the number of drones considered in truck-and-drone literature. On the other hand, the truck does not have to be synchronized with returning robots, which adds flexibility to meet time windows (not considered by

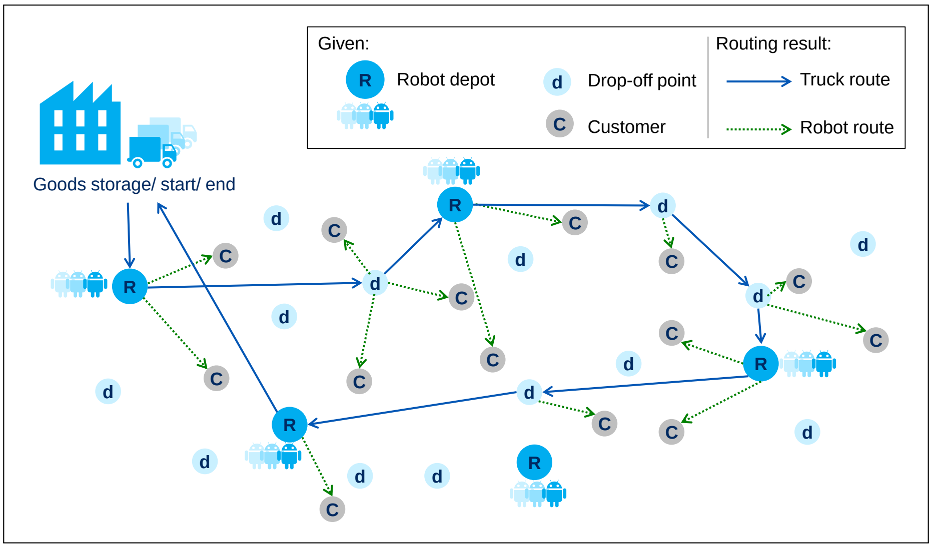

Figure 3.3: Illustrative truck-and-robot tour

the mentioned drone publications) and achieve large reductions in truck mileage. For a more extensive overview on truck-and-drone routing we refer to Macrina et al. (2020), Boysen et al. (2021) and Rojas Viloria et al. (2020).

As highlighted by Otto et al. (2018) drones have some limitations in big cities, such as the capacity of the air space, availability of safe landing or drop-off space, noise regulation, and safety (e.g., flying above busy pedestrian zones) as well as privacy concerns (e.g., flying near homes while filming). Consequently, companies are now piloting this approach in rural areas rather than in cities (The Guardian, 2019), where drones can leverage their advantage of higher speed. The delivery robots used in our application have several advantages for the use in cities. They are quiet, inherently safe due to the low speed, robust against weather conditions and vandalism, and they can park without consuming electrical energy. Due to the different dynamics of driving vs. flying, robots typically have a longer range, higher payload, and cheaper sensors and batteries. However, as they are slower than drones, existing concepts for drones cannot be applied directly to robots due to the underlying distribution process (see Section 3.3.2). The disadvantage of lower speed of robots is eliminated by the combination with trucks and depots. We further refer to Otto et al. (2018) for a more detailed literature overview on drone delivery, the limitations of this approach and obstacles to overcome. To summarize, truck-and-drone concepts have several limitations in cities and their routing approaches cannot be applied directly for truck-and-robot routing. Robots have advantages in cities, while drones may be more beneficial for last-mile deliveries in rural areas.

Autonomous electric cars Networks of autonomous electric cars are another approach to reduce costs and emissions in last-mile delivery. Agatz et al. (2017) provide a valuable overview on advantages and challenges of autonomous vehicle platoons serving ad hoc passenger and freight transportation requests. The system consists of a pool of autonomous vehicles that can move individually or form a train (also called "platoon" or "flexible road train") with several others. While this could improve traffic flow and reduce the use of private cars, it is computationally hard to assess due to its complexity. The idea of a platoon with vehicles joining and leaving on the route is similar to the truck-and-robot distribution, where robots are picked up and released. However, there are not yet satisfying modelling and routing approaches for these concepts (Agatz et al., 2017). This is due to the high problem complexity, given that the vehicles have a high speed, range to move on their own, and are able to join or leave a platoon anywhere on the route without stopping. To obtain insights on performance of autonomous vehicle platoons in urban freight delivery, Haas and Friedrich (2017) apply traffic micro-simulation. While their results provide first insights into benefits and challenges, the system analyzed of three nodes and a roundabout does not provide a realistic problem size for application in practice. In addition, several technical and non-technical issues of autonomous driving remain, such as reliability of sensors and real-time processing, testing and validation as well as regulatory and insurance-related questions (Hussain and Zeadally, 2019). To summarize, platooning with autonomous electric cars has some similarities with truck-and-robot delivery, but there are no routing approaches available that could be transferred.

Parcel lockers Another related approach in last-mile delivery is the use of parcel lockers as hubs or storage locations. The most common way to use parcel lockers is to replace home delivery with pickup from a locker by the customer. This could be a way to reduce costs and emissions for less urgent deliveries that customers do not mind picking up close to their homes. For example, Veenstra et al. (2018) solve the facility location of parcel lockers and the vehicle routing problem of delivering either to the customers or a nearby locker. Parcel lockers are already in use on a large scale, e.g. by DHL (2020). In a further use case, lockers are applied as a micro-hub to exchange parcels between vehicles. Fikar et al. (2018) show through agent-based simulation that this approach can improve service quality and reduce delays in food delivery with cargo bikes. Enthoven et al. (2020) consider a two-echelon system of trucks and cargo bikes, using the lockers for cross-docking. Generally, our problem is similar to such two-echelon problems, but more complexity is added in our case as vehicles from the second-tier are transported by the first-tier vehicle. This leads to even more interdependencies and synchronization between the two vehicle types. The system of parcel lockers can be extended by the use of autonomous vehicles with several lockers that can park close to the customers and wait for them to retrieve their parcels. This is a promising concept for regular (next day) deliveries (McKinsey & Company, 2016). To summarize, delivery to parcel lockers is a promising solution for non-urgent deliveries and parcel lockers as micro-hubs have similarities to the truck-and-robot concept. However, we consider the truck-and-robot concept for urgent premium deliveries that has additional dependencies and requires specific routing approaches.

3.2.2 Autonomous last-mile delivery with robots

The literature on autonomous delivery robots is still very limited. It can be classified as hub-and-robot concepts and truck-and-robot concepts. While in the former robots move without an additional carrier, in the latter the support of delivery trucks is essential.

Hub-and-robot concepts Bakach et al. (2021b) provide a cost assessment for a concept involving robots. They propose a two-tier last-mile delivery system where a truck brings goods to local hubs in which the goods are stored and loaded into robots. The robots then make pendulum tours to the customers. This is simulated with and without time windows. The key difference versus the problem considered in this paper is that the robots are not transported on the truck but stay around one fixed hub. Furthermore, the hubs have a different role to that of our depots, as goods are stored and automatically loaded into robots there. The authors propose a mixed integer program (MIP) to first define the lowest possible number of hubs across a set of problem instances. In the next step, for a given number of hubs, the customers are assigned to hubs and individual robots such that robot mileage is minimized, while the maximum robot availability and range is limited. Note that this sequential approach is only possible if the flexibility of moving robots on the truck is given up. The authors provide a cost estimate for their solutions that indicates the potential to reduce operating costs by 70 - 90% compared to "truck only" deliveries. However, robot amortization and maintenance are not included in this comparison. Poeting et al. (2019b) assess the same concept as Bakach et al. (2021b), with only one robot per hub. A truck delivers most parcels directly to customers and only up to 3% of the parcels to the hubs for delivery by robots. Their focus is on planning a suitable truck tour such that customer time windows and hub opening hours are met. Poeting et al. (2019a) similarly propose an MIP to schedule robot pendulum tours from hubs with several robots such that customer time windows are met with minimal earliness and tardiness. They evidence that their MIP approach is suitable for up to 20 customers and evaluate the impact of the robot quantity. Sonneberg et al. (2019) propose an MIP to plan cost-optimal tours with robots only and assess the benefits of additional robot compartments (that enable a robot to serve more customers on the same tour). As relevant costs they consider a rental fee per robot per day, labor costs for loading the robots and transport costs per distance unit. The model is applied to instances with ten customers. As expected, additional compartments drastically reduce the distance traveled, number of robots needed and total costs, since the robots no longer return to the hub after every delivery. In comparison, our problem covers a more general case and could be reduced to such hub concepts by setting the maximum number of robots aboard the truck to zero. Furthermore, we investigate whether an integrated, cost-optimal selection of truck-and-robot routes can achieve savings in operational costs and eliminate the need for expensive hub and storage facilities.

Truck-and-robot concepts Jennings and Figliozzi (2019) propose a system where a truck drops off robots in the customers' area on several rounds and picks them up to return them to the DC on later rounds. They apply continuous approximation to estimate travel distances and times, but do not optimize for routing. Further, the authors do not consider robot cost and utilization, which will suffer from the robots' long waiting times in this concept. Boysen et al. (2018b) propose the delivery concept based on robot depots, as depicted in Figure 3.3. They assess a system of one truck, several depots, up to 40 customers and a similar number of defined drop-off points. In contrast to our area of application where time windows are necessary, a deadline for delivery is given for each customer and therefore early deliveries are possible. Their solution approach consists of a multi-start local search heuristic for truck tours and an MIP to find optimal robot schedules for each truck tour. The authors identify key drivers for service quality and show that several additional trucks would be needed to satisfy the same customer demands with traditional truck delivery. However, their objective exclusively considers the number of late deliveries, neglecting the actual delay times that are necessary to measure service levels. There is further no total cost evaluation that allows a realistic cost assessment with respect to truck routing and corresponding robot costs. The consideration of all relevant cost aspects significantly changes the search for optimal truck routes. Furthermore, robots are assumed to be available in unlimited numbers. This leads to non-feasible solutions with respect to a practical application, as the robot fleet and thus the number of robots per depot is limited. Alfandari et al. (2019) have compared the binary lateness measure for the truck-and-robot concept by Boysen et al. (2018b) with two alternatives (maximum and average delay) and developed a Branch-and-Benders-cut scheme to solve problems within an hour. While their numerical method proves to be efficient, the cost of robot use remains unknown.

Research gap and contribution Table 3.1 summarizes related literature. It includes characteristics on the use of support vehicles (SVs, i.e. robots or drones), i.e., if SVs (i) are transported by truck (SV on truck), (ii) are stored in a location (SV storage), (iii) availability in the storage is limited (SV avail.), and (iv) SV return to the truck after delivery (SV return).

This paper is the first to consider a large but limited number of support vehicles, which can be picked up and transported by the truck, delivery time windows and total costs. The truck-and-robot concept is one of the most promising new last-mile delivery concepts in city logistics and is technically ready for implementation. Other approaches have considerable limitations for use in dense urban areas. First publications have demonstrated the computational performance and potential of the truck-and-robot system for smaller problem sizes. However, none of the current publications identifies and takes into account all decision-relevant costs to assess the profitability of the truck-and-robot concept (i.e., truck routing and robot usage costs). Moreover, practical constraints (as the limited availability of robots) and hard time windows (required for attended home delivery) have not yet been considered. We therefore contribute to the existing literature by closing this gap. We deduce and integrate empirically collected costs for both trucks and robots and extend the concept by incorporating the mentioned restrictions in practice. In this way we are the first to present a cost-based evaluation of this innovative last-mile delivery concept, which is a prerequisite for assessing possible applications and their profitability. As such, it is not only of academic interest but also of high practical relevance, as it enables field tests on a larger scale.

3.3 Problem description

This section describes the technology required, the distribution processes involved as well as the associated costs for the truck-and-robot concept. It concludes by formulating the formal representation of the distribution problem.

Publication	Specifics	Objective	Aspects considered in optimization: Cust.	SV	TW	SV on truck	SV storage	SV avail.	SV return	Costs
Approaches with a truck and several drones										
Chang and Lee (2018)	Clusters	Delivery time	100	10	-	✓	-	-	✓	-
Salama and Srinivas (2020)	Clusters	Return time and routing cost	35	6	-	✓	-	-	✓	✓
Moshref-Javadi et al. (2020b)	Clusters	Customer waiting time	100	3	-	✓	-	-	✓	-
Kim and Moon (2019)	Drone hubs	Delivery time	80	na	-	-	✓	✓	-	-
Phan et al. (2018)	Multi-drone-TSP	Costs	100	3	-	✓	-	-	✓	✓
Murray and Raj (2020)	Multi-drone-TSP	Return time	100	4	-	✓	-	-	✓	-
Moshref-Javadi et al. (2020a)	Multi-drone-TSP	Customer waiting time	100	3	-	✓	-	-	✓	-
Hub-and-robot systems										
Sonneberg et al. (2019)	Varied robot sizes	Robot transport costs	10	3	✓	-	✓	✓	-	✓
Poeting et al. (2019a)	Varying no. of robots	Deviation from time window	20	5	✓	-	✓	✓	-	-
Poeting et al. (2019b)	Two-tier system with robots on second tier	Length of truck tour on tier one	na	na	✓	-	✓	-	-	-
Bakach et al. (2021b)	Two-tier system with robots on second tier	Number of hubs, robot time	300	95	✓	-	✓	✓	-	-
Truck-and-robot systems										
Jennings and Figliozzi (2019)	Sequential tours for dropping off and collecting robots	*None*: continuous approximation for tour length	na	na	-	-	-	-	-	-
Boysen et al. (2018b)	Deadlines	Number of delays	40	40	(✓)[1]	✓	✓	-	-	-
Alfandari et al. (2019)	Deadlines	3 lateness measures	100	100	(✓)[1]	✓	✓	-	-	-
This paper	Time windows	Total cost	125	125	✓	✓	✓	✓	-	✓

Cust.: maximum number of customers served, SV: support vehicles used (robots/ drones), TW: time windows, ✓: considered, -: not considered
[1] deadlines considered

Table 3.1: Overview on existing delivery robot literature and related truck-and-drone approaches

3.3.1 Truck-and-robot related technology

The key principle of the truck-and-robot concept is that a truck acts as mother ship that can pick up, transport, load and drop off delivery robots. Autonomous delivery robots are designed to carry out a single customer delivery at a time. The compartment is locked during transportation and can be opened at the customer location via an access code or using a smartphone app to retrieve the freight. The robots are equipped with several cameras, GPS, additional sensors for distance measurement and a mobile internet connection to move autonomously and prevent abuse or theft. This allows safe autonomous driving on sidewalks at pedestrian speed and, when needed, manual remote control. Once a parcel is delivered, the robot returns to a designated robot depot. The truck, in its role as mother ship, helps to reduce the distance traveled by the slow-moving robots. This reduces transportation times and increases the robots' utilization. A typical truck setup could be to use the complete floor of the loading space for approximately six to ten robots and the space above for shelves that carry the parcels. For instance, Daimler's "Vans and Robots" setup (see Figure 3.4) provides space for 54 delivery boxes and up to eight robots. A ramp allows the robots to enter and leave the truck either from the back or the side. An employee is needed to manually drive the truck and load the robots with freight. We refer to Hoffmann and Prause (2018) for more details on robot technology.

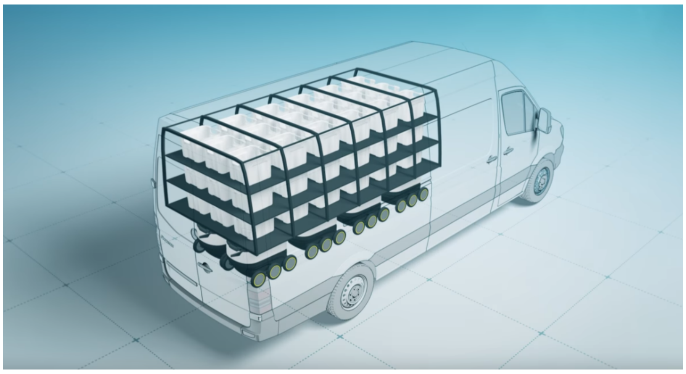

Figure 3.4: Example of truck layout with robots on the floor and goods in the racks above (Mercedes-Benz Vans, 2016)

3.3.2 Truck-and-robot distribution concept

The truck-and-robot distribution is based on a combination of a delivery truck, robots and a network of small robot charging stations, also called robot depots (see Figure 3.3). The concept is intended for attended home deliveries in inner cities. The distribution process consists of a truck tour and various robot tours, one for each customer. The truck departs from a given starting point (e.g., the DC) with all parcels to be delivered on board and visits several drop-off locations (blue arrows in Figure 3.3) to release robots for delivery. A drop-off location is either a robot depot (denoted as "R" in Figure 3.3) or a designated drop-off point (denoted as "d" in Figure 3.3), where trucks are able to stop and unload robots. In the case of robot depots, the trucks are able to pick up new robots for later deliveries or release robots with deliveries right from the depot. Once a robot is released for delivery, it visits a single customer (green dotted lines in Figure 3.3). We consider attended home deliveries, and consequently the delivery (i.e., arrival at the customer location) has to happen within the time window agreed upon with the respective customer. As soon as a robot arrives at the customer location, the customer is notified and can pick up the delivery. In the event that a robot arrives early (e.g., before the assigned time window), it must wait until the customer is available to receive the delivery. This is another advantage as the cost of the robot's waiting time is lower than the cost of a delivery person waiting or making multiple delivery attempts. After the parcel has been retrieved, the robot returns to the closest robot depot (not displayed in Figure 3.3 for sake of clarity), where it can recharge and wait for the next delivery tour. Since robots are comparatively slow, the advantage of this concept is that the truck does not have to wait for them. On its route, it will repeatedly pick up additional robots waiting at the depots, load them with parcels and drop them off close to the respective customers. As stated above, robots can also be loaded with parcels at depots and be sent to nearby customers without transporting them on the truck. The challenge of this concept is the definition of the truck route, i.e., to decide on where to stop the truck to drop off the robots. Once a truck route is defined, it is necessary to decide where to start each customer delivery such that late deliveries and robot mileage are minimized, while respecting the truck's maximum robot capacity and the robot availability of each depot along its way.

To summarize, the truck-and-robot concept presented relies on a network of depots and drop-off locations, robots fulfill the complete customer demand, and robots return to a robot depot, not to a truck, as they move at pedestrian speed.

3.3.3 Decision-relevant costs

In the following we derive the decision-relevant costs for truck-and-robot delivery. This is necessary to extend the current literature with the cost-based approach and also obtain managerial insights about the benefits of the truck-and-robot concept in general.

Truck-related costs One of the main drivers for potential cost savings within the truck-and-robot concept is the reduction of truck mileage compared to classical truck deliveries. The cost factors of the truck utilization are the driver's salary (which is incurred at an hourly rate), fuel consumption, tolls and truck amortization (which are proportional to the distance covered). These rates apply to the time and distance of the truck‘s entire round trip back to the starting point. The truck-and-robot system primarily aims at reducing truck-related costs the following way: the truck does not have to drive to every individual customer and the driver does not need to spend time carrying parcels and waiting for customers.

Robot-related costs The cost of robots is the primary cost increase compared to a traditional truck delivery. As is common for machines, an hourly machine rate for the robots is applied for the entire time the robot is loaded, travels to the customer, is unloaded and returns to the closest depot. This is also in line with a possible concept of a third party service provider for robots. In this concept, robot manufacturers (e.g., Kiwibot (2020) and Starship (2019)) offer their robots as a service for logistics companies at a predefined hourly rate. It incurs for the entire time between a robot's launch by the truck driver and its return to the closest depot after the delivery was made. Since we consider attended home deliveries as a premium shipping service (e.g., as offered by grocery stores and Amazon (2019)) a delivery cannot happen before the agreed time window. A robot must wait at the customer until the beginning of the time window in the event of an early arrival. The hourly cost rate covers amortization, maintenance, electrical energy, rent for the depot space and charging stations. The fact that for electric vehicles more than 50% of the total cost of ownership is incurred for amortization and only 22% for electrical energy highlights the need to consider usage time as the main cost driver for robots (Bekel and Pauliuk, 2019).

Service-related costs Service-related costs account for any delayed deliveries. This means that a late delivery will incur hourly penalty costs if a consumer does not receive the parcel within the agreed time window. For instance, some companies offer a refund of premium shipping fees if an order arrives late (e.g., Amazon (2019)).

This also implies a trade-off decision between service dimensions (i.e., on time) and logistics costs.

3.3.4 Decision model

Based on the characteristics described, we present the truck-and-robot model to minimize total costs, while respecting a limited robot fleet. The basis of a truck-and-robot routing problem consists of the locations to be visited by the truck and robots. The available locations can be divided into the following sets:

- Set of robot depot locations R: At these locations, the truck can load additional robots up to the depot's robot availability and/or launch available robots for delivery.
- Set of drop-off locations D: At these locations, the truck can only stop and launch the robots it has aboard. There are no additional robots available to load. This can be interpreted as a robot depot with a robot availability of 0.
- Set of customers C: Indicates the customer locations with a delivery request within the planning horizon being considered. These are visited exclusively by the robots.

Since the truck could potentially visit a drop-off or robot depot location $a, a \in D \cup R$, several times (i.e., releasing robots at different times), we allow this by duplicating locations. This results in the index sets $\hat{D}$ and $\hat{R}$ of duplicate locations. For simplicity, we refer to the set of all duplicate locations as $\hat{L} := \hat{D} \cup \hat{R}$. For every distinct location $a, a \in D \cup R$, we call the index set of its duplicates $I_a \subset \hat{L}$. Finally, we define I_a^m as the set of indices in I_a that are less or equal to $m \in I_a$.

Using these sets, a truck is characterized by its starting position γ, its initial number of loaded robots δ and its maximum robot capacity K. Subtours are not allowed and each truck stop is associated with a distinct arrival time t_i, $i \in \hat{L}$. The initial number of available robots in a robot depot is denoted by $r_a, a \in D \cup R$. These robots can be retrieved from and replenished into the depot by the truck. We do not consider the time when a robot has returned to the depots for robot availability.

Robots often return to a depot only after the truck tour is completed due to their low speed compared to the truck. Waiting for robots to return is not economical due to the high costs of the truck driver. Furthermore, the return times would be hard to predict in practice as customers could take longer to retrieve their goods. The travel times between locations i and j, $i, j \in \hat{L}$, are specified by ϑ^t_{ij} for the truck and between locations $i, i \in \hat{L}$, and customers $k, k \in C$, by ϑ^r_{ik} for the robots. Accordingly, λ_{ij} indicates the travel distance of a truck between the corresponding locations. Please note that any fixed processing time that occurs at a stop (e.g., for walking around the truck, loading and unloading the robots) can be modeled by adding it to the respective travel times. This processing time can be specific to every stop. The deliveries for all customers $k,\ k \in C$, have to happen within a defined time window of length ϵ, which ends with the deadline d_k (i.e., the time window of a customer is given by $[d_k - \epsilon, d_k]$). All customers are served exclusively by robots.

A solution is defined by the following decision variables. First, s_{ij} indicates whether the truck travels from location i to j, $i, j \in \hat{L}$. For every location duplicate $i, i \in \hat{L}$, x_{ik} indicates whether a customer k is served from there (i.e., a robot is launched from this location to drive to the customer) or not. To ensure feasibility and assess actual costs, we additionally need the following auxiliary decision variables. If a robot arrives before the designated time window, it has to wait until the start of the time window (causing robot usage costs). The corresponding waiting time is denoted by auxiliary variable $w_k, k \in C$. If the deadline is missed, the delay at customer k is indicated by auxiliary variable $v_k, k \in C$. The auxiliary variable q_i indicates the number of robots on the truck upon departure from location $i, i \in \hat{L}$. In line with this, auxiliary variable e_i indicates the quantity of robots taken out of a robot depot $i, i \in \hat{R}$.

The cost of truck routing and robot scheduling is defined as follows:

- The truck costs incur for the time (at the rate c^{t}) and distance (at the rate c^{d}) of the entire round trip back to the starting point. This includes travel and processing times for loading the robots.
- Robots travel from depot/drop-off locations to customers and return to the closest depot afterwards, i.e., the hourly cost (at rate c^{r}) for a robot incurs for the time

from loading the robot to its arrival at the depot closest to the customer served. Service times for loading and unloading are included in the travel times.

- Deliveries can only happen after the beginning of the corresponding time window due to customer availability. An earlier arrival causes waiting time (and thus usage cost at the rate c^{r}) for the robot.
- Deliveries after the time window lead to hourly delay costs at the rate c^{l} as customers have to wait for their goods.

Index sets	
C	Set of customers
D	Set of distinct drop-off locations
R	Set of distinct robot depot locations
$\hat{D}$	Set of drop-off locations including duplicates
$\hat{R}$	Set of robot depot locations including duplicates
$\hat{L}$	Set of all locations the truck can visit, including duplicates, $\hat{L} := \hat{D} \cup \hat{R}$
I_a	Set of duplicate indices $i, i \in \hat{L}$, of one distinct location $a, a \in D \cup R$
I_a^m	Set of elements $i \in I_a$ with $i \leq m$
Parameters	
d_k	Deadline for customer $k, k \in C$
K	Maximum robot capacity of a truck
r_a	Initial number of available robots in location $a, a \in R$
γ	Starting position of the truck, with $\gamma \notin \hat{L}$
δ	Initial number of robots aboard the truck
ϵ	Length of time windows
ϑ_{ij}^t	Truck travel time from location i to location $j, i, j \in \hat{L}$
ϑ_{ik}^r	Robot travel time from location $i, i \in \hat{L}$, to customer $k, k \in C$
λ_{ij}	Distance between locations i and $j, i, j \in \hat{L}$
Cost parameters	
c^{l}	Cost of delays per time unit
c^{d}	Cost of truck per distance unit
c^{t} (c^{r})	Cost of truck (robot) per time unit
Decision variables	
s_{ij}	Binary: 1, if truck goes from location i to location j; 0 otherwise
x_{ik}	Binary: 1, if customer k is supplied from location i; 0 otherwise
Auxiliary variables	
e_i	Number of robots taken out of depot location $i, i \in \hat{R}$
q_i	Number of robots aboard the truck at departure from location $i, i \in \hat{L}$
t_i	Arrival time of truck at location $i, i \in \hat{L}$
v_k	Delay of delivery to customer $k, k \in C$
w_k	Waiting time for robot at customer $k, k \in C$

Table 3.2: Notation

Table 3.2 summarizes the notation used. The objective function and the restrictions of the model are then formulated as follows.

$$\min \mathrm{TC}(S, X, E, Q, T, V, W) =$$
$$\sum_{i \in \hat{L}} \sum_{j \in \hat{L}} (c^{\mathrm{d}} \cdot \lambda_{ij} + c^{\mathrm{t}} \cdot \vartheta^{t}_{ij}) \cdot s_{ij} + \sum_{i \in \hat{L}} \sum_{k \in C} c^{\mathrm{r}} \cdot \vartheta^{r}_{ik} \cdot x_{ik} + \sum_{k \in C} (c^{\mathrm{l}} \cdot v_k + c^{\mathrm{r}} \cdot w_k) \quad (3.1)$$

subject to

$$\sum_{i \in \hat{L}} x_{ik} = 1 \qquad \forall k \in C \qquad (3.2)$$
$$\sum_{k \in C} x_{jk} \leq M \cdot \sum_{i \in \hat{L} \cup \{\gamma\}} s_{ij} \qquad \forall j \in \hat{L} \qquad (3.3)$$
$$\sum_{j \in \hat{L}} s_{\gamma,j} \leq 1 \qquad (3.4)$$
$$\sum_{i \in \hat{L} \cup \{\gamma\}} s_{ij} = \sum_{i \in \hat{L} \cup \{\gamma\}} s_{ji} \qquad \forall j \in \hat{L} \cup \{\gamma\} \qquad (3.5)$$
$$t_\gamma = 0 \qquad (3.6)$$
$$t_j \geq t_i + \vartheta^{t}_{ij} - M \cdot (1 - s_{ij}) \qquad \forall j \in \hat{L}; i \in \hat{L} \cup \{\gamma\} \qquad (3.7)$$
$$v_k \geq t_i + \vartheta^{r}_{ik} - d_k - M \cdot (1 - x_{ik}) \qquad \forall k \in C; i \in \hat{L} \qquad (3.8)$$
$$w_k \geq d_k - t_i - \vartheta^{r}_{ik} - \epsilon - M \cdot (1 - x_{ik}) \qquad \forall k \in C; i \in \hat{L} \qquad (3.9)$$
$$q_\gamma = \delta \qquad (3.10)$$
$$q_j \leq q_i + e_j - \sum_{k \in C} x_{jk} + M \cdot (1 - s_{ij}) \qquad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{R} \qquad (3.11)$$
$$q_j \leq q_i - \sum_{k \in C} x_{jk} + M \cdot (1 - s_{ij}) \qquad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{D} \qquad (3.12)$$
$$t_i \leq t_j \qquad \forall a \in R; i, j \in I_a : i \leq j \qquad (3.13)$$
$$\sum_{h \in \hat{L} \cup \{\gamma\}} s_{hi} \geq \sum_{h \in \hat{L} \cup \{\gamma\}} s_{hj} \qquad \forall a \in R; i, j \in I_a : i \leq j \qquad (3.14)$$

$$r_a - \sum_{i \in I_a^m} e_i \geq 0 \qquad \forall a \in R; m \in I_a \qquad (3.15)$$

$$s_{ij} \in \{0,1\} \qquad \forall i,j \in \hat{L} \cup \{\gamma\} : i \neq j \qquad (3.16)$$

$$s_{ii} = 0 \qquad \forall i \in \hat{L} \cup \{\gamma\} \qquad (3.17)$$

$$x_{ik} \in \{0,1\} \qquad \forall i \in \hat{L}; k \in C \qquad (3.18)$$

$$e_i \in \mathbb{Z} \qquad \forall i \in \hat{R} \qquad (3.19)$$

$$t_i \geq 0 \qquad \forall i \in \hat{L} \qquad (3.20)$$

$$q_i \in [0, K] \qquad \forall i \in \hat{L} \qquad (3.21)$$

$$v_k, w_k \geq 0 \qquad \forall k \in C \qquad (3.22)$$

The objective function (3.1) minimizes total costs TC. The first term of the objective function considers the truck costs, which depend on the traveling segments s_{ij} selected and the corresponding costs for distance and travel time. The second term sums up the robot costs dependent on associated travel times, and the last term sums up the costs for delayed deliveries and for robot waiting times if the assigned time window is not met. Please note that the costs of the robots' return to the closest depot do not depend on the scheduling decisions as they are known in advance, and thus added ex-post. Every customer must be supplied by exactly one robot (see Constraints (3.2)), and robots can only start from locations where the truck has stopped (see Constraints (3.3)). Constraint (3.4) ensures that only one truck can start in γ, and if it arrives at a location, it must also depart from there, which is denoted by Constraints (3.5). Constraints (3.5) also ensure the truck returns to the starting point γ. Based on the tour, Constraints (3.6) and (3.7) define the arrival times at every truck stop. These constraints further prevent subtours as a feasible arrival time t_i for a location $i, i \in \hat{L}$, visited twice, does not exist. Constraints (3.8) define the delay duration for every customer (in the event of late delivery) and Constraints (3.9) the robots' waiting time. Waiting times occur in the event of early arrivals as we consider attended home deliveries and customers are assumed to be available only during the agreed time windows. Both constraints use the fact that the arrival time at a customer k supplied from location j (i.e., $x_{jk} = 1$) equals $t_j + \vartheta^r_{jk}$. Constraints (3.10)–(3.12) keep track of the number of robots on the truck. The truck departs from the starting point with the initial number of robots. On later stops, robots launched are subtracted and robots loaded in depots are added. Constraints (3.13) and (3.14) ensure without loss of generality that duplicates of the same location are visited in ascending index order. This fact is then used by Constraints (3.15) to ensure that the number of robots in a depot after visiting duplicate m is not below 0. By enforcing ascending order of duplicates in Constraints (3.13) and (3.14), we ensure that no e_i of unvisited stops is included in Constraints (3.15) for any m that is visited. The variable e_i of unvisited stops could otherwise imply infinite robot supply, as they are not constrained by Constraints (3.11). Finally, Constraints (3.16)–(3.22) define the variables.

A solution consists of a truck route from the starting point γ through several depots and drop-off points and an allocation of each customer to one of the locations on that route, from which its parcel will be delivered by a robot. Consequently, the solution π can be denoted as a tuple of locations Y, where $y(u) \in R \cup D$ is the u-th stop, and a matrix $X = (x_{uk})$, defining whether customer k is supplied from stop u. Note that a location $a \in R \cup D$ can occur on the route several times and $y(1) = \gamma$ always holds.

The MIP presented is an NP-hard combinatorial problem (Boysen et al., 2018b). The truck must visit a location up to $|C|$ times, which means we have to solve the problem for $|C| \cdot (|R| + |D|)$ duplicate locations. In a real-world application, e.g., with $C = 50$, $R = 25$ and $D = 50$, these would amount to 3,750 locations, which makes it intractable to solve in acceptable time with conventional solvers. For this reason we propose a heuristic.

3.4 Solution approach

In this Section, we introduce our Truck-and-Robot Cost-optimal Routing approach (TRC). The task of finding an optimal truck tour is closely related to the classical vehicle routing problem (VRP), with the addition that the truck can visit a location several times or not at all, and that at the same time the robot schedules need to be defined. The proposed heuristic is therefore based on principles known from VRPs by generating an initial tour and improving the truck tour using a specific search procedure. The structure of our approach is inspired by Boysen et al. (2018b) as it has been shown that it is efficient for the truck-and-robot problem to minimize lateness. The specific algorithms and operators have been developed to reflect our setting and model. More precisely, we aim at minimizing total logistics costs, and therefore develop a tailored solution approach. This means that our approach uses a cost-specific start heuristic as well as specialized search operators to take into account the different cost aspects for the optimization. We extend the existing model by considering additional constraints for a limited availability of robots, and present a repair step for non-feasible routing solutions. Moreover, we introduce an innovative heuristic for the subproblem of scheduling robots.

Figure 3.5 summarizes our three-step approach. After an initial truck route has been determined using priority rules in *Step 1*, an iterative approach is applied that alternates Steps 2 and 3. *Step 2a* ensures the feasibility of the truck route with respect to robot availability before the robot scheduling subproblem is solved for each feasible truck tour in *Step 2b* to define the corresponding robot movements (i.e., which customers are served from which location on the truck tour). We present an exact and a heuristic approach to solve the robot scheduling. With the solution obtained, the remaining problem is to find the best truck tour, which takes place in *Step 3* using a tailored heuristic based on a local search algorithm.

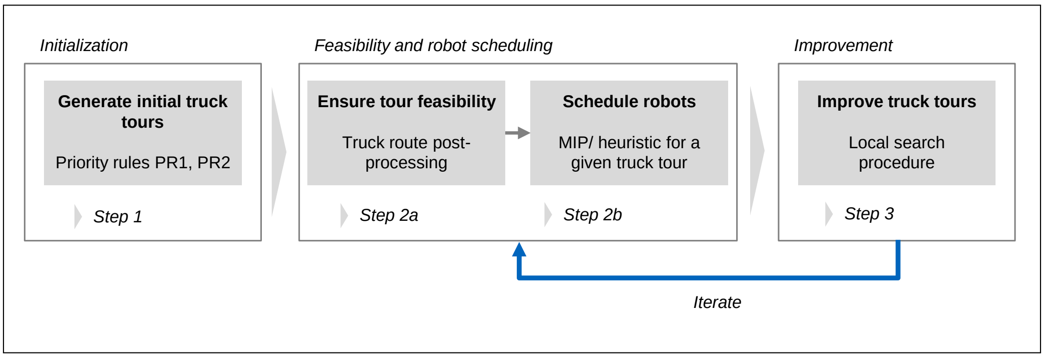

Figure 3.5: Illustrative structure of the applied solution method

Step 1: Initial solution for truck tours To obtain a start solution for the truck tours we apply the priority rules "Move to the position that is cost-optimal for the highest number of customers per the location's distance" (PR1) and "Move to the position from which most customers can be reached in time" (PR2). PR1 considers robot and delay costs but ignores truck-related costs. As a result it tends to lead to longer truck routes than optimal. PR2 aims to minimize delays, as they can be a pivotal cost driver. With robot and truck costs ignored, it leads to shorter than optimal truck tours at the cost of inefficient robot use. Combined, the two rules incorporate all key cost drivers and provide starting points on both sides of the optimum (routes that are too long and too short) to the local search. PR1 and PR2 are detailed in the following.

PR1 is based on the delivery costs of each customer k from a possible next drop-off or depot location $a, a \in R \cup D$. These costs consist of the robot travel and waiting time and potential delay costs. These are driven by the arrival time of the truck at location $a, a \in R \cup D$, and the robot travel time ϑ^r_{ak} between location a, and the customer $k, k \in C$. Truck costs are not considered as they cannot be clearly allocated to a single customer. We select the next (first) truck stop as follows. Based on the current time and truck location, the delivery costs for every location-customer combination ($a, a \in R \cup D$; $k, k \in C$) is calculated. Next, for every customer, the delivery costs at potential next stops is compared to the delivery costs of stops already visited (including the current stop). If the delivery costs are minimal for one of the stops visited, the customer is assigned to that stop and will not be considered for the further assignment process. Finally, for every location $a, a \in R \cup D$ that has not yet been visited, the number of customers with their cost minimum in this location is divided by the distance from the current truck position to derive a score. The location with the highest score is then selected as the next truck stop and the customers with their cost minimum in this location are assigned to it. The procedure is repeated until all customers $k, k \in C$, are assigned to a location $a, a \in R \cup D$.

