

## **A mixed truck and robot delivery approach for the daily supply of customers**

**Andreas Heimfarth, Manuel Ostermeier, Alexander Hübner**

### **Angaben zur Veröffentlichung / Publication details:**

Heimfarth, Andreas, Manuel Ostermeier, and Alexander Hübner. 2021. "A mixed truck and robot delivery approach for the daily supply of customers." SSRN Electronic Journal. Elsevier BV. <https://doi.org/10.2139/ssrn.3815759>.

### **Nutzungsbedingungen / Terms of use:**

**licgercopyright**

Dieses Dokument wird unter folgenden Bedingungen zur Verfügung gestellt: / This document is made available under the following conditions:

**Deutsches Urheberrecht**

Weitere Informationen finden Sie unter: / For more information see:

<https://www.uni-augsburg.de/de/organisation/bibliothek/publizieren-zitieren-archivieren/publizieren>



# A mixed truck and robot delivery approach for the daily supply of customers

Heimfarth, Andreas<sup>1</sup>, Ostermeier, Manuel <sup>\*,1</sup>, Hübner, Alexander<sup>1</sup>

*Technical University of Munich  
Supply and Value Chain Management  
Am Essigberg 3, 94315 Straubing, Germany*

---

## 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.

*Keywords:* Logistics, Vehicle routing, Autonomous vehicles, Last-mile, General variable neighborhood search

---

\*Corresponding author; e-mail: manuel.ostermeier@tum.de

<sup>1</sup>Technical University of Munich, Supply and Value Chain Management, Am Essigberg 3, 94315 Straubing, Germany

## 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., 2016]. 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., 2018; Ostermeier et al., 2021].

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 electron-

ics, 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. [2018]; Ostermeier et al. [2021]; Bakach et al. [2021]). 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 2. Section 3 discusses related literature and highlights the differences versus other last-mile delivery concepts. Section 4 presents the formal model of the MTR-RP. We detail our GVNS approach in Section 5. Section 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 7 summarizes our findings and presents opportunities for future research.

## 2. Problem description

This section outlines how a truck and robots are combined for attended home deliveries with time windows. Section 2.1 introduces the related technology, on which the problem is based. We then describe the MTR delivery concept in Section 2.2.

### 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 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.



Figure 1: Specialized truck with freight containers and delivery robots [Mercedes-Benz Vans, 2016]

## 2.2. Concept of mixed truck and robot deliveries (MTR)

*Conventional TnR.* In line with Boysen et al. [2018] and Ostermeier et al. [2021], 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 (see Figure 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 2), and robot tours visiting a single customer each (dotted arrows in Figure 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 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., 2018; Ostermeier et al., 2021].

*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 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 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.

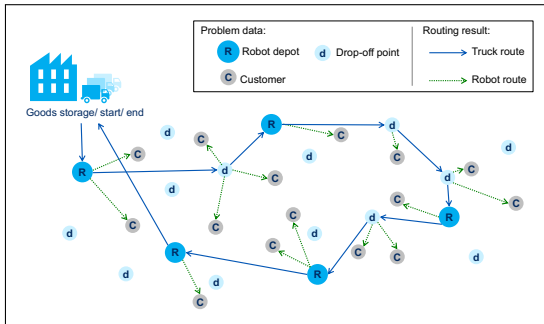


Figure 2: TnR tour (with all deliveries by robots)

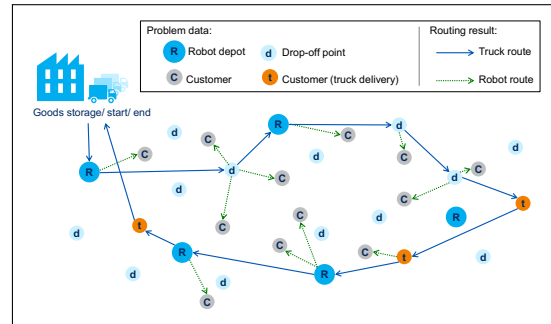


Figure 3: MTR tour (incl. deliveries by truck)

*Decision problem structure.* MTR routing requires simultaneous decisions on different routing problem aspects. To illustrate this, Figure 4 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). 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). 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), 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.

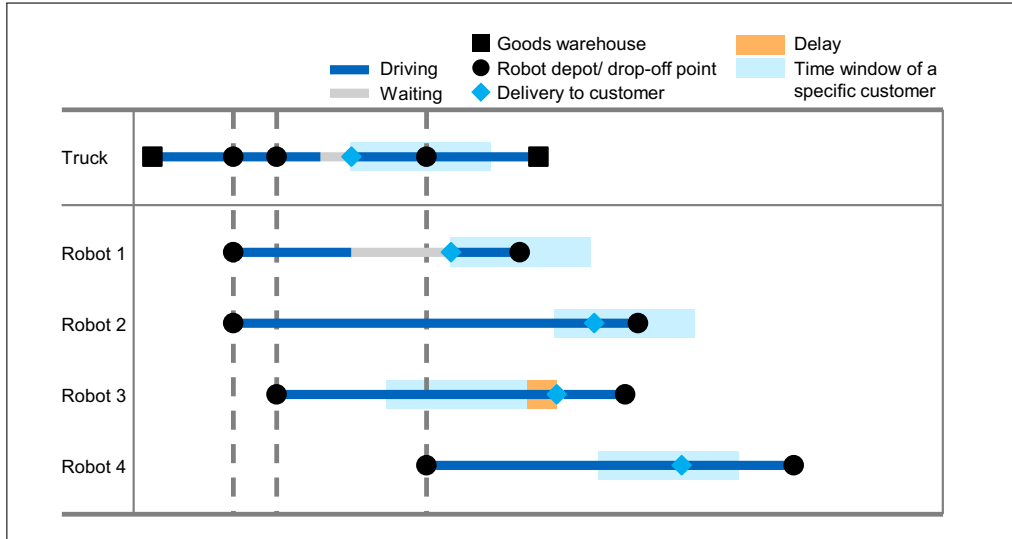


Figure 4: Gantt chart of an MTR tour (example)

### 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. [2021]. 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 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. [2021] 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. [2018] 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. [2021] 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.

