The multi-vehicle truck-and-robot routing problem for last-mile delivery

Manuel Ostermeier, Andreas Heimfarth, Alexander Hübner

Angaben zur Veröffentlichung / Publication details:

Ostermeier, Manuel, Andreas Heimfarth, and Alexander Hübner. 2023. "The multi-vehicle truck-and-robot routing problem for last-mile delivery." European Journal of Operational Research 310 (2): 680–97.
https://doi.org/10.1016/j.ejor.2023.03.031.

The multi-vehicle truck-and-robot routing problem for last-mile delivery

Manuel Ostermeier[a,*], Andreas Heimfarth[b], Alexander Hübner[b]

[a] University of Augsburg, Resilient Operations, Universitätsstraße 12, Augsburg 86159, Germany
[b] Technical University of Munich, Supply and Value Chain Management, Am Essigberg 3, Straubing 94315, Germany

1. Introduction

The global autonomous last-mile delivery market is forecast to grow seven-fold by 2027, with ground vehicles accounting for 85% of it (Grand View Research, 2020). Some retailers expect 80% of their deliveries to be autonomous by 2025 (Bennett, 2020). Consumers are increasingly ordering products online to benefit from the comfort of home deliveries. Statista (2022), for example, forecasts annual growth of online food sales in the U.S. by 24% until 2027. However, classic trucks for last-mile delivery are increasingly obstructing traffic flow in urban areas and driving up local emissions. Enhanced services such as same-day delivery and tighter delivery time windows pose even more challenges for logistics systems, while cost pressure increases (Arslan et al., 2019; Buldeo Rai et al., 2019; van Heeswijk et al., 2019; Hübner et al., 2016; Ishfaq et al., 2016) and labor shortage grows (McKinsey & Company, 2021). Last-mile deliveries are therefore becoming increasingly important (Boysen et al., 2021; Otto et al., 2018). Innovative delivery concepts are needed to reduce traffic congestion, CO_2 emissions, and noise, and to enable cost-efficient and customer-friendly services (Hübner et al., 2019; Orenstein et al., 2019).

A promising approach to address these matters is applying autonomous robots carried by trucks in urban areas, known as the truck-and-robot concept with robot depots (see e.g., Boysen et al., 2018b). The central idea of the concept is to overcome the issue of the slow delivery speed of autonomously driving delivery robots. Delivery trucks act as motherships and transport parcels together with multiple robots that are picked up during the tour from robot depots. The motherships enable fast and timely deliveries as they cover larger transportation distances. The trucks release one or multiple robots at drop-off points or robot depots. The robots then take care of the "last-mile" delivery and travel to the customer's home at pedestrian speed, deliver the order and return to the next robot depot. The customers can choose a delivery time window during which they are at home and retrieve their order from the robot. Daimler (2019), for instance, has developed and successfully tested customized trucks paired with delivery robots.

* Corresponding author.
E-mail address: manuel.ostermeier@uni-a.de (M. Ostermeier).

Delivery robots have further been successfully implemented in different settings by several companies (see e.g., Kiwibot, 2020; Marble, 2019; Starship, 2019), who usually offer them as a rental service to logistics service providers.

The objective of truck-and-robot routing with robot depots is to plan a truck route and schedule robot deliveries to satisfy the complete demand, respect truck capacities and robot availability, and minimize the total costs arising from the travel costs of trucks and robots and potential service-related costs (e.g., for delays). The emerging but still small body of literature on truck-and-robot concepts with depots focuses on identifying central problem aspects and related benefits, such as the reduction of emissions and costs (Ostermeier et al., 2022) or service quality (Alfandari et al., 2022; Boysen et al., 2018b). Current literature addresses the basic problem, where only one truck is available to transport robots and goods. For large and densely populated delivery areas, a single truck cannot fulfill the complete demand within a defined delivery period when strict time windows have to be met. This calls for generalizing the concept to multiple trucks, i.e., a fleet of delivery trucks. The single truck problem includes the (i) routing of one truck and (ii) robot scheduling, but neither the (iii) assignment of customers to trucks nor (iv) the simultaneous routing of multiple trucks is considered. However, the assignment of customers to tours and the routing of multiple tours have a significant impact on the robot scheduling (i.e., where