PR2 considers only customers $k, k \in C$, that can be reached from a given drop-off or depot location $a, a \in R \cup D$, before their deadline d_k. It selects the location $a, a \in R \cup D$, with the highest number of customers that can be reached on time as next stop. At every stop, all customers who can be reached before the deadline terminates are served, and the selection of the next location $a, a \in R \cup D$, is repeated with the remaining customers. This procedure terminates as soon as there are no more customers that can be served in time. This rule was proposed by Boysen et al. (2018b) and proved to be effective for our approach as well.

Step 2a: Feasibility and robot availability Both priority rules ignore robot availability at any given location $a, a \in R \cup D$, and an unlimited number of customers can be served. This may result in non-feasible routes. In contrast to the routing approach by Boysen et al. (2018b) we must ensure that the total number of available robots NR (initial number of robots on the truck δ plus all robots at depots visited on the tour $r_a, a \in R$) is equal to or larger than the number of customers $|C|$. To do so, we post-process the results obtained by PR1 and PR2 by adding the closest additional depots if necessary. The post-processing is described in Algorithm 3.1.

Algorithm 3.1 Truck route post-processing for feasibility

Input: (Truck tour Y, number of customers $|C|$, initial robot availability on truck δ and in depots r_a)

```
feasible = false;
while not feasible do
    NR = δ + Σ_{a∈Y} r_a;
    if NR < |C| then
        append closest depot a ∈ R, a ∉ Y;
    else
        feasible = true;
    end if
end while
return Y
```

Step 2b: Robot scheduling for given truck route As a next step we solve the corresponding scheduling of robots for the given truck route. We propose two alternatives. The first one is based on the exact solution of a scheduling MIP, the second one develops a heuristic.

Alternative 1: Exact solution of MIP for robot scheduling An MIP for the subproblem of robot scheduling assigns customers to the truck stops (i.e. depots and drop-off points on the route). This is essential to evaluate the total costs. Using the MIP, we provide the basis for the solution evaluation and thus for the acceptance of improved solutions within the improvement heuristic. This complements the truck tour to a full solution, i.e., it provides the total costs for a given tour and the starting point of each robot delivery. In contrast to the complete problem, we do not need duplicates of robot drop-off (D) and depot locations (R). Table 3.3 presents the additional notation of truck tour parameters and decision variables.

Truck tour parameters	
U	Index set of stops on the truck tour $u \in \{1, 2, ...\}$
$y(u)$	u-th stop of the truck route, $y(u) \in R \cup D$
t_u	Arrival time at truck stop number $u, u \in U$
c^{T}_{uk}	Cost of serving customer $k, k \in C$ from stop $u, u \in U$
Decision and auxiliary variables	
x_{uk}	Binary: 1, if customer $k, k \in K$, is supplied from stop $u, u \in U$; 0 otherwise
q_u	Number of robots aboard the truck at departure from stop $u, u \in U$
r_{au}	Number of available robots in location $a, a \in R \cup D$, after the u-th truck stop

Table 3.3: Additional parameters and variables for the MIP

For a given truck route Y and the corresponding index set of stops U, the time of each truck stop $t_u, u \in U$, is determined by Equations (3.23) and (3.24). The parameter $y(u)$ indicates the actual location of the u-th stop, i.e., $y(u) \in R \cup D$. Based on the time of each stop, we precalculate the total cost c^{T}_{uk} of supplying a customer k from the corresponding stop u as defined in (3.25). The costs include the robot usage cost per time (c^{r}) depending on the travel time from the truck stop to the customer, the potential waiting time for the beginning of the delivery time window, and the time to return to the closest depot $\min\limits_{a \in R}(\vartheta^r_{ak})$. Delay costs are also included.

$$t_1 = 0 \tag{3.23}$$

$$t_u = t_{u-1} + \vartheta^t_{y(u),y(u-1)} \qquad \forall u \in U \setminus \{1\} \tag{3.24}$$

$$\begin{aligned} c^{\mathrm{T}}_{uk} &= c^{\mathrm{r}} \cdot (\vartheta^r_{y(u)k} + (d_k - \epsilon - t_u - \vartheta^r_{y(u)k})^+ + \min_{a \in R}(\vartheta^r_{ak})) \\ &+ c^{\mathrm{l}} \cdot (t_u + \vartheta^r_{y(u)k} - d_k)^+ \qquad \forall u \in U, k \in C \end{aligned} \tag{3.25}$$

The auxiliary variable r_{au} indicates the number of robots available at location $a, a \in R \cup D$, after the u-th stop and q_u indicates the number of robots aboard the truck after departing from u. The resulting MIP for minimizing the costs of robot scheduling for a given truck tour follows.

$$\min \ F(Q, X, R) = \sum_{u \in U} \sum_{k \in C} x_{uk} \cdot c^{\mathrm{T}}_{uk} \tag{3.26}$$

subject to

$$\sum_{u \in U} x_{uk} = 1 \qquad \forall k \in C \tag{3.27}$$

$$r_{au} = r_{au-1} \qquad \forall a \in R, u \in U : a \neq y(u) \tag{3.28}$$

$$r_{au} \leq r_{a,u-1} + q_{u-1} - q_u - \sum_{k \in C} x_{uk} \qquad \forall a \in R \cup D, u \in U : a = y(u) \tag{3.29}$$

$$q_0 = \delta \tag{3.30}$$

$$r_{a0} = r_a \qquad \forall a \in R \tag{3.31}$$

$$r_{au} = 0 \qquad \forall a \in D, u \in U \tag{3.32}$$

$$x_{uk} \in \{0, 1\} \qquad \forall u \in U, k \in C \tag{3.33}$$

$$r_{au} \geq 0 \qquad \forall a \in R, u \in U \tag{3.34}$$

$$0 \leq q_u \leq K \qquad \forall u \in U \tag{3.35}$$

Constraints (3.27) ensure that exactly one robot is launched to each customer. Constraints (3.28) and (3.29) keep track of the number of robots in the depots and on the truck after every truck stop. Constraints (3.30) and (3.31) define the initial quantity of robots in each depot and on the truck. Constraints (3.32) ensure that robots cannot be stored at drop-off locations. Constraints (3.33)–(3.35) define the range of variables.

Due to the problem structure, the corresponding LP solution satisfies the integer constraints in most cases. The problem can therefore be solved many times within the entire heuristic at affordable computational effort. Nevertheless, the MIP for the robot scheduling constitutes a potential bottleneck in terms of computation time. For this reason, we propose an alternative heuristic approach in the following. Numerical analysis will be applied to assess the performance in terms of computation time vs. solution quality of both variants.

Alternative 2: Heuristic for robot scheduling The robot scheduling heuristic (RSH) relies on (i) calculating a lower bound of the robot schedule cost, (ii) fixing the number of started robots per stop, and (iii) swapping customers between stops.

(i) The RSH starts with assigning each customer to the cost-optimal truck stop without considering robot availability. Here we need to differentiate two cases: First, if the resulting solution is above the total costs of the current best-known solution, the given truck tour cannot be part of a better solution. It will be disregarded and the search continues with Step 3. Second, if this solution is below the total costs of the current best-known solution, we further test if this robot schedule is feasible with respect to robot availability in the depots and on the truck. If so, this robot schedule and the given truck tour are the new best-known solution. If not, we search for a feasible robot schedule for the given tour in step (ii).

(ii) Customers are sequentially assigned to stops on the tour using regret-insertion. The regret is defined as the cost difference of the best and second best possible stop to which a customer can be assigned. Since stops can run out of robots simultaneously (e.g., several consecutive drop-off points as soon as the truck runs empty), we consider as possible second-best stops only those after the best one, starting at the first depot. The customer with the highest regret is assigned to the best possible stop. After each assignment, the robot availability and truck capacity have to be updated, and new regret values are calculated. This step is repeated until all customers are assigned and therefore a feasible robot schedule is obtained.

(iii) Finally, we improve the robot schedule by swapping pairs of customers if this leads to a cost decrease. Please note that this step does not impact robot availability and therefore feasibility remains. We further focus on the customers currently not assigned to their best stop to speed up the search. This step concludes the robot scheduling and provides the total costs of the current solution.

In the following, we denote the approach based on this heuristic as *TRC-RSH* and the approach based on exact robot scheduling as *TRC*. In the numerical analysis, we will show which of the alternatives is beneficial under which conditions.

Step 3: Local search for improving truck routing A local search (LS) procedure is applied to improve the solutions. The truck tours obtained are post-processed in each iteration to obtain feasible tours (Step 2a), and the corresponding robot scheduling (Step 2b) is solved. This means that any truck tour Y obtained within the LS corresponds to exactly one complete solution $\pi = (Y', X)$, which is the result of the feasibility check and the subsequent robot scheduling. Please note that if Y is feasible, $Y' = Y$ holds. This enables us to find the best possible objective value for any given truck tour as the respective robot scheduling solution is optimal. We denote the complete solutions (i.e., truck tour after feasibility check and corresponding robot schedule) for the initial tours Y_{pr1} and Y_{pr2} as π_{pr1} and π_{pr2}. Each of these two start solutions is then evaluated $\kappa/2$ times by the LS procedure as given in Algorithm 3.2. The best solution is chosen across all κ search cycles. The LS is applied for κ^{ls} iterations with the respective starting solution. The following operators within the LS are used in order to improve the given truck route:

- *Best-cost removal:* The stop whose removal leads to the best estimated total cost is removed. This operator aims at reducing unnecessary detours early on the tour, which may cause delays on later stops. The total costs are estimated without solving the robot scheduling, using the following procedure. First, for all customers assigned to the remaining stops, the new arrival time is calculated based on the new, shorter truck tour. This new arrival time can lead to an updated robot waiting time or delay time, and thus a change in cost. Customers who were previously assigned to the stop that has been removed must be redistributed. We do this by calculating the hypothetical robot and delay cost (based on arrival time) for every combination of stops on the truck route and customers to be redistributed. All of these customers are then assigned to the stop where their respective cost is minimal (without enforcing the number of available robots at that stop). Finally, the truck time and distance are updated together with their corresponding costs. After assessing these hypothetical new total costs for the removal of every stop, the stop with minimal cost is selected for removal.

- *Depot insertion:* After a random stop of the current tour, the unused depot that leads to the minimal deviation is inserted. This operator contributes to the degree of freedom for the robot schedule as this increases the number of available robots and their possible starting locations.
- *Random removal:* A random stop is removed from the current tour.
- *Random insertion:* A random stop is inserted at a random position of the current tour.
- *Random swap:* Two random stops of the tour are swapped.

Algorithm 3.2 Local search procedure

```
Input: (Starting solutions π_pr1, π_pr2; number of iterations κ)
  π_best = null; // best solution found
  for i = 1 to κ do
    if i ≤ κ/2 then
      π = π_pr1.clone();
    else
      π = π_pr2.clone();
    end if
    π_local = LS(#iterations: κ^ls, start solution: π); // local search with κ^ls cycles
    if π_best = null or Z(π_local) ≤Z(π_best) then
      π_best = π_local; // save best result
    end if
  end for
  return π_best
```

To enlarge the search area in the event of local minima, two instead of one of these operations are performed sequentially within a single iteration if no improvement has been made during the last 50 LS cycles and three operations after 100 LS cycles without improvement. Our tests have shown that this enables the best-known solutions to be improved, and that the best solutions were also found faster and more robustly across all κ cycles.

3.5 Numerical analysis

This section analyzes the performance of our solution approach and provides managerial insights. Section 3.5.1 describes the generation of our problem instances, which are available online at `http://www.vrp-rep.org/datasets/item/2020-0005.html`. Using these instances, we show the efficiency of our *TRC* approach in Section 3.5.2. This section compares it first to MIP solutions using Gurobi for small instances (see Section 3.5.2). We further investigate the efficiency of the exact vs. heuristic approach for the robot scheduling (see Section 3.5.2). Additionally, we asses the efficiency with the state-of-the-art approach from literature in Section 3.5.2. Section 3.5.2 extends the comparison to other instances provided in recent literature. In Section 3.5.3, we compare the truck-and-robot system's performance to traditional truck deliveries. Section 3.5.4 details the cost structure of the solutions and resulting implications for the system's ability to meet changing demand settings. We have implemented our algorithm and the corresponding benchmark approach in Python (using PyCharm 2018.3.5 Professional Edition) with Gurobi (version 8.0.1) as a solver for the robot scheduling MIP. All computations were executed on a 64-bit PC with an Intel Core i7-8650U CPU (4×1.9 GHz), 16 GB RAM and Windows 10 Enterprise.

3.5.1 Instance generation and parameter setting

This section details the generation of the problem instances. We consider Munich as a model delivery area for deriving actual delivery situations. We assume a delivery area that resembles half of the city center (a square area with side length of 4 km) and in which $|C| = 50$ customers are served per tour. For a realistic spatial distribution we randomly choose 50 of all known building locations in the northern half of Munich using Open Street Maps (see OpenStreetMap Foundation (2019)) and distribute the $|R| = 25$ depots in an equidistant manner. The drop-off points are uniform-randomly distributed and the truck is assigned to one random depot or drop-off location as a starting point. For the deadlines we assume the company offers different delivery times for customers, depending on their location, and assigns orders to trucks such that early deadlines are closer to the truck's starting position and later deadlines further away (see similar approach in Boysen et al. (2018b)). To simulate this, every deadline is calculated as $d_k = t_{k,min} \cdot \rho_k$, where $t_{k,min}$ is the time needed to directly go from the starting point to the customer by truck (excluding any handling times) and ρ_k is a factor drawn from a uniform distribution in the interval $[\rho_{min}, \rho_{max}]$ (in our example $[5, 8]$) for every customer individually. This procedure assumes a given vehicle allocation such that viable tours can be planned. The initial number of robots is set to $r_a = 10$ for every depot $a, a \in R$, if not stated otherwise. We further use the setting for the remaining parameters as given by Boysen et al. (2018b). The truck can carry up to $K = 8$ robots and is fully loaded at the beginning ($\delta = 8$). The average speed is 30 km/h for the truck and 5 km/h for the robots. The handling time per stop is assumed to be 40 sec. The truck costs are estimated using fuel, labor, investment costs and amortization, resulting in a distance cost of $c^{\text{d}} = 0.20$ €/km and a time cost of $c^{\text{t}} = 30$ €/h. For the delays, we refer to the common offer to refund the premium shipping fee in the event of late delivery. In our example, a fee of 5 € is refunded for a one hour delay. Robot costs are calculated based on the target purchasing price of 2,000 a€s reported on Condliffe (2018). Amortization time is assumed to be five years. The robots are in operation 50 weeks per year, six days per week and eight hours per day. Furthermore, the utilization rate is 50% and there is a markup for maintenance, electricity, etc., of 50%. Using these estimates, we derive a cost rate of $c^{\text{r}} = 0.50$ €/h, with c^{r}= 2,000 /€(5 · 50 · 6 · 8 h · 50% utilization) · (1 + 50% markup). Lastly, we set the run-time parameters of our heuristic to $\kappa = 16$ search cycles and $\kappa^{\text{ls}} = 500$, as this robustly produced results that were close to the best-known one across all κ cycles. The search cycles were executed in parallel using multiprocessing. Table 3.4 summarizes the parameters applied that have been collected empirically. We apply this data throughout all experiments except for a further comparison on lateness in Section 3.5.2, where we build the experiments upon the data instances of Boysen et al. (2018b).

Parameter	Description	Value
Index set sizes		
$\lvert C \rvert$	Number of customers	50
$\lvert D \rvert$	Number of drop-off locations	48
$\lvert R \rvert$	Number of robot locations	25
Constraints		
K	Truck's maximum robot capacity	8
r	Initial number of available robots per depot	10
δ	Initial number of robots aboard the truck	8
ϵ	Length of time windows	10 min
μ	Fixed processing time at every stop	40 s
$[\rho_{min}, \rho_{max}]$	Deadline factor interval	[5, 8]
ω_t	Average truck speed	30 km/h
ω_r	Average robot speed	5 km/h
Cost factors		
c^{d}	Distance cost of truck	0.20 €/km
c^{t}	Time cost of truck	30 €/h
c^{r}	Cost of robots	0.50 €/h
c^{l}	Cost of delays	5.00 €/h

Table 3.4: Empirically estimated values applied as default values in a problem instance

3.5.2 Efficiency of suggested solution approach

Comparison to exact solution of MIP

In our first performance comparison, we assess the solution quality and runtime of our *TRC* approach by solving the MIP presented exactly (see Equations (3.1) to (3.22)) using Gurobi. The MIP solution using Gurobi is only feasible and computationally tractable for small instances. We use instances with 9, 12, 15 and 18 customers and 16 depots each. The number of drop-off points is derived as $2 \cdot |C|$. The number of robots per depot r, the robot capacity of the truck K, and initial number of robots aboard δ are set at $|C|/3$. The remaining parameters are as described in Table 3.4. We consider two duplicates of each location, i.e. the truck can visit a location at most twice. This has proved to be sufficient to find the optimal solution. The runtime limit of Gurobi is set at 30 minutes. Table 3.5 compares TRC to the MIP for 20 instances per problem size, showing the percentage of instances for which the optimal and best-known solutions were found, the average gap of solutions to Gurobi‘s lower bound, the runtime and average savings obtained by applying TRC instead of Gurobi. The comparison shows that the MIP solution with Gurobi becomes computationally intractable for more than 12 customers. TRC in contrast remains fast, identifies better solutions on average and in total finds 14 out of 19 proven optima (i.e., 18% of all 80 instances).

	Optima found [%]		Best solution found [%]		∅ Gap to lower bound [%]		∅ Runtime [sec]		∅ Improvement by TRC [%]	
Customer	TRC	Gurobi	TRC	Gurobi	TRC	Gurobi	TRC	Gurobi	Cost	Time
9	50	65	85	70	9	12	48	995	4	95
12	20	30	90	40	20	25	63	1,524	6	96
15	0	0	95	30	33	41	85	1,800	13	95
18	0	0	100	5	33	50	101	1,800	25	94
Total	18	24	93	36	25	36	75	1,530	14	95

Table 3.5: Performance of our TRC approach compared to solution of the MIP (Equations (3.1) to (3.22)) with Gurobi (20 instances per row)

Comparison of alternative solution approaches

Next, we compare the performance of TRC and TRC-RSH, i.e., the efficiency if RHS is used for robot scheduling instead of the MIP in Step 2b. Table 3.6 summarizes the results across 20 instances per parameter setting.

Instances		Average change due to RSH [%]		Standard deviation
Customers	Robots per depot	Computation time	Objective value	of objective value change [ppt]
25	10	-71	0.00	0.0
50	10	-56	0.00	0.2
75	10	-46	-0.03	0.9
100	10	-29	0.05	1.7
125	10	-23	-0.39	2.4
100	5	-14	4.77	9.5
100	30	0	-0.06	1.7

Table 3.6: Performance comparison of TRC-RSH vs. TRC

Applying TRC-RSH is up to 71% faster, in particular for smaller test cases. However, the advantage of computational time decreases with a growing number of customers. With high number of customers and high number of robots, the TRC becomes as fast as the TRC-RSH. Whereas there is no significant difference in solutions between

TRS-RSH and TRC for instances with up to 100 customers and 10 robots per depot, the objective values deviate increasingly and we obtain slightly better or worse solutions by TRS-RSH. This can be attributed to the stochastic search within our LS. Since RSH evaluates some truck tours with higher-than-actual costs, it can avoid local minima that prevent finding a better solution. On the other hand, it can also prevent the LS from finding the best solution. Both effects occur increasingly with larger instances, as can be seen in the increasing standard deviation. This effect is reinforced with very tight robot availability (see e.g., example with 100 customers and 5 robots per depot). A lower robot availability reinforces RSH's disadvantages in robot scheduling and therefore leads to worse objective values. On the other hand, when robots are not limiting at all, RSH's time advantage disappears, since the trivial robot schedule to assign every customer to their cheapest stop is often feasible.

In conclusion, TRC-RSH has the potential to save computation time in cases where robot availability is not extremely low. As one of our contributions is to consider robot availability, we will rely on TRC in our further experiments as it provides more robust results and performs better for scenarios with more restricted robot availability. In practice, the choice between TRC-RSH and TRC can be made based on the number of customers and the availability of robots.

Comparison to benchmark approach

We compare our solution approach to the state-of-the-art approach by Boysen et al. (2018b). Their approach (denoted as *TRL*) aims at minimizing the number of customers who receive their order late, i.e., it only evaluates whether orders are delivered late, but does not account for the absolute time of delay. The multi-start local search procedure of TRL relies on a neighborhood search with standard operators for VRPs. We reimplemented this approach and use the parameter setting as indicated by the authors for the "large dataset". We adapt our TRC approach to minimize the number of late deliveries as well (denoted as *TRC lateness*), and set the local search limit to $\kappa^{\text{ls}} = 2,000$. TRL lateness and TRC lateness are ultimately compared with TRC applied for its intended total cost objective (denoted as *TRC* *cost*). For both TRL and TRC lateness, the total costs are obtained by an ex-post calculation on the final solution using the cost factors stated. Figure 3.6 shows the average results (of a sample of 20 instances per data point) for 25 to 125 customers.

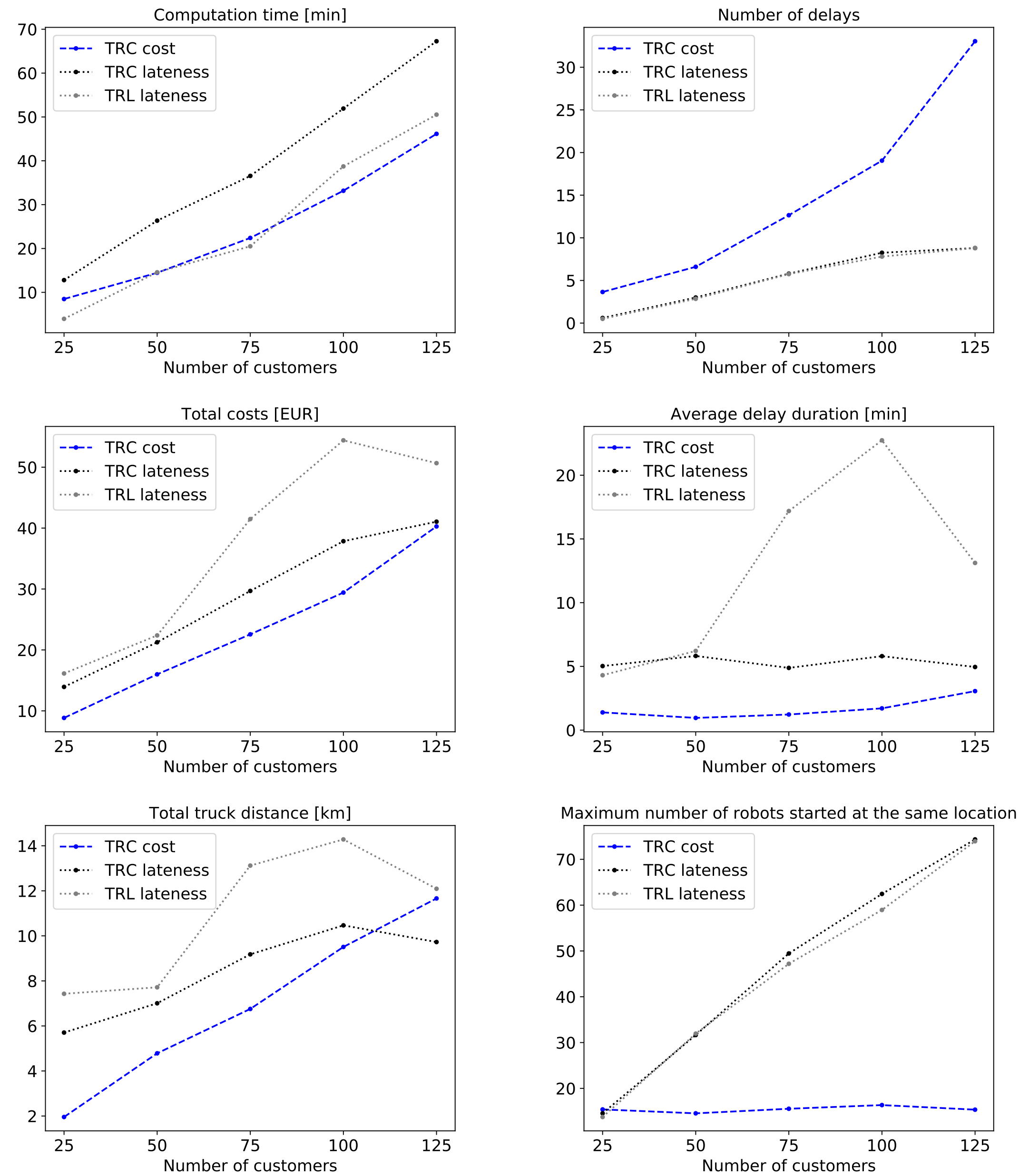

Figure 3.6: Performance comparison of our TRC method and the TRL routine by Boysen et al. (2018b) applied for the two objectives cost and lateness

Performance comparison for minimizing lateness The results show that our TRC approach also performs very well if applied for the altered objective of minimizing lateness. In comparison to TRL, it is able to match the given results both in terms of computation time and objective value. The higher computation times for *TRC lateness* can be attributed to cost-specific starting rules and search operators. As its rules and operators are focused on total cost, it takes TRC slightly longer to converge to solutions with the minimal number of delays. The increase in runtime for all methods is mostly driven by the fact that the MIP for robot scheduling must be solved for longer truck tours with an increasing number of customers.

Impact of total cost perspective Considering TRC applied to the cost objective, we see that its computation times are comparable to TRL. Looking at the solution structure, the comparison between TRC with cost objective and TRL further highlights the impact of a total cost perspective. *TRC cost* leads to a higher number of delays. However, the average delay duration (of all delays > 0) is drastically reduced compared to the TRL results. This is because for TRL, every delayed customer is "lost" and there is no incentive to shorten an unavoidable delay. In contrast, TRC shortens unavoidable delays. Comparing the total costs achieved by all approaches, we see that minimizing lateness while ignoring the other cost factors leads to a significant increase in actual costs. By way of example, in the case of 75 customers, *TRC cost* reduces total costs by 46%, average delay duration by 93% and truck mileage by 49% compared to TRL. The difference in total costs is lower when comparing *TRC cost* to *TRC lateness* since the latter already applies our cost-specific operators and therefore respects total costs while minimizing lateness. For the large data set of 125 customers, the total costs of both approaches are almost equal. The reason for this is the impact of robot availability, which we detail in the following.

Impact of limited robot availability In *TRC cost* we take into account the limited availability of robots at depots (i.e., 10 robots per depot), which represents an additional restriction. This means that the more customers are to be served, the more constraining the limited robot availability gets for the routing decision. In the case of 125 customers, for instance, the truck must visit at least 12 depots to supply all customers (see increased truck distance and delay duration). The restriction reduces the cost advantage at first glance but leads to a more realistic truck tour where the maximum number of robots started at a single depot is reduced from 74 to 16. Thanks to our combination of cost optimization and limited robot availability, the maximum number of robots started remains constant as the number of customers goes up. This makes the concept economically feasible, as depots must be affordable and easy to integrate into existing traffic space and the robot fleet size should be realistic to keep investment costs as low as possible.

Performance comparison based on instances from literature

We further assess the algorithmic performance with data instances of Boysen et al. (2018b). The so called “large dataset” comprises 200 instances with 40 customers each. We compare our TRC to the TRL approach of Boysen et al. (2018b) based on their lateness objective (see Section 3.5.2).

The comparison in Table 3.7 confirms the results from Section 3.5.2. *TRC lateness* finds objective values comparable to *TRL lateness*. Out of 200 instances, 182 instances were solved with identical objective values by both approaches. TRC found a better solution for four instances and a worse solution for 14 instances. Due to the problem- and cost-specific search operators used in our TRC approach, it takes more effort to find lateness-optimal solutions, but it is able to find pareto improvements with respect to the associated costs: total costs, covered truck distance, and the absolute time of delays are decreased despite the objective of minimizing the number of delays. TRC‘s computation times are five to six minutes higher. As already seen in Figure 3.6, TRL is faster for smaller instances (as it is the case here with 40 customers), but the relative difference is much lower for larger instances with more customers. Both approaches are consequently comparable with respect to time-efficiency.

Criteria	TRL lateness Boysen et al. (2018b)	TRC lateness this paper	Change TRC vs. TRL [%]
# of best known solutions found	196	186	-5.1
Avg. objective value [number of delays]	0.99	1.03	+4.0
Avg. total costs (ex-post evaluation) [€]	27.49	25.25	-8.1
Avg. truck distance [km]	10.09	8.49	-15.9
Avg. delay per customer [sec]	14.38	10.33	-28.2
Avg. computation time [sec]	217.10	563.13	+159

Table 3.7: Performance of our TRC approach compared to the TRL approach of Boysen et al. (2018b) based on the 200 instances of the "large dataset" of Boysen et al. (2018b)

3.5.3 Analysis of truck-and-robot performance

A detailed cost comparison with traditional delivery modes is a prerequisite for the adoption of truck-and-robot systems. As such, we analyze different parameter settings for the truck-and-robot concept and compare the TRC results with the solution for classical truck-only delivery, modeled as a vehicle routing problem (VRP) with time windows. We solve the VRP (see Appendix A) using the Gurobi solver in Python with a time limit of three hours. We adapt our problem setting to solve the VRP in reasonable time and apply the following to reduce computational effort. First, there are no waiting times. We assume that the driver can always leave the parcel at the door should he/she arrive before the time window, which reduces the time windows to deadlines. This drastically increases flexibility for the truck routes. Furthermore, we define the same processing time of 40 sec. per customer, which is an optimistic time for manual deliveries. Finally, we allow the use of up to four vehicles without incurring any fixed cost per vehicle. The solutions show that less than four vehicles are used in more than 90% of the cases and thus additional vehicles do not provide better solutions. These simplifying assumptions work in favor of the truck-only deliveries.

Influence of varying robot fleet size and general observations

First we study the impact of robot availability. Figure 3.7 shows how the truck-and-robot concept solved by our TRC (labeled *TRC cost*) performs compared to the VRP subject to robot availability per depot $r_a, a \in R$. We analyze the impact on total costs, truck distance, the average delay per customer, and the number of delays. The reported numbers are again average values from 20 instances per data point. Since truck-only delivery does not use any robots, its performance is not affected by a change in robot availability. Further, since the truck-only delivery could not be solved to proven optimality within the defined time limit of three hours in all cases, we report the values of the best-known solution (labeled *VRP*) and the lower bound of the total cost (labeled *VRP LB*).

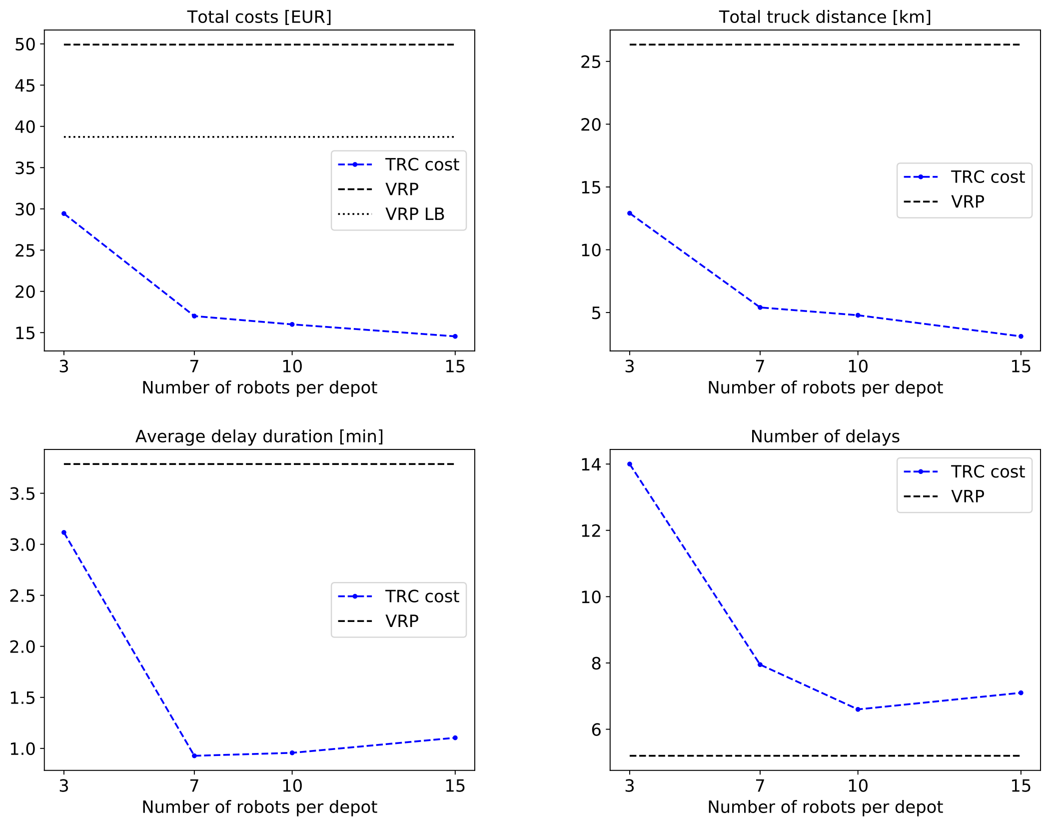

Figure 3.7: Comparison of truck-only delivery to TRC for a varying number of robots per depot

In our default case with 10 robots per depot, TRC leads to a cost saving of between 59% and 68% (compared to the lower bound and the best-known solution, respectively). In comparison to the best-known VRP solution, the truck distance is reduced by 82%. This correlates with the reduction in CO2 emissions when diesel trucks are applied. While the number of delays increases by 27%, the delay duration per customer is reduced by 68%. This shows that the VRP does not avoid delays even if four delivery trucks are used. In contrast, the truck-and-robot approach is able to adapt its solution in order to minimize the total delay time by accepting smaller delays for a few more customers, and thus reducing total costs. Finally, in the case of truck-only delivery, an average of three trucks is required to serve all customers compared to only one truck using the truck-and-robot concept.

While cost and truck mileage for the truck-and-robot system are very competitive even for a small robot fleet, a minimum number of robots is required per depot (in our case 7). Above this minimum fleet size, the improvements in total costs achieved are small, but a reduction of the robot fleet below this level leads to a significant increase in total costs. The number of delays as well as total time of delay is minimal for ten robots per depot. Neither an increase nor a decrease leads to better solutions. The reasons for this are twofold. First, an increase in the number of robots per depot above 10 causes more delays, as the trade-off between truck cost and delay cost leads to shorter truck tours that cause delays due to longer robot trips. Second, the number of available robots per depot defines the shortest possible truck tour such that the truck can pick up enough robots. If the number of robots is reduced, the truck tour becomes longer and the probability of late deliveries from later stops rises. This effect also explains the substantial increase in truck distance, delay and cost for the case there are three robots per depot. Even if all customers were located in the same area, the truck would have to visit $(|C| - \delta)/r$ depots, i.e., 14 depots in this case. This leads to high truck cost and a late robot launch for the customers served last. The effects of robot availability observed highlight the need to carefully size the robot fleet for the cost-optimal supply of customers. On the other hand our results indicate that the concept could be piloted with a very small robot fleet for a use case in which small delays are still acceptable. In summary, our results show that the proposed truck-and-robot concept outperforms classic truck delivery in terms of both cost and service criteria.

Influence of varying depot density

Since the availability of robot depots is essential for the truck-and-robot concept, we show the impact of changing depot density. For this experiment we keep the total number of available robots as close as possible to our default setting in Section 3.5.1 by adapting the number of robots per depot to 28/16/10/7 for the instances with 9/16/25/36 depots, respectively. The results are summarized in Figure 3.8.