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

✓: considered, -: not considered

Table 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 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. Formulation of the MTR-RP

This section introduces the mathematical formulation of the MTR-RP. The notation used is summarized in Table 2.

<i>Index sets</i>	
$C$	Set of all customers $k \in C$
$C^m$ ( $C^r$ )	Subset of customers requiring truck (robot) delivery, with $C^m \cup C^r \subseteq C$
$C^o$	Subset of customers indifferent regarding truck or robot delivery, with $C^o \subseteq C$
$D$ ( $R$ )	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: $\hat{L} := C^m \cup C^o \cup \hat{D} \cup \hat{R}$
$I_a$	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 $i \leq m$
<i>Problem parameters</i>	
$d_k$	Deadline for customer $k, k \in C$
$K$	Maximum robot capacity of a truck
$r_a$	Initial amount of available robots in location $a, a \in R$
$\gamma$ ( $\bar{\gamma}$ )	Start (end) position of the truck, with $\gamma, \bar{\gamma} \notin \hat{L}$
$\delta$	Initial number of robots aboard the truck
$\epsilon_k$	Length of time window of customer $k$
$\lambda_{i,j}$	Distance between locations $i$ and $j, i, j \in \hat{L}$
$\vartheta_{i,j}^t$	Truck travel time from location $i$ to location $j, i, j \in \hat{L}$
$\vartheta_{i,k}^r$	Robot travel time from location $i, i \in \hat{L}$ , to customer $k, k \in C$
$\vartheta_k^b$	Robot travel time from customer $k$ back to the closest robot depot
<i>Cost parameters</i>	
$c^l$	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
<i>Decision variables</i>	
$s_{i,j}$	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
<i>Auxiliary variables</i>	
$t_i$	Arrival time of truck at location $i$
$q_i$	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 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  $C = C^m \cup C^r \cup C^o$ . Every customer  $k \in C^r \cup C^o$  can be served by one robot, every customer  $k \in C^m \cup C^o$  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  $\hat{L} := C^m \cup C^o \cup \hat{D} \cup \hat{R}$ . 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  $\gamma$  (e.g., a goods warehouse,  $\gamma \notin \hat{L}$ ) with  $\delta$  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  $\epsilon_k$ . 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  $\vartheta_{i,j}^t$  for the truck and  $\vartheta_{i,k}^r$  for the robots. We further denote the robot travel time from customer  $k$ , back to the closest depot as  $\vartheta_k^b$ . 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.

$$\begin{aligned} \min F(S, X, T, V, L, E, W) = \\ = c^t t_\gamma + \sum_{i \in \hat{L} \cup \{\gamma\}} \sum_{j \in \hat{L} \cup \{\bar{\gamma}\}} c^d \lambda_{i,j} s_{i,j} + \sum_{i \in \hat{L}} \sum_{k \in C^r \cup C^o} c^r (\vartheta_{i,k}^r + \vartheta_k^b) x_{i,k} + \sum_{k \in C} (c^l v_k + c^r w_k) \end{aligned} \quad (1)$$

subject to

$$\sum_{i \in \hat{L}} x_{i,k} + \sum_{i \in \hat{L} \cup \{\gamma\}} s_{i,k} = 1 \quad \forall k \in C^o \cup C^m \quad (2)$$

$$\sum_{i \in \hat{L}} x_{i,k} = 1 \quad \forall k \in C^r \quad (3)$$

$$\sum_{k \in C} x_{j,k} \leq M \sum_{i \in \hat{L} \cup \{\gamma\}} s_{i,j} \quad \forall j \in \hat{L} \quad (4)$$

$$\sum_{j \in \hat{L}} s_{\gamma,j} \leq 1 \quad (5)$$

$$\sum_{i \in \hat{L} \cup \{\gamma\}} s_{i,j} = \sum_{i \in \hat{L} \cup \{\bar{\gamma}\}} s_{j,i} \quad \forall j \in \hat{L} \quad (6)$$

$$t_\gamma = 0 \quad (7)$$

$$t_j \geq t_i + \vartheta_{i,j}^t - M(1 - s_{i,j}) \quad \forall j \in \hat{L} \cup \{\bar{\gamma}\}; i \in \hat{L} \cup \{\gamma\} \quad (8)$$

$$t_k \geq d_k - \epsilon \quad \forall k \in C^m \quad (9)$$

$$t_k \geq d_k - \epsilon - M(1 - \sum_{i \in \hat{L} \cup \{\gamma\}} s_{i,k}) \quad \forall k \in C^o \quad (10)$$

$$q_\gamma = \delta \quad (11)$$

$$q_j \leq q_i + e_j - \sum_{k \in C} x_{j,k} + M(1 - s_{i,j}) \quad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{R} \quad (12)$$