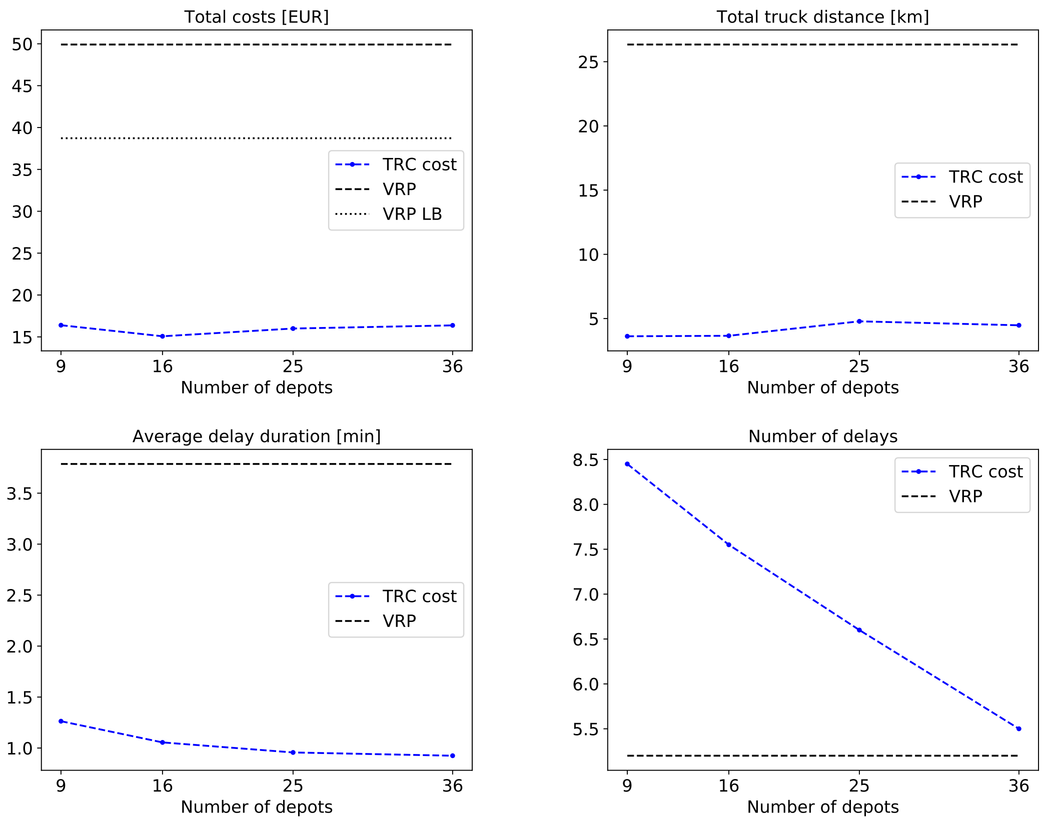

Figure 3.8: Comparison of truck-only delivery to TRC for varying robot depot density

The results show that the impact of depot density on cost and truck distance is small. In our analysis, the number of robots per depot increases as the number of depots declines, which leads to shorter truck tours (see also Section 3.5.3). In detail, mileage is reduced by 24% in the case of nine depots vs. our default case with 25. Total costs vary between -2% (25 vs. 36 depots) and +6% (25 vs. nine depots) only, and do not reveal a clear trend. Note that due to the equidistant arrangement of depots assumed, a change in their number affects all depot coordinates, and thus a solution with fewer depots may be better if those depots happen to be closer to the random customer locations. This explains the cost minimum for 16 depots. The most striking observation is the increase in the number of delays together with total time of delay. If fewer depots are available, the best solution accepts an increase in delays in favor of shorter tours. This observation is in line with our results from Section 3.5.3. In summary, fewer depots can be attractive with regard to cost and emissions but significantly reduce on-time deliveries. This highlights the advantage of the truck-and-robot concept with depots vs. other approaches that involve only one large robot hub in which robots are loaded and launched.

Influence of varying robot costs

The cost of a single robot is an important input factor for the truck-and-robot distribution, and may vary significantly for different situations. This depends on market dynamics, volumes produced and technical details of the different robot models. We therefore analyze the impact of varying robot costs on the overall problem. The results are illustrated in Figure 3.9.

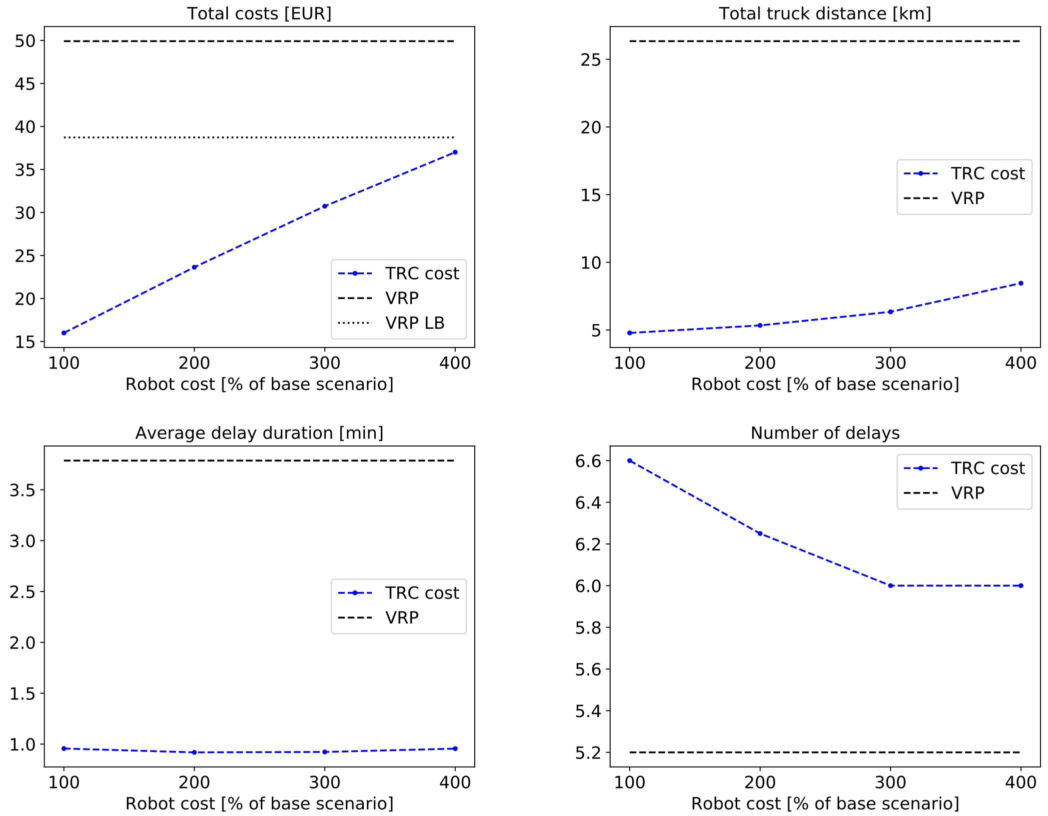

Figure 3.9: Comparison of truck-only delivery to TRC for varying robot costs

As could be expected, total costs show a linear increase as the robot cost rate rises. However, even if our basic rate of 0.50 €/h is quadrupled to 2.00 €/h, total costs increase by only 131% and remain competitive compared to truck-only delivery. This means that even with significantly higher robot costs, the truck-and-robot concept enables distribution with lower costs compared to the lower bound of the VRP. The robot cost increase is only partly mitigated via longer truck tours (i.e., the truck visits more stops close to customers such that robot time is reduced). In the case where robot costs are 2.00 €/h, we found that robot time was reduced by 34% compared to the base case, leading to a 164% increase in overall costs for robots. Additional deviations on the truck tour to reduce robot time would lead to delays for customers who are served from late stops and are thus inefficient. Up to 300% of the original cost rate, we can again observe the positive effect of longer truck tours on delays (also see Section 3.5.3). Our two delay measures decrease by 9% each with higher robot costs as more distance is covered by the truck and this reduces robot travel times and consequently delays. The increase in the truck distance of 77% is still acceptable considering the great reduction in travel distance compared to traditional truck delivery.

3.5.4 Cost impact of changing demand

The number of orders may increase as the service becomes more popular with consumers. Consequently we also analyze the impact of changing demand on the cost structure. Figure 3.10 provides a breakdown of the costs depending on the number of customers served. Please note that waiting cost and costs for robot travel times are summarized as robot costs.

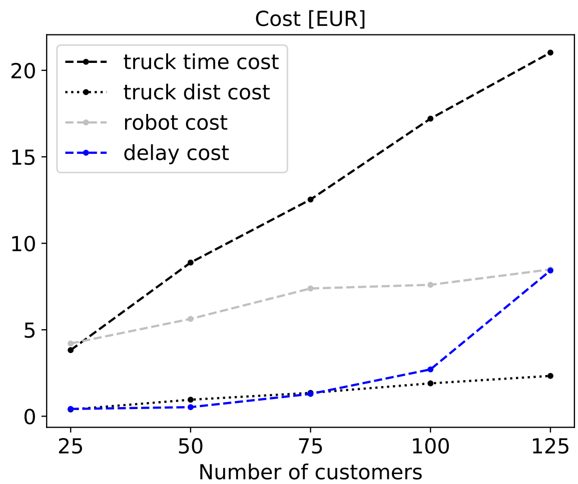

Figure 3.10: Cost breakdown depending on number of customers

The truck usage cost (both for time and distance) increases linearly with the number of customers. As the number of customers quintuples from 25 to 125, so does the truck cost (+450%). The robot costs (including a growing share of 20 - 38% waiting costs), on the other hand, only increase moderately. Consequently, a change in order volumes within a certain range does not change the robot fleet needed or its utilization but mostly the shift and tour length of the truck. In the event of lower order volumes, the truck-and-robot concept therefore helps to save variable truck costs (e.g., fuel and personnel cost) as the use of robots enables short delivery tours. Furthermore, the delay cost increases moderately for scenarios with up to 100 customers as well. As a consequence, the truck-and-robot concept robustly reveals good service performance for the corresponding scenarios. Afterwards, the delays increase significantly for supplying another 25 customers (125 customer case), meaning that a higher number of robots per depot would be necessary to decrease the number of delays (see Section 3.5.3). In conclusion, these results underline the flexibility and advantages of the truck-and-robot system but also identify the limits for on-time deliveries and the need to adapt the robot system to the given requirements.

3.6 Conclusion

The truck-and-robot concept is an innovative solution for last-mile delivery. Our model extends this concept by identifying relevant costs and therefore enabling total cost evaluation. Further, we include a setting with practical relevance where the available robot fleet is limited. A specialized heuristic is presented to address this problem based on problem-specific construction heuristics, search operators for truck tours and an MIP to find optimal robot schedules for given truck tours and robot availability. Additionally, we introduce a heuristic alternative for the robot scheduling step that further decreases runtimes. In numerical experiments we analyze the main characteristics of the truck-and-robot concept and their impact on total costs. We show the efficiency of our approach via comparison with a benchmark method and highlight the need for a total cost perspective. Further, we compare the truck-and-robot approach to classic delivery by trucks. In summary, the findings from the experiments can provide guidance for planning and operating a truck-and-robot system. A truck-and-robot system can reduce costs by up to two-thirds compared to classic truck delivery and at the same time reduce the truck fleet required. The savings potential depends mostly on the truck driver labor costs and the robot purchase prices. However, even for significantly higher robot costs in the early stage of the innovation cycle we show that (i) the concept is attractive for companies, as total costs are below the cost of conventional truck-only delivery, (ii) high level of on-time deliveries is not opposed to cost-optimal routing, and (iii) truck distance and thus local emissions are reduced by more than 60% as a by-product (without incurring additional costs). For a given delivery area, a certain minimum number of robots is required to ensure high service quality. Beyond that number, improvements due to additional robots are small. Depot density has little impact on cost and emissions but moderate impact on service quality.

There are numerous opportunities for future research. To begin with, the approach presented can help evaluate the effects of urban planning decisions on a truck-and-robot system. Certain zones could be forbidden for the truck or the robots, for example, while in other zones robots could be allowed to travel at higher speed. Such constraints can be modelled by adapting the corresponding index sets and travel times. Moreover, the robot fleet needed could be further reduced by allowing robots to travel between depots such that robot availability is increased at locations visited by the truck. Robot availability could be further improved by considering the robots' return from customers to depots (with stochastic arrival times). To enable this, decisions on the robot movements must be made simultaneously with the delivery routing decisions. In addition, deliveries without time windows or for bulky goods that require manual delivery can be included in the planning problem. For the heuristic, this means customers are then potential or required stops on the truck route. Another interesting aspect is the allocation of customers to trucks and the start time of the truck, which we assume as given. In practice, these decisions must be made such that they enable our method to find efficient tours. Lastly, solving truck-and-robot problems requires tailored solution approaches and innovative algorithms. There are further approaches for related VRP variants that could be adapted and tested for the truck-and-robot concept.

4 A mixed truck and robot delivery approach

Co-autors: Manuel Ostermeier, Alexander Hübner
Under review by the *European Journal of Operational Research*

Abstract Innovative last-mile logistics solutions are needed to reduce delivery costs, traffic congestion, and pollution in cities. A promising concept in this context are truck-and-robot systems, as they enable significant cost and traffic reduction compared to classic truck deliveries. The system relies on small autonomous delivery robots to cover the last meters to a customer. Existing truck-and-robot concepts to date consider home deliveries by robots, while trucks are only used to transport robots and not for deliveries. This assumption disregards the fact that regular truck deliveries are still needed for some delivery requests, such as for the delivery of bulky items, or for customers who do not accept robots. Our research addresses this issue and proposes a mixed truck and robot delivery concept in which both robots and the delivery truck can visit customers. Our tailored solution approach is based on a General Variable Neighborhood Search that efficiently solves the routing problem and outperforms existing truck-and-robot routing algorithms. The numerical experiments show that this approach enables cost reductions of up to 43% compared to classical truck deliveries and up to 22% compared to a truck-and-robot system that does not allow deliveries by both truck and robots on the same tour. Further analyses reveal additional benefits of such mixed tours and the robustness of our approach for different problem settings.

4.1 Introduction

Traffic congestion and pollution are growing problems in cities around the world. Home deliveries are contributing to this problem due to the increasing volume of online orders (Ishfaq et al., 2016; Wollenburg et al., 2018; Allen et al., 2018), particularly as usual deliveries are still conducted by a diesel truck. New concepts are needed to enable the projected growth of delivery volumes and prevent urban traffic from collapsing (Agatz et al., 2008; Orenstein et al., 2019; Hübner et al., 2019). While attended home deliveries are convenient for customers, they account for a large share of logistics costs (Kuhn and Sternbeck, 2013; Hübner et al., 2016a). The complexity of planning deliveries is growing with access restrictions in inner cities (e.g., diesel suspensions) and the growing application of time windows of attended home deliveries. This increases customer service and reduces the number of failed deliveries, i.e., deliveries that are not accepted as customers are not at home. In addition, the COVID-19 pandemic has not only increased the home deliveries, but also created consumer preferences for deliveries without human interaction and challenged companies to protect their workforce.

Delivery by truck and robots is a promising approach to address these issues as well as to flexibly accommodate customers' time window preferences. Autonomous delivery robots (e.g., by Starship (2019) and Marble (2019)) can transport a single parcel or grocery bag to customers. They are designed to travel short distances at pedestrian speed. Due to their lower speed and limited range, delivery robots are combined with specialized trucks to enable a fast and efficient delivery process. This means that a truck transports the corresponding goods for delivery together with robots and releases the robots at dedicated drop-off locations for the actual home delivery. Daimler (2019) has tested such a concept and has shown that it potentially decreases lead time and traffic. Baum et al. (2019) predict that delivery robots will likely be introduced on a larger scale soon due to their low production costs and limited legal obstacles. Recent routing literature shows the suitability and cost efficiency of the combination of trucks and robots and provides methods for cost-optimal routing (Boysen et al., 2018b; Ostermeier et al., 2021a).

Existing truck-and-robot (TnR) concepts exclusively consider robots for final delivery to customers. In practice, however, there are multiple reasons for deliveries requiring human interaction and therefore final delivery by a person. First, some customers may be unable or unwilling to interact with the robot and to retrieve the goods from it, such as elderly or disabled persons. Second, the delivery of some goods would be forbidden or risky via a robot. This includes valuables, drugs and hazardous substances such as cleansing agents, paint, pesticides, etc. Third, individual orders may be too bulky to fit into the robot compartment. This can be the case with some electronics, household and do-it-yourself products, and even groceries being delivered in bulk. According to Forbes (2019), 10 - 25% of Amazon deliveries could not be handled by aerial drones, whose size restrictions are similar to those of delivery robots. Up to one in four orders must therefore be delivered without the use of robots and completed by conventional delivery by truck and human driver. Moreover, even when an order is suitable for robot delivery, the possibility of choosing between truck or robot increases routing flexibility and may yield cost reductions.

In the related routing approaches for attended home delivery, the prevailing literature deals either with a vehicle routing problem (VRP) for truck delivery (e.g., Toth and Vigo (2001); Laporte (2009)) or a TnR routing problem with delivery by robots (see e.g., Boysen et al. (2018b); Ostermeier et al. (2021a); Bakach et al. (2021b)). This means that only truck or robot deliveries are considered, ignoring requirements and the potential benefits of combining deliveries by robot and truck as described above. A new approach that provides this additional flexibility is therefore needed. We close this gap in literature by proposing the Mixed Truck and Robot (MTR) delivery concept, leading to the Mixed Truck and Robot Routing Problem (MTR-RP). This is a generalization of the TnR routing problem and determines which customers are supplied via truck, which customers are approached via robots, and how these deliveries are integrated into the delivery tour. In this application, the truck not only transports the robots to drop-off locations, but is also deployed for direct customer deliveries. This additional option increases the complexity of routing. As such, we solve the MTR-RP with a variant of General Variable Neighborhood Search (GVNS) that incorporates problem-specific insights into the operators. Furthermore, the MTR-RP is different from truck-and-drone concepts, as first a small number of drones are used during a tour, whereas with MTR, the truck picks up multiple new robots during the tour and second, the drones return to the truck, whereas robots return to a depot.

The delivery concept with robots is innovative and we therefore first outline the detailed problem characteristics based on existing concepts and technology in Section 4.2. Section 4.3 discusses related literature and highlights the differences versus other last-mile delivery concepts. Section 4.4 presents the formal model of the MTR-RP. We detail our GVNS approach in Section 4.5. Section 4.6 presents numerical experiments to compare our approach to existing routing frameworks and to analyze the impact of the additional delivery mode by truck. Section 4.7 summarizes our findings and presents opportunities for future research.

4.2 Problem description

This section outlines how a truck and robots are combined for attended home deliveries with time windows. Section 4.2.1 introduces the related technology, on which the problem is based. We then describe the MTR delivery concept in Section 4.2.2.

4.2.1 Technical properties of robots and customized trucks

Delivery robots navigate autonomously on sidewalks and bike lanes but can be remote controlled in the event of problems. To do so, most models rely on several cameras, map data and GPS. In addition, many robots use lidar, ultrasound and radar. For communication, LTE and WiFi are widely-used, at times also touch displays and speakers (Baum et al., 2019). The sensors enable autonomous driving and help prevent theft or vandalism. Recent studies show that robot technology is ready for industry applications. Starship (2019) reports successful tests in more than 80 cities worldwide, and Jaller et al. (2020) discuss robot models that are already in use in the US and Europe. Baum et al. (2019) count 19 different models, of which the majority have already been tested in the field. According to their overview, most robots operate at pedestrian speed, i.e., at 6 to 8 km/h. The maximum range lies between 6 and 77 km (Jennings and Figliozzi, 2019). The payload varies from one parcel and 10 kg to 20 parcels and 70 kg. When a robot arrives at the delivery destination, customers are notified (e.g., via mobile phone) and can unlock the robot's compartment with a code to retrieve the order (Starship, 2019; Marble, 2019).

Given the relatively low speed of robots, companies such as Daimler (2019) have developed customized trucks to transport them. Otherwise robots would have to drive the complete distance from the warehouse to the customer and back. In large delivery areas, this would imply long travel times, issues with lead times and meeting short-term time windows, and low robot utilization. The trucks transport robots to overcome larger distances (e.g., between the warehouse and city center) and release them at dedicated drop-off locations. This enables the efficient use of delivery robots, especially in urban areas. Trucks typically provide space for around eight robots on their floor and enable autonomous pick-up and drop-off via automatic doors and ramps. A shelf system above the floor can be used to carry goods for delivery. It is only driving the truck and loading robots that remain manual tasks. Figure 4.1 shows a typical truck setup. For robot deliveries, the truck driver enters the front part of the cargo bay, retrieves the goods from the shelf system, loads them into robots, and these then leave the truck via a ramp to the side. Direct deliveries by the driver (i.e., without a robot) can therefore easily be included in this system. These orders could be loaded to the rear of the shelf system, for instance, and when the driver arrives at the customer location, (s)he picks up the order from the back door and walks to the customer.

4.2.2 Concept of mixed truck and robot deliveries (MTR)

Conventional TnR In line with Boysen et al. (2018b) and Ostermeier et al. (2021a), TnR is a system in which the delivery robots are transported by truck and therefore the times and locations of both vehicle types are coupled. The central element of this concept is that robots are carried by truck and dropped off close to customers

Figure 4.1: Specialized truck with freight containers and delivery robots (Mercedes-Benz Vans, 2016)

(see Figure 4.2). The distribution process therefore consists of a truck tour, visiting different robot drop-off locations (i.e., a location where the truck can safely stop and release robots onto the sidewalk, see solid arrows in Figure 4.2), and robot tours visiting a single customer each (dotted arrows in Figure 4.2). Some of these drop-off locations are so-called robot depots, where robots are stored and charged. Trucks can both pick up robots at robot depots for later drop-off or load and release robots directly for delivery without transporting them. The number of available robots per depot is limited. Each robot returns to a nearby robot depot after it has delivered its parcel (not displayed in Figure 4.2 for sake of readability). At the depot (which consists only of an outdoor charging station and parking space), it is again charged and waits for the next delivery. Other drop-off points are spots where trucks can stop and release robots for delivery, but no robots are stored. This concept reduces the truck mileage and increases the driver's productivity, which makes it attractive from a cost and environmental perspective (Boysen et al., 2018b; Ostermeier et al., 2021a).

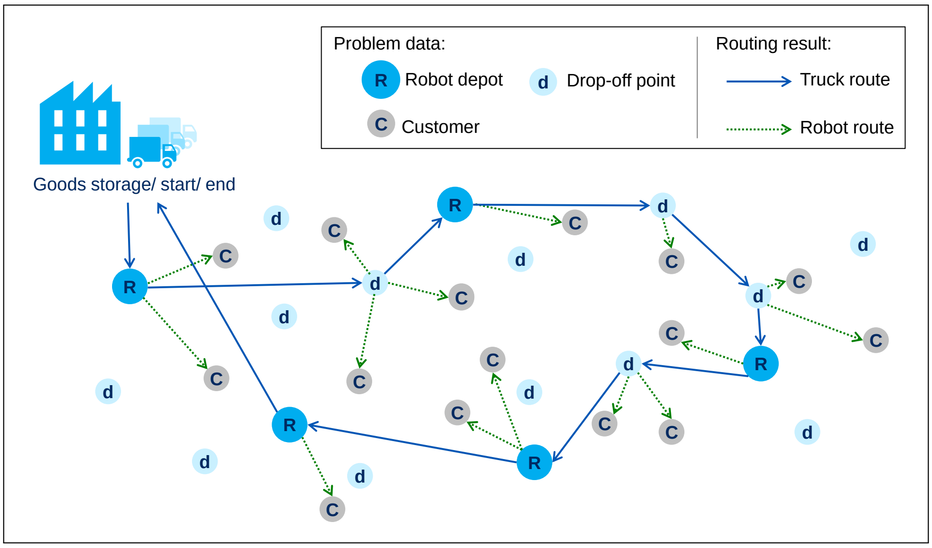

Figure 4.2: TnR tour (with all deliveries by robots)

MTR concept In the conventional TnR concept described above, the truck acts solely as a taxi for robots and does not deliver parcels directly to customers. However, some deliveries are not suitable for robot delivery and must be made by a delivery person. This is necessary for bulky goods that do not fit into the robot's compartment, and goods that must be handed over personally, such as valuables and drugs. A customer could also choose not to receive robot deliveries based on personal preferences or skills. In these cases, a delivery truck has to visit the customer within the respective time window. This can be done by a separate delivery tour (as in prevailing truck-only concepts) or by employing the truck used for robot drop-off to directly approach those customers (as shown in Figure 4.3). Using one truck for both delivery modes has the potential to reduce the fleet needed and the costs and emissions caused for serving a set of customers. Besides customers requiring truck delivery, there are customers who can be visited by either truck or robot. Visiting those customers by truck can in some cases further decrease costs as it may lead to shorter tours or reduce robot use and delays. Note that when the truck stops at a customer, it can launch robots to other customers from there as well. As a consequence, we extend the existing TnR concept to account for both delivery types.

The stops for truck delivery have to be integrated into the truck routes for dropping off robots (see solid arrows in Figure 4.3). This complicates the search for optimal truck tours, since truck deliveries also have to take place within the designated time windows. Early arrivals at customer locations cause waiting times for the truck and late arrivals cause delay costs in the form of reduced future revenues (due to lower customer satisfaction) or the granting of rebates. The admission of additional truck deliveries therefore causes new dependencies and increases the problem complexity.

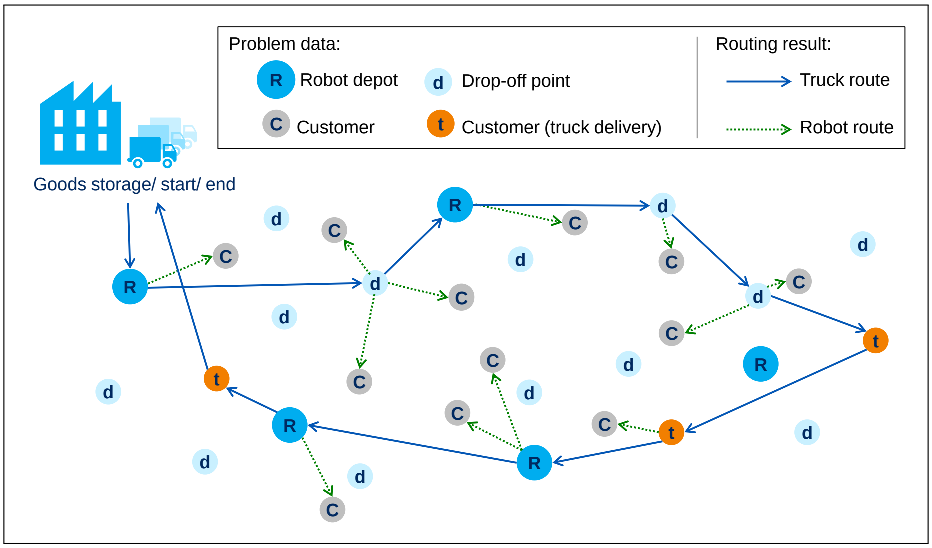

Figure 4.3: MTR tour (incl. deliveries by truck)

Decision problem structure MTR routing requires simultaneous decisions on different routing problem aspects. To illustrate this, Figure 4.4 shows the different vehicles' actions in a truck-and-robot tour over time. For the truck, it includes driving between the goods warehouse, robot depots and drop-off points and customers, as well as potential waiting time at customers. For the robot, it comprises travel time between drop-off points, customers and depots, and potential waiting time. For the truck, there is a mileage-based cost (mainly for fuel) and a time-based cost (for the driver's salary). These have to be considered separately since the truck might have to wait if it reaches a customer before the time window (see diamond in the truck lane of Figure 4.4). The robots start from a depot or drop-off point visited by the truck, drive to a customer and must also wait for the time window in the event of early arrival (see Robot 1 in Figure 4.4). After the delivery, the robots return to the closest depot. A time-based robot fee applies during this entire time. If an order arrives late (see Robot 3 in Figure 4.4), a delay cost is incurred, consisting of a rebate granted to the customer or that accounts for penalties for reduced customer satisfaction. A feasible solution must ensure all customers are served after the start of their respective time window by truck or robot, depending on the request. The decision problem at hand aims to minimize total delivery costs. To achieve this, it is necessary to define (i) which customers are served via truck, which via robot, (ii) which robot depot and drop-off locations are visited during the truck tour, (iii) in which sequence these locations are visited, and (iv) from which stop on the tour each robot delivery is started. The truck starts and ends at the goods warehouse, whereas a robot starts from either a depot or a drop-off location and, after meeting the customer, returns to the closest depot. Besides required travel times and synchronization of truck and robot actions, the decision is constrained by the number of robots available on the truck and in each robot depot.

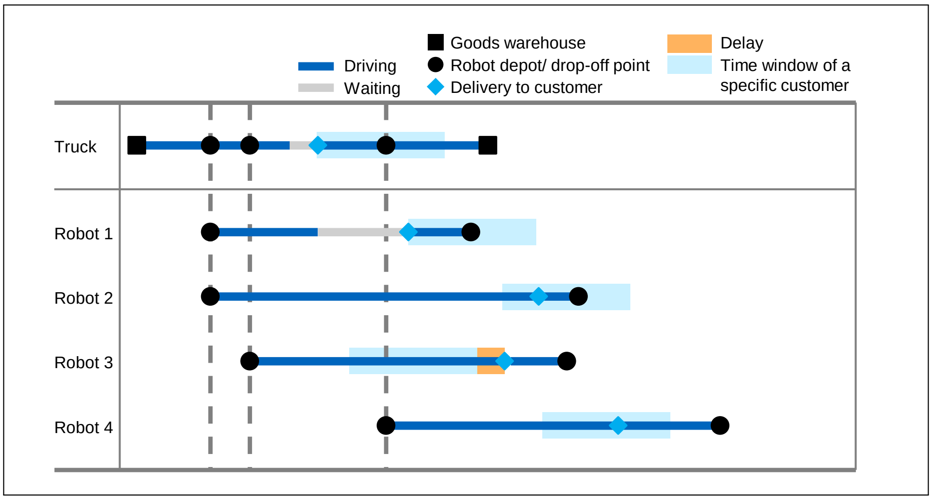

Figure 4.4: Gantt chart of an MTR tour (example)

4.3 Review of related literature

This section provides an overview of related routing approaches for robot-based deliveries. We first highlight the similarities and differences of related concepts, namely truck-and-drone delivery and delivery with covering options. These concepts share the idea of two vehicle types making deliveries together. Next, we provide a summary of robot routing literature, separated into hub-and-robot and TnR concepts. We conclude by highlighting the gap in existing literature.

(i) Truck-and-drone delivery Truck delivery supported by drones has received a lot of attention in recent publications (e.g., Ulmer and Thomas (2018), Sacramento et al. (2019), Agatz et al. (2018)). A truck visits customers to make deliveries and a drone serves other customers not visited by the truck. Initially the truck transports the drone. While the truck stops to make a delivery, the drone can start with a parcel, serve one customer and meet the truck again at a later customer on the truck route. This can be repeated several times. Since every drone delivery starts at a customer served by the truck, the highest possible share of drone deliveries is 50% (Murray and Chu, 2015; Agatz et al., 2018; Ha et al., 2018; de Freitas and Penna, 2020). Even for an extended scenario with up to four drones on the truck, solved by Murray and Raj (2020), the share of truck deliveries must remain above 20%. The authors further note that adding drones leads to diminishing marginal improvements, since too many drones cause long take-off and landing queues at the truck. A key difference between drone concepts and the robot concept considered is therefore the lower number of autonomous vehicles (drones), and their return to the truck instead of dedicated depots. The MTR concept has a higher potential to reduce truck mileage as a truck can launch multiple robots at each stop. Furthermore, the truck picks up further robots during the tour from robot depots, whereas the pertinent applications in truck-and-drone routing rely on a given number of drones on the truck. A further difference is that the truck stops in the MTR concept are optional stops at depots, drop-off points, and further customer locations. Routing approaches for truck-and-drone are as such not directly applicable to MTR since they rely on the fact that many customers need to be visited by truck and the truck does not have other (optional) locations to visit. Pertinent heuristics improve the solution of the traveling salesman problem (TSP) by reassigning customers to the drone (Murray and Chu, 2015; Agatz et al., 2018; Ha et al., 2018; de Freitas and Penna, 2020; Kitjacharoenchai et al., 2019; Sacramento et al., 2019; Murray and Raj, 2020). A detailed analysis of the differences between truck-and-drone and TnR is performed by Ostermeier et al. (2021a). Alongside these differences, practical advantages of robots are their high safety level, robustness in any weather conditions and fewer regulatory obstacles due to slow driving instead of flying. These strengths could soon enable the large-scale practical application of delivery robots in cities (Baum et al., 2019). In summary, delivery robots and drones are used in different setups (based on their strengths) and problem specifics. We refer to Otto et al. (2018) and Macrina et al. (2020) for a detailed overview of the truck-and-drone concept and its challenges.

(ii) Delivery with covering options Enthoven et al. (2020) introduce the two-echelon vehicle routing problem with covering options (2E-VRP-CO). In this last-mile delivery application, the truck on the first echelon can either deliver a parcel to a satellite location, from where cargo bikes bring it to the customers, or to a covering location (i.e., a parcel locker) from which nearby customers can pick up the parcel. Similar to the MTR-RP, the truck only needs to visit a subset of given potential locations, and the delivery type which makes the last mile has to be defined. The proposed solution approach relies on an Adaptive Large Neighborhood Search (ALNS) with tailored operators. Several aspects of our MTR-RP are more complex, however, despite the similarities. First, robots can only be applied to attended home delivery and thus have to meet time windows. Second, the robots move aboard the truck, which is not the case in a two-echelon setup. In the two-echelon case, each potential truck stop has a fixed number of bikes available and there are only a few of these stops. Finally, both vehicle types of the MTR-RP can visit customers, whereas in the 2E-VRP-CO this is only possible for cargo bikes. These differences add dependencies to the truck schedule, as robots can only launch from a location while the truck is present and the truck must meet the customer's time window. Similarly, other two-echelon models fall short of characteristics required in the MTR-RP.

(iii) Hub-and-robot The first concepts developed involving robots can be described as hub-and-robot concepts. Their principle is that robots move between a fixed hub and customers. They do so independently of other means of transportation. Consequently, hubs have the ability to store goods and load the robots, which requires a more sophisticated infrastructure compared to the robot depots (i.e., charging stations) in the TnR or MTR case. Bakach et al. (2021b) propose a mixed integer program (MIP) to allocate customers to hubs and robots. Their objective is to minimize the number of hubs and robot mileage required, while respecting the robots' maximum range. Poeting et al. (2019b) and Poeting et al. (2019a) optimally solve an MIP for truck tours visiting hubs and customers and a schedule of pendulum robot tours from these hubs to customers. Sonneberg et al. (2019) minimize the costs of tours for robots with several compartments applying an MIP. Due to their nature, hub-and-robot systems do not consider mixed delivery but only robot deliveries paired with an existing hub infrastructure.

(iv) TnR The MTR-RP originates from TnR systems. These concepts constitute a more complex routing problem than the hub-and-robot concept due to the coupling of truck and robot movements. To date, three publications explicitly deal with TnR routing. In the seminal paper, Boysen et al. (2018b) introduce the idea of robot depots to eliminate truck waiting time and aim to minimize the number of delayed deliveries. The system analyzed consists of 40 customers and several depots and drop-off points. They solve the problem with a multi-start local search (LS) procedure and show that a TnR system with one truck can replace several traditional delivery vehicles while maintaining service quality. The authors do not incorporate truck deliveries in their approach nor do they provide a quantification of financial and environmental benefits. Some simplifications are assumed (e.g., unlimited robot availability at every depot). Alfandari et al. (2019) build on this work by analyzing alternative delay measures and proposing a Branch-and-Benders-cut scheme for faster computation. Ostermeier et al. (2021a) have extended the problem to account for limitations in robot availability at every depot and minimize total logistics costs, including both truck- and robot-specific costs. Again, the problem is restricted to robot delivery only, while direct truck deliveries are not considered. The authors propose a local search to deal with the increased complexity. In their experiments the concept reduces costs by up to 68% and truck mileage by up to 82% compared to classical truck delivery.

Furthermore, Simoni et al. (2020) propose a delivery mode similar to truck-and-drone, in which a robot leaves the truck at a customer location, makes one or two deliveries and meets the truck again at a later customer on the truck route. Accordingly, their solution approach relies on finding good TSP tours within a local search with adaptive perturbation and then optimally inserting robot tours with dynamic programming. Due to the limited speed of robots, a large share of customers is still served by truck and the reported savings potential of around 20% is lower than savings achieved by the above TnR variants. Jennings and Figliozzi (2019) and similarly Figliozzi and Jennings (2020), assess a TnR system based on continuous approximation and conclude that it has the potential to reduce truck mileage. They do not solve a specific routing problem, but estimate the system's performance based on average distances and speeds.

			Aspects considered in modeling and optimization				
Publication	Objective	Methodology	Delays	Robot availab.	Costs	Truck delivery	Truck/robot selection
Boysen et al. (2018b)	Number of late deliveries	Local search	✓	-	-	-	-
Alfandari et al. (2019)	3 different delay measures	Branch-and-Benders-cut	✓	-	-	-	-
Ostermeier et al. (2021a)	Total costs	Local search	✓	✓	✓	-	-
This paper	Total costs	GVNS	✓	✓	✓	✓	✓

✓: considered, -: not considered

Table 4.1: Summary of existing TnR routing literature

Research gap In summary, the MTR concept leads to a routing problem that requires problem-tailored solution approaches. Approaches for the concepts mentioned in paragraphs (i) to (iv) do not yet include the necessary specifics of the MTR-RP, in particular time windows, a large fleet of smaller vehicles transported by truck and a selection of alternative delivery modes to the customer. For a more detailed review of last-mile delivery concepts we refer to Boysen et al. (2021).

There are only three publications on TnR routing and none of them enables mixed truck and robot deliveries (see Table 4.1). All publications dealing with this innovative last-mile delivery concept focus on robot deliveries, while the truck does not visit customers directly, but only stops at given drop-off locations. However, in a practical application the combination of both delivery modes is needed to ensure that all types of orders can be processed on the same truck tour to reduce costs. We therefore extend the existing literature by addressing the MTR-RP, in which truck deliveries are incorporated when required and a decision between truck and robot delivery is made if both modes are feasible. The corresponding decision model is presented in the next section.

4.4 Formulation of the MTR-RP

This section introduces the mathematical formulation of the MTR-RP. The notation used is summarized in Table 4.2.

Index sets	
C	Set of all customers $k \in C$
C^{m} (C^{r})	Subset of customers requiring truck (robot) delivery, with $C^{\mathrm{m}} \cup C^{\mathrm{r}} \subseteq C$
C^{o}	Subset of customers indifferent regarding truck or robot delivery, with $C^{\mathrm{o}} \subseteq C$
D (R)	Set of distinct robot drop-off points (robot depots)
$\hat{D}$ ($\hat{R}$)	Set of robot drop-off points (robot depots) including duplicates
$\hat{L}$	Set of all (duplicate) locations reachable by truck: $\hat{L} := C^{\mathrm{m}} \cup C^{\mathrm{o}} \cup \hat{D} \cup \hat{R}$
I_a	Set of duplicate indices $i, i \in \hat{D} \cup \hat{R}$, of one distinct location $a, a \in D \cup R$
I_a^m	Set of elements $i \in I_a$ with $i \leq m$
Problem parameters	
d_k	Deadline for customer $k, k \in C$
K	Maximum robot capacity of a truck
r_a	Initial amount of available robots in location $a, a \in R$
γ ($\bar{\gamma}$)	Start (end) position of the truck, with $\gamma, \bar{\gamma} \notin \hat{L}$
δ	Initial number of robots aboard the truck
ϵ_k	Length of time window of customer k
$\lambda_{i,j}$	Distance between locations i and $j, i, j \in \hat{L}$
$\vartheta^{\mathrm{t}}_{i,j}$	Truck travel time from location i to location $j, i, j \in \hat{L}$
$\vartheta^{\mathrm{r}}_{i,k}$	Robot travel time from location $i, i \in \hat{L}$, to customer $k, k \in C$
$\vartheta^{\mathrm{b}}_{k}$	Robot travel time from customer k back to the closest robot depot
Cost parameters	
c^{l}	Cost of delay per time unit
c^{d}	Cost of truck per distance unit
c^{t} (c^{r})	Cost of truck (robot) per time unit
Decision variables	
$s_{i,j}$	Binary: 1, if truck travels from location i to location j; 0 otherwise
$x_{i,k}$	Binary: 1, if customer k is supplied by a robot from location i; 0 otherwise
Auxiliary variables	
t_i	Arrival time of truck at location i

Continued on next page

Table 4.2 – *Continued from previous page*

q_i	Number of robots aboard the truck after visiting location i
e_i	Number of robots taken out of depot location $i, i \in \hat{R}$
v_k	Delay of delivery for customer k
w_k	Waiting time for robot at customer k

Table 4.2: Notation of the MTR-RP

The following sets form the basis of the MTR-RP. The set of customers C consists of three disjointed subsets: customers with mandatory truck delivery C^{m}, customers requiring robot delivery C^{r}, and customers for which the delivery mode is optional C^{o} (i.e., both truck and robot delivery are possible), with $C = C^{\text{m}} \cup C^{\text{r}} \cup C^{\text{o}}$. Every customer $k \in C^{\text{r}} \cup C^{\text{o}}$ can be served by one robot, every customer $k \in C^{\text{m}} \cup C^{\text{o}}$ by the truck. The truck-and-robot infrastructure consists of a set of robot drop-off locations D, where the truck can start robots, and a set of robot depots R, where the truck can pick up and start robots. We further duplicate drop-off and depot locations to allow multiple visits of the same depot or drop-off point. This results in the duplicate sets $\hat{D}$ and $\hat{R}$. For clarity, we summarize all (duplicate) locations that can be visited by truck in $\hat{L} := C^{\text{m}} \cup C^{\text{o}} \cup \hat{D} \cup \hat{R}$. For every distinct location $a, a \in D \cup R$, we denote the set of its duplicates as $I_a, I_a \subset \hat{L}$, and the set of indices in I_a that are less or equal to $m, m \in I_a$, as I_a^m. The set I_a^m is required to keep track of the order in which duplicates are visited and to enforce the constraint on available robots after every visit.

The truck starts in γ (e.g., a goods warehouse, $\gamma \notin \hat{L}$) with δ robots on board and has a maximum capacity of K robots ($\delta \leq K$). It is already loaded with the goods to be delivered. In every robot depot $a, a \in R$, there are an initial number of robots r_a available. Every customer $k, k \in C$, has a delivery time window defined by a deadline d_k and the time window length ϵ_k. The delivery cannot take place before the customers' time window starts (i.e., not before $d_k - \epsilon_k$). In this case truck or robot waiting time applies. If it occurs after the deadline, delay costs at the rate of c^{l} are incurred. The distance between locations i and j is denoted by $\lambda_{i,j}$, the resulting travel times by $\vartheta^{\text{t}}_{i,j}$ for the truck and $\vartheta^{\text{r}}_{i,k}$ for the robots. We further denote the robot travel time from customer k, back to the closest depot as ϑ^{b}_k. Note that the costs of the robots' return to the closest depot is a parameter for each customer supplied by robot as the closest depot is known in advance. Any processing time for loading and unloading is added to these times. We introduce the dummy end location $\bar{\gamma}$ (typically equal to the starting location, $\bar{\gamma} \notin \hat{L}$) to track total truck time. This is necessary since the truck may have to wait to meet a time window for delivery. The total truck time that is needed to assess truck usage costs is thus the arrival time at the end node $\bar{\gamma}$, indicated by $t_{\bar{\gamma}}$. The time-based cost rate of the truck is denoted as c^{t} and the distance-based cost rate c^{d}. A time-based machine rate c^{r} is assumed for the use of robots. It is incurred while loading the robot, its travel to the customer, waiting for the beginning of the time window (if necessary), unloading by the customer, and the return to the closest depot.

In the course of minimizing total costs, we further define the following decision variables. The binary variable $s_{i,j}$ indicates whether the truck travels from location i to location j or not. The binary variable $x_{i,k}$ defines whether customer k is supplied by robot from location i, i.e., whether a robot travels from i to k. To track feasibility and costs of a solution, the following auxiliary decision variables are needed. The variable t_i defines the arrival time of the truck at location $i, i \in \hat{L}$, and q_i the quantity of robots aboard the truck when leaving the location. The quantity of robots taken out of depot $i, i \in \hat{R}$ (i.e., loaded on the truck or directly started towards a customer) is defined by e_i. For every customer k, v_k indicates the duration of delay (in the event of late arrival) and w_k the robot waiting time (in the event of early arrival). We then formulate the MTR-RP as follows.

$$\min TC = c^{\mathrm{t}} t_{\bar{\gamma}} + \sum_{i \in \hat{L} \cup \{\gamma\}} \sum_{j \in \hat{L} \cup \{\bar{\gamma}\}} c^{\mathrm{d}} \lambda_{i,j} s_{i,j} + \sum_{i \in \hat{L}} \sum_{k \in C^{\mathrm{r}} \cup C^{\mathrm{o}}} c^{\mathrm{r}} (\vartheta^{\mathrm{r}}_{i,k} + \vartheta^{\mathrm{b}}_{k}) x_{i,k} + \sum_{k \in C} (c^{\mathrm{l}} v_k + c^{\mathrm{r}} w_k) \quad (4.1)$$

subject to

$$\sum_{i\in\hat{L}} x_{i,k} + \sum_{i\in\hat{L}\cup\{\gamma\}} s_{i,k} = 1 \qquad \forall k \in C^{\text{o}} \cup C^{\text{m}} \tag{4.2}$$

$$\sum_{i\in\hat{L}} x_{i,k} = 1 \qquad \forall k \in C^{\text{r}} \tag{4.3}$$

$$\sum_{k\in C} x_{j,k} \leq M \sum_{i\in\hat{L}\cup\{\gamma\}} s_{i,j} \qquad \forall j \in \hat{L} \tag{4.4}$$

$$\sum_{j\in\hat{L}} s_{\gamma,j} \leq 1 \tag{4.5}$$

$$\sum_{i\in\hat{L}\cup\{\gamma\}} s_{i,j} = \sum_{i\in\hat{L}\cup\{\bar{\gamma}\}} s_{j,i} \qquad \forall j \in \hat{L} \tag{4.6}$$

$$t_\gamma = 0 \tag{4.7}$$

$$t_j \geq t_i + \vartheta^{\text{t}}_{i,j} - M(1 - s_{i,j}) \qquad \forall j \in \hat{L} \cup \{\bar{\gamma}\}; i \in \hat{L} \cup \{\gamma\} \tag{4.8}$$

$$t_k \geq d_k - \epsilon \qquad \forall k \in C^{\text{m}} \tag{4.9}$$

$$t_k \geq d_k - \epsilon - M(1 - \sum_{i\in\hat{L}\cup\{\gamma\}} s_{i,k}) \qquad \forall k \in C^{\text{o}} \tag{4.10}$$

$$q_\gamma = \delta \tag{4.11}$$

$$q_j \leq q_i + e_j - \sum_{k\in C} x_{j,k} + M(1 - s_{i,j}) \qquad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{R} \tag{4.12}$$

$$q_j \leq q_i - \sum_{k\in C} x_{j,k} + M(1 - s_{i,j}) \qquad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{D} \cup C^{\text{m}} \cup C^{\text{o}} \tag{4.13}$$

$$v_k \geq t_k - d_k \qquad \forall k \in C^{\text{m}} \cup C^{\text{o}} \tag{4.14}$$

$$v_k \geq t_j + \vartheta^{\text{r}}_{j,k} - d_k - M(1 - x_{j,k}) \qquad \forall k \in C^{\text{r}} \cup C^{\text{o}}, j \in \hat{L} \tag{4.15}$$

$$w_k \geq (d_k - \epsilon) - t_j - \vartheta^{r}_{j,k} - M(1 - x_{j,k}) \qquad \forall k \in C^{\text{r}} \cup C^{\text{o}}, j \in \hat{L} \tag{4.16}$$

$$t_i \leq t_j \qquad \forall a \in R; i, j \in I_a : i \leq j \tag{4.17}$$

$$\sum_{h\in\hat{L}\cup\{\gamma\}} s_{h,i} \geq \sum_{h\in\hat{L}\cup\{\gamma\}} s_{h,j} \qquad \forall a \in R; i, j \in I_a : i \leq j \tag{4.18}$$

$$r_a - \sum_{i\in I^m_a} e_i \geq 0 \qquad \forall a \in R; m \in I_a \tag{4.19}$$

$$s_{i,j} \in \{0, 1\} \qquad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{L} \cup \{\bar{\gamma}\} : i \neq j \tag{4.20}$$

$$s_{i,i} = 0 \qquad \forall i \in \hat{L} \tag{4.21}$$

$$x_{i,k} \in \{0, 1\} \qquad \forall i \in \hat{L}; k \in C^{\text{r}} \cup C^{\text{o}} \tag{4.22}$$

$$x_{i,k} = 0 \qquad \forall i \in \hat{L}; k \in C^{\text{m}} \tag{4.23}$$

$$e_i \in \mathbb{Z} \qquad \forall i \in \hat{R} \quad (4.24)$$

$$t_i \geq 0 \qquad \forall i \in \hat{L} \cup \{\bar{\gamma}\} \quad (4.25)$$

$$q_i \in \{0, ..., K\} \qquad \forall i \in \hat{L} \quad (4.26)$$

$$v_k, w_k \geq 0 \qquad \forall k \in C \quad (4.27)$$

The objective function (4.1) minimizes total costs. The first term considers the cost of truck time (at cost rate c^{t}). It compromises the total truck time including travel time between locations and potential waiting time if customers are approached too early. The second term covers the truck's distance costs (at cost rate c^{d}). The third term comprises the robot costs dependent on associated travel times to the customer and back to the closest depot (at cost rate c^{r}). The last term of the objective function sums up the cost of possible delayed deliveries (cost rate c^{l}) and robot waiting times across all customers. Constraint (4.2) ensures exactly one visit by either truck or robot for every customer $k \in C^{\text{o}} \cup C^{\text{m}}$. Similarly, constraint (4.3) ensures that each customer who requires a robot delivery is visited by exactly one robot. Constraint (4.4) states that robots can only be launched from stops that are actually visited by truck. Constraint (4.5) defines that the truck only leaves once from the starting point, and (4.6) ensures that if the truck reaches a location, it must also leave it. Constraints (4.7) and (4.8) determine the truck arrival time at every stop based on travel times. This also prevents a second visit to the same (duplicate) stop. Constraint (4.9) ensures that a required truck delivery is not made before the respective time window and (4.10) does so for optional truck deliveries in case they are made by truck (and not by robot). The following constraints (4.11), (4.12) and (4.13) handle the number of robots aboard the truck when leaving the starting point, a depot or any other location, respectively. Constraint (4.14) defines the delay for customers receiving truck delivery. Constraints (4.15) and (4.16) define the delay and waiting time for customers receiving robot deliveries. Constraints (4.17) and (4.18) ensure without loss of generality that duplicates of the same location are visited in ascending order of their index. This fact is then used by constraint (4.19) to track the robot stock in every depot and to ensure that the stock is ≥ 0. Finally, the variable domains are defined by constraints (4.20) to (4.27).

The MTR-RP extends the classical TnR problem, i.e., without truck deliveries, in several ways: Some customers must be served by truck, others can be. This means that the total number of robots started (tracked by (4.11), (4.12) and (4.13)) is not predetermined but part of the decision problem. Moreover, total truck time is no longer based merely on the legs $s_{i,j}$ traveled since the truck may have to wait for the beginning of a time window ((4.9) and (4.10)). We need to determine the usage time of a truck instead by using the return time to the warehouse $t_{\bar{\gamma}}$, and add the term $t_{\bar{\gamma}}c^{\mathrm{t}}$ to the objective function. Since the optimal $t_{\bar{\gamma}}$ is determined via the recursive constraints (4.8), (4.9) and (4.10), this is computationally expensive even for small instances.

4.5 Solution approach

The MTR-RP generalizes the NP-hard TnR routing problem and therefore constitutes an NP-hard optimization problem by itself (see Boysen et al. (2018b)). Since even small instances cannot be solved exactly, we propose a tailored solution approach, denoted as *MTR heuristic*, that is based on a GVNS framework (see Mladenović and Hansen (1997); Hansen and Mladenović (2001)). VNS formulations have been used successfully for many variants of routing problems (e.g., Kovacs et al. (2014a); Henke et al. (2015); Ostermeier et al. (2020)) as they provide a high degree of flexibility and can be tailored to the given problem specifics. The key benefit of GVNS for this application (compared to the local search previously used for TnR, e.g., in Boysen et al. (2018b) and Ostermeier et al. (2021a)) is that complete neighborhoods are evaluated in a defined order. This is necessary for finding improvements as the objective function is sensitive to small changes in the truck route, which can lead to long waiting times or delays. Furthermore, defining an order of assessed neighborhoods enables us to incorporate problem-specific knowledge, such as truck distance as a key cost driver (Ostermeier et al., 2021a). An overview of our solution framework is shown in Figure 4.5.

We generate an initial truck tour with one of two possible start procedures, depending on the given problem instance (see Section 4.5.1). This truck tour is then evaluated

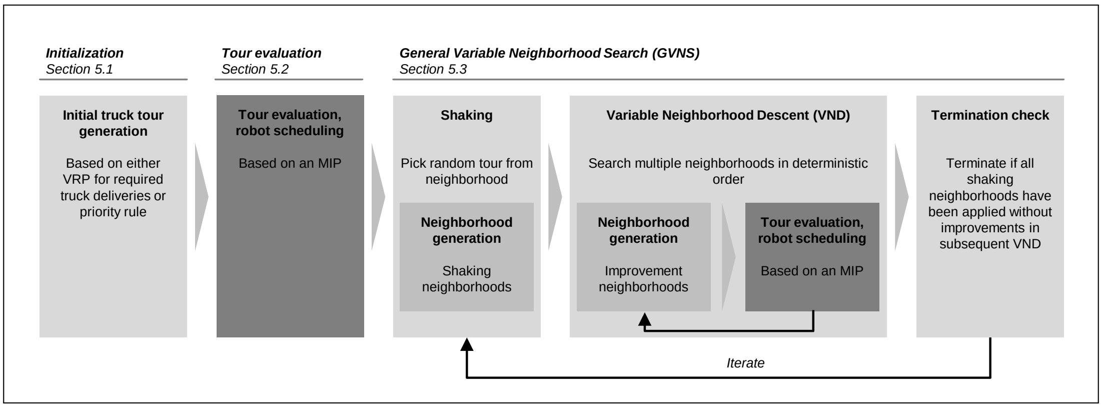

Figure 4.5: Structure of the MTR heuristic proposed

and complemented to a full solution by finding the optimal robot schedule using an MIP (see Section 4.5.2). Next, a GVNS is used to improve truck tours with respect to depots visited, drop-off locations and direct truck deliveries (see Section 4.5.3). It consists of a shaking step and a subsequent Variable Neighborhood Descent (VND). Within the GVNS, tours are again assessed by the robot scheduling MIP from Section 4.5.2.

4.5.1 Initial truck tour generation

There are start heuristics for classical VRPs (i.e., truck delivery only) and TnR routing (i.e., robot delivery only) available in current literature. Our approach combines these two modes and thus chooses between truck and robot delivery based on efficiency. We found in our numerical experiments that above a certain number of mandatory truck deliveries, the order of these deliveries is crucial for solution quality. Below a certain number of truck deliveries, the robot deliveries have a greater impact on the solution and total costs. Leveraging these insights, we propose two alternative principles for generating start solutions, depending on the number of truck deliveries required. They differ in terms of which deliveries are considered and in which order.

Robot deliveries first, truck deliveries second In the event of less than σ mandatory truck deliveries, we generate a tour that includes both robot and truck deliveries in a two-step approach. First, stops at drop-off and depot locations are sequentially appended to the tour based on the *priority rule (PR)* “go to the location from which most robot deliveries can be started such that they reach customers on time”. Truck delivery customers are ignored in the first step. As soon as robot customers are assigned to a stop, they are not considered for later stops. This rule results in a sequence of depot and drop-off points, which could be non-feasible since robot availability is not yet considered. In the second step, the truck deliveries required are inserted sequentially, each customer at the position of the tour where the smallest deviation is caused. We therefore obtain a complete tour consisting of drop-off locations and stops at truck delivery customers.

Only truck deliveries In the event of at least σ truck deliveries, we solve a *VRP with time windows* (see model provided in Appendix B) for truck delivery customers, thus ignoring robot deliveries completely. The corresponding VRP can be solved optimally for small problem sizes, while for larger problem sizes the best solution found within a given time limit τ is used. This results in a truck tour that visits all customers requiring truck delivery, starting from the start location. This route then serves as starting solution for the GVNS. Despite lacking the consideration of robot drop-off locations, this enables us to obtain an efficient basis for the truck routing as the direct truck deliveries are decisive for the final tour, including drop-off and depot locations.

4.5.2 Tour evaluation and robot scheduling

Feasibility of truck tours All solutions obtained (including the start solution) need to be assessed with respect to robot availability to prevent non-feasible tours. A truck tour is only feasible if the total number of available robots (initial number of robots on the truck δ plus all robots at depots visited on the tour $r_a, a \in R$) is equal to or larger than the number of customers not visited by truck (i.e., customers that are not on the truck route). We append the closest unvisited depot to the end of the tour as long as the number of available robots is not sufficient.

Robot scheduling for given truck route Once feasibility is ensured, the corresponding robot movements for the truck route in question must be defined, i.e., all remaining customers must be assigned to a truck stop, from which the corresponding robot will start. This transforms the truck tour into a full solution. We apply an MIP proposed by Boysen et al. (2018b) and enhanced by Ostermeier et al. (2021a) to assign customers to the truck stops on the route. This is necessary to evaluate the quality of a route that has been found. In contrast to the MIP from Section 4.4, which included the decision on truck movements, we do not need duplicates of robot drop-off (D) and depot locations (R). This leads to $L := C^{\mathrm{m}} \cup C^{\mathrm{o}} \cup D \cup R$ being the set of all locations potentially reachable by truck. We assume the truck tour to be given as a tuple Y, where $y(u)$ is the location of the u-th stop, $y(u) \in L$. Note that we exclude all customers who are served by direct truck delivery from the assignment. We denote the set of remaining customers to be served by robot as $\tilde{C}$, with $\tilde{C} \subseteq C^{\mathrm{o}} \cup C^{\mathrm{r}}$. Table 4.3 summarizes the notation of truck tour parameters and decision variables.

Truck tour parameters	
U	Index set of stops on the truck tour $u \in \{1, 2, ...\}$
Y	Tuple of truck stops, where element $y(u)$ is the u-th stop of the truck tour, $y(u) \in L$
$\tilde{C}$	Set of customers not visited by truck (i.e., not in Y)
t_u	Arrival time at truck stop $u, u \in U$
$c^{\mathrm{T}}_{u,k}$	Cost of serving customer $k, k \in \tilde{C}$, from stop $u, u \in U$
Decision and auxiliary variables	
$x_{u,k}$	Binary: 1, if customer $k, k \in \tilde{C}$, is supplied from stop $u, u \in U$; 0 otherwise
q_u	Number of robots aboard the truck at departure from stop $u, u \in U$
$r_{a,u}$	Number of available robots in location $a, a \in L$, after the u-th truck stop

Table 4.3: Additional parameters and variables for robot scheduling

The actual arrival time at each truck stop $t_u, u \in U$ for a given tour Y can be calculated using Equations (4.28)–(4.30). Equation (4.28) states that the truck tour starts at time zero. For drop-off and depot locations, only truck travel times determine the arrival time (Equation (4.29)). For customer locations, the beginning of the respective time window also has to be considered to prevent premature deliveries (Equation (4.30)).

$$t_1 = 0 \tag{4.28}$$

$$t_u = t_{u-1} + \vartheta^t_{y(u),y(u-1)} \qquad \forall u : y(u) \in D \cup R \tag{4.29}$$

$$t_u = max(t_{u-1} + \vartheta^t_{y(u),y(u-1)}, d_{y(u)} - \epsilon_k) \qquad \forall u : y(u) \in C \tag{4.30}$$

Based on arrival times, the total cost $c^{\mathrm{T}}_{u,k}$ of supplying a customer k from stop u is denoted by Equation (4.31). It comprises the robot usage cost (at rate c^{r}) for travel time, waiting time at the customer (in the event the robot arrives before the time window) and the time to return to the closest depot ϑ^{b}_k. Finally, delay costs are added.

$$\begin{aligned} c^{\mathrm{T}}_{u,k} := {} & c^{\mathrm{r}}(\vartheta^r_{y(u),k} + (d_k - \epsilon_k - t_u - \vartheta^r_{y(u),k})^+ + \vartheta^b_k) \\ & + c^{\mathrm{l}}(t_u + \vartheta^r_{y(u),k} - d_k)^+ \qquad \forall u \in U, k \in \tilde{C} \end{aligned} \tag{4.31}$$

The variables $x_{u,k}$, $r_{a,u}$ and q_u define where each customer's robot is started, how many robots are available in each location and on the truck after every stop. The robot scheduling MIP can then be formulated as follows.

$$\min \ F(Q, X, R) = \sum_{u \in U} \sum_{k \in \tilde{C}} x_{u,k} \cdot c^{\mathrm{T}}_{u,k} \tag{4.32}$$

subject to

$$\sum_{u \in U} x_{u,k} = 1 \qquad \forall k \in \tilde{C} \tag{4.33}$$

$$r_{a,u} = r_{a,u-1} \qquad \forall a \in R, u \in U : a \neq y(u) \tag{4.34}$$

$$r_{a,u} \leq r_{a,u-1} + q_{u-1} - q_u - \sum_{k \in \tilde{C}} x_{u,k} \qquad \forall a \in L, u \in U : a = y(u) \tag{4.35}$$

$$q_0 = \delta \tag{4.36}$$

$$r_{a,0} = r_a \qquad \forall a \in R \tag{4.37}$$

$$r_{a,u} = 0 \qquad \forall a \in L \setminus R, u \in U \tag{4.38}$$

$$x_{u,k} \in \{0,1\} \qquad \forall u \in U, k \in \tilde{C} \tag{4.39}$$

$$r_{a,u} \geq 0 \qquad \forall a \in R, u \in U \tag{4.40}$$

$$0 \leq q_u \leq K \qquad \forall u \in U \tag{4.41}$$

The objective function (4.32) minimizes total robot and delay costs. Constraint (4.33) ensures that exactly one robot is sent to each remaining customer. Constraint (4.34) states that if a depot is not visited, the number of available robots remains the same. Constraint (4.35) keeps track of the number of robots in locations visited and aboard the truck after every stop. Equations (4.36) and (4.37) define the initial number of robots in the depots and on the truck. Constraint (4.38) ensures that robots cannot be stored at drop-off locations or customers. Constraints (4.39)–(4.41) define the variable domains.

4.5.3 General Variable Neighborhood Search

For improving the truck tour, we apply a GVNS as described by Hansen and Mladenović (2018), which tries to improve the initial routing solution by exploiting problem-specific knowledge. It conducts several cycles of shaking and subsequent VND. Both the shaking and the VND rely on neighborhoods. These are defined by operators, such that every neighborhood contains all truck tours that can be generated by applying the respective operator to the incumbent truck tour. Algorithm 4.1 summarizes the GVNS applied. The inner while loop constitutes the VND (with its improvement neighborhood k_i), the outer one conducts the shaking (with shaking neighborhood k_s) and stores the best known solution. The parameter α in the for loop determines the number of VND iterations for every shaking neighborhood. To evaluate truck tours, the GVNS repeatedly uses the robot scheduling MIP.

Algorithm 4.1 GVNS procedure (adapted from Hansen and Mladenović (2018))

Input: Starting solution π_s
  $\pi_{best} = \pi_s$; // best solution found
  $k_s = 1$ // shaking neighborhood
  while $k_s \leq$ number of shaking neighborhoods **do**
    improvement = false
    // perform several VND runs with same shaking neighborhood:
    for $j = 1$ to α **do**
      $k_i = 1$
      $\pi_{current}$ = random(shakeneighborhood(π_{best}, k_s)) // shaking neighborhood k_s
      while $k_i \leq$ number of VND neighborhoods **do**
        π_{k_i} = best(improveneighborhood($\pi_{current}, k_i$) // improvement neighborhood k_i
        if Z(π_{k_i}) <Z(π_{best}) **then**
          $\pi_{best} = \pi_{k_i}$
          $k_i = 1$
          improvement = true
        else
          $k_i + = 1$; // next neighborhood
        end if
      end while
      if improvement = true **then**
        $k_s = 1$
        break
      end if
    end for
    $k_s + = 1$
  end whilereturn π_{best}

Shaking The shaking phase of the GVNS is used to diversify the search. Neighborhoods are obtained by varying the truck tour of a previously generated solution and reoptimizing the robot movements. The neighborhoods are applied in the given order, one in each shaking phase, and used to generate α new solutions. For each of these solutions we apply a separate VND in the next step. When a shaking step has led to an improvement, the process restarts from the first neighborhood. The search is complete after all shaking neighborhoods have been used without improvements.

- **Depot insertion.** Inserts a new robot depot into the tour. Since robot availability is crucial for finding an efficient robot schedule, selecting different depots can enable tour improvements.
- **Detour insertion.** This operator inserts a drop-off point or a customer with optional truck delivery into the truck tour that leads to a detour of half the delivery area's side length or above. It is used to diversify the search by causing a large change in the current truck tour.
- **Swap stop.** This operator swaps two random stops (of which each can be a drop-off point, robot depot or customer) of a truck tour. This may again lead to large detours and thus widens the search space.
- **Stop relocation.** This operator shifts a stop to a later or earlier point on the tour.
- **Customer reshuffling.** This operator instigates the most extensive tour change. It first removes all non-customer stops and then reshuffles the visits at customers according to their original arrival times (i.e., before stop removal). This means that every truck delivery customer supplied late is shifted to earlier positions on the tour such that the deadline is met. The resulting new tour is added to the neighborhood for every combination of late customer and possible earlier position on the tour. In the event that this results in more than n^{shuffle} tours, only the option that minimizes the tour distance is added for each late customer. The following VND will then construct a new solution around the reshuffled truck deliveries. This operator makes use of the fact that particularly the order of truck deliveries required defines the solution quality. The complete procedure of customer reshuffling is presented in Algorithm 4.2.

Algorithm 4.2 Customer reshuffling

Input: Current tour Y
save original arrival times t_k of every customer k supplied by truck
remove non-customer stops from Y
for $u = 1$ to u_{max} **do**
$k = y(u)$;
if $t_k > d_k$ **then**
// customer was delivered late
$pool_k = \{\}$
$pool_k$.add(shift(Y, k, 1)) // shift customer to first stop after starting point
for $i = 2$ to $u - 1$ **do**
$\hat{k} = y(i - 1)$
if $t_{\hat{k}} < d_k$ **then**
// time at previous stop is before deadline
$pool_k$.add(shift(Y, k, i)) // shift customer to stop i
end if
end for
end if
end for
if no. of tours in all pools $> n^{\text{shuffle}}$ **then**
$pool_{final} = \bigcup$\{shortest tour from every pool\}
else
$pool_{final} = \bigcup$\{all pools\}
end ifreturn $pool_{final}$

VND The VND is used to improve the truck tour. It relies on multiple neighborhoods of the incumbent solution that are searched sequentially. The VND restarts from the first neighborhood when a better solution is found. This continues until all neighborhoods of the incumbent solution have been searched and no improvement has been found. Each neighborhood contains all tours that can result from applying its operator to the incumbent tour.

- **Remove a non-depot.** Removes a drop-off point or a customer with optional truck delivery from the current truck tour. Since truck distance is a main cost driver, this often leads to improvements. Required truck deliveries cannot be removed in this step.

- **Remove a depot.** Removes a depot from the current truck tour. The removal of a depot may lead to non-feasible solutions. In this case additional depots will be appended within the feasibility check.
- **Add depot.** Adds a new depot to the existing truck tour. Additional depots can increase robot availability on parts of the tour and lead to better robot schedules at reduced costs.
- **Add a non-depot.** Adds a drop-off point to the existing truck tour. This may reduce robot travel times by bringing the truck closer to nearby customers.
- **Swap two stops.** By changing the order of stops, truck distance can be reduced or delays at the later stop can be avoided.
- **Relocate a stop.** This operator primarily aims at improving arrival times at customers. In particular when the truck arrives at a customer too early and is forced to wait for the time window, shifting this customer to a later point of the tour can reduce total time and delays.

The order of improvement neighborhoods ensures that tours are kept short, and that we start with the smallest neighborhoods. This reduces the computational effort by limiting the number and complexity of the robot scheduling MIP (equations (4.32)-(4.41)) that has to be solved to evaluate the tours. Since in neighborhoods "add depot" and "add non-depot", several hundreds of combinations of inserted location and insertion position of the tour exist, neighborhoods are limited to the $n^{\max}$ shortest tours. This again reduces computational effort based on known problem characteristics.

4.6 Numerical examples

This section analyzes the performance of our MTR heuristic. First, we describe the instances and parameters used in our experiments (Section 4.6.1). Next, we compare our approach to a benchmark (Section 4.6.2) to assess the performance of our algorithm. Further experiments assess the impact of both required and optional truck deliveries. We compare different fulfillment concepts for home delivery depending on the share of truck deliveries required (Section 4.6.3) and analyze the impact of time windows on the routing (Section 4.6.4). Finally, we discuss the impact of customer distribution (Section 4.6.5), and cost rates for the truck and delays (Section 4.6.6). Our approach was implemented in Python (using PyCharm 2018.3.5 Professional Edition) with Gurobi (version 8.0.1) as MIP solver and executed on a 64-bit PC with an Intel Core i7-8650U CPU (4 × 1.9 GHz), 16 GB RAM, and Windows 10 Enterprise.

4.6.1 Instance and parameter setting

In our numerical experiments we aim at analyzing the performance of our MTR heuristic in comparison to related approaches. To enable a fair comparison and to evaluate the impact of direct deliveries we leverage the test data provided by Ostermeier et al. (2021a) (`http://www.vrp-rep.org/datasets/item/2020-0005.html`). The data set comprises 160 instances for TnR routing and resembles the general setting of our problem but ignores the possibility of direct truck deliveries. The data setting is as follows. Customer locations are picked randomly from all buildings in a 4 km^2 area in northern Munich (Germany), using OpenStreetMap (OpenStreetMap Foundation, 2019) to create instances with $|C| = 50$ customers. To account for direct truck deliveries, we assume that the first 12% of customers (which the instances list in random order) require truck delivery ($|C^{\mathrm{m}}|/|C|$=0.12). The remaining customers require robot delivery ($C^{\mathrm{r}} = C \setminus C^{\mathrm{m}}$). This means there are no optional truck deliveries in the default case ($C^{\mathrm{o}} = \emptyset$). The impact of optional truck deliveries will be analyzed separately. Note that our assumption for $|C^{\mathrm{m}}|/|C|$ is in line with the estimate reported by Forbes (2019), that technically 75 - 90% of Amazon deliveries could be made by autonomous vehicles, and will be subject to a sensitivity analysis in the following. There are $|R|$=25 evenly distributed robot depots, and $|D|$ = 48 uniform-randomly distributed drop-off points in the area. All delivery time windows have the same length $\epsilon = 10$ min. The end of a customer‘s time window is generated based on the direct travel time of the truck from its random starting position to the customer. This travel time is multiplied by a uniform-randomly distributed factor from the interval $[\rho_{min}, \rho_{max}] = [5, 8]$. This procedure simulates an assignment of customers to vehicles such that reasonable tours are made possible.

The initial number of robots is $r_a = 10$ for every depot $a, a \in R$. The capacity of the truck is $K = 8$ robots and it is fully loaded at the start ($\delta = 8$). The average speed of the truck is 30 km/h and the average speed of the robots 5 km/h. A handling time per truck stop of $\mu = 40$ sec is assumed in addition to travel times. There are 20 instances generated for each setup. All results presented show the average of the corresponding 20 solutions. We further apply the cost rates empirically quantified by Ostermeier et al. (2021a). These are $c^{\text{d}} = 0.20$ €/km and $c^{\text{t}} = 30$ €/h for the truck, $c^{\text{r}} = 0.50$ €/h for robot use and $c^{\text{l}} = 5$ €/h for delivery delays.

Lastly, we allow $\alpha = 4$ VND iterations per shaking neighborhood (executed in parallel), a maximum of $n^{\text{shuffle}} = 4$ for the *customer reshuffling* shaking neighborhood and a maximum VND neighborhood size of $n^{\max} = 90$ tours. The threshold for the selection of the start heuristic is set to $\sigma = 2$ and its time limit τ to 3 minutes.

4.6.2 Performance comparison

There are no existing solution approaches to MTR and only a couple of publications on TnR (see literature analysis). To the best of our knowledge, Boysen et al. (2018b) and Ostermeier et al. (2021a) provide the currently most developed approaches in this research area. As we provide a generalization of the problem, we will use a special case of our problem that is equivalent to the problems in the benchmark. We compare our MTR heuristic to the LS approach by Ostermeier et al. (2021a), as the authors study the TnR concept a with total cost objective, i.e., without the possibility of truck deliveries. Their numerical studies show that their LS approach outperforms the approach by Boysen et al. (2018b) in finding cost-optimal tours. However, due to its structure, the LS is not suitable for incorporating truck deliveries and all customers must be visited by robot. We consequently apply our MTR heuristic to solve both instances without truck deliveries and additionally instances with the restriction of 12% truck deliveries required. In the special case of our setting without truck deliveries, the problem is identical to the one solved by Ostermeier et al. (2021a). The MTR heuristic reaches a solution quality differing only 0.3 - 2.0% from the LS in these cases. The computation times for different problem sizes are shown in Figure 4.6. The scenario with 12% truck deliveries is labeled 'MTR 0.12' and the one without truck deliveries 'MTR 0.0'. We see that the MTR approach outperforms the LS when only robot deliveries are required, reducing the computation time by up to 94% (25 customers). This shows that despite the focus of our MTR approach on a mixed delivery structure, it works efficiently and effectively for a related problem without direct truck deliveries. When truck deliveries are required, the computation effort increases, but remains at a level acceptable for an application in practice.

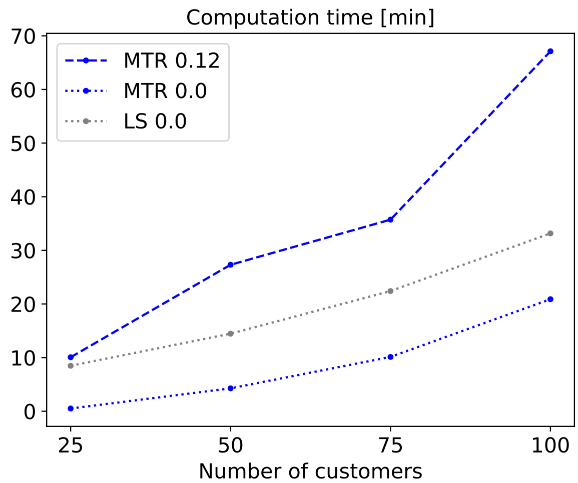

Figure 4.6: Comparison of our MTR approach to the benchmark (i.e., LS approach by Ostermeier et al. (2021a)) for a share of 0% and 12% of truck deliveries

Additionally, we found that the MIP for the entire MTR-RP ((4.1)–(4.27)) could not be solved exactly within three hours for six customers, even if stops are not duplicated, branching is supported by a relaxed MIP version and a feasible start solution is provided to the solver. An average MIP gap of 52% remained.

4.6.3 Comparison of delivery concepts

This section compares the delivery concepts given in Table 4.4 for a varying share of truck deliveries required.