$$q_j \leq q_i - \sum_{k \in C} x_{j,k} + M(1 - s_{i,j}) \quad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{D} \cup C^m \cup C^o \quad (13)$$

$$v_k \geq t_k - d_k \quad \forall k \in C^m \cup C^o \quad (14)$$

$$v_k \geq t_j + \vartheta_{j,k}^r - d_k - M(1 - x_{j,k}) \quad \forall k \in C^r \cup C^o, j \in \hat{L} \quad (15)$$

$$w_k \geq (d_k - \epsilon) - t_j - \vartheta_{j,k}^r - M(1 - x_{j,k}) \quad \forall k \in C^r \cup C^o, j \in \hat{L} \quad (16)$$

$$t_i \leq t_j \quad \forall a \in R; i, j \in I_a : i \leq j \quad (17)$$

$$\sum_{h \in \hat{L} \cup \{\gamma\}} s_{h,i} \geq \sum_{h \in \hat{L} \cup \{\gamma\}} s_{h,j} \quad \forall a \in R; i, j \in I_a : i \leq j \quad (18)$$

$$r_a - \sum_{i \in I_a^m} e_i \geq 0 \quad \forall a \in R; m \in I_a \quad (19)$$

$$s_{i,j} \in \{0, 1\} \quad \forall i \in \hat{L} \cup \{\gamma\}; j \in \hat{L} \cup \{\bar{\gamma}\} : i \neq j \quad (20)$$

$$s_{i,i} = 0 \quad \forall i \in \hat{L} \quad (21)$$

$$x_{i,k} \in \{0, 1\} \quad \forall i \in \hat{L}; k \in C^r \cup C^o \quad (22)$$

$$x_{i,k} = 0 \quad \forall i \in \hat{L}; k \in C^m \quad (23)$$

$$e_i \in \mathbb{Z} \quad \forall i \in \hat{R} \quad (24)$$

$$t_i \geq 0 \quad \forall i \in \hat{L} \cup \{\bar{\gamma}\} \quad (25)$$

$$q_i \in \{0, \dots, K\} \quad \forall i \in \hat{L} \quad (26)$$

$$v_k, w_k \geq 0 \quad \forall k \in C \quad (27)$$

The objective function (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 (2) ensures exactly one visit by either truck or robot for every customer  $k \in C^o \cup C^m$ . Similarly, constraint (3) ensures that each customer who requires a robot delivery is visited by exactly one robot. Constraint (4) states that robots can only be launched from stops that are actually visited by truck. Constraint (5) defines that the truck only leaves once from the starting point, and (6) ensures that if the truck reaches a location, it must also leave it. Constraints (7) and (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 (9) ensures that a required truck delivery is not made before the respective time window and (10) does so for optional truck deliveries in case they are made by truck (and not by robot). The following constraints (11), (12) and (13) handle the number of robots aboard the truck when leaving the starting point, a depot or any other location, respectively. Constraint (14) defines the delay for customers receiving

truck delivery. Constraints (15) and (16) define the delay and waiting time for customers receiving robot deliveries. Constraints (17) and (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 (19) to track the robot stock in every depot and to ensure that the stock is  $\geq 0$ . Finally, the variable domains are defined by constraints (20) to (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 (11), (12) and (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 ((9) and (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  $t_{\bar{\gamma}}c^t$  to the objective function. Since the optimal  $t_{\bar{\gamma}}$  is determined via the recursive constraints (8), (9) and (10), this is computationally expensive even for small instances.

## 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. [2018]). 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. [2014]; 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. [2018] and Ostermeier et al. [2021]) 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., 2021]. An overview of our solution framework is shown in Figure 5.

We generate an initial truck tour with one of two possible start procedures, depending on the given problem instance (see Section 5.1). This truck tour is then evaluated and complemented to a full solution by finding the optimal robot schedule using an MIP (see Section 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 5.3). It consists of a shaking step and a subsequent Variable Neighborhood

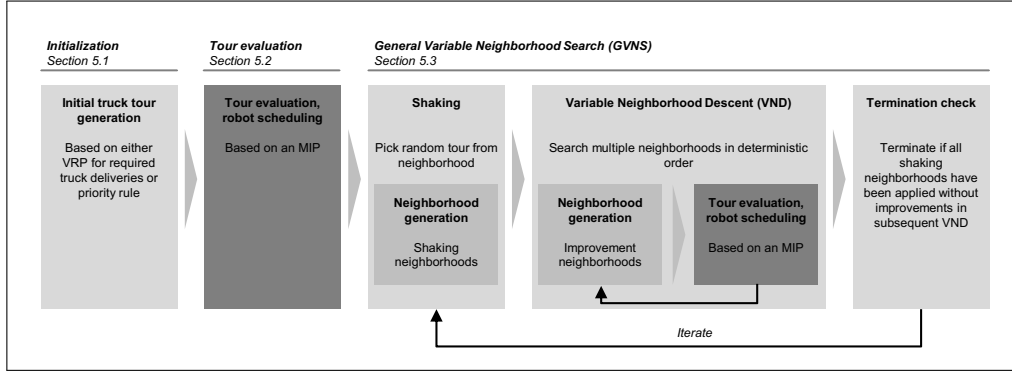


Figure 5: Structure of the MTR heuristic proposed

Descent (VND). Within the GVNS, tours are again assessed by the robot scheduling MIP from Section 5.2.

### 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  $\sigma$  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  $\sigma$  truck deliveries, we solve a *VRP with time windows* (see model provided in Appendix A) 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  $\tau$  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.