Figure 4.7 shows the total costs, computation times, average delay and total truck distance for the concepts analyzed. We henceforth highlight the default setting

Concept	Description	Rationale	Solution approach
TD: Truck-only delivery	Only deliveries by truck to all customers	Benchmark to assess MTR benefits	MIP from Appendix B
MTR: Mixed truck and robot delivery	Tour with mandatory truck deliveries and all other deliveries by robot ($C^r = C \setminus C^m$)	Approach of this paper	MTR heuristic
MTR OT: MTR with optional truck deliveries	MTR extended by optional truck deliveries, i.e., in addition to mandatory truck deliveries, all other deliveries can be made by truck ($C^o = C \setminus C^m$)	Approach of this paper and assessing optional truck delivery	MTR heuristic
STR: Separate truck and robot tours	Separate planning of one TD tour for truck deliveries and one TnR tour for robot deliveries (i.e., two simultaneous tours)	Serves as benchmark to assess benefits of MTR heuristic vs. existing TnR heuristics	TD tour by MIP from Appendix B; TnR tour by MTR heuristic

Table 4.4: Overview of delivery concepts

described in Section 4.6.1 with a bold x-label. Note that TD was solved without consideration of the earliest delivery time, i.e., delivery can occur before the time window to reduce computational complexity. This leads to an advantage for TD and an underestimation of the improvements due to MTR. Despite this simplification, optimality could not be proved within the computation time limit of three hours. We therefore report properties of the best solutions found and the lower bound of the objective value (‘TD LB‘). Further, MTR and STR are identical for 0% of truck deliveries.

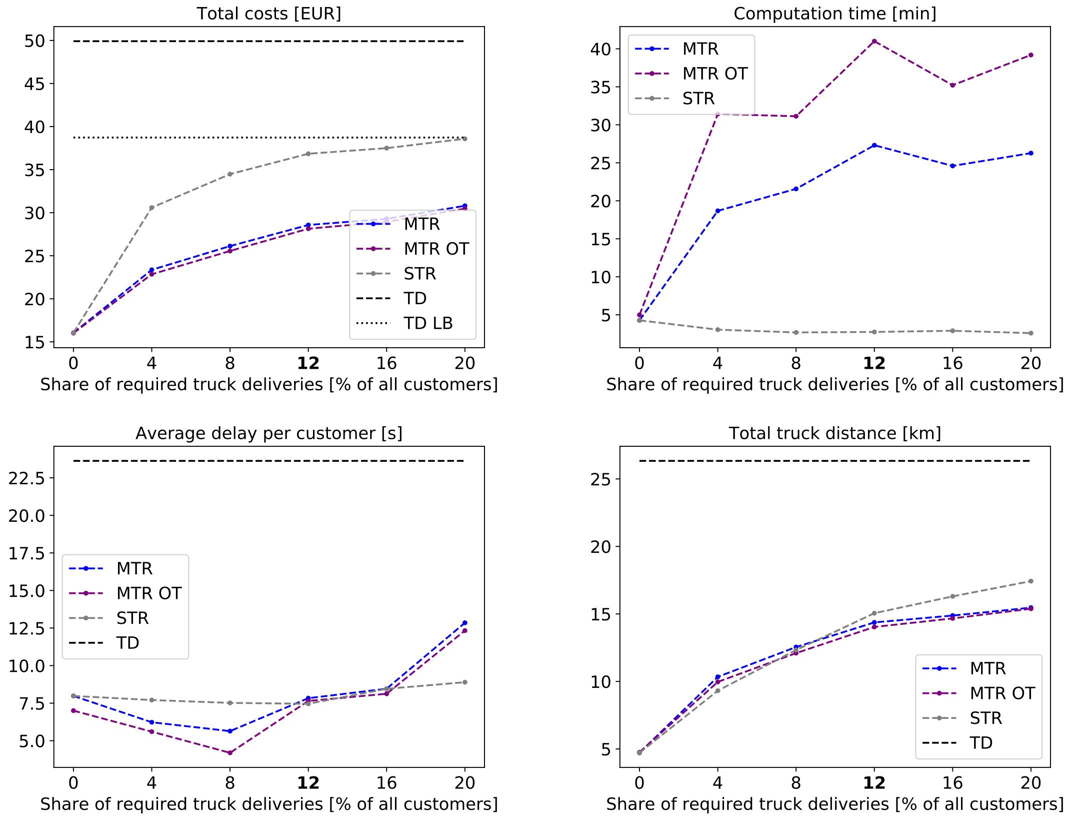

Figure 4.7: Comparison of different delivery modes for a varying share of required truck deliveries

Computation time Runtime increases significantly for a mixed planning (i.e., MTR and MTR OT) as soon as truck deliveries are required. The actual locations of individual customers are the main driver of computation times. Single customers can significantly increase the problem complexity and the respective runtimes if they require truck delivery and cause a large detour for the truck. For example, for MTR, the standard deviation across the 20 instances relative to the average objective value increases by 25% when the share of truck deliveries increases from 0 to 12%. In line with this, runtimes for MTR OT are higher due to the potential additional truck stops. STR on the other hand reveals a decrease in runtime as more truck deliveries are outsourced to a separate routing problem.

Total costs All robot concepts outperform a solution with truck deliveries only (TD). This is inline with current literature (see Ostermeier et al. (2021a)). MTR OT is the option with the lowest total costs in all examples. Total costs increase significantly for all concepts involving robots as soon as truck deliveries are required (i.e., comparing 0 and 4% truck deliveries required). In the STR case, this is due to the truck delivery tour needed in addition to the robot delivery tour. In the MTR and MTR OT cases, it can be attributed to reduced flexibility given the stops required for truck delivery.

A further increase in the share of truck deliveries leads to a moderate increase in total costs. Comparing a combined truck and robot delivery to a separate delivery (i.e., MTR vs. STR) of more than 4% truck deliveries results in cost savings of between 20 and 24% in favor of a mixed delivery. This highlights the advantage of our MTR heuristic's ability to combine truck and robot deliveries into one tour. The cost advantage of additional optional truck deliveries (i.e., MTR OT vs. MTR) is lower with up to 2% savings. Compared to TD, MTR reduces costs by 43% in the default case with 12% truck deliveries. This highlights the attractiveness of delivery by trucks and robots even for situations in which not all deliveries can be made by robot.

Delay and truck distance The logistical performance with respect to delays is comparable for all robot concepts. This shows that all deliveries can be made by a single tour without compromising on delivery performance. MTR and MTR OT show a minimum delay when 8% of truck deliveries are considered. The reason for this is that including additional stops at truck delivery customers (and thus forcing the truck to make a longer tour) can improve punctuality as less distance needs to be covered by robots. A further increase in truck deliveries then leads to additional delays caused by longer truck tours and a later launch of robots at the last drop-off points. The latter effect also leads to a decreasing advantage of MTR OT when more than 8% of deliveries are required by truck. The truck is already overwhelmed serving the customers who require truck deliveries such that optional truck deliveries are hardly made in addition. The development of covered truck distance is similar across the three concepts using robots. It shows a flattening increase for an increasing number of truck deliveries. In the default case, MTR reduces truck mileage by 45% compared to TD, showing that it is able to reduce pollution and traffic even when truck deliveries are necessary. The steady increase in mileage can be reasoned by the tight time windows considered. The truck in the MTR scenario must go on a criss-cross route to satisfy all the time windows at customer stops. We therefore analyze a changing time window structure in the following.

4.6.4 Analysis of the time window structure

Time windows limit the degree of freedom for the routing. This section analyzes the impact of the time window length for both truck and robot deliveries. We analyze both customer groups separately since the impact of a customer's time window on the overall solution is higher when the truck needs to visit the customer and meet the time window. This can lead to detours or waiting time affecting all other deliveries as well, while a robot delivery has little effect on other deliveries. As the cost advantage of MTR OT compared to MTR is only around 2% in our tests, we restrict our remaining analyses to the comparison of MTR vs. STR for better readability.

Time window length for truck deliveries Figure 4.8 shows the performance of MTR vs. STR depending on the change in time window length for truck deliveries. Every time window change is made symmetrically, i.e., in the case of a 10 min change, start and end of the time windows are shifted by 5 min each. 0 corresponds to the default case.

Cost and computation time are reduced if time windows become wider due to increased flexibility. The cost decrease runs in parallel for MTR and STR such that MTR‘s cost advantage is stable at 21 - 23%. The driver of the cost decrease is reduced truck usage both for MTR and STR. STR achieves only a moderate truck distance reduction, but at the same time reduces delays and keeps robot use stable since the separate robot delivery tour is not affected. MTR achieves a larger distance reduction at the cost of increasing delays and robot use. This means that although the time windows become wider, MTR uses this opportunity to further reduce truck

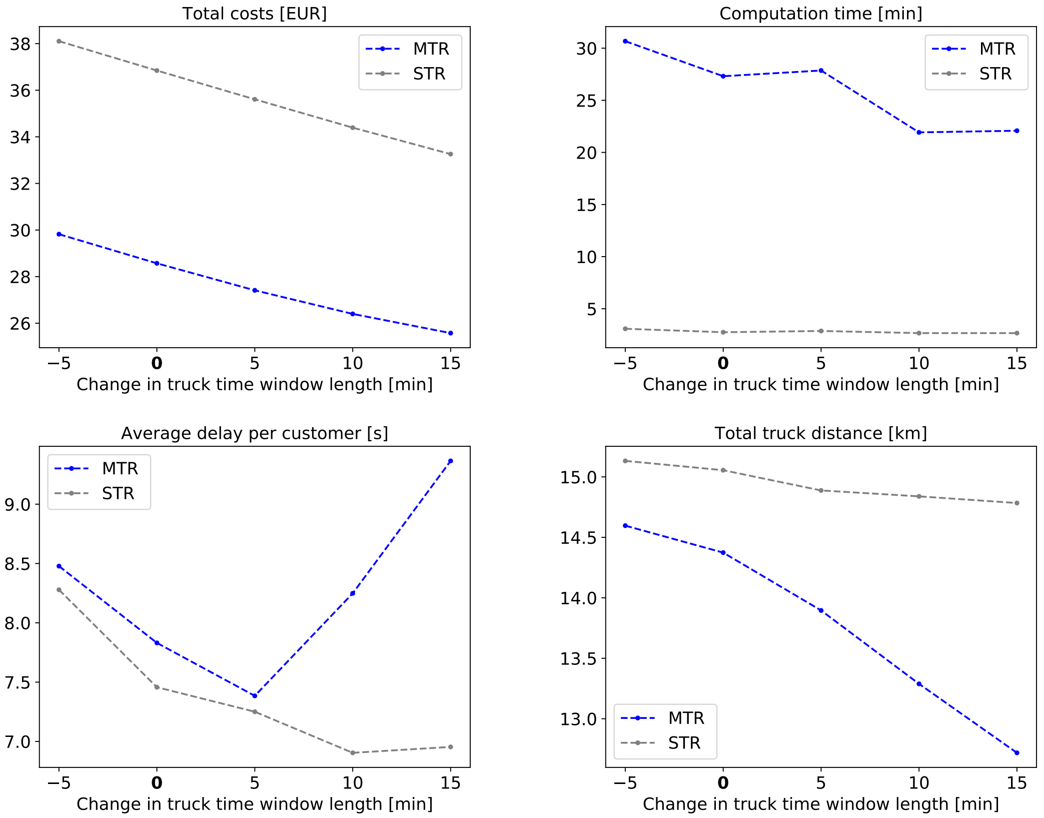

Figure 4.8: Comparison of MTR vs. STR for varying length of truck delivery time windows

distance and allow longer robot travel, resulting in a very small increase in delays. Additionally, we considered a scenario without time windows for truck deliveries. Even in this scenario, a cost saving of 19% is achieved by MTR compared to STR. This is possible as truck deliveries can be added freely at beneficial points of the route such that deviations are minimized.

Time window length for robot deliveries We further analyze the impact of robot delivery time windows. The results for the corresponding changes are shown in Figure 4.9.

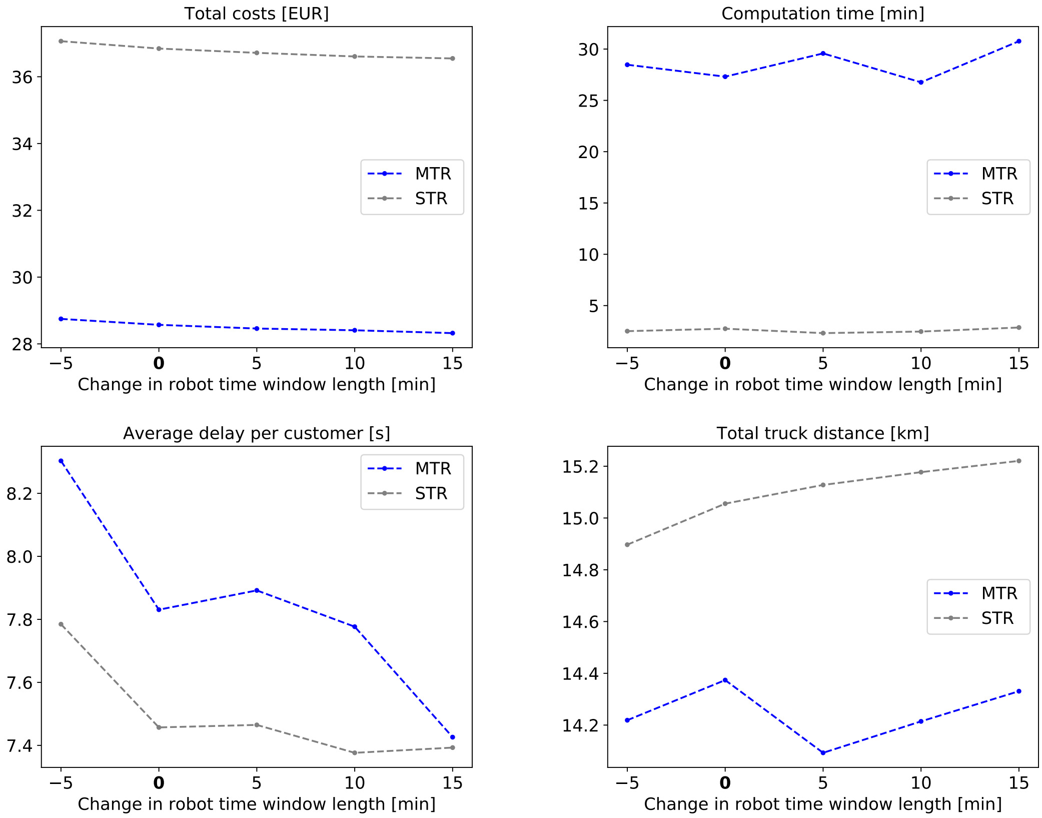

Figure 4.9: Comparison of MTR vs. STR for varying length of robot delivery time windows

As could be expected, costs of the MTR are hardly affected by these changes since truck tours are dominated by truck deliveries. The only effect of wider time windows is reduction in delays. For STR, the TnR route changes slightly. The distance becomes longer, while robot cost and delays decrease. This leads to a minor cost reduction as robot deliveries only account for 38% of total costs and truck deliveries are not affected. In practice, this means both approaches can fulfill tight time windows for robot deliveries at little additional cost. The MTR approach outperforms the STR concept with separate planning of truck and robot deliveries across all scenarios.

4.6.5 Impact of customer distribution

The spatial distribution of customers can have a strong impact on a concept‘s performance. We therefore analyze total costs of MTR vs. STR for different distribution types. The uniform distribution of our default setting is compared to two alternatives: a concentrated distribution, where customers are located centrally in a 2×2 km^2 square area, and a clustered distribution, where two customer clusters a considered, one in the lower left and one in the upper right quadrant of the original 4×4 km^2 square area. The distribution of depots and drop-off points remains unchanged. The number of customers is varied from 25 to 100 (where our default case corresponds to the uniform distribution of 50 customers). The MIP used to solve the truck delivery tour part of STR could not be solved to proven optimality within three hours in the 100-customer case. The best-known solutions are reported. The results are summarized in Figure 4.10.

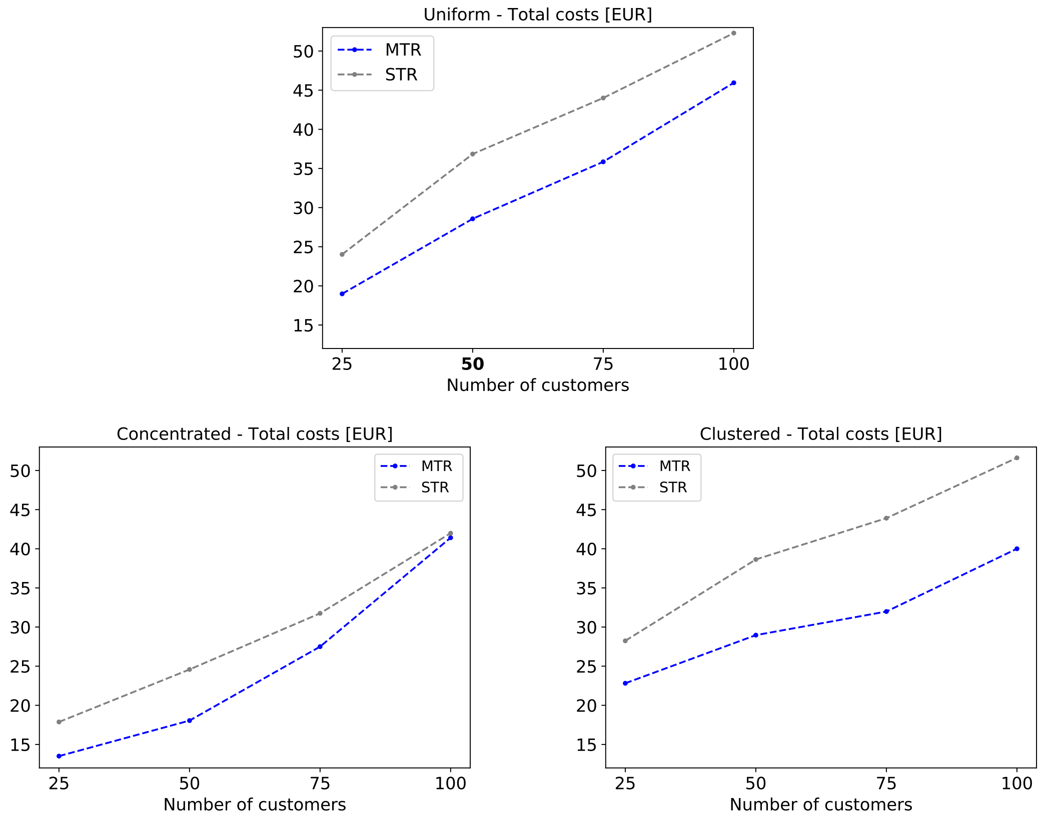

Figure 4.10: Cost comparison of MTR vs. STR for varying customer distributions

Total costs show a near linear increase for both concepts. The MTR approach is able to sustain or even expand its cost advantage for an increasing number of customers when customers are distributed uniformly or clustered. MTR‘s savings decrease in the concentrated distribution scenario, culminating in almost equal results for 100 customers. Concentrated customers are beneficial for both MTR and STR as long as the robot depot density is high enough. For 25 and 50 customers, depots in customer proximity provide enough robots to serve all customers (10 robots per depot). In the case of 100 customers (of which 88 receive robot delivery), the truck is forced to leave the customer area to pick up robots from remote depots. Both STR and MTR suffer from this effect such that the cost difference decreases. Despite this effect we can state that combining truck and robot deliveries within our MTR approach leads to significant savings in most cases, and to equal costs in the worst-case scenario.

4.6.6 Impact of costs

Impact of truck costs The hourly cost rate of the truck is mostly driven by the driver‘s salary. We therefore provide a sensitivity analysis on the truck cost rate c^t, which corresponds to a Western European salary level in our default case. Figure 4.11 displays our findings. Total costs increase proportionally for both approaches, leading to stable cost savings of 22 - 24% through MTR. STR is more sensitive to changing costs. The higher the truck costs, the higher the delays. The increase in delays goes along with a decrease in truck distance. The MTR solution on the other hand is not sensitive to changing costs with respect to delays and truck distance. In the 10 €/h scenario, the MTR approach therefore results in 10% less mileage at a cost of a 50% higher average delay compared to STR.

Impact of delay costs We have shown that increasing truck costs may lead to increasing delays within the MTR approach. In our final test we therefore assess how MTR performs for varying delay costs c^l. The results are summarized in Figure 4.12. The cost curves show that MTR savings slightly decrease as the importance of delays increases. However, MTR achieves cost savings of 15% even for a 100 €/h delay cost rate. Since the applied instances are chosen to be challenging with respect to

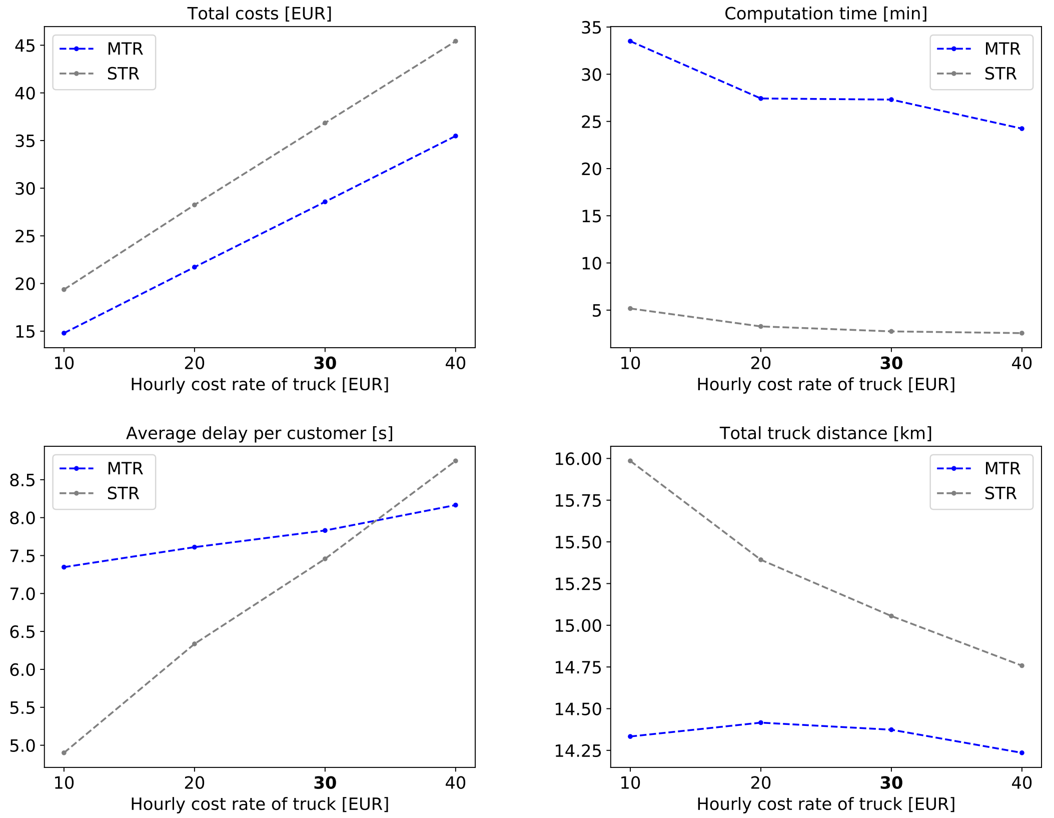

Figure 4.11: Comparison of MTR vs. STR for varying hourly truck cost rates

delivery times, neither of the two approaches can eliminate delays completely. STR is able to reduce delays more as it uses two vehicles instead of one. The price of this is an increasing truck distance, while MTR's truck distance is stable. In summary, the STR concept minimizes delays compared to our MTR approach, but at the cost of longer truck tours. From a total cost perspective, MTR enables significant cost savings even if the costs of delays are high.

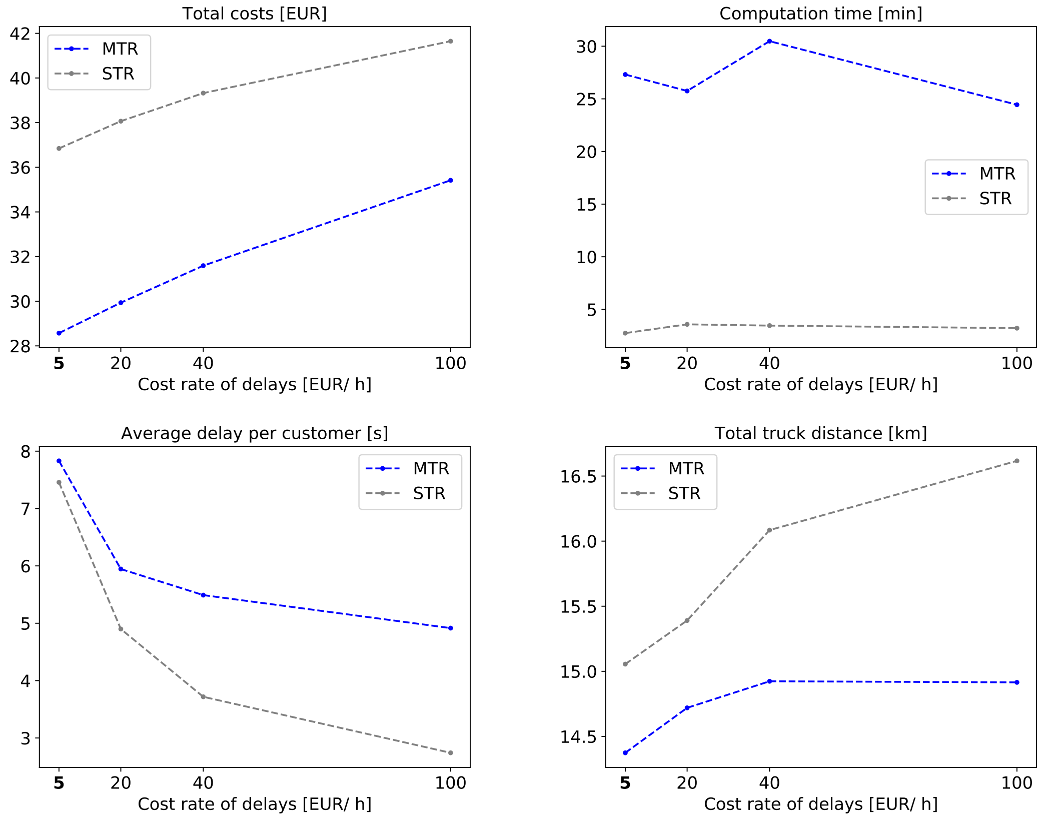

Figure 4.12: Comparison of MTR vs. STR for varying delay cost rates

4.7 Conclusion

Our work shows that the MTR concept is a valuable extension of the existing TnR concept to enable further applications in the retail industry. It combines autonomous robot deliveries with classical truck deliveries (e.g., for bulky orders). We present a comprehensive model formulation for this home delivery concept and solve it using a tailored GVNS solution framework. The GVNS is competitive compared to existing TnR routing algorithms as it outperforms the prevailing LS approach in terms of runtime and equals its solution quality for a robot-only delivery. The extension presented enables practitioners to assess and operate an MTR system that can completely replace classical truck tours.

Our analyses show that the MTR concept reduces costs and truck mileage by more than 40% compared to classical truck delivery, even when a share of customers has to be supplied by truck. To give some further detail, the experiments show that (i) direct truck deliveries have a large impact on costs and solution structure (e.g., 46% higher costs and 119% higher mileage due to 4% of truck deliveries with MTR), (ii) by including direct truck deliveries in the tour, our approach leads to savings of up to 24% compared to a separation of truck and robot deliveries, and (iii) adapting the time windows for truck deliveries can help to further reduce costs and travel distance. Additional analyses highlight the benefits of a mixed delivery concept and show that the MTR results are robust across different settings.

While we address an important extension for TnR delivery, there are several other aspects that can be assessed in future research. Our model could test technical additions and infrastructure specifics such as faster robot travel on bike lanes. Robot movements between depots may further help to increase robot availability in depots visited by truck. The exchange of robots between depots might therefore be a next step. In line with this, our model could be extended to include the pickup of robots at drop-off points on the tour. This means that robots could be sent to locations other than robot depots. Stochastic travel times and pickups from customers could be considered to generalize the problem. Other innovative last-mile delivery concepts could be compared to MTR to derive guidance on which concept and fleet mix to implement in which setting. To date, the TnR and MTR routing approaches have focused on a single truck tour. The use of multiple tours and the corresponding allocation of customers to different tours is required in settings with higher order volumes. Ultimately, the problem presented demonstrates situations of high complexity and unique structure for which alternative solution approaches can be tested. Those could assist in accelerating computation, dealing with larger problem sizes or evidencing optimality.

5 Vehicle assignment for Truck-and-Robot Deliveries

Co-autors: Manuel Ostermeier, Alexander Hübner
Under review by *Transportation Science*

Abstract Last-mile delivery with autonomous robots launched from dedicated delivery trucks is a promising, innovative approach to reduce logistics costs and inner-city traffic. Existing approaches for the truck-and-robot concept focus on the use of a single truck. This limits the practical use of this concept for actual industry applications in bigger cities. The demand for home deliveries is steadily increasing and with that also the requirements for potential delivery by robots. The truck-and-robot concept therefore needs to be extended to meet the growing demand, and to offer an attractive alternative to truck deliveries on a larger scale. We present an extended truck-and-robot delivery concept to cover the more general case of multiple delivery tours. This means that several trucks are available for distribution, and customer orders must be clustered to delivery tours. We formulate this problem as *multi-vehicle truck-and-robot routing problem*. This extension includes an approach for clustering customers to truck tours. We develop a tailored heuristic solution approach based on an innovative neighborhood search, the Set Improvement Neighborhood Search (SINS). We show that tour costs can be reduced by up to 24% using our integrated approach of customer clustering, truck routing and robot scheduling compared to a sequential cluster-first-route-second approach. Complementary experiments show a 62% savings potential compared to conventional truck delivery and analyze the impact of changing customer distributions, delivery time windows, and truck or robot availability.

5.1 Introduction

The global autonomous last-mile delivery market is forecast to grow seven-fold by 2027, with ground vehicles accounting for 85% of it (Grand View Research, 2020). Some retailers expect 80% of their deliveries to be autonomous by 2025 (Bennett, 2020). The application of autonomous delivery robots has the potential to fundamentally revolutionize last-mile logistics. At the same time, consumers are increasingly ordering products online to benefit from the comfort of home deliveries. Statista (2021), for example, forecasts a growth of online food sales in the U.S. by one-third from 2020 to 2023. However, classical trucks for last-mile delivery are increasingly obstructing traffic flow in urban areas and driving up local emissions. Enhanced services such as same-day delivery and tighter delivery time windows pose even more challenges for logistics systems, while cost pressure increases (Ishfaq et al., 2016; Hübner et al., 2016a; Buldeo Rai et al., 2019; Arslan et al., 2019; van Heeswijk et al., 2019). Last-mile deliveries are therefore becoming increasingly important (McKinsey & Company, 2016; Otto et al., 2018; Boysen et al., 2021). Innovative delivery concepts are needed to reduce traffic congestion, CO_2 emissions, as well as noise, and to enable cost-efficient and customer-friendly services (Orenstein et al., 2019; Hübner et al., 2019). A promising approach in this respect is applying autonomous robots carried by trucks in urban areas, known as the truck-and-robot concept (see e.g., Boysen et al. (2018b) and Alfandari et al. (2019)). Delivery trucks act as motherships and transport parcels together with the robots – also known as ground drones. The robots carry out the actual home deliveries and are released in customer vicinity to travel the "last mile" to the customer's home. The customers can choose a delivery time window, during which they are at home and can retrieve their order from the robot. Daimler (2019), for instance, has developed and successfully tested customized trucks paired with delivery robots. Delivery robots have further been successfully implemented in different settings by a number of companies (see e.g., Marble (2019), Starship (2019) and Kiwibot (2020)), who usually offer them as a rental service to logistics service providers.

The objective of truck-and-robot routing is to plan a truck route and schedule robot deliveries to satisfy the complete demand, respect truck and robot capacities, and minimize the total costs arising from the travel costs of trucks and robots and potential service-related costs (e.g., for delays). The emerging but currently still small body of literature on truck-and-robot concepts focuses on the identification of central problem aspects and related benefits, such as the reduction of emissions and costs (Ostermeier et al., 2021a) or service quality (Boysen et al., 2018a; Alfandari et al., 2019). Current literature addresses the basic problem, where only one truck is available to transport robots and goods. Given the steady growth of e-commerce and its home delivery, a single truck may not be sufficient to fulfill the complete demand within a defined delivery period. This calls for an extension of the concept to multiple trucks. The generalized problem employs not just one truck but a fleet of delivery trucks and additionally incorporates the clustering of customers to truck tours and the routing of multiple tours.

This paper addresses this generalization and formulates the first Multi-Vehicle Truck-and-Robot Routing Problem (MVTR-RP) as an extension of the basic problem with only one truck. Prevailing solution approaches of the single-truck problem (e.g., Boysen et al. (2018b) or Alfandari et al. (2019)) are not designed to solve this extended problem as neither the clustering for customers to various truck tours is considered nor the routing of various trucks with different start times. We introduce a tailored solution approach for the $\mathcal{NP}$-hard problem of simultaneously solving the clustering of customers to truck tours, routing of trucks and robot scheduling. Our approach relies on a specifically developed neighborhood search algorithm – denoted as Set Improvement Neighborhood Search (SINS) – that improves a set of tours by optimally choosing a new set out of the neighborhoods of all incumbent tours. It does this by creating and testing large pools of potential truck tours, from which the optimal set is chosen and to which customers are clustered.

The remainder of this paper is organized as follows. We outline the delivery concept and develop the formal problem description in Section 5.2. Section 5.3 reviews related literature and Section 5.4 proposes a tailored heuristic. Section 5.5 analyses the numerical efficiency and develops managerial insights. We summarize our findings and outline potential future research areas in Section 5.6.

5.2 Problem description and formal model

This section details the multi-vehicle truck-and-robot concept. We first outline the logistical setup of both the single- and multi-vehicle concept in Section 5.2.1. This builds the basis for introducing the formal model (Section 5.2.2). In our context the term "vehicle" refers to a goods and robots transporting truck, whereas the term "robots" refers to autonomously driving ground robots.

5.2.1 Technological and logistical setup

The basic truck-and-robot concept with a single truck The truck-and-robot concept combines the use of autonomous delivery robots with specialized delivery trucks to launch robots in customer proximity for attended home delivery. The delivery trucks are specialized vans (see Figure 5.1) that act as mothership for the transportation of parcels and robots (Jennings and Figliozzi, 2019; Boysen et al., 2018b; Alfandari et al., 2019). Robots can enter the truck via a ramp from the back, be loaded by the driver in the front part of the truck, and leave it via another ramp to the side. The truck's capacity is limited with respect to robots and storage boxes. An example setup allows up to eight robots and 54 storage boxes (see, e.g., Mercedes-Benz Vans (2016)). A variety of different robot models are used for deliveries (see e.g., Jaller et al. (2020); Baum et al. (2019)), mainly differing in size and travel speed.

The other key component of the truck-and-robot system are robot depots, small charging stations in the customer area operated by a robot provider. The provider rents robots to multiple logistics service companies. Robots wait at these charging stations until they are needed by a logistics provider, who picks them up by truck and pays a time-dependent rental fee. Figure 5.2 illustrates the example of a truck-and-robot tour with a single truck. The truck tour of the logistics provider starts at a goods warehouse where all parcels for delivery and the initial number of robots are loaded. The truck then visits dedicated locations to pick up and release robots within the delivery area. Additional potential truck stops where robots can be released

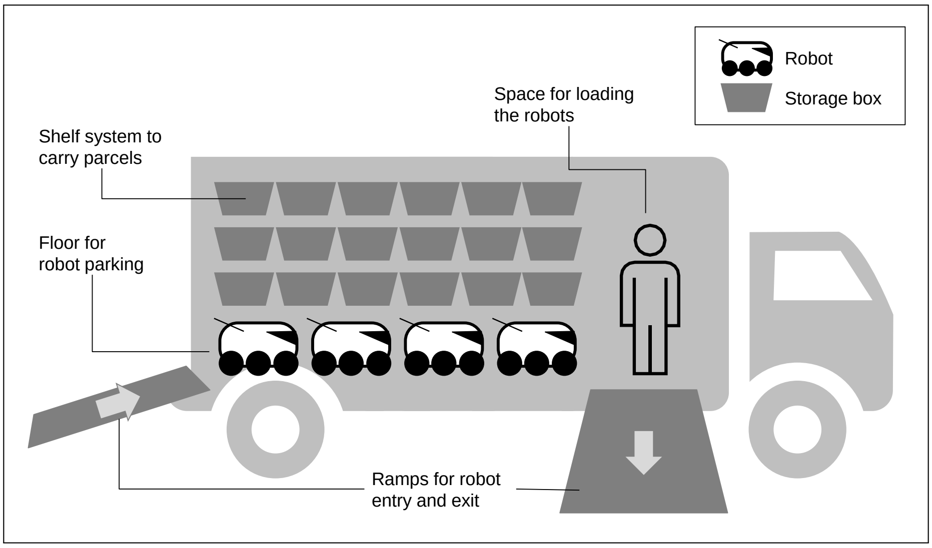

Figure 5.1: Specialized truck for robot deliveries

are defined as drop-off points. In contrast to depots, robots can only be released at drop-off points but no new robots can be loaded. Drop-off points and depots are predetermined locations due to infrastructural requirements. The truck never waits for dropped off robots, but picks up new robots waiting at the robot depots. Once released, robots move autonomously on sidewalks at pedestrian speed and deliver parcels to customer doors. Customers are notified on arrival and receive their delivery by unlocking the compartment after receiving a code. Since the customers must be present, they can choose a delivery time window during the purchase. After a customer has retrieved the parcel, the robots return to the closest robot depot, from where the robot provider can rent it to another logistics company (not shown in Figure 5.2 for better readability). Relevant costs are the truck costs (i.e., travel times and distances from warehouse to robot depots, drop-off locations and return to warehouse) and robot travel times from the drop-off location to customers. The truck costs include time-based costs for the driver and distance-based travel costs. As robots are usually rented as a service from a robot provider and charged by usage time, a time-based fee applies for the actual travel time. The application of time windows imposes further cost considerations. If a robot arrives before the time window, it must wait for the customer. If it arrives after the time window, delay costs (as opportunity costs for reduced satisfaction or rebates on delivery fees) are incurred.

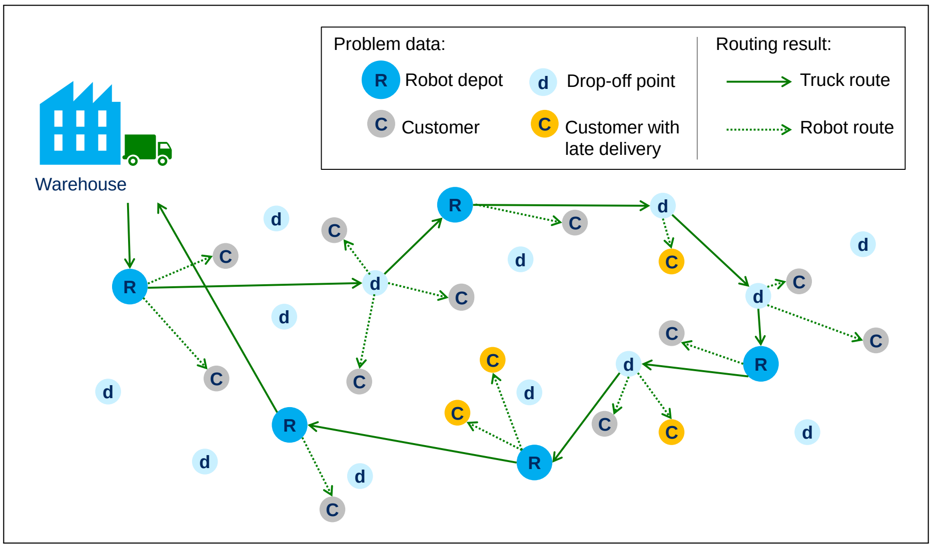

Figure 5.2: Truck-and-robot tour with one truck (example)

The routing of trucks is a central aspect in the truck-and-robot concept. Specifically, it needs to be decided which of the given robot depots and drop-off points are visited by the truck and in which sequence. In contrast to the Traveling Salesman Problem (TSP), the number of stops and the stop locations are part of the decision problem. At the same time, the assignment of customers to truck stops needs to be considered, i.e., the drop-off location as the starting point of each customer's delivery. This is also called *robot scheduling* and determines the arrival and usage time of each robot. This is further constrained by robot availability at robot depots. We denote the problem of routing one truck with robots as Single-Vehicle Truck-and-Robot Routing Problem (SVTR-RP).

Multi-vehicle truck-and-robot routing A single truck is only able to supply a given maximum number of customers, as its parcel capacity is limited. A single truck may not be sufficient in a real-life setting for last-mile delivery in urban areas with a large number of customers. An increasing demand volume together with the need to carry out deliveries simultaneously (given the time windows) will require additional trucks to avoid late or failed deliveries. Figure 5.3 illustrates an example with two truck tours. In comparison to Figure 5.2, two trucks can serve more customers and avoid delays. The extension to the multi-vehicle problem, denoted above as MVTR-RP, enables the simultaneous delivery to different customers by multiple tours. This increases the flexibility and the fulfillment capacity, but it also increases the problem complexity. In the MVTR-RP, multiple tours need to be determined, i.e., their start times, the respective stops and their sequence. Multiple tours access the same resources (robot depots), and robot availability must be monitored. In addition, customers need to not only be *assigned* to truck stops (as in the basic SVTR-RP) but *clustered* (i.e., allocated) to delivery tours in the first place. This means a simultaneous decision is required on the (i) *clustering of customers to truck tours*, the (ii) *routing of trucks* (i.e., selection of robot depots and drop-off locations as well as sequencing of the stops), and the (iii) *robot scheduling* (i.e., assignment of customers to drop-off locations of the tour, from where the respective robot starts). All three decisions are interdependent. For example, the reallocation of one customer to another tour impacts routing and robot scheduling. Furthermore, it is not sufficient to cluster customers based on similar locations and time windows, since two customers far from each other with different deadlines could fit well on one tour.

To summarize, the underlying routing problem is specified by a given number of customers with a known demand and given time windows that need to be supplied with a certain number of robots launched by trucks. The trucks travel from a warehouse to robot depots to pick up robots and to drop-off points to launch the robots, where each robot serves one customer. The objective is to minimize travel costs of trucks and robots and potential service-related costs (e.g., for delays) by clustering customers, routing truck tours and defining robot schedules that satisfy the entire demand as well as maintaining truck and robot capacities.

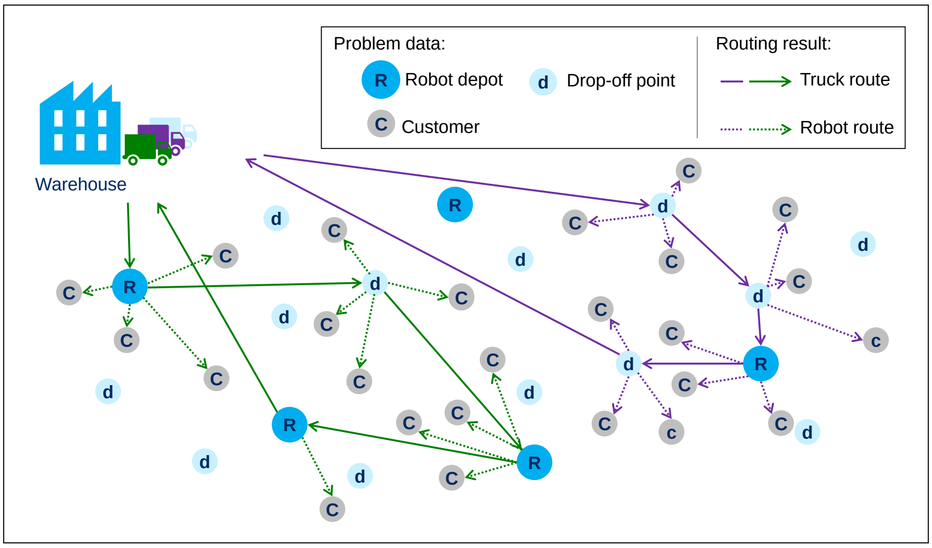

Figure 5.3: Two separate truck-and-robot tours

5.2.2 Mathematical model of the Multi-Vehicle Truck-and-Robot Routing Problem

Subsequent to the problem description we formulate the mathematical model of the MVTR-RP. Table 5.1 summarizes the notation used.

Index sets	
C	Set of customers, $k \in C$
D $(\hat{D})$	Set of distinct drop-off points (including duplicates)
R $(\hat{R})$	Set of distinct robot depots (including duplicates)
$\hat{L}$	Set of all (duplicate) locations from which robots can be started: $\hat{L} := \hat{D} \cup \hat{R}$
Ω $(\bar{\Omega})$	Set of duplicate start (end) positions, one duplicate ω ($\bar{\omega}$) for each truck, with $\omega, \bar{\omega} \notin \hat{L}$
I_a	Set of duplicate indices $i, i \in \hat{R}$, of one distinct robot depot $a, a \in R$
I_a^m	Set of elements $i \in I_a$ with $i \leq m$
Problem parameters	
d_k	Deadline for customer $k, k \in C$
Q (G)	Maximum robot (parcel volume) capacity of a truck
β_a	Initial amount of available robots in location $a, a \in R$
δ	Initial number of robots aboard a truck at start
ϵ	Length of time windows

Continued on next page

Table 5.1 – *Continued from previous page*

η_k	Volume of parcels per customer k
$\vartheta^v_{i,j}$	Truck travel time from location i to location j with $i, j \in \hat{L} \cup \Omega \cup \bar{\Omega}$
$\vartheta^r_{i,k}$	Robot travel time from location $i, i \in \hat{L}$, to customer $k, k \in C$
$\lambda_{i,j}$	Distance between locations i and j with $i, j \in \hat{L} \cup \Omega \cup \bar{\Omega}$
Cost parameters	
c^{late}	Cost of delays per time unit
c^{dist}	Cost of truck travel per distance unit
c^{veh} (c^{rob})	Cost of truck (robot) per time unit
Decision variables	
$s_{i,j}$	Binary: 1, if a truck travels from location i to location j; 0 otherwise
t_i	Arrival time of a truck at location i
$x_{i,k}$	Binary: 1, if customer k is supplied from location i; 0 otherwise
Auxiliary variables	
e_i	Number of robots taken out of depot location $i, i \in \hat{R}$
g_i	Volume of parcels aboard a truck when arriving at location i
q_i	Number of robots aboard a truck at departure from location i
l_k	Lateness (delay) time of delivery to customer k
w_k	Waiting time for robot at customer k

Table 5.1: Notation of the MVTR-RP

Index sets The MVTR-RP is based on the location sets of customers (C), robot depots (R) and drop-off points (D). To allow for each robot depot and each drop-off point to be visited several times, we duplicate the elements in R and D, resulting in $\hat{R}$ and $\hat{D}$ as the corresponding sets of duplicates. All duplicate locations are summarized by the set $\hat{L} := \hat{D} \cup \hat{R}$. Duplicates enable multiple visits at the same location by several trucks. This means that one or several trucks can retrieve robots from the same robot depot on multiple occasions. The set of the truck's start and end locations is denoted by Ω and $\bar{\Omega}$, each containing one (duplicate) location for every available truck. This way the number and usage time of trucks can be tracked although all trucks move through the same location network $\hat{L}$. Finally, to keep track of available robots in the unique robot depots, we define the set I_a of all duplicate locations i to the depot a (with $i \in \hat{L}, a \in R$), and the set I^m_a of indices in $i \in I_a$ with $i \leq m$ for a given number m.

Parameters and costs Between two locations i and j we define the distance as $\lambda_{i,j}$ and travel times $\vartheta^v_{i,j}$ and $\vartheta^r_{i,j}$ for the trucks and robots, respectively. The travel times include any processing times that occur at each stop. Each customer $k, k \in C$, has a time window defined by the delivery deadline d_k and the time window length ϵ, which is the same for all customers. Every robot depot $a, a \in R$, has an initial number β_a of robots available. We consider a homogeneous truck fleet where each truck has a maximum robot capacity Q and starts with δ robots aboard. G denotes a truck's parcel capacity and η_k the parcel volume of customer k. We further assume that each customer order fits into a single robot. Finally, time-dependent cost rates c^{veh} and c^{rob} for the trucks and the robots apply. For the truck, this mostly represents the driver's salary, as we assume drivers can perform other value-adding tasks or reduce overtime when tours become shorter. The costs for the robots consist of the time-based rental fee charged by the robot provider. Each truck further incurs costs c^{dist} per distance. Delayed deliveries are priced at a time-based lateness rate c^{late}.

Decision variables The binary variable $s_{i,j}$ defines whether a truck travels from location i to location j. The variable t_i stands for a truck's arrival time at each location. The binary variable $x_{i,k}$ defines whether customer k is served from location i, i.e., whether a robot travels between the two. Further, the following auxiliary decision variables are applied. The variable q_i defines the number of robots on the truck and e_i the number taken from a depot at the truck's departure from stop i. The variable g_i represents the parcel volume aboard the truck when arriving at location i. Variable l_k tracks the delay time if the delivery at customer k occurs after the deadline, and w_k represents a robot's waiting time if it arrives early, i.e., before $d_k - \epsilon$. The decision problem is then defined as follows.

$$\begin{aligned} \min RC = c^{\text{veh}}\Big(\sum_{\bar{\omega}\in\bar{\Omega}} t_{\bar{\omega}} - \sum_{\omega\in\Omega} t_{\omega}\Big) + \sum_{i\in\hat{L}\cup\Omega}\ \sum_{j\in\hat{L}\cup\bar{\Omega}} c^{\text{dist}}\lambda_{i,j}s_{i,j} \\ + \sum_{i\in\hat{L}}\sum_{k\in C} c^{\text{rob}}\vartheta^{r}_{i,k}x_{i,k} + \sum_{k\in C}\big(c^{\text{late}}l_k + c^{\text{rob}}w_k\big) \qquad (5.1) \end{aligned}$$

subject to

$$\sum_{i \in \hat{L}} x_{i,k} = 1 \qquad \forall k \in C \tag{5.2}$$

$$x_{j,k} \leq \sum_{i \in \hat{L} \cup \Omega} s_{i,j} \qquad \forall j \in \hat{L}, k \in C \tag{5.3}$$

$$\sum_{j \in \hat{L} \cup \bar{\Omega}} s_{\omega,j} = 1 \qquad \forall \omega \in \Omega \tag{5.4}$$

$$\sum_{j \in \hat{L} \cup \Omega} s_{j,\bar{\omega}} = 1 \qquad \forall \bar{\omega} \in \bar{\Omega} \tag{5.5}$$

$$\sum_{i \in \hat{L} \cup \Omega} s_{i,j} = \sum_{i \in \hat{L} \cup \bar{\Omega}} s_{j,i} \qquad \forall j \in \hat{L} \tag{5.6}$$

$$t_j \geq t_i + \vartheta^v_{i,j} - M \cdot (1 - s_{i,j}) \qquad \forall j \in \hat{L} \cup \bar{\Omega}; i \in \hat{L} \cup \Omega \tag{5.7}$$

$$l_k \geq t_i + \vartheta^r_{i,k} - d_k - M \cdot (1 - x_{i,k}) \qquad \forall k \in C; i \in \hat{L} \tag{5.8}$$

$$w_k \geq d_k - t_i - \vartheta^r_{i,k} - \epsilon - M \cdot (1 - x_{i,k}) \qquad \forall k \in C; i \in \hat{L} \tag{5.9}$$

$$q_\omega = \delta \qquad \forall \omega \in \Omega \tag{5.10}$$

$$q_j \leq q_i + e_j - \sum_{k \in C} x_{j,k} + M \cdot (1 - s_{i,j}) \qquad \forall i \in \hat{L} \cup \Omega; j \in \hat{R} \tag{5.11}$$

$$q_j \leq q_i - \sum_{k \in C} x_{j,k} + M \cdot (1 - s_{i,j}) \qquad \forall i \in \hat{L} \cup \Omega; j \in \hat{D} \tag{5.12}$$

$$g_{\bar{\omega}} = 0 \qquad \forall \bar{\omega} \in \bar{\Omega} \tag{5.13}$$

$$g_j \geq g_i + \sum_{k \in C} \eta_k x_{j,k} - M \cdot (1 - s_{j,i}) \qquad \forall i \in \hat{L} \cup \bar{\Omega}; j \in \hat{L} \tag{5.14}$$

$$t_i \leq t_j \qquad \forall a \in R; i, j \in I_a : i \leq j \tag{5.15}$$

$$\sum_{h \in \hat{L} \cup \Omega} s_{h,i} \geq \sum_{h \in \hat{L} \cup \Omega} s_{h,j} \qquad \forall a \in R; i, j \in I_a : i \leq j \tag{5.16}$$

$$\beta_a - \sum_{i \in I_a^m} e_i \geq 0 \qquad \forall a \in R; m \in I_a \tag{5.17}$$

$$s_{i,j} \in \{0, 1\} \qquad \forall i \in \hat{L} \cup \Omega, j \in \hat{L} \cup \bar{\Omega} : i \neq j \tag{5.18}$$

$$s_{i,i} = 0 \qquad \forall i \in \hat{L} \tag{5.19}$$

$$x_{i,k} \in \{0, 1\} \qquad \forall i \in \hat{L}; k \in C \tag{5.20}$$

$$e_i \in \mathbb{Z} \qquad \forall i \in \hat{R} \tag{5.21}$$

$$t_i \geq 0 \qquad \forall i \in \hat{L} \cup \Omega \cup \bar{\Omega} \tag{5.22}$$

$$q_i \in [0, Q] \qquad \forall i \in \hat{L} \tag{5.23}$$

$$g_i \in [0, G] \qquad \forall i \in \hat{L} \qquad (5.24)$$

$$l_k, w_k \geq 0 \qquad \forall k \in C \qquad (5.25)$$

The objective function (5.1) minimizes total routing costs (RC). The first term indicates the time-dependent usage costs across all trucks, i.e., the total time for each truck from departure to arrival at the warehouse. The second term sums up total distance costs of all legs traveled by the trucks. The third term considers the costs of robot travel from truck stops to customers. The last term adds the costs of robot waiting and delay times. Note that the robots' return time from a customer to the closest depot does not depend on the routing decisions and is therefore not decision relevant. Constraints (5.2) ensure that every customer is served exactly once, while Constraints (5.3) make sure a robot starts only from stops visited by a truck. Constraints (5.4) and (5.5) ensure that only one truck starts from each start location and also returns to it. Trucks that are not required will stay at their start location, i.e., $s_{\omega,\bar{\omega}} = 1$ and $t_\omega = t_{\bar{\omega}}$. Constraints (5.6) represent the flow constraint, stating that the truck leaves every location $j, j \in \hat{L}$ as often as it arrives there. Constraints (5.7) calculate the truck arrival times based on associated travel times. Note that these constraints ensure that every duplicate stop is visited only once and only by one truck. This means that a sufficient number of duplicates is needed for every stop (in the worst case this could be $|C|$). Constraints (5.8) and (5.9) calculate the delay and robot waiting time for each delivery. Equations (5.10) define the number of available robots aboard the truck at departure. Constraints (5.11) and (5.12) keep track of the robots aboard a truck after each stop, depending on whether the stop is a robot depot or a drop-off point. Constraints (5.13) and (5.14) ensure adherence to truck capacities by keeping track of the total quantity of parcels aboard the truck arriving at location j. This is done in a recursive manner by defining that the tour ends with an empty truck. Constraints (5.15) and (5.16) enforce (without loss of generality) that duplicates of the same location are visited in ascending order of their index. This fact is then used to determine the robots in each depot after every visit (left side of Constraints (5.17)). This limits robot availability at depots even if they are visited by different trucks. Finally, constraints (5.18) - (5.25) define the variable domains.

5.3 Related literature

Having derived the distribution system and the formal decision problem, this section reviews related literature. We refer to Savelsbergh and van Woensel (2016), Olsson et al. (2019) and Boysen et al. (2021) for a detailed overview on current last-mile delivery concepts. The literature related to our problem can be classified into two streams. The first stream is clearly related to our setting and defined by 'truck-and-robot' as a system in which the robots are transported aboard the truck, make a delivery and return to a robot depot. We first review this literature (and related concepts) in Section 5.3.1. As this is an emerging area with only a small body of literature and all state-of-the art publications consider only a single truck and not yet multiple trucks, we extend our review to multi-vehicle problems in Section 5.3.2. This constitutes a second related stream and includes various other means of transportation such as drones and cargo bikes that are combined with multiple trucks. Finally, we blend both streams, analyze them in the context of the MVTR-RP, and identify the research gap in Section 5.3.3.

5.3.1 Robot concepts with a single truck

Current publications for truck-and-robot systems are based on the SVTR-RP, i.e., limited to one truck. All approaches consider time windows or delivery deadlines. In a seminal paper, Boysen et al. (2018b) discuss such a basic truck-and-robot concept. The authors minimize the number of late deliveries and assume unlimited robot availability. Their solution approach is based on a local search procedure, and their analysis shows the potential benefits of the concept compared to standard truck delivery. Alfandari et al. (2019) build on this work by proposing alternative delay measures and a Branch-and-Benders-cut scheme for routing. Ostermeier et al. (2021a) further extend the decision model by minimizing total delivery costs and incorporating limitations on robot availability. Their solution approach relies on a local search as well and solves instances with up to 125 customers. In numerical experiments, the system of one truck with robots reduces costs and emissions by more than 50% compared to normal truck delivery. Heimfarth et al. (2021) generalize the concept by including manual delivery by the truck driver. They apply a General Variable Neighborhood Search (GVNS) to minimize costs. This work considers the most general case of the SVTR-RP to date, and the proposed GVNS framework proved very efficient compared to other existing local search approaches.

Another concept based on robots relies on hubs (also called satellite locations) in which goods arriving by truck can be stored. It requires more infrastructure and workforce since goods must be stored and robots loaded at the hubs when the truck is already gone. Bakach et al. (2021b), for instance, propose a two-tier delivery system where a truck supplies local hubs in which goods are stored and loaded into robots. The robots then make pendulum tours to the customers. Poeting et al. (2019b), Poeting et al. (2019a), Sonneberg et al. (2019) and Bakach et al. (2021a) are further examples of hub-and-robot settings. The key difference vs. truck-and-robot is that the robots are not transported on the truck but stay around one fixed hub. These variations reduce the complexity but induce practical disadvantages, such as long driving and waiting times for vehicles.

5.3.2 Related concepts with multiple trucks

As there is no existing approach on truck-and-robot concepts with multiple vehicles, we expand our literature analysis to related concepts that employ multiple vehicles and combine trucks with innovative transportation technologies. We analyze these works with respect to our setting, focusing on clustering customers to truck tours. All these concepts share with our use case that trucks are combined with smaller vehicles that have a limited range, namely *robots*, *drones* or *cargo bikes*.

Robot-based concepts with multiple vehicles A hub-and-robot approach is taken by Liu et al. (2020a), who consider delivery robots starting at and returning to fixed hubs. They assume larger robots that can carry more than one order. The customers are assigned to hubs based on k-means clustering. The remaining problem is a vehicle routing problem (VRP) for trucks supplying goods to the satellites and a VRP for the robots of each hub. Liu et al. (2020b) propose a similar concept, with the additional possibility of customers picking up their order at a hub. Time windows are not considered in both publications. The concept of multiple trucks with robot sidekicks is proposed by Chen et al. (2021b). It relies on robots transported by trucks and considers time windows, but does not use robot depots. In this concept the truck makes deliveries to customers and robots make pendulum tours to other customers nearby in the meantime, or while the truck waits for the return of the robots. The problem is solved with a cluster-first-route-second approach. Chen et al. (2021a) analyze the same problem and propose an ALNS for simultaneous clustering and routing. However, the savings potential is smaller compared to using robot depots. Chen et al. (2021b) report savings of 4 - 17% compared to normal truck deliveries, whereas Ostermeier et al. (2021a) for example identified more than 50% savings compared to truck deliveries.

Truck-and-drone with multiple vehicles A large body of truck-and-drone literature was published in recent years (see e.g., review of Otto et al. (2018)). In these applications, a truck usually carries between one and four drones (see Murray and Chu (2015), Agatz et al. (2018), Moshref-Javadi et al. (2020a) and Murray and Raj (2020) as examples for single-truck problems), which depart from the truck at a customer location to serve another customer and join the truck again at the start point or at a another customer. The solution approaches for this particular application first solve a TSP (in the single truck case) or VRP (e.g., see Kitjacharoenchai et al. (2019)) for the trucks, and then select some customers for drone delivery. So far only Li et al. (2020) consider multiple trucks and drones and deliveries with time windows. Their concept is based on a fleet of trucks delivering to customers directly and drones starting from the truck at the visit of customer locations or from the goods warehouse. Each drone can serve several customers per tour and then returns to its starting point, i.e., the truck waits for drones to return. Dayarian et al. (2020) propose a fleet of trucks and drones for same-day delivery, in which the latter supply further parcels to the trucks along their route as new orders are placed. This leads to a problem similar to a VRP, with the additional decision on the meeting points for drone resupply. The truck-and-drone concepts differ from robot approaches, as they consider a small and fixed number of drones that only serve some of the customers, and have to return to the truck. For a more detailed overview on drone delivery we refer to Otto et al. (2018), Dayarian et al. (2020) and Macrina et al. (2020), and to Ostermeier et al. (2021a) for an analysis of differences between truck-and-drone vs. truck-and-robot routing.

Cargo bikes and multiple vehicles A further related problem setting is the application of cargo bikes. Cargo bikes can be combined with trucks to reduce emissions and traffic in inner cities. In most concepts the goods are handed over from trucks to bikes at predefined satellite locations. This means there has to be synchronization between the two vehicle types, resulting in a two-echelon problem. In contrast to truck-and-robot, the bikes cannot be transported on the truck and they can visit many customers in a row, which also requires definition of the bike tour. Anderluh et al. (2017) consider a delivery area with two zones in which customers require delivery by truck or by bike, respectively. Mühlbauer and Fontaine (2021) analyze a system where all deliveries are made by bike. Both publications simultaneously solve the clustering of customers and routing of trucks and bikes. While cargo bike deliveries have similarities with the truck-and-robot concept, there are several fundamental differences. First, time windows have not been taken into account so far. Further, each potential truck stop has a fixed number of bikes available and there are only a few stops. Finally, as trucks cannot pick up and transport bikes, this dramatically reduces the solution space. This is also the key difference between truck-and-robot and two-echelon problems.

5.3.3 Research gap and contribution

Looking at the first stream, there is no approach in pertinent literature on related truck-and-robot concepts for clustering customers when multiple trucks are applied. Current literature on truck-and-robot routing only covers the SVTR-RP. However, in large delivery areas and with the growing order volumes or tighter deadlines, several trucks will become necessary. This leads to the research gap in truck-and-robot literature when it comes to clustering customers to multiple trucks, which depends on the truck routing and robot scheduling.

Approaches to customer clustering exist only for other related applications. However, looking at the second related stream, we can conclude that the distinctive features of the truck-and-robot concept complicate the transfer of available approaches to the MVTR-RP. The MVTR-RP has several specifics that differ from other delivery concepts. First, there are many potential locations for robot pickup and drop-off to be considered for truck routing. Second, these locations cannot store goods, which leads to synchronization between trucks and robots. Third, the truck has limited capacity for two types of objects, goods and robots. Finally, deliveries must occur within time windows, as goods are retrieved from the robots manually in attended home deliveries. These problem specifics make it difficult to apply existing clustering and routing approaches to the MVTR-RP. To summarize, none of the publications on related problems consider the customer clustering to tours and the combination of coupled truck and robot movements, robot pickup along the tour and time windows. There is consequently a research gap when it comes to multi-vehicle settings and the associated MVTR-RP.

5.4 Solution approach

The MVTR-RP generalizes the $\mathcal{NP}$-hard SVTR-RP (see Boysen et al. (2018a)) and hence the MVTR-RP also constitutes an $\mathcal{NP}$-hard problem. That means practically relevant problem sizes cannot be solved to optimality in reasonable time with exact approaches. An advanced heuristic for customer clustering, truck routing and robot scheduling is needed to solve the problem efficiently. As we introduce a new decision problem, there is no reference approach available that jointly includes clustering, routing and robot scheduling. We therefore introduce an innovative heuristic, the Truck-and-Robot Clustering and Routing (TRCR), which is built on an initialization and improvement phase (see Figure 5.4). The initialization generates a route for each available truck and allocates each customer to one of these initial truck routes (denoted as *clustering*), dependent on customers' locations and time windows (Section 5.4.1). The route of each customer cluster is then determined using a variable neighborhood descent (VND) framework (Section 5.4.1). Based on the initialization, the improvement phase, denoted as Set Improvement Neighborhood

Search (SINS), seeks improved clustering, routing and scheduling. It iteratively generates neighborhoods containing variations of each incumbent truck tour (Section 5.4.2), and solves an MIP to test whether a better set of truck-and-robot routes can be obtained by combining these variations (Section 5.4.2). As long as a better set can be found, this process is repeated, otherwise SINS terminates. SINS makes use of the fact that improvements can often be achieved by simultaneously changing two tours and reallocating customers between them. Since it uses an MIP in a metaheuristic fashion, it belongs to the growing field of math-heuristics.

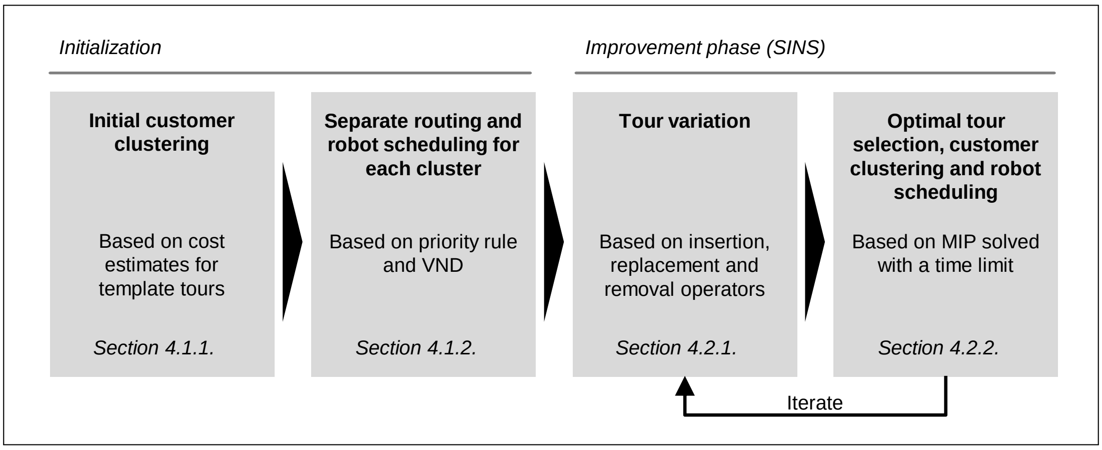

Figure 5.4: Overview of the TRCR heuristic

5.4.1 Initialization phase

The initialization provides a start solution consisting of multiple tours serving all customers. It first clusters customers and then solves the routing and robot scheduling problem for each cluster.

Initial customer clustering

The initial clustering is based on template tours, which is a concept that has been successfully applied to VRPs (see e.g., Kovacs et al. (2014b)). The idea is to *(i) generate template tours* based on given problem specifics, and *(ii) cluster customers* such that they are allocated to template tours based on approximated tour costs.

(i) Template tour generation The building of template tours is based on insights from typical truck-and-robot delivery tours found in our experiments: the first stop regularly lies outside a given range away from the start, the tour proceeds in an arc-shape, and finally returns to the start. Template tours only consider robot depots as these are essential for the supply with robots and the tour building. The tour generation is defined as follows. Candidate depots are selected that are between an inner and outer circle around the start position. The inner circle defines an area including a share of σ_1 robot depots, the outer one an area including a share of σ_2 depots (see Figure 5.5). All depots between these circles (see grey area) act as candidates for the template routes. Next, we select the convex hull of these candidate depots, i.e., only depots that form the convex hull are part of the template tours (see solid blue line in Figure 5.5). Finally, we generate all possible tours that move along the convex hull and visit all of its depots exactly once. All depots on the convex hull are included in the tour and visited in their sequence as given by the convex hull, i.e., no shortcuts are allowed. This results in one template tour for every possible first depot and for each possible direction (clockwise/counterclockwise). This leads to $6 \cdot 2 = 12$ template tours for the example in Figure 5.5. The generation of template tours can be applied for different shares of σ_1 and σ_2 to create a larger pool of templates. All tours created are then used as input for the customer clustering. Please note that the template tours are only used to cluster customers, not for actual routing.

(ii) Customer clustering Customers have two characteristics related to routing: location and time windows. In the clustering, we take these two into account. This means two customers with differing locations and time windows may be clustered to the same tour if they fit into the tour sequence and arrival times. In total we have n clusters, each corresponding to one of the template tours. Inspired by the VRP clustering heuristic by Fisher and Jaikumar (1981), we solve an MIP to cluster customers for the later route building based on a cost approximation for each customer-tour combination. At this stage we ignore robot availability and truck capacity. The notation for the clustering MIP is presented in Table 5.2.

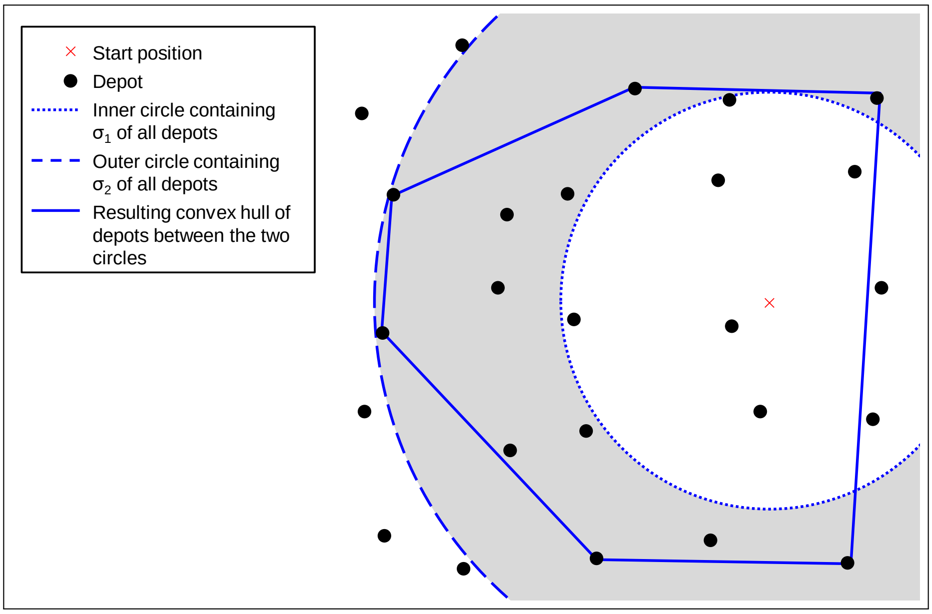

Figure 5.5: Template tour generation

Sets and parameters	
C	Set of customers, $k \in C$
T	Set of template tours, $\tau \in T$
n	Number of available trucks
$c_{k,\tau}$	Cost of supplying customer k from tour τ
Decision variables	
$x_{k,\tau}$	Binary: 1, if customer $k, k \in C$, is supplied from tour $\tau, \tau \in T$; 0 otherwise
z_{τ}	Binary: 1, if template tour $\tau, \tau \in T$ is used; 0 otherwise

Table 5.2: Notation for the initial customer clustering based on cost approximations

The clustering of customers is based on an approximation of costs for serving the customer from a template route. We estimate that all depots and m^e equidistant points between two consecutive depots on the tour are potential stops. This accounts for the possibility of visiting drop-off points between two depots. The coordinates and arrival times of the points between depots are obtained via linear interpolation.

The approximated cost of serving a customer from a given point of a tour τ is defined as the sum of the robot costs for travel time to the customer and potential waiting time, the cost of a potential delay, and a share of the truck costs incurred up to that point, i.e., the truck costs divided by the average number of customers per available truck, $|C|/n$. In a preprocessing phase we calculate the costs for all theoretical points on the template tour and select the minimal cost for each customer and template tour, denoted by $c_{k,\tau}$. The decision model for the customer clustering can then be formulated as follows.

$$\min \; F(X) = \sum_{k\in C}\sum_{\tau\in T} c_{k,\tau} x_{k,\tau} \tag{5.26}$$

subject to

$$\sum_{\tau\in T} x_{k,\tau} = 1 \qquad \forall k \in C \tag{5.27}$$

$$x_{k,\tau} \le z_\tau \qquad \forall k \in C, \tau \in T \tag{5.28}$$

$$\sum_{\tau\in T} z_\tau = n \tag{5.29}$$

$$x_{k,\tau} \in \{0,1\} \qquad \forall k \in C, \tau \in T \tag{5.30}$$

$$z_\tau \in \{0,1\} \qquad \forall \tau \in T \tag{5.31}$$

The objective function (5.26) minimizes the total clustering costs of customer-tour combinations. Constraints (5.27) ensure that each customer is served once, (5.28) mandate that customers are served only via tours that are actually used, and (5.29) defines the number of template tours used as a basis for the improvement phase. Finally, (5.30) and (5.31) define the variable domains. The solutions of this step result in allocating each customer to one cluster. The customer clusters found are then used in the next step to determine actual routes for the customers in question.

Separate routing and robot scheduling for each cluster

Subsequent to the clustering, actual truck tours are determined by solving the routing problem for each customer cluster. As such, it resembles an SVTR-RP and we build upon known efficient approaches. We apply an adapted version of the heuristic proposed by Heimfarth et al. (2021) for the solution of an SVTR-RP. We use the priority rule "go to the location from which most robot deliveries can be started such that they reach customers in time" to generate an initial truck tour and a VND for improvement. After a truck tour is defined, the optimal robot schedule is determined. Here we apply a robot scheduling MIP which is a special case of our "tour selection and customer clustering" MIP that will be introduced later in Section 5.4.2, when only one truck with unlimited parcel capacity is used. For the sake of streamlining the algorithmic description, we pick up on the robot scheduling approach later below when detailing the improvement phase.

After sequentially appending stops based on the priority rule mentioned, the resulting tour is used as start solution for the VND framework, which sequentially tests complete neighborhoods of the incumbent tour. If a better tour is found, it is accepted as the new incumbent tour and the VND restarts from the first neighborhood. If no better tour is found, the VND accepts a random solution from the neighborhood with an initial probability of p, which is decreased by Δp in every iteration. This random acceptance of worse tours widens the search space in the early stage of the VND. If no new solution is accepted, the VND proceeds with the next neighborhood on the list. It terminates when all neighborhoods have been evaluated in a row without accepting a new solution. For a more detailed description of VND we refer to Hansen and Mladenović (2018). The neighborhoods evaluated within the VND are defined by the following operators (adapted from Heimfarth et al. (2021)) and evaluated in the order presented. Each neighborhood is limited to the m^{VND} cheapest tours (in terms of truck costs) to limit computational efforts.

1. **Remove drop-off point.** Removes a drop-off point from the current tour. Since truck distance is a main cost driver, this often leads to improvements.