### 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  $\delta$  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. [2018] and enhanced by Ostermeier et al. [2021] 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, which included the decision on truck movements, we do not need duplicates of robot drop-off ( $D$ ) and depot locations ( $R$ ). This leads to  $L := C^m \cup C^o \cup D \cup R$  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  $\tilde{C} \subseteq C^o \cup C^r$ . Table 3 summarizes the notation of truck tour parameters and decision variables.

<i>Truck tour parameters</i>	
$U$	Index set of stops on the truck tour $u \in \{1, 2, \dots\}$
$Y$	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 $u, u \in U$
$c_{u,k}^T$	Cost of serving customer $k, k \in \tilde{C}$ , from stop $u, u \in U$
<i>Decision and auxiliary variables</i>	
$x_{u,k}$	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 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 (28)–(30). Equation (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 (29)). For customer locations, the beginning of the respective time window also has to be considered to prevent premature deliveries (Equation (30)).

$$t_1 = 0 \tag{28}$$

$$t_u = t_{u-1} + \vartheta_{y(u),y(u-1)}^t \quad \forall u : y(u) \in D \cup R \tag{29}$$

$$t_u = \max(t_{u-1} + \vartheta_{y(u),y(u-1)}^t, d_{y(u)} - \epsilon_k) \quad \forall u : y(u) \in C \tag{30}$$

Based on arrival times, the total cost  $c_{u,k}^T$  of supplying a customer  $k$  from stop  $u$  is denoted by Equation (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  $\vartheta_k^b$ . Finally, delay costs are added.

$$\begin{aligned} c_{u,k}^T &:= c^r(\vartheta_{y(u),k}^r + (d_k - \epsilon_k - t_u - \vartheta_{y(u),k}^r)^+ + \vartheta_k^b) \\ &+ c^l(t_u + \vartheta_{y(u),k}^r - d_k)^+ \quad \forall u \in U, k \in \tilde{C} \end{aligned} \tag{31}$$

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.

$$\min F(Q, X, R) = \sum_{u \in U} \sum_{k \in \tilde{C}} x_{u,k} \cdot c_{u,k}^T \tag{32}$$

subject to

$$\sum_{u \in U} x_{u,k} = 1 \quad \forall k \in \tilde{C} \tag{33}$$

$$r_{a,u} = r_{a,u-1} \quad \forall a \in R, u \in U : a \neq y(u) \tag{34}$$

$$r_{a,u} \leq r_{a,u-1} + q_{u-1} - q_u - \sum_{k \in \tilde{C}} x_{u,k} \quad \forall a \in L, u \in U : a = y(u) \tag{35}$$

$$q_0 = \delta \tag{36}$$

$$r_{a,0} = r_a \quad \forall a \in R \tag{37}$$

$$r_{a,u} = 0 \quad \forall a \in L \setminus R, u \in U \quad (38)$$

$$x_{u,k} \in \{0, 1\} \quad \forall u \in U, k \in \tilde{C} \quad (39)$$

$$r_{a,u} \geq 0 \quad \forall a \in R, u \in U \quad (40)$$

$$0 \leq q_u \leq K \quad \forall u \in U \quad (41)$$

The objective function (32) minimizes total robot and delay costs. Constraint (33) ensures that exactly one robot is sent to each remaining customer. Constraint (34) states that if a depot is not visited, the number of available robots remains the same. Constraint (35) keeps track of the number of robots in locations visited and aboard the truck after every stop. Equations (36) and (37) define the initial number of robots in the depots and on the truck. Constraint (38) ensures that robots cannot be stored at drop-off locations or customers. Constraints (39)–(41) define the variable domains.

### 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 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  $\alpha$  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.

*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  $\alpha$  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.

---

**Algorithm 1** GVNS procedure (adapted from Hansen and Mladenović [2018])

---

**Input:** Starting solution  $\pi_s$   
 $\pi_{best} = \pi_s$ ; // 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  $\alpha$  **do**  
     $k_i = 1$   
     $\pi_{current} = \text{random}(\text{shakeneighborhood}(\pi_{best}, k_s))$  // shaking neighborhood  $k_s$   
    **while**  $k_i \leq$  number of VND neighborhoods **do**  
       $\pi_{k_i} = \text{best}(\text{improvementneighborhood}(\pi_{current}, k_i))$  // improvement neighborhood  $k_i$   
      **if**  $Z(\pi_{k_i}) < Z(\pi_{best})$  **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 while**  
**return**  $\pi_{best}$

---

- **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  $n^{\text{shuffle}}$  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 2.

---

**Algorithm 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  
         $pool_k = \{\}$   
         $pool_k.add(\text{shift}(Y, k, 1))$  // 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  
                 $pool_k.add(\text{shift}(Y, k, i))$  // shift customer to stop  $i$   
            **end if**  
        **end for**  
    **end if**  
**end for**  
**if** no. of tours in all pools  $> n^{\text{shuffle}}$  **then**  
     $pool_{final} = \bigcup \{\text{shortest tour from every pool}\}$   
**else**  
     $pool_{final} = \bigcup \{\text{all pools}\}$   
**end if**  
**return**  $pool_{final}$

---

*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 (32)-(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.

## 6. Numerical examples

This section analyzes the performance of our MTR heuristic. First, we describe the instances and parameters used in our experiments (Section 6.1). Next, we compare our approach to a benchmark (Section 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 6.3) and analyze the impact of time windows on the routing (Section 6.4). Finally, we discuss the impact of customer distribution (Section 6.5), and cost rates for the truck and delays (Section 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 ( $4 \times 1.9$  GHz), 16 GB RAM, and Windows 10 Enterprise.

### 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. [2021] (<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  $4 \text{ km}^2$  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 ( $|C^m|/|C|=0.12$ ). The remaining customers require robot delivery ( $C^r = C \setminus C^m$ ). This means there are no optional truck deliveries in the default case ( $C^o = \emptyset$ ). The impact of optional truck deliveries will be analyzed separately. Note that our assumption for  $|C^m|/|C|$  is in line with the estimate reported by Forbes [2019], that technically 75 to 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. [2021]. These are  $c^d = 0.20$  €/km and  $c^t = 30$  €/h for the truck,  $c^r = 0.50$  €/h for robot use and  $c^l = 5$  €/h for delivery delays.

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

## 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. [2018] and Ostermeier et al. [2021] 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. [2021], as the authors study the TnR concept 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. [2018] 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. [2021]. The MTR heuristic reaches a solution quality differing only 0.3 to 2.0% from the LS in these cases. The computation times for different problem sizes are shown in Figure 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.

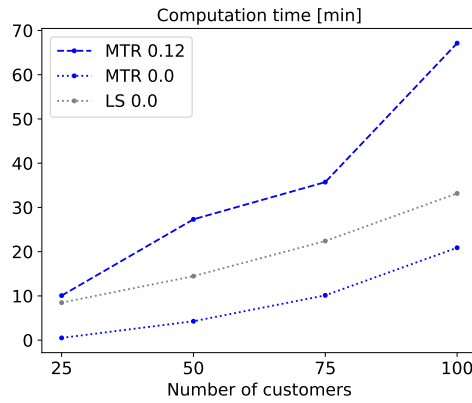


Figure 6: Comparison of our MTR approach to the benchmark (i.e., LS approach by Ostermeier et al. [2021]) for a share of 0% and 12% of truck deliveries

Additionally, we found that the MIP for the entire MTR-RP ((1)–(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.

### 6.3. Comparison of delivery concepts

This section compares the delivery concepts given in Table 4 for a varying share of truck deliveries required.

Figure 7 shows the total costs, computation times, average delay and total truck distance for the concepts analyzed. We henceforth highlight the default setting described in Section 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,

Concept	Description	Rationale	Solution approach
<b>TD:</b> Truck-only delivery	Only deliveries by truck to all customers	Benchmark to assess MTR benefits	MIP from Appendix A
<b>MTR:</b> 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
<b>MTR OT:</b> 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
<b>STR:</b> 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 A; TnR tour by MTR heuristic

Table 4: Overview of delivery concepts

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.

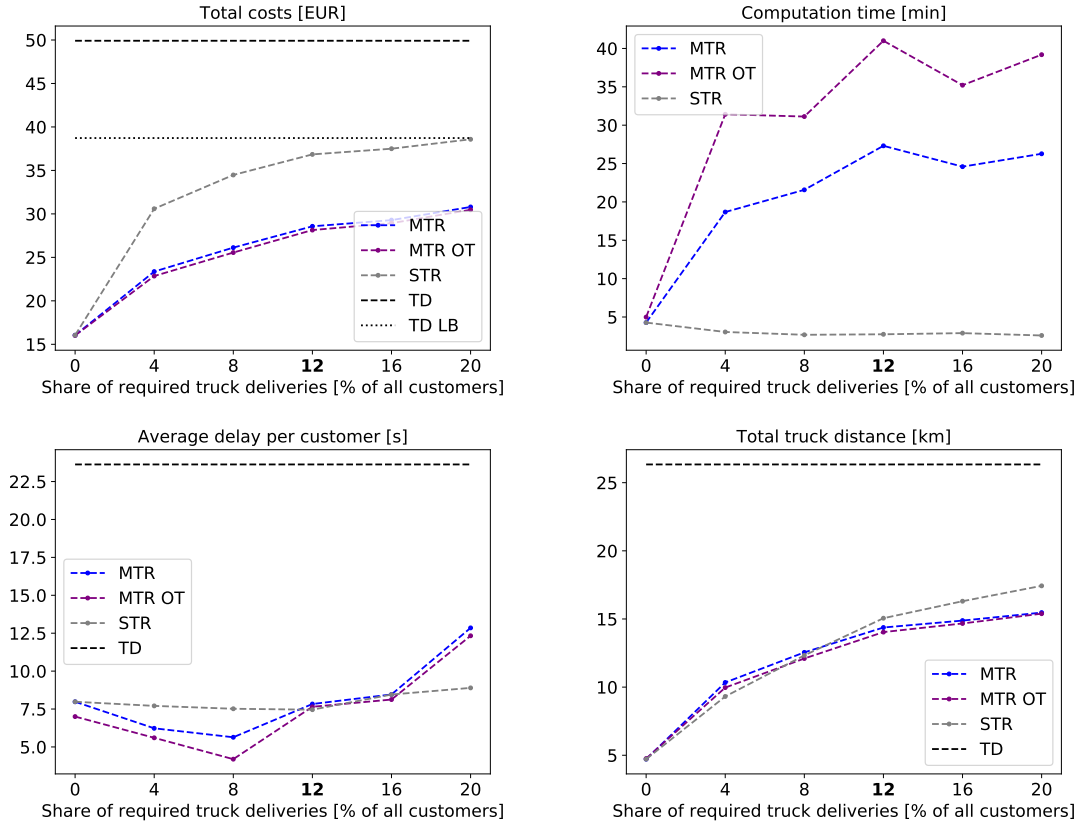


Figure 7: Comparison of different delivery modes for a varying share of required truck deliveries

*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. [2021]). 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.