2. **Remove depot.** Removes a depot. Depot removal may lead to non-feasible tours with respect to robot availability. In the event of non-feasibility, the closest depot is appended to the end of the tour.

3. **Add depot.** Adds a new depot to the existing tour. Additional depots can increase robot availability on parts of the tour and thus lead to better robot schedules at reduced costs.

4. **Add drop-off point.** Adds a new drop-off point to the tour. This may reduce robot usage or delays by bringing robots closer to the customers.

5. **Swap two stops.** Swaps two stops, i.e., both robot depots and drop-off locations. By swapping two stops, truck distance can be reduced or delays of deliveries starting at the later stop can be avoided.

6. **Relocate stop.** Shifts a single stop within a tour. Depending on whether the stop is shifted to an earlier or later point on the tour, the delays occurring at this stop or following stops can be reduced.

The VND results in n individual truck-and-robot routes serving each customer exactly once. As we generate each tour separately, the routes can still rely on the same robots in the same depots such that in combination this may lead to non-feasible solutions considering all tours. Nevertheless, the tours found provide an efficient starting solution for the improvement phase.

5.4.2 Improvement phase

Using the initial routing for the customer clusters found, the improvement phase searches for an improved customer clustering, routes and robot schedules. It therefore varies given tours and the corresponding clustering of customers to these tours.

Tour variation

In this step, we generate tour candidates as a potential basis for improvements. To build this set, a pool of variations is generated for each incumbent tour. Each pool is built by applying the following operators, which were most effective in our pretests. The operators are applied to modify the stops at a tour by changing depots and drop-off points. Since depots are crucial for robot availability, particular emphasis is given to inserting new depots into a tour.

- **Remove up to two stops.** Every possible tour resulting from the removal of one or two stops is added to the pool.
- **Insert a depot.** The m^{VAR} cheapest truck tours (based on costs for truck time and distance) obtained by inserting a depot are added to the pool.
- **Insert a drop-off point.** The m^{VAR} cheapest truck tours obtained by inserting a drop-off point are added to the pool.
- **Replace a stop by a depot.** The m^{VAR} cheapest truck tours obtained by replacing an existing stop with a depot are added to the pool.
- **Insert two new depots.** The m^{VAR} cheapest truck tours obtained by inserting two depots that are not yet on the tour are added to the pool.
- **Adapt departure time.** Shift the departure of the truck to an earlier or later start time.

The start time of a tour significantly impacts times and costs, and often leads to improvements due to new options for customer supply. Hence, the last operator speeds up the search in the event that a significant change of the departure time is beneficial (which could otherwise only be achieved in several iterations when adding or removing a maximum of two stops). Tour variation results in one pool for each incumbent tour.

Before these pools can be passed to the next step of the algorithm to select the best combination of tours, we need to identify non-feasible tour combinations and exclude them. This is necessary as robot availability has been relaxed so far and may be violated. In the MVTR-RP, multiple tours access the same robot depots and respective robot availability, i.e., robot availability is shared between tours (see

Equation (5.17)). For a computationally efficient implementation of this constraint, we assume that robot depots can be refilled automatically (i.e., by transferring robots between depots) within a given refill time. No two tours may access the same robot depot during this time. We introduce the parameter ϑ^{f} to indicate the refill time, thus defining a waiting time until the next tour may access a given robot depot. As a consequence, we forbid the selection of two tours for the routing solution if these tours visit the same depot with less than ϑ^{f} time in between. Note that a very large ϑ^{f} ensures that depots are not shared between trucks at all and no refill is assumed. We exclude the combination of any two tours from the same pool (i.e., derived from the same incumbent tour) in the same way, as this proved inefficient. The rationale behind this is that incumbent tours can be simultaneously adapted to make improvements, but choosing several very similar tours proved unattractive.

However, excluding tour combinations may lead to overall non-feasibility when selecting tours in the next step, as it can happen that no two tours can be combined due to the use of common depots. This may lead to insufficient depot visits to ensure a sufficient robot availability. We therefore create another variant of each incumbent tour by sequentially replacing every depot also included in another incumbent tour by the closest depot not included in any incumbent tour. This results in one additional variation for each of the n incumbent tours and prevents non-feasibility. The result of this step is a set of T tours and a matrix $b_{\tau,\chi}$ defining whether any two tours τ and χ can be used at the same time.

Optimal tour selection, customer clustering and robot scheduling

This step selects the optimal set of tours and provides a feasible solution for the MVTR-RP based on the tours created within the tour variation step. This means that it simultaneously defines the truck tours used, the customer clustering (i.e., from which truck each customer's robot starts) and the corresponding robot schedule (i.e., from which stop each customer's robot starts). Table 5.3 summarizes the notation of truck tour parameters and decision variables.

Problem parameters	
T	Set of potential tours, $\tau \in T$
U_τ	Index set of stops on the truck tour τ; $\tau \in T$; $u \in \{1, 2, ...\}$
Y_τ	Tuple of truck stops on tour $\tau, \tau \in T$, where element $y_\tau(u)$ is the u-th stop of τ; $y_\tau(u) \in L$
n	Number of available trucks, $n \leq \|T\|$
$c^{\mathrm{T}}_{k,\tau,u}$	Costs (for robot travel/ waiting time and potential delay) of supplying customer k from tour τ, at stop u, $u \in U_\tau$
c^{f}_τ	Total truck costs (incl. time and distance) of using tour τ
$b_{\tau,\chi}$	1, if only tour τ or χ can be used; 0 if both can be used, $\tau, \chi \in T$
Decision variables	
$x_{k,\tau,u}$	Binary: 1, if customer $k, k \in C$, is supplied from tour $\tau, \tau \in T$, at stop u, $u \in U_\tau$; 0 otherwise
z_τ	Binary: 1, if tour $\tau, \tau \in T$, is used; 0 otherwise
$q_{\tau,u}$	Number of robots aboard the truck on tour $\tau, \tau \in T$, at departure from stop $u, u \in U_\tau$
$\beta_{a,\tau,u}$	Number of available robots in location $a, a \in L$, for tour τ after the u-th stop, $u \in U_\tau$

Table 5.3: Notation for the optimal customer clustering to given tours

Since a set of potential truck tours τ is given, drop-off (D) and robot depot locations (R) do not need to be duplicated, and $L := D \cup R$ is the set of all locations potentially reachable by truck. A truck tour τ is defined by a tuple Y_τ, where $y_\tau(u)$ is the location of the u-th stop, $y_\tau(u) \in L$. From the set of all potential tours, we want to select a maximum of n tours and allocate all customers to the stops of these tours in a cost-minimal manner. For each tour $\tau \in T$, we pre-calculate the arrival times $\psi_{\tau,u}$ at its stops u based on the truck travel times. With known robot travel times and customer deadlines we can then calculate the robot travel and the waiting and delay costs $c^{\mathrm{T}}_{k,\tau,u}$ of serving a customer k from stop u on tour τ as shown in Equation (5.32). Its first term represents the robot costs for travelling from the stop to the customer and waiting at the customer if needed. The second term adds the cost of a potential delay of the delivery. Note again, as the robot always returns to the closest depot, the return costs are not decision relevant.

$$
\begin{aligned}
c^{\mathrm{T}}_{k,\tau,u} &:= c^{\mathrm{rob}}(\vartheta^r_{y_\tau(u),k} + (d_k - \epsilon - \psi_{\tau,u} - \vartheta^r_{y_\tau(u),k})^+) \\
&+ c^{\mathrm{late}}(\psi_{\tau,u} + \vartheta^r_{y(u),k} - d_k)^+ \qquad \forall u \in U, k \in C
\end{aligned}
\tag{5.32}
$$

Furthermore, every tour is associated with fixed total tour costs c^{f}_{τ} incurred for the truck's travel time and distance. The decision variables $x_{k,\tau,u}$ define whether customer k is served from stop u of tour τ. Variable z_τ states whether tour τ is used at all. The auxiliary variables $q_{\tau,u}$ and $\beta_{a,\tau,u}$ keep track of available robots on a truck during its tour and in the depots visited by a tour. The objective function and constraints are formulated as follows:

$$\min \; F(X) = \sum_{k\in C}\sum_{\tau\in T}\sum_{u\in U_\tau} c_{k,\tau,u}\cdot x_{k,\tau,u} + \sum_{\tau\in T} c^{f}_{\tau} z_\tau \tag{5.33}$$

subject to

$$\sum_{\tau\in T}\sum_{u\in U_\tau} x_{k,\tau,u} = 1 \qquad \forall k\in C \tag{5.34}$$

$$\sum_{k\in C}\sum_{u\in U_\tau} \eta_k x_{k,\tau,u} \le G z_\tau \qquad \forall \tau\in T \tag{5.35}$$

$$z_\tau + z_\chi \le 1 \qquad \forall \tau,\chi\in T : b_{\tau,\chi}=1 \tag{5.36}$$

$$\sum_{\tau\in T} z_\tau \le n \tag{5.37}$$

$$\beta_{a,\tau,u} \le \beta_{a,\tau,u-1} + q_{\tau,u-1} - q_{\tau,u} - \sum_{k\in C} x_{k,\tau,u} \qquad \forall a\in L, \tau\in T, u\in U_\tau : a = y_t(u) \tag{5.38}$$

$$\beta_{a,\tau,u} = \beta_{a,\tau,u-1} \qquad \forall a\in R, \tau\in T, u\in U_\tau : a \ne y_t(u) \tag{5.39}$$

$$q_{\tau,0} = \delta \qquad \forall \tau\in T \tag{5.40}$$

$$\beta_{a,\tau,0} = \beta_a \qquad \forall a\in R, \tau\in T \tag{5.41}$$

$$\beta_{a,\tau,u} = 0 \qquad \forall a\in D, \tau\in T, u\in U_\tau \tag{5.42}$$

$$x_{k,\tau,u} \in \{0,1\} \qquad \forall k\in C, \tau\in T, u\in U_\tau \tag{5.43}$$

$$z_\tau \in \{0,1\} \qquad \forall \tau\in T \tag{5.44}$$

$$\beta_{a,\tau,u} \ge 0 \qquad \forall a\in R, \tau\in T, u\in U_\tau \tag{5.45}$$

$$0 \le q_{\tau,u} \le Q \qquad \forall \tau\in T, u\in U_\tau \tag{5.46}$$

The objective function (5.33) minimizes the sum of the costs of robot travel, waiting and potential delay and the costs of all truck tours selected. Constraints (5.34) ensure that every customer is supplied by exactly one robot and Constraints (5.35) that robots are only started from tours used, not exceeding the goods capacity of each truck. Constraints (5.36) allow the definition of tour combinations excluded and (5.37) limit the total number of tours. Constraints (5.38) and (5.39) keep track of the robots available to a tour in each location and on the truck, depending on whether the location is visited (i.e., $a = y_\tau(u)$) or not. Note that due to Constraints (5.36), there is no coupling of robot availability between tours required. The variable $\beta_{a,\tau,u}$ is only needed since the same tour could visit a robot depot several times. Equations (5.40) state the initial number of robots aboard the trucks, and (5.41) define the same for each robot depot. Constraints (5.42) ensure robots cannot be stored at drop-off points. Finally, Constraints (5.43)-(5.46) define the variable domains.

Iterations The MIP returns a feasible solution for the MVTR-RP. We set a runtime limit of m^{MIP} for the MIP to ensure time-efficient iterations. This is done as the solver otherwise spends considerable time to prove an optimum while this does not lead to better solutions. If one tour of the previous solution was eliminated (i.e., none of this tour‘s variations was chosen) and thus less than n tours are selected, this tour will be added to the potential tours τ in future iterations. This ensures that both an increase and decrease of tours is possible in subsequent iterations. When no improvement is found for the first time, the time limit for the MIP solver is increased and the next improvement iteration starts. When no improvement is found for the second time, the heuristic terminates and the current set of tours (together with the customer assignment to the stops of these tours identified by the MIP) are returned as the best solution.

5.5 Numerical studies

This section completes numerical studies to obtain insights into computational performance of the TRCR and managerial implications related to the MVTR-RP. We describe the parameter setting applied in our experiments in Section 5.5.1. Section 5.5.2 investigates the performance of TRCR compared to benchmark approaches. We further provide managerial insights by means of sensitivity analyses on depot refill times, time window distributions, and fleet sizes (Section 5.5.3).

5.5.1 Instances, parameter setting and test bed

We apply a 4×4 km delivery area that resembles the northern half of the Munich (Germany) city center. In the default data set, we assume a customer set $|C| = 50$, and randomly select 50 building locations in the delivery area obtained from OpenStreetMap Foundation (2019). Up to $n = 3$ trucks are available for this delivery area. $|R| = 25$ depots are first distributed in an equidistant manner and then slightly shifted by a random distance between 0 and 500 meters in south-north and east-west direction. $|D| = 48$ drop-off points are distributed by a random uniform distribution. The warehouse is randomly selected from the set of depots and drop-off locations. For the determination of customer deadlines we assume a random-uniform distribution. The interval for the deadlines is defined as $[t^{\mathrm{M}} \cdot \rho_{min}, t^{\mathrm{M}} \cdot \rho_{max}]$, where t^{M} is the time needed to travel from the starting point to the furthest customer by truck. The parameters are set to $\rho_{min} = 3$ and $\rho_{max} = 6$ in the default case. The initial number of robots is set to $\beta_a = 0.08 \cdot |C|$ in every depot $a, a \in R$ and the depot refill time to $\vartheta^{\mathrm{f}} = 15$ min. Similarly, the truck's capacity and initial number of loaded robots is set ($Q = \delta = 0.08 \cdot |C|$). The parcel volume per customer is $\eta = 1$ and the truck's parcel capacity is $G = 100$. The average speed is 30 km/h for the truck and 5 km/h for the robots. A handling time per stop of 40 sec is added to the resulting travel times. Following the costs empirically derived by Ostermeier et al. (2021a), we assume the cost rates of $c^{\text{dist}} = 0.20$ €/km and $c^{\text{veh}} = 30$ €/h for the truck, $c^{\text{late}} = 5$ € for delays and $c^{\text{rob}} = 1.0$ €/h for robot use. Please note that we also apply the robot costs to the return time from the customer to the closest depot to enable a fair comparison of total costs. As explained above, these costs are not decision relevant (as known a priori) but contribute to the total costs of the MVTR-RP.

Within the TRCR, the applied values of σ_1 and σ_2 (share of depots defining circle areas) in the *initial truck tour generation* are chosen as $[0.0, 0.3]$ and $[0.3, 0.8]$. We set the number of points considered as potential stops between two depots in the start heuristic to $m^{\mathrm{e}} = 5$. The VND is parameterized with the probability of accepting a worse solution $p = 0.5$, its decrease per cycle $\Delta p = 0.001$ and the maximum neighborhood size $m^{\mathrm{VND}} = 20$. The maximum number of tours for each operator used in the *tour variation* step is set to $m^{\mathrm{VAR}} = 20$, and the time limit m^{MIP} for solving the *optimal tour selection, customer clustering and robot scheduling* to 1 min at the start and 10 min after the increase. The default setting of the base scenario is highlighted in the following charts with a bold x-label. We generate 20 instances for each set of parameters, each with different locations and deadlines. Henceforth, each data point shown represents the average across 20 instances.

Our approach was implemented in Python (using PyCharm 2018.3.5 Professional Edition) with Gurobi (version 8.0.1) as MIP solver and executed on a 64-bit PC with an Intel Core i7-8650U CPU (4×1.9 GHz), 16 GB RAM, and Windows 10 Enterprise.

5.5.2 Efficiency of the solution approach

The SVTR-RP already constitutes an $\mathcal{NP}$-hard problem and thus no comparison for practically relevant problem sizes of the more complex MVTR-RP is possible. For example, the optimization of an MVTR-RP instance with six customers was terminated with an optimality gap of 70% after 10 hours of computation. As there is no existing routing approach for the MVTR-RP, we apply the GVNS framework for the SVTR-RP by Heimfarth et al. (2021) as a benchmark for single truck problems. This benchmark approach (denoted as *GVNS*) is most suitable as it is aimed at total logistics costs as well. Moreover, we extend the GVNS with a customer clustering to tours in a second step to enable a comparison for the MVTR-RP. For the customer clustering, we solve a clustering MIP (see Appendix C) for a given number of trucks, assuming standard truck deliveries. To reduce computation times, we relax the time window constraint to deadlines within the MIP. We further limit the search to 60 minutes. This is then input to the GVNS for subsequent routing of each cluster. We denote this benchmark approach *MIP&GVNS*. Compared to the integrated approach of TRCR with simultaneous customer clustering, truck routing and robot scheduling, the MIP&GVNS can be described as a cluster-first-route-second approach. As an alternative clustering, we also tested the clustering heuristic by Fisher and Jaikumar (1981), followed again by the GVNS. However, the resulting solution quality is worse than in the case of the MIP&GVNS. Please note that MIP&GVNS does not prevent the different trucks from using the same robots at depots, and as such provides a simplified search in favor of the benchmark approach.

Special case of a single truck We set the number of available trucks to $n = 1$ to compare TRCR directly to the specialized GVNS by Heimfarth et al. (2021). Figure 5.6 shows the comparison for different instance sizes. In this special case, TRCR comes close to the solution quality of GVNS, which has been developed specifically for such settings. Our approach leads to costs that are only 1% higher on average. This is an acceptable gap as TRCR is designed to solve the MVTR-RP with a larger pool of tours for several available vehicles. The GVNS on the other hand is tailored to a single truck and consequently uses tailored operators to improve the SVTR-RP. The computation time is two to three times (up to 8 minutes) higher, as TRCR also solves the *tour selection, customer clustering and robot scheduling* MIP, although this is not needed in the case of a single truck. In summary, this validates TRCR's ability to find good solutions compared to state-of-the-art approaches for the SVTR-RP.

Performance comparison for multiple trucks The routing of multiple trucks is at the core of the MVTR-RP. Figure 5.7 shows the computation times and logistical performance obtained from our TRCR (simultaneous approach) and the MIP&GVNS (cluster-first-route-second approach) for different numbers of customers. The TRCR is 23 - 60% faster, especially since the clustering MIP always reaches its time limit and then requires additional time for the routing. TRCR computation times are generally at a level acceptable for practical use, as the tours can be planned during picking time in the warehouse. Moreover, TRCR is able to reduce total costs by 18

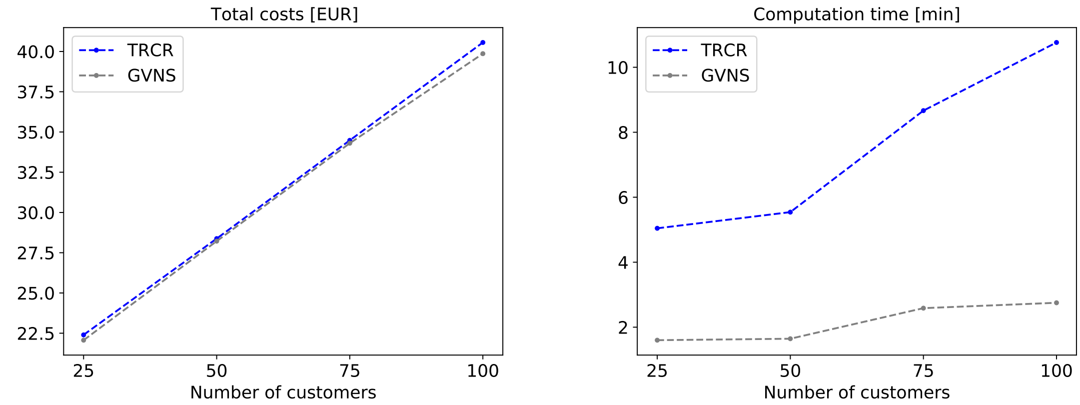

Figure 5.6: Comparison of TRCR vs. GVNS routing for SVTR-RP, average of 20 instances

- 24%. The MIP&GVNS results in the use of too many trucks and a suboptimal clustering of customers resulting in longer delays and truck distances. Consequently, TRCR leads to a pareto improvement compared to the cluster-first-route-second benchmark. Furthermore, the average objective value reduction in the improvement phase of TRCR is 18%, which highlights the effectiveness of both our start heuristic and improvement phase.

Benchmark analysis with varying spatial customer distribution To assess the robustness of the results of the TRCR, we apply different customer distributions and compare TRCR again with the MIP&GVNS. We denote the default case as *uniform* and create two further spatial distributions. First, we select only building locations from the lower left and upper right quadrant of the selected 4×4 km square area, resulting in a *clustered* distribution. By selecting only building locations from the central 2×2 km square, we obtain a *concentrated* distribution. Table 5.4 shows that TRCR performs very well for all customer distribution settings, reducing total costs by 12% (concentrated) to 23% (uniform).

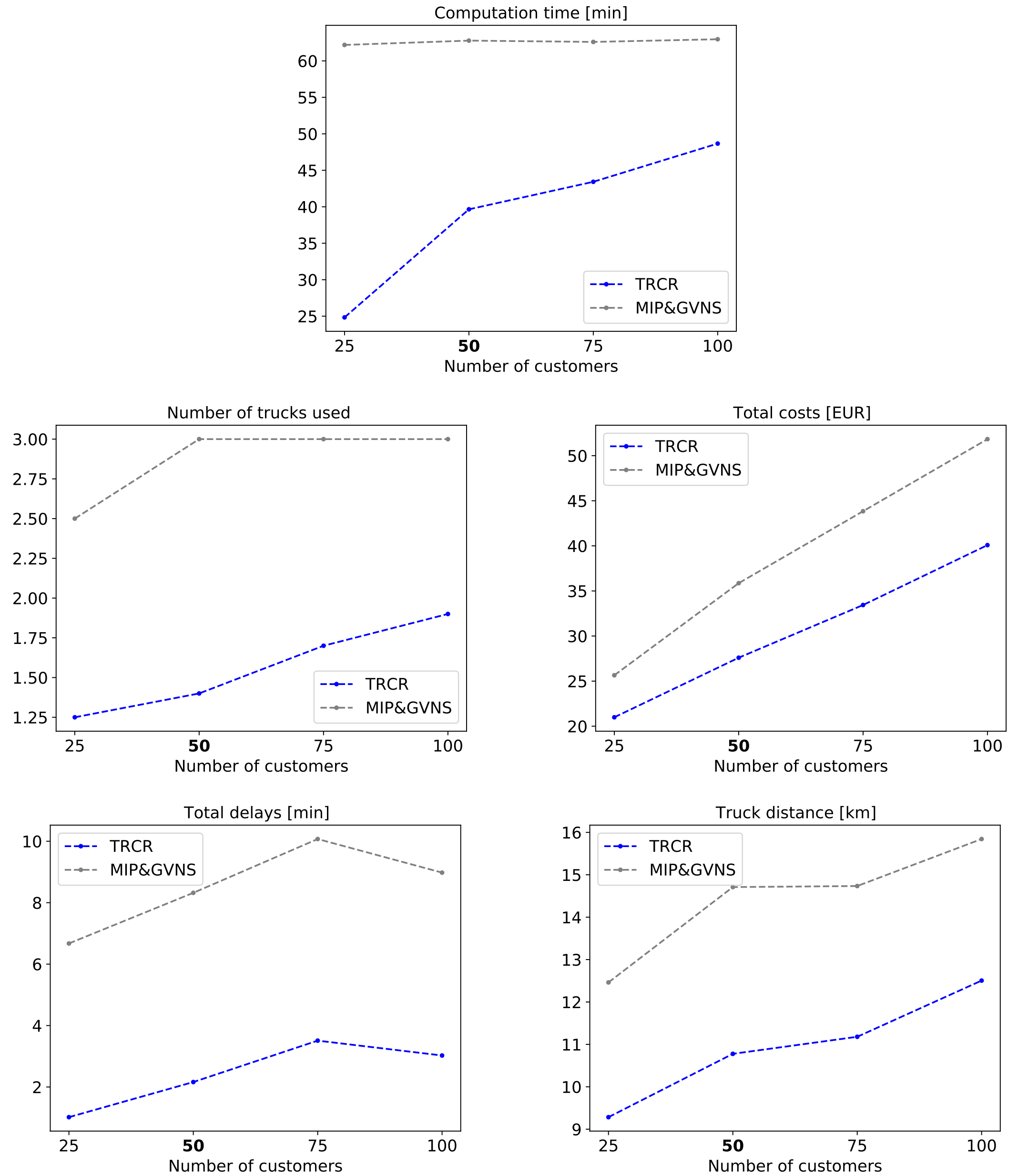

Figure 5.7: Comparison of TRCR vs. MIP&GVNS for MVTR-RP, average of 20 instances

Benchmark analysis with varying truck capacity Limited truck capacity can lead to solutions with more trucks or less efficient routes. In the previous section,

Customer distribution	Improvement of TRCR vs. MIP&GVNS, in % of MIP&GVNS: Total costs	Computation time	Number of trucks used	Truck distance
Uniform	23	37	53	27
Clusters	20	58	42	20
Concentrated	12	42	41	25

Table 5.4: TRCR vs. MIP&GVNS for different spatial customer distributions, average of 20 instances

it was only the deadlines that motivated the use of additional trucks. We therefore reduce the trucks' parcel capacity G for the case with 100 customers of $G = 100$ to $G = 80$ and $G = 60$. With one parcel volume unit per customer ($\eta = 1$), G customers can be served per truck. The results are summarized in Figure 5.8. TRCR results are very stable across different truck capacities. We obtain the following insights when comparing the TRCR solutions with varying truck sizes. Reducing capacity from 100 to 60 increases the average cost by 0.3% and the computation time by 23%. As expected, the number of tours (+8%) and the truck mileage increase (+4%). The delays increase only slightly, at a low absolute level. Overall, this shows that truck capacity does not have a crucial impact on logistical performance as long as there is sufficient total capacity across the trucks to serve customers. The TRCR efficiently handles instances with tighter truck capacities as well. Finally, the TRCR again outperforms the MIP&GVNS solutions. The latter do not change across capacity sizes, as none of the truck tours obtained serve more than 60 customers.

5.5.3 Managerial insights

Having shown the efficiency of the proposed algorithm, we now analyze the impact of hierarchical and strategic managerial decisions that are entered into the MVTR-RP as input parameters and compare the MVTR-RP to conventional truck deliveries without robots. First, the assumption of depots being refilled is analyzed, as this is a key assumption in the truck-and-robot concept and also in our solution approach.

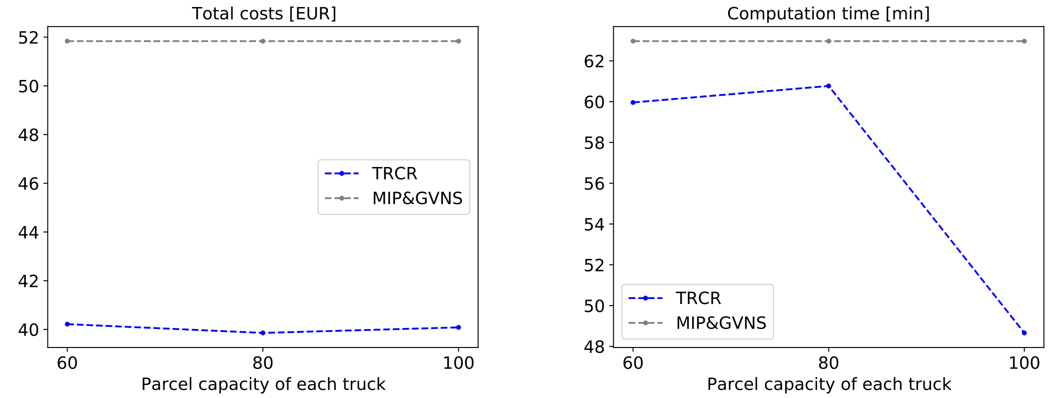

Figure 5.8: TRCR vs. MIP&GVNS for varying truck capacity and 100 customers, average of 20 instances

Next, the time window distribution is varied, since time windows are expected to have a strong influence on the customer clustering. Finally, we detail the benefits of additional trucks (and thus one further advantage of TRCR) in the case of tight time windows. To highlight the overall attractiveness of truck-and-robot systems, we henceforth compare the performance to traditional truck deliveries (i.e., all customers are supplied by trucks). We apply the MIP (see Appendix C) for this purpose, and we report the best-known solution (labeled *VRP*) and the lower bound of the costs (labeled as *VRP LB*) after 60 minutes of computation.

Robot depot refill assumptions and comparison with truck delivery Robot availability at robot depots is limited. The time required to refill a depot with new robots after a truck visit could have a significant impact on solution quality if visiting a single depot several times could be beneficial. We analyze the impact of this time on a truck-and-robot system with 25 (default case as described above) and 12 depots, which also leads to less available robots. The two scenarios are denoted TRCR-12 and TRCR-25. Figure 5.9 shows that there is hardly any impact on costs and other performance metrics with varying refill times for solutions obtained with TRCR. In particular, even if the time is set to 0 (i.e., two trucks arriving at the depot at the same time can both access the depot's full robot availability), this does not lead to significant improvements. This means that the assumption in our solution approach (which prevents two trucks visiting the same depot within the depot refill time) does not worsen the results. It further shows that there is no benefit in ensuring an immediate refill of visited depots in practice. Reducing the number of depots from 25 to 12 leads to longer truck distances traveled and a cost increase of 17% on average. Comparison with the conventional truck-only delivery reveals cost savings of 62% and a truck mileage reduction of 71% due to the truck-and-robot concept. This corresponds to a 71% reduction of local emissions if diesel trucks are used. The MIP is terminated after 60 min and results in a 45% average MIP gap. However, even compared to the lower bound, TRCR has a cost savings potential of 30%.

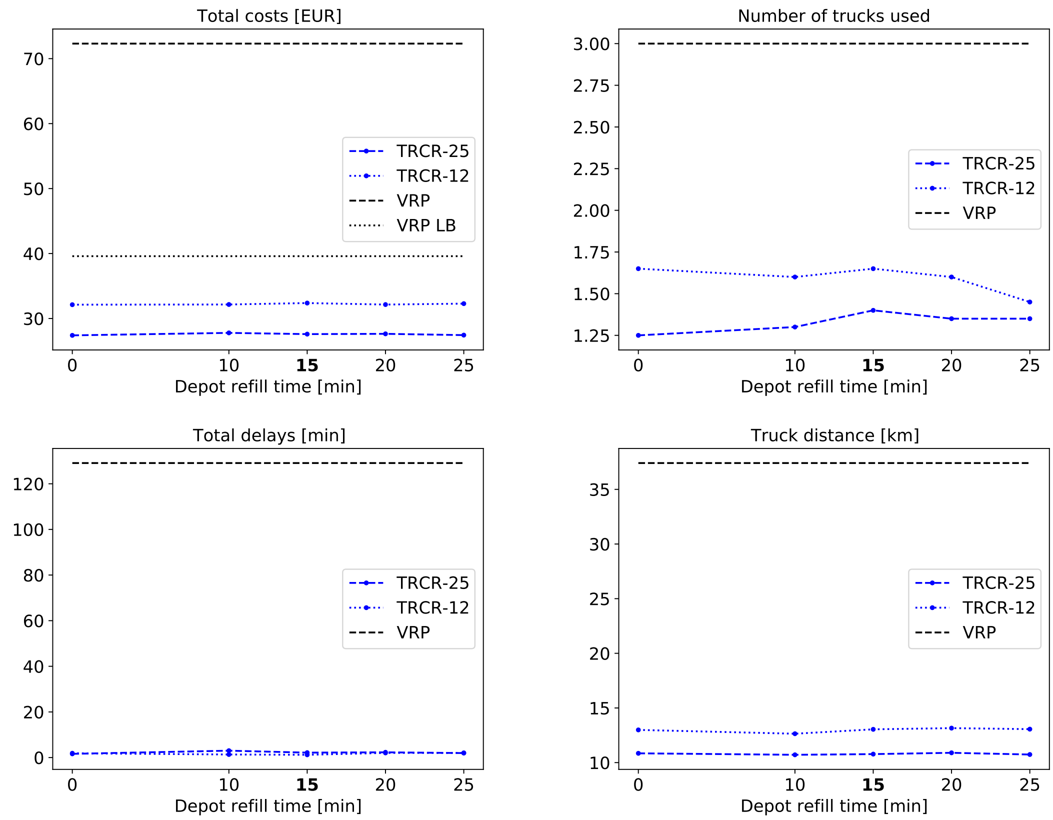

Figure 5.9: System performance for different depot refill times, average of 20 instances

Time window structure The key feature of TRCR is the clustering of customers, which must happen based on their locations and time windows, as these are the only differences between any two customers. Therefore, besides the influence of locations (see Section 5.5.2), we further assess the impact of time windows, which are expected to have a strong impact on costs and logistical performance (see e.g., Heimfarth et al. (2021)). First, the effect of earlier and later time windows is assessed. Next, customers in the same region are offered similar time windows.

(i) Early vs. late time windows. One important question when designing a last-mile delivery system is how fast a delivery can reach the customer. We therefore test the system's performance with earlier or later time windows, i.e., different values of the deadline factor interval $[\rho_{min}, \rho_{max}]$. Figure 5.10 summarizes results for a ρ_{min} of 1 (earlier), 3 (default) and 5 (later). With $\rho_{max} = \rho_{min} + 2$ the span of all deadlines remains constant. Comparing the results of TRCR for $\rho_{min} = 3$ vs. 5, we see that additional trucks can ensure that earlier deadlines are met at scant additional costs. On the other hand, if time windows are too early ($\rho_{min} = 1$ instead of 3), even increasing the number of trucks used by 61% cannot prevent a 19-fold increase in delays and further leads to a 35% longer distance travelled by the trucks. Together, these effects increase total costs by 22%. There is little variance in the number of trucks used: for $\rho_{min} = 1$, this number is 2 or 3 in all solutions; for $\rho_{min} = 3$ or 5, only 1 or 2 trucks are used. This shows that a truck-and-robot system must be designed specifically to meet the lead times promised. In times of high demand, offering later time windows to customers can relieve the system. The scenario with truck deliveries (again denoted as *VRP*) benefits from later deadlines as well, as it can reduce its high level of delays. However, it does so by further increasing truck distance. The cost savings of using the truck-and-robot system remain high as a result, at 54% with the later deadlines.

(ii) Location-based time windows. Assigning similar time windows to adjacent customers could potentially enable shorter truck tours and reduce robot waiting time and delays. We therefore test two policies for making such an assignment. Each policy splits the customers into two equal groups and assigns each group to one half of the deadline factor interval such that the first half of all deadlines is in one region and the second half in the other region. The first policy (denoted *distance*) splits the customers based on their distance from the warehouse and assigns the closer customers the earlier deadlines. The second policy (denoted *angle*) finds a

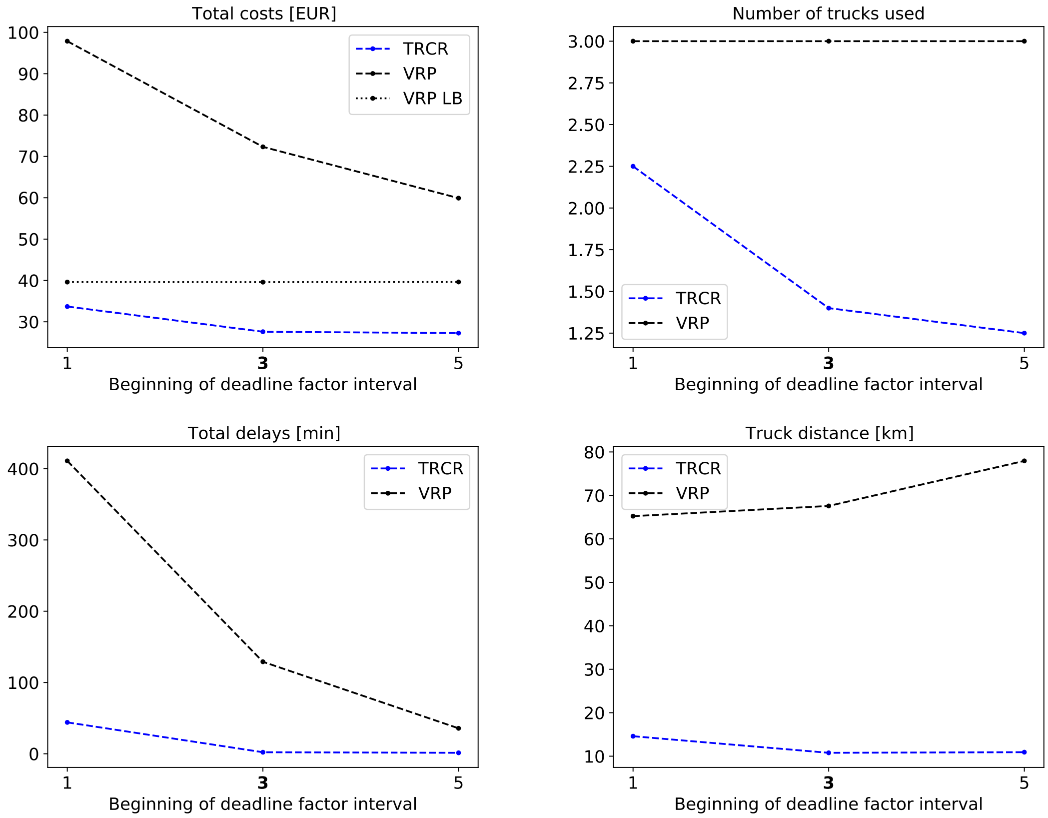

Figure 5.10: System performance for different deadline factor intervals

straight line through the warehouse that separates the customers into two equal groups and randomly assigns one to the earlier and one to the later half of the deadlines. We compare the TRCR results for these scenarios with the default case of completely random deadlines. Both types of zone lead to a decrease in delays (angle -68% and distance -85%). Interestingly, assigning zones based on distance does so while at the same time reducing the number of trucks used (-4%) and their total distance traveled (-4%). This shows that the same truck can serve the closer customers with early deadlines first and then take care of the customers who are further away with later deadlines. In the *angle* case, more trucks (7%) are needed and they cover a larger distance (5%). Consequently, this time window distribution does not seem to facilitate efficient truck tours. The total costs in this scenario remain unchanged. Only the *distance* scenario enables a 2% cost reduction in total, thus leading to a pareto improvement (all metrics considered are improved). It is therefore a favorable policy to offer customers a time window based on their distance from the warehouse.