#### 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 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.

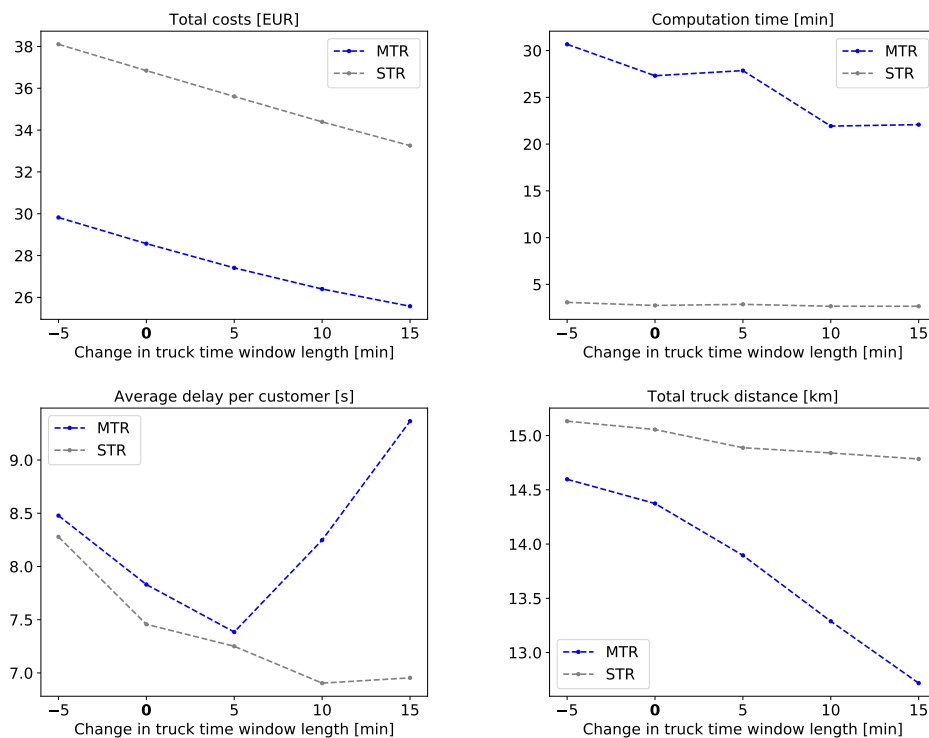


Figure 8: Comparison of MTR vs. STR for varying length of truck delivery time windows

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 to 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 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 9.

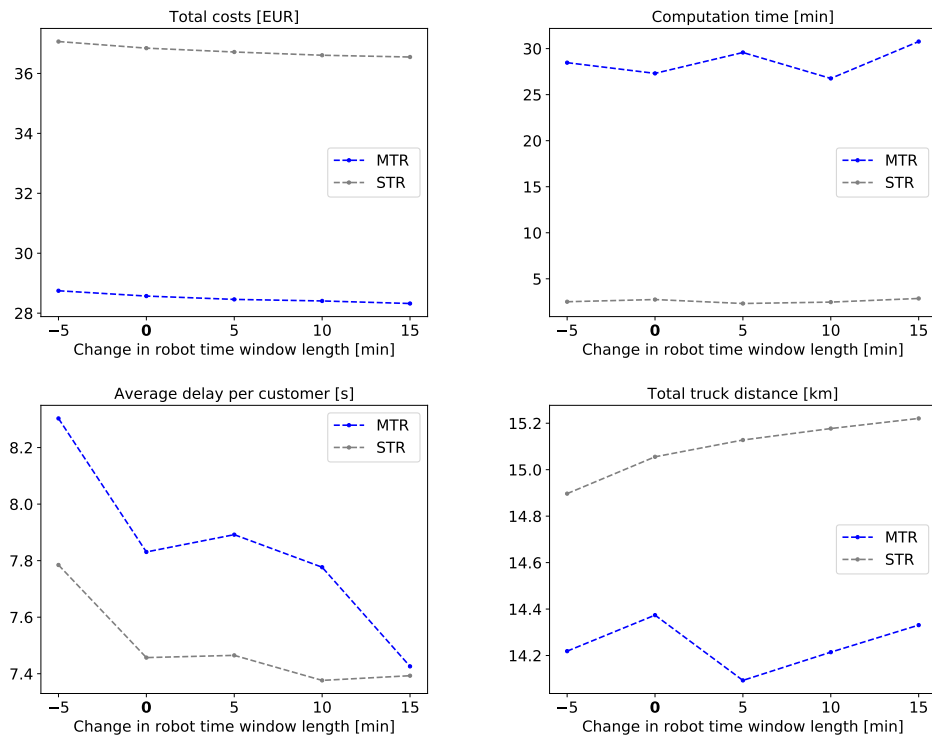


Figure 9: Comparison of MTR vs. STR for varying length of robot delivery time windows

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.

### 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  $2 \times 2$  km<sup>2</sup> square area, and a clustered distribution, where two customer clusters are considered, one in the lower left and one in the upper right quadrant of the original  $4 \times 4$  km<sup>2</sup> 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 10.

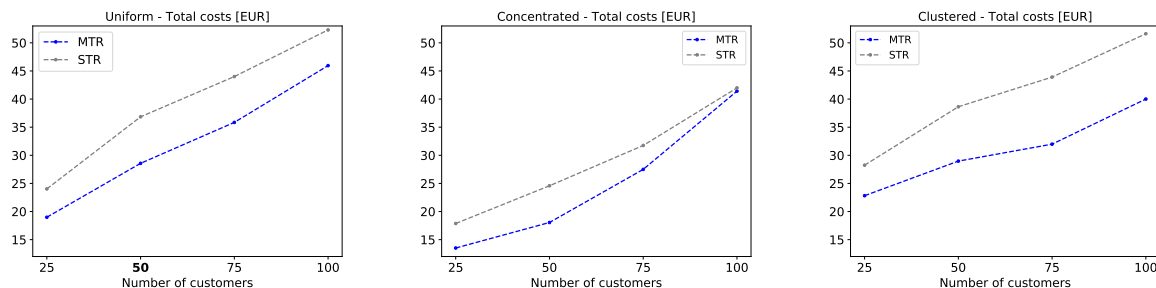


Figure 10: Cost comparison of MTR vs. STR for varying customer distributions

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.

### 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 11 displays our findings. Total costs increase proportionally for both approaches, leading to stable cost savings of 22 to 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.

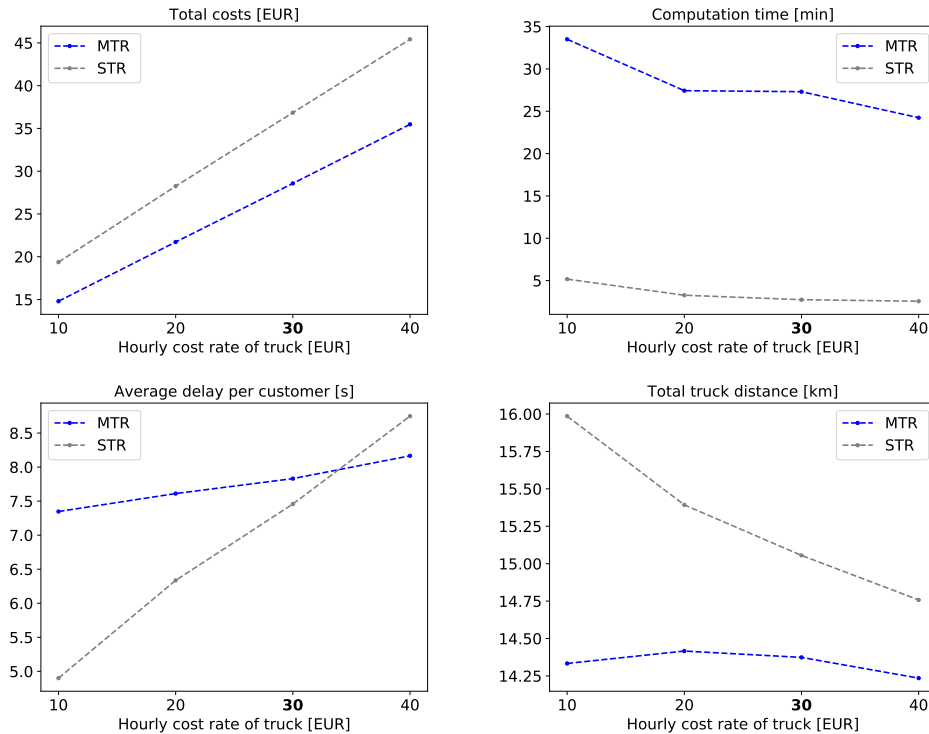


Figure 11: Comparison of MTR vs. STR for varying hourly truck cost rates

*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  $c^d$ . The results are summarized in Figure 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 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.

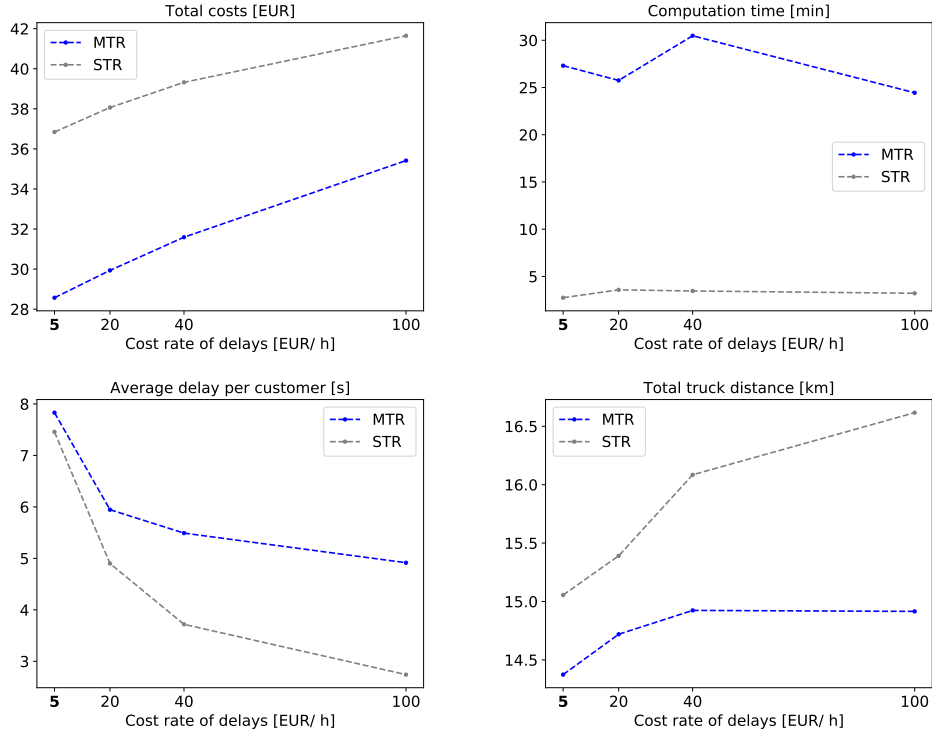


Figure 12: Comparison of MTR vs. STR for varying delay cost rates

## 7. Conclusion

Our work shows that the MTR concept is a valuable extension of the existing ThR 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 ThR 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.