Fleet size One key question when implementing a truck-and-robot system is how many trucks will be needed. This is also necessary to plan the shifts. We therefore investigate the effect of the number of available trucks on the system's performance. The instances used were generated with a deadline factor interval of $[\rho_{min}, \rho_{max}] = [1, 3]$. This leads to best solutions using two or three trucks in the default case of three available trucks. Table 5.5 shows the results with a varying number of available trucks. Our derived cost function for trucks includes a distance-based fee for trucks. That means that the fixed costs for a truck are transferred to mileage costs. With a total lifetime mileage, the total costs of a truck (incl. variable and fixed costs) can be calculated per distance unit (see Hübner and Ostermeier (2019) for a similar approach). TRCR tends to use more trucks whenever they become available even if this does not really reduce total costs: these are reduced by 24% when a second truck is added, mostly due to a 74% reduction in delays. After that, additional available trucks have no effect on total costs, as the reduction in delays is outweighed by the increase in truck distance. This shows that it can be necessary to apply several trucks and TRCR can help to optimally size the truck fleet for a given demand scenario. In the VRP case (not shown in Table 5.5), all available trucks are used to reduce the high level of delays. Consequently, fewer available trucks result in a high cost increase driven by delays.

	Number of available trucks			
	1	2	3	4
Total costs [EUR]	44	34	34	34
Number of trucks used	1.0	2.0	2.3	2.5
Total delays [min]	191	49	44	41
Truck distance [km]	12.4	14.2	14.6	14.6

Table 5.5: Performance change of TRCR for different available truck fleets, average of 20 instances

5.6 Conclusion

Truck-and-robot systems will contribute to reduce costs, emissions and traffic congestion caused by last-mile delivery. Multiple trucks combined with robots are needed to enable large scale application and short lead times. Since state-of-the-art literature is limited to single-truck problems (e.g., Boysen et al. (2018a), Alfandari et al. (2019) or Ostermeier et al. (2021a)), we present an extension to truck-and-robot routing by allowing the use of multiple trucks. The resulting *multi-vehicle truck-and-robot routing problem* (MVTR-RP) requires the simultaneous clustering of customers to truck tours and routing of each tour. To simultaneously solve the customer clustering, truck routing and robot scheduling problem, we propose a novel heuristic, the Set Improvement Neighborhood Search (SINS). The approach further relies on tailored start heuristics, problem-specific neighborhood operators and estimates leading to solvable MIPs. Our numerical experiments show that this simultaneous solution is 23 to 60% faster and yields 18 to 24% better solutions than a benchmark that is based on a cluster-first-route-second approach. The solution approach yields robust results overall, demonstrating the advantages of the truck-and-robot concept compared to various benchmarks and alternative delivery concepts. Further numerical studies show that using multiple trucks instead of one for the same delivery area and time windows reduces delays by about 75% on average, only slightly increases total truck distance by about 15%, and hence results in total costs improvements of about 20 - 25%. This highlights the benefits of multiple trucks. We were able to further identify cost reductions by 62% and truck emissions by 71% compared to conventional truck delivery. This improvement potential further increases with more challenging time windows. Sizing the truck fleet, defining the time windows customers can choose and the total number of available robot depots are identified as key decisions for the hierarchical planning of truck-and-robot operations.

As we deal with an innovative delivery concept, there are still many promising opportunities for future research. One could build on our approach to develop exact solution approaches. Furthermore, the SINS approach of optimally choosing a set of tours from the neighborhoods of all incumbent tours has proven effective and could be transferred to other problems, for which the solution is a set of objects.

Further modifications to the truck-and-robot concept and the instances applied can be investigated as well, such as alternative delivery modes (by driver, by drone etc.) and different spatial situations. Our results show the attractiveness of truck-and-robot in a city, but the system‘s high flexibility could make it adaptable to more rural settings, too. This could require the introduction of additional delivery modes. The concept can be further extended to pickup and delivery operations, as customer returns are becoming increasingly important for logistics operations. A heterogeneous fleet of trucks with varying capacity and also range limitations should be assessed. This would be particularly relevant if an existing truck fleet is successively replaced with electric vehicles. It would require additional decisions and constraints in the tour selection step. More generally, there are plenty of applications and approaches for clustering. One could adapt our approach to related clustering applications or examine the effectiveness of existing clustering approaches for the MVTR-RP in a cluster-first-route-second setting. Machine learning methods seem a promising way forward in view of the complex interdependencies of the various problem aspects involved.

6 Conclusion and outlook

6.1 Conclusion

Every individual contribution concludes by summarizing the methodological and managerial insights and proposing extensions and improvements as subject to further research. In this section, our findings are stated in a more aggregated way and viewed in a wider context.

Potential of truck-and-robot delivery Within the work presented, we assessed the economical and environmental advantages of truck-and-robot deliveries. In the scenarios analyzed, the concept displays the potential to reduce costs and emissions by more than 50%. It demonstrates a high flexibility and robust performance across different settings. The improvements observed mean a substantial relief for urban traffic and pollution and at the same time make fast premium deliveries viable and affordable for many new use cases. As an example, truck-and-robot could be used to extend the delivery areas of restaurants and local stores. We therefore believe truck-and-robot should be tested in the field on a larger scale. Our heuristics can help to assess the implications of truck-and-robot use in specific settings prior to implementation and ensure efficient operations afterwards.

The key success factors for the implementation of truck-and-robot systems are the following. Autonomous driving must work reliably most of the time. This reduces remote control effort and thus the cost of operating robots. Furthermore, it would enable deregulation of robot use, which is still needed in many locations before robots are allowed to move completely unsupervised. Another prerequisite for the concept are robot charging stations in cities. These could be included into regular electric car charging stations and (if appropriately standardized) shared with other small electric vehicles such as scooters and bikes. Another possibility is to leverage a retailer‘s or logistics company‘s existing store infrastructure to provide charging stations. Besides these technical and regulatory requirements, it is crucial that a robot fleet reach a certain scale and utilization to be run profitably. The only companies who could reach this are very large online retailers or dedicated robot sharing providers, who offer short term rentals (similar to car sharing) to any retailer, logistics provider and consumer. Finally, consumers’ sentiment towards robots will play a crucial role for the adoption of the technology. This includes the willingness to retrieve an order from a robot, but also the public acceptance of robots moving on the sidewalk.

Methodological findings Different heuristic approaches are presented in the three publications. All of them have in common that truck routes are improved via search operators and that robot movements for given truck tours are solved in an exact manner. Obtaining exact solutions with a solver for parts of the problems is a strong driver of runtime, but it proved very advantageous for solution quality in our different tests during the development phases. Furthermore, it makes resulting tours and weaknesses of the overall heuristic more interpretable.

The most complex problem is solved in Contribution 3, as it considers multiple trucks. We think that some of its heuristic components can be transferred to similar problems. First, the principle of clustering customers with delivery time windows based on their fit to a pool of given candidate tours can be used in any routing problem involving several vehicles and time windows. Second, we proposed the concept of Set Improvement Neighborhood Search (SINS), which tries to improve a set of objects by choosing a new set from the neighborhoods of all objects. This could be applied not only to various routing decisions, but whenever a set of similar objects must be defined, e.g. when configuring machines or vehicles.

6.2 Future areas of research

There are numerous opportunities for future research. The truck-and-robot concept can be further extended, e.g. to pick-up and delivery scenarios. The role of a potential robot fleet operator could further be investigated, including problems such as network balancing and dynamic pricing. Further offerings based on shared robots can be developed, such as direct delivery from stores and restaurants. While these use cases are less complex from a routing perspective, it would be interesting to assess their profitability and implications for urban traffic. Further, a life cycle sustainability assessment of delivery robots should be conducted to account for the impact of robot manufacturing.

Methodologically, truck-and-robot delivery provides further areas of research as well. E.g., our heuristics can be enhanced and tuned to reduce computation times. As computing power increases, more and more sophisticated approaches such as Genetic Algorithms become feasible for this problem. Due to this development, it may soon be possible to further investigate the performance of our heuristics compared to exact methods for relevant problem sizes. Another promising approach to routing lies in machine learning methods. Reinforcement learning, as an example, has been used for small routing problems successfully and could soon be adapted to truck-and-robot routing problems as well. Further, our solution approaches can potentially be transferred to related applications involving several coupled vehicles, such as truck-and-drone delivery, ride sharing and flexible road trains. Even besides urban logistics, there are fields with similar problem structures, where vehicles depart from a mothership and their return is negligible. This can be relevant in disaster response via drones, where it is most important that drugs, food or other important supplies arrive on time, even if the drone cannot return from the destination due to its range. A similar problem can be found in research applications, where autonomous exploration vehicles (e.g., mars rovers) are brought to a remote area to remain there and report results via radio.

Finally, the field of sustainable city logistics offers a multitude of research topics. This can involve innovative vehicle concepts or smart ways of bundling and distributing passengers and freight. Further advancements in the areas of propulsion technology, autonomous driving, IT platforms and real-time processing can help achieve this. Even more broadly, ways to eliminate the need of transportation contribute to the solution. This includes 3D printing, near-shoring of industrial and agricultural production and technologies for engaging remotely for work, leisure and shopping.

Acknowledgements I want to thank Alexander Hübner and Manuel Ostermeier for the great collaboration and all the guidance they provided during this PhD project. It was an inspiration and fun to work with them. Further, I owe a debt of gratitude to the industry experts who have provided empirical data and practical insights as well as encouraged our research topic. I also thank all editors and reviewers of the scientific journals for their advice on improving this work.

Finally I want to thank my parents Helmut and Gabriele. They have always supported and encouraged me in the best way I could imagine.

Bibliography

Agatz, N., Bazzan, A. L. C., Kutadinata, R., Mattfeld, D. C., Sester, M., Winter, S., Wolfson, O., 2017. Autonomous car and ride sharing: flexible road trains. In: Hoel, E. G. (Ed.), 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL GIS 2017). The Association for Computing Machinery, New York, New York, pp. 1–4.

Agatz, N., Bouman, P., Schmidt, M., 2018. Optimization approaches for the traveling salesman problem with drone. Transportation Science 52 (4), 965–981.

Agatz, N., Campbell, A., Fleischmann, M., Savelsbergh, M., 2011. Time slot management in attended home delivery. Transportation Science 45 (3), 435–449.

Agatz, N. A., Fleischmann, M., van Nunen, J. A., 2008. E-fulfillment and multi-channel distribution - a review. European Journal of Operational Research 187 (2), 339 – 356.

Alfandari, L., Ljubic, I., de Melo da Silva, M., 2019. On last-mile delivery with autonomous robots. Working paper.

Allen, J., Piecyk, M., Piotrowska, M., McLeod, F., Cherrett, T., Ghali, K., Nguyen, T., Bektas, T., Bates, O., Friday, A., Wise, S., Austwick, M., 2018. Understanding the impact of e-commerce on last-mile light goods vehicle activity in urban areas: The case of london. Transportation Research Part D: Transport and Environment 61, 325–338.

Amazon, 2019. Premiumversand.
URL `https://www.amazon.de/gp/help/customer/display.html?nodeId=202013930`

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 25 (2), 345–376.

Arslan, A. M., Agatz, N., Kroon, L., Zuidwijk, R., 2019. Crowdsourced delivery—a dynamic pickup and delivery problem with ad hoc drivers. Transportation Science 53 (1), 222–235.

Bakach, I., Campbell, A. M., Ehmke, J. F., 2021a. Robot-based last-mile routing problem with stochastic travel times and level of service zones. Working paper.

Bakach, I., Campbell, A. M., Ehmke, J. F., 2021b. A two–tier urban delivery network with robot–based deliveries. Networks, 1–23.

Baum, L., Assmann, T., Strubelt, H., 2019. State of the art - automated micro-vehicles for urban logistics. IFAC-PapersOnLine 52 (13), 2455–2462.

Bekel, K., Pauliuk, S., 2019. Prospective cost and environmental impact assessment of battery and fuel cell electric vehicles in germany. The International Journal of Life Cycle Assessment 24 (12), 2220–2237.

Bennett, T., 2020. Autonomous last mile delivery robots are gaining traction, slowly. URL `https://www.sharedmobility.news/autonomous-last-mile-delivery-robots-are-gaining-traction-slowly/`

Bouman, P., Agatz, N., Schmidt, M., 2018. Dynamic programming approaches for the traveling salesman problem with drone. Networks 72 (4), 528–542.

Boysen, N., Briskorn, D., Fedtke, S., Schwerdfeger, S., 2018a. Drone delivery from trucks: Drone scheduling for given truck routes. Networks 72 (4), 506–527.

Boysen, N., Fedtke, S., Schwerdfeger, S., 2021. Last-mile delivery concepts: a survey from an operational research perspective. OR Spectrum (43), 1–58.

Boysen, N., Schwerdfeger, S., Weidinger, F., 2018b. Scheduling last-mile deliveries with truck-based autonomous robots. European Journal of Operational Research 271 (3), 1085–1099.

Buldeo Rai, H., Verlinde, S., Macharis, C., 2019. The "next day, free delivery" myth unravelled. International Journal of Retail & Distribution Management 47 (1), 39–54.

Chang, Y. S., Lee, H. J., 2018. Optimal delivery routing with wider drone-delivery areas along a shorter truck-route. Expert Systems with Applications 104, 307–317.

Chen, C., Demir, E., Huang, Y., 2021a. An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and delivery robots. European Journal of Operational Research, DOI: 10.1016/j.ejor.2021.02.027.

Chen, C., Demir, E., Huang, Y., Qiu, R., 2021b. The adoption of self-driving delivery robots in last mile logistics. Transportation Research Part E: Logistics and Transportation Review 146, DOI: 10.1016/j.tre.2020.102214.

Coelho, B. N., Coelho, V. N., Coelho, I. M., Ochi, L. S., Haghnazar K., R., Zuidema, D., Lima, M. S., da Costa, A. R., 2017. A multi-objective green uav routing problem. Computers & Operations Research 88, 306–315.

Condliffe, J., 2018. Why sidewalk delivery robots still need safety drivers.
URL `https://www.technologyreview.com/f/610107/why-sidewalk-delivery-robots-still-need-safety-drivers-too/`

Daimler, 2019. Vans & robots.
URL `https://www.daimler.com/innovation/specials/future-transportation-vans/paketbote-2-0.html`

Dayarian, I., Savelsbergh, M., Clarke, J.-P., 2020. Same-day delivery with drone resupply. Transportation Science, DOI: 10.1287/trsc.2019.0944.

de Freitas, J. C., Penna, P. H. V., 2020. A variable neighborhood search for flying sidekick traveling salesman problem. International Transactions in Operational Research 27 (1), 267–290.

DHL, 2020. Dhl locker.
URL `https://www.dhlparcel.nl/en/consumer/dhl-locker`

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

Figliozzi, M., Jennings, D., 2020. Autonomous delivery robots and their potential impacts on urban freight energy consumption and emissions. Transportation Research Procedia 46, 21–28.

Fikar, C., Hirsch, P., Gronalt, M., 2018. A decision support system to investigate dynamic last-mile distribution facilitating cargo-bikes. International Journal of Logistics Research and Applications 21 (3), 300–317.

Fisher, M. L., Jaikumar, R., 1981. A generalized assignment heuristic for vehicle routing. Networks 11, 109–124.

Forbes, 2019. Amazon's new delivery drone will start shipping packages 'in a matter of months'.
URL `https://www.forbes.com/sites/jilliandonfro/2019/06/05/amazon-new-delivery-drone-remars-warehouse-robots-alexa-prediction/#7c0c6f7d145f`

Glatzel, C., Großpietsch, J., Hübner, A., 2012. Higher margins through efficient supply chains. Akzente (2), 16–21.

Grand View Research, 2020. Autonomous last mile delivery market size, share & trends analysis report.
URL `https://www.grandviewresearch.com/industry-analysis/autonomous-last-mile-delivery-market`

Ha, Q. M., Deville, Y., Pham, Q. D., Hà, M. H., 2018. On the min-cost traveling salesman problem with drone. Transportation Research Part C: Emerging Technologies 86, 597–621.

Haas, I., Friedrich, B., 2017. Developing a micro-simulation tool for autonomous connected vehicle platoons used in city logistics. Transportation Research Procedia 27, 1203–1210.

Hansen, P., Mladenović, N., 2001. Variable neighborhood search: Principles and applications. European Journal of Operational Research 130, 449–467.

Hansen, P., Mladenović, N., 2018. Variable neighborhood search. In: Martí, R., Pardalos, P. M., Resende, M. G. C. (Eds.), Handbook of Heuristics. Vol. 328. Springer International Publishing, Cham, pp. 759–787.

Heimfarth, A., Ostermeier, M., Hübner, A., 2021. A mixed truck and robot delivery approach for the daily supply of customers. Working paper TU Munich.

Henke, T., Speranza, M. G., Wäscher, G., 2015. The multi-compartment vehicle routing problem with flexible compartment sizes. European Journal of Operational Research 246, 730–743.

Hoffmann, T., Prause, G., 2018. On the regulatory framework for last-mile delivery robots. Machines 6 (3), 33.

Hübner, A., Holzapfel, A., Kuhn, H., Obermair, E., 2019. Distribution in omnichannel grocery retailing: An analysis of concepts realized. In: Gallino, S., Moreno, A. (Eds.), Operations in an omnichannel world. Springer series in supply chain management. Springer Nature Switzerland AG, Cham, pp. 283–310.

Hübner, A., Kuhn, H., Wollenburg, J., 2016a. Last mile fulfilment and distribution in omni-channel grocery retailing. International Journal of Retail & Distribution Management 44 (3), 228–247.

Hübner, A., Kuhn, H., Wollenburg, J., 2016b. Last mile fulfilment and distribution in omni-channel grocery retailing: A strategic planning framework. International Journal of Retail & Distribution Management 44, 1–21.

Hübner, A., Ostermeier, M., 2019. A multi-compartment vehicle routing problem with loading and unloading costs. Transportation Science 53 (1), 282–300.

Hussain, R., Zeadally, S., 2019. Autonomous cars: Research results, issues, and future challenges. IEEE Communications Surveys & Tutorials 21 (2), 1275–1313.

Ishfaq, R., Defee, C. C., Gibson, B. J., Raja, U., 2016. Realignment of the physical distribution process in omni-channel fulfillment. International Journal of Physical Distribution & Logistics Management 46 (6/7), 543–561.

Jaller, M., Otero-Palencia, C., Pahwa, A., 2020. Automation, electrification, and shared mobility in urban freight: opportunities and challenges. Transportation Research Procedia 46, 13–20.

Jennings, D., Figliozzi, M., 2019. A study of sidewalk autonomous delivery robots and their potential impacts on freight efficiency and travel. Transportation Research Record 2673 (6), 317–326.

Kim, S., Moon, I., 2019. Traveling salesman problem with a drone station. IEEE Transactions on Systems, Man, and Cybernetics: Systems 49 (1), 42–52.

Kitjacharoenchai, P., Ventresca, M., Moshref-Javadi, M., Lee, S., Tanchoco, J. M., Brunese, P. A., 2019. Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. Computers & Industrial Engineering 129, 14–30.

Kiwibot, 2020. Company website.
URL `https://www.kiwibot.com/`

Klein, R., Neugebauer, M., Ratkovitch, D., Steinhardt, C., 2019. Differentiated time slot pricing under routing considerations in attended home delivery. Transportation Science 53 (1), 236–255.

Köhler, C., Ehmke, J. F., Campbell, A. M., 2020. Flexible time window management for attended home deliveries. Omega 91, 102023.

Konstantakopoulos, G. D., Gayialis, S. P., Kechagias, E. P., 2020. Vehicle routing problem and related algorithms for logistics distribution: a literature review and classification. Operational Research.

Kovacs, A. A., Golden, B. L., Hartl, R. F., Parragh, S. N., 2014a. The generalized consistent vehicle routing problem. Transportation Science 49, 796–816.

Kovacs, A. A., Parragh, S. N., Hartl, R. F., 2014b. A template-based adaptive large neighborhood search for the consistent vehicle routing problem. Networks 63 (1), 60–81.

Kuhn, H., Sternbeck, M. G., 2013. Integrative retail logistics: An exploratory study. Operations Management Research 6 (1-2), 2–18.

Laporte, G., 2009. Fifty years of vehicle routing. Transportation Science 43 (4), 408–416.

Li, H., Wang, H., Chen, J., Bai, M., 2020. Two-echelon vehicle routing problem with time windows and mobile satellites. Transportation Research Part B: Methodological 138, 179–201.

Liu, D., Deng, Z., Mao, X., Yang, Y., Kaisar, E. I., 2020a. Two-echelon vehicle-routing problem: Optimization of autonomous delivery vehicle-assisted e-grocery distribution. IEEE Access 8, 108705–108719.

Liu, D., Deng, Z., Zhang, W., Wang, Y., Kaisar, E. I., 2020b. Design of sustainable urban electronic grocery distribution network. Alexandria Engineering Journal 60 (1), 145–157.

Macrina, G., Di Puglia Pugliese, L., Guerriero, F., Laporte, G., 2020. Drone-aided routing: A literature review. Transportation Research Part C: Emerging Technologies 120, 102762.

Marble, 2019. Company website.
URL https://www.marble.io/

McKinsey & Company, 2016. Parcel delivery - the future of last mile.
URL https://www.mckinsey.com/~/media/mckinsey/industries/travel%20transport%20and%20logistics/our%20insights/how%20customer%20demands%20are%20reshaping%20last%20mile%20delivery/parcel_delivery_the_future_of_last_mile.ashx

Mercedes-Benz Vans, 2016. Vans & robots: Efficient delivery with the mothership concept.
URL https://youtu.be/yUMOLzCsifs

Mladenović, N., Hansen, P., 1997. Variable neighborhood search. Computers & Operations Research 24, 1097–1100.

Moshref-Javadi, M., Hemmati, A., Winkenbach, M., 2020a. A truck and drones model for last-mile delivery: A mathematical model and heuristic approach. Applied Mathematical Modelling 80, 290–318.

Moshref-Javadi, M., Lee, S., Winkenbach, M., 2020b. Design and evaluation of a multi-trip delivery model with truck and drones. Transportation Research Part E: Logistics and Transportation Review 136, 101887.

Mühlbauer, F., Fontaine, P., 2021. A parallelised large neighbourhood search heuristic for the asymmetric two-echelon vehicle routing problem with swap containers for cargo-bicycles. European Journal of Operational Research 289 (2), 742–757.

Murray, C. C., Chu, A. G., 2015. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. Transportation Research Part C: Emerging Technologies 54, 86–109.

Murray, C. C., Raj, R., 2020. The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. Transportation Research Part C: Emerging Technologies 110, 368–398.

Olsson, J., Hellström, D., Pålsson, H., 2019. Framework of last mile logistics research: A systematic review of the literature. Sustainability 11 (24), 7131.

OpenStreetMap Foundation, 2019. Main page.
URL `https://www.openstreetmap.org`

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 8, 683–711.

Ostermeier, M., Heimfarth, A., Hübner, A., 2021a. Cost-optimal truck-and-robot routing for last-mile delivery. Networks, 1–26.

Ostermeier, M., Heimfarth, A., Hübner, A., 2021b. The multi-vehicle truck-and-robot routing problem for last-mile delivery. Working paper TU Munich.

Ostermeier, M., Holzapfel, A., Kuhn, H., Schubert, D., 2020. Integrated zone picking and vehicle routing operations with restricted intermediate storage. Working paper.

Otto, A., Agatz, N., Campbell, J., Golden, B., Pesch, E., 2018. Optimization approaches for civil applications of unmanned aerial vehicles (uavs) or aerial drones: A survey. Networks 72 (4), 411–458.

Phan, A. T., Nguyen, T. D., Pham, Q. D., 2018. Traveling salesman problem with multiple drones. In: Unknown (Ed.), Proceedings of the Ninth International Symposium on Information and Communication Technology - SoICT 2018. ACM Press, New York, New York, USA, pp. 46–53.

Poeting, M., Schaudt, S., Clausen, U., 2019a. A comprehensive case study in last-mile delivery concepts for parcel robots. In: Mustafee, N., Bae, K.-H., Lazarova-Molnar, S., Rabe, M., Szabo, C., Haas, P., Son, Y.-J. (Eds.), Proceedings of the 2019 Winter Simulation Conference. IEEE, Piscataway, NJ, United States.

Poeting, M., Schaudt, S., Clausen, U., 2019b. Simulation of an optimized last-mile parcel delivery network involving delivery robots. In: Clausen, U., Langkau, S., Kreuz, F. (Eds.), Advances in Production, Logistics and Traffic. Vol. 17 of Lecture Notes in Logistics. Springer International Publishing, Cham, pp. 1–19.

Rojas Viloria, D., Solano-Charris, E. L., Muñoz-Villamizar, A., Montoya-Torres, J. R., 2020. Unmanned aerial vehicles/drones in vehicle routing problems: a literature review. International Transactions in Operational Research 28 (4), 1626–1657.

Sacramento, D., Pisinger, D., Ropke, S., 2019. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. Transportation Research Part C: Emerging Technologies 102, 289–315.

Salama, M., Srinivas, S., 2020. Joint optimization of customer location clustering and drone-based routing for last-mile deliveries. Transportation Research Part C: Emerging Technologies 114, 620–642.

Savelsbergh, M., van Woensel, T., 2016. 50th anniversary invited article—city logistics: Challenges and opportunities. Transportation Science 50 (2), 579–590.

Simoni, M. D., Kutanoglu, E., Claudel, C. G., 2020. Optimization and analysis of a robot-assisted last mile delivery system. Transportation Research Part E: Logistics and Transportation Review 142, 102049.

Sonneberg, M.-O., Leyerer, M., Kleinschmidt, A., Knigge, F., Breitner, M. H., 2019. Autonomous unmanned ground vehicles for urban logistics: Optimization of last mile delivery operations. In: Bui, T. (Ed.), Proceedings of the 52nd Hawaii International Conference on System Sciences. Proceedings of the Annual Hawaii International Conference on System Sciences. University of Hawaii at Manoa, Maui, Hawaii.

Starship, 2019. Company website.
URL `https://www.starship.xyz/`

Statista, 2021. U.s. consumers: Online grocery shopping - statistics & facts.
URL `https://www.statista.com/topics/1915/us-consumers-online-grocery-shopping/`

The Guardian, 2019. Google's world-first drone delivery business wins approval in canberra.
URL `https://www.theguardian.com/australia-news/2019/apr/09/googles-world-first-drone-delivery-business-wins-approval-in-canberra`

Toth, P., Vigo, D., 2001. The vehicle routing problem. SIAM monographs on discrete mathematics and applications. Society for Industrial and Applied Mathematics, Philadelphia.

Ulmer, M., 2017. Delivery deadlines in same-day delivery. Logistics Research 10 (3), 1–15.

Ulmer, M. W., Thomas, B. W., 2018. Same-day delivery with heterogeneous fleets of drones and vehicles. Networks 72 (4), 475–505.

United Nations, 2018. World urbanization prospects: The 2018 revision.
URL `https://esa.un.org/unpd/wup/Publications`

van Heeswijk, W. J. A., Mes, M. R. K., Schutten, J. M. J., 2019. The delivery dispatching problem with time windows for urban consolidation centers. Transportation Science 53 (1), 203–221.

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 268 (2), 703–715.

Wollenburg, J., Hübner, A., Trautrims, A., Kuhn, H., 2018. From bricks-and-mortar to bricks-and-clicks: Logistics networks in omni-channel grocery retailing. International Journal of Physical Distribution and Logistics Management 48 (4), 415–438.

World Economic Forum, 2020. The future of the last-mile ecosystem: Transition roadmaps for public- and private-sector players.
URL `http://www3.weforum.org/docs/WEF_Future_of_the_last_mile_ecosystem.pdf`

Yoo, H. D., Chankov, S. M., 2018. Drone-delivery using autonomous mobility: An innovative approach to future last-mile delivery problems. In: 2018 IEEE

International Conference on Industrial Engineering and Engineering Management (IEEM). IEEE, pp. 1216–1220.

Appendix

A MIP model for the VRP with deadlines

We introduce the following MIP model as a benchmark for the truck-and-robot concept. It minimizes the cost of traditional truck-based manual delivery, assuming the same cost factors as in the truck-and-robot case. We assume that the driver can leave the parcel at the door in the event that he/she arrives before the time window, which reduces the time windows to deadlines and favors the VRP. Furthermore, we assume the same processing time of 40 sec. per customer, which is again on the optimistic side and favors the VRP.

The notation is as defined by Tables 3.2 and 3.4. We additionally introduce the number of available vehicles f and the total costs for truck travel from location i to location j as c_{ij}^{VRP}. It is calculated using $c_{ij}^{\text{VRP}} = \lambda_{ij} \cdot (c^{\text{d}} + c^{\text{t}}/\omega_t)$. The binary variable s_{ij} is 1 if a vehicle goes from location i to location j and 0 otherwise. Finally, t_k denotes the arrival time at customer k.

This leads to the objective function (1), which incorporates the cost of truck travel and delays. Constraint (2) ensures that every customer is visited exactly once. (3) and (4) keep track of the arrival times at customers and (5) derives the delay from the arrival time. Constraints (6) and (7) establish flow constraints for the trucks at every stop. Equations (8) and (9) define the solution space.

$$\min \sum_{i \in C \cup \{\gamma\}} \sum_{j \in C \cup \{\gamma\}} s_{ij} \cdot c_{ij}^{\text{VRP}} + \sum_{k \in C} v_k \cdot c^{\text{l}} \tag{1}$$

subject to

$$\sum_{i \in C \cup \{\gamma\}} s_{ik} = 1 \qquad \forall k \in C \tag{2}$$

$$\vartheta_{\gamma k}^{t} - M \cdot (1 - s_{\gamma k}) \leq t_k \qquad \forall k \in C \tag{3}$$

$$t_i + \vartheta_{ij}^{t} - M \cdot (1 - s_{ij}) \leq t_j \qquad \forall i, j \in C \tag{4}$$

$$t_k - d_k \leq v_k \qquad \forall k \in C \qquad (5)$$

$$\sum_{i \in C \cup \{\gamma\}} s_{ik} = \sum_{i \in C \cup \{\gamma\}} s_{ki} \qquad \forall k \in C \qquad (6)$$

$$\sum_{k \in C} s_{\gamma k} \leq f \qquad (7)$$

$$s_{ij} \in \{0, 1\} \qquad \forall i, j \in C \qquad (8)$$

$$t_k \geq 0;\ v_k \geq 0 \qquad \forall k \in C \qquad (9)$$

B MIP model for the VRP with time windows

For solving the VRP with time windows, we introduce the following MIP model, which we adapted from Ostermeier et al. (2021a) to incorporate time windows instead of only deadlines. It minimizes the cost of traditional truck delivery assuming the same cost factors as in the MTR case. We further assume the same processing time of 40 sec. for every customer k (included in ϑ^t_{ik}). We introduce the set of available vehicles F, which contains only one vehicle in our case. The binary decision variable s_{fij} is 1 if vehicle f travels from location i to location j and 0 otherwise. Finally, auxiliary decision variable t_k denotes the arrival time at customer k and t^T_f the total tour time of vehicle f. This leads to the objective function (10), which incorporates the cost of truck distance, truck time and delays. Constraints (11) ensure every customer is visited exactly once. (12) and (13) keep track of the earliest possible arrival times at customers. Constraints (14) ensure no customer is served before his/ her time window and Constraints (15) derive the delays from the arrival times. (16) define the total operating time of each truck. (17) and (18) establish flow constraints for the trucks at every stop. Constraints (19) to (21) define the solution space.

$$\min \sum_{f \in F} \sum_{i \in C \cup \{\gamma\}} \sum_{j \in C \cup \{\gamma\}} c^{\mathrm{d}} \lambda_{ij} s_{fij} + \sum_{f \in F} c^{\mathrm{t}} t^T_f + \sum_{k \in C} c^{\mathrm{l}} v_k \qquad (10)$$

subject to

$$\sum_{i \in C \cup \{\gamma\}} \sum_{f \in F} s_{fik} = 1 \qquad \forall k \in C \qquad (11)$$

$$t_k \geq \vartheta^t_{\gamma k} - M \cdot (1 - s_{f\gamma k}) \qquad \forall k \in C, f \in F \qquad (12)$$

$$t_j \geq t_i + \vartheta^t_{ij} - M \cdot (1 - s_{fij}) \qquad \forall i, j \in C, f \in F \qquad (13)$$

$$t_k \geq d_k - \epsilon \qquad \forall k \in C \qquad (14)$$

$$v_k \geq t_k - d_k \qquad \forall k \in C \qquad (15)$$

$$t^T_f \geq t_k + \vartheta^t_{k\gamma} - M \cdot (1 - s_{fk\gamma}) \qquad \forall k \in C, f \in F \qquad (16)$$

$$\sum_{i \in C \cup \{\gamma\}} s_{fik} = \sum_{i \in C \cup \{\gamma\}} s_{fki} \qquad \forall k \in C, f \in F \qquad (17)$$

$$\sum_{k \in C} s_{f\gamma k} \leq 1 \qquad \forall f \in F \qquad (18)$$

$$s_{fij} \in \{0, 1\} \qquad \forall i, j \in C, f \in F \qquad (19)$$

$$t_k \geq 0; \ v_k \geq 0 \qquad \forall k \in C \qquad (20)$$

$$t^T_f \geq 0 \qquad \forall f \in F \qquad (21)$$

C MIP model for customer clustering of the benchmark approach

The MIP minimizes the cost of traditional truck delivery assuming the same cost factors as in the truck-and-robot case. We consider only deadlines instead of time windows to reduce computational complexity. Our experiments showed that this leads to the best results in the *MIP&GVNS* experiments while also working in favor of the benchmark. Given early deliveries are possible, we can further fix the start time of each vehicle to the earliest possible time, t_ω=0, without worsening the objective value.

We introduce the set of available vehicles F. The binary decision variable $s_{f,i,j}$ is 1 if vehicle f travels from location i to location j, and 0 otherwise. The auxiliary decision variable t_k denotes the arrival time at customer k and t^T_f the total tour time of vehicle f. This leads to the objective function (22), which sums up the cost of truck distance, truck time and delays. Constraints (23) ensure that every customer is visited exactly once. Constraints (24) keep track of the earliest possible arrival times at customers. Constraints (25) derive the delay from the arrival time. (26) defines the total operating time of each truck. (27) and (28) establish flow constraints for the trucks at every stop. Constraints (29) limit the truck capacity, and Constraints (30) to (33) define the solution space.

$$\min \sum_{f \in F} \sum_{i \in C \cup \{\omega\}} \sum_{j \in C \cup \{\omega\}} c^{\text{dist}} \lambda_{i,j} s_{f,i,j} + \sum_{f \in F} c^{\text{veh}} t_f^T + \sum_{k \in C} c^{\text{late}} l_k \tag{22}$$

subject to

$$\sum_{i \in C \cup \{\omega\}} \sum_{f \in F} s_{f,i,k} = 1 \qquad \forall k \in C \tag{23}$$

$$t_j \geq t_i + \vartheta^t_{i,j} - M \cdot (1 - s_{f,i,j}) \qquad \forall i \in C \cup \{\omega\}, j \in C, f \in F \tag{24}$$

$$l_k \geq t_k - d_k \qquad \forall k \in C \tag{25}$$

$$t_f^T \geq t_k + \vartheta^t_{k,\omega} - M \cdot (1 - s_{f,k,\omega}) \qquad \forall k \in C, f \in F \tag{26}$$

$$\sum_{i \in C \cup \{\omega\}} s_{f,i,k} = \sum_{i \in C \cup \{\omega\}} s_{f,k,i} \qquad \forall k \in C, f \in F \tag{27}$$

$$\sum_{k \in C} s_{f,\omega,k} \leq 1 \qquad \forall f \in F \tag{28}$$

$$\sum_{i \in C \cup \{\omega\}} \sum_{k \in C} \eta_k s_{f,i,k} \leq G \qquad \forall f \in F \tag{29}$$

$$s_{f,i,j} \in \{0,1\} \qquad \forall i,j \in C \cup \{\omega\}, f \in F \tag{30}$$

$$t_\omega = 0 \tag{31}$$

$$t_k \geq 0; \;\; l_k \geq 0 \qquad \forall k \in C \tag{32}$$

$$t_f^T \geq 0 \qquad \forall f \in F \tag{33}$$

Declaration of Authorship

I hereby declare that the thesis submitted is my own unaided work. All direct or indirect sources used are acknowledged as references.

I am aware that the thesis in digital form can be examined for the use of unauthorized aid and in order to determine whether the thesis as a whole or parts incorporated in it may be deemed as plagiarism. For the comparison of my work with existing sources I agree that it shall be entered in a database where it shall also remain after examination, to enable comparison with future theses submitted. Further rights of reproduction and usage, however, are not granted here.

This paper was not previously presented to another examination board and has not been published.

City, Date

Signature