## References

- Agatz, N., Bouman, P., Schmidt, M., 2018. Optimization approaches for the traveling salesman problem with drone. *Transportation Science* 52 (4), 965–981.
- 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.
- Bakach, I., Campbell, A. M., Ehmke, J. F., 2021. A two-tier urban delivery network with robot-based deliveries. *Networks*, 1–23 (online). doi: <https://doi.org/10.1002/net.22024>.
- Baum, L., Assmann, T., Strubelt, H., 2019. State of the art - automated micro-vehicles for urban logistics. *IFAC-PapersOnLine* 52 (13), 2455–2462.
- 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., 2018. Scheduling last-mile deliveries with truck-based autonomous robots. *European Journal of Operational Research* 271 (3), 1085–1099.
- Daimler, 2019. Vans & robots.  
URL <https://www.daimler.com/innovation/specials/future-transportation-vans/paketbote-2-0.html>
- 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.
- Enthoven, D. L., Jargalsaikhan, B., Roodbergen, K. J., uit het Broek, M. A. J., Schrottenboer, 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.
- 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-remarks-warehouse-robots-alexa-prediction/#7c0c6f7d145f>
- 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.
- 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.
- 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.
- Hübner, A., Holzzapfel, 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., 2016. Last mile fulfilment and distribution in omni-channel grocery retailing. *International Journal of Retail & Distribution Management* 44 (3), 228–247.
- 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.
- 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.
- Kovacs, A. A., Golden, B. L., Hartl, R. F., Parragh, S. N., 2014. The generalized consistent vehicle routing problem. *Transportation Science* 49, 796–816.
- 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.
- 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/>
- 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.

- 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.
- 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., 2021. Cost-optimal routing of a truck-and-robot system with limited robot availability. *Networks*, 1–26.
- 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.
- 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.
- 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.
- 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/>
- 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. W., Thomas, B. W., 2018. Same-day delivery with heterogeneous fleets of drones and vehicles. *Networks* 72 (4), 475–505.
- Wollenburg, J., Hübner, A., Trautrimis, 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.

## Appendix A. A 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. [2021] 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  $\vartheta_{ik}^t$ ). 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  $t_f^T$  the total tour time of vehicle  $f$ . This leads to the objective function (A.1), which incorporates the cost of truck distance, truck time and delays. Constraints (A.2) ensure every customer is visited exactly once. (A.3) and (A.4) keep track of the earliest possible arrival times at customers. Constraints (A.5) ensure no customer is served before his/ her time window and Constraints (A.6) derive the delays from the

arrival times. (A.7) define the total operating time of each truck. (A.8) and (A.9) establish flow constraints for the trucks at every stop. Constraints (A.10) to (A.12) define the solution space.

$$\min \sum_{f \in F} \sum_{i \in \text{CU}\{\gamma\}} \sum_{j \in \text{CU}\{\gamma\}} c^d \lambda_{ij} s_{fij} + \sum_{f \in F} c^t t_f^T + \sum_{k \in C} c^l v_k \quad (\text{A.1})$$

subject to

$$\sum_{i \in \text{CU}\{\gamma\}} \sum_{f \in F} s_{fik} = 1 \quad \forall k \in C \quad (\text{A.2})$$

$$t_k \geq \vartheta_{\gamma k}^t - M \cdot (1 - s_{f\gamma k}) \quad \forall k \in C, f \in F \quad (\text{A.3})$$

$$t_j \geq t_i + \vartheta_{ij}^t - M \cdot (1 - s_{fij}) \quad \forall i, j \in C, f \in F \quad (\text{A.4})$$

$$t_k \geq d_k - \epsilon \quad \forall k \in C \quad (\text{A.5})$$

$$v_k \geq t_k - d_k \quad \forall k \in C \quad (\text{A.6})$$

$$t_f^T \geq t_k + \vartheta_{k\gamma}^t - M \cdot (1 - s_{fk\gamma}) \quad \forall k \in C, f \in F \quad (\text{A.7})$$

$$\sum_{i \in \text{CU}\{\gamma\}} s_{fik} = \sum_{i \in \text{CU}\{\gamma\}} s_{fki} \quad \forall k \in C, f \in F \quad (\text{A.8})$$

$$\sum_{k \in C} s_{f\gamma k} \leq 1 \quad \forall f \in F \quad (\text{A.9})$$

$$s_{fij} \in \{0, 1\} \quad \forall i, j \in C, f \in F \quad (\text{A.10})$$

$$t_k \geq 0; v_k \geq 0 \quad \forall k \in C \quad (\text{A.11})$$

$$t_f^T \geq 0 \quad \forall f \in F \quad (\text{A.12})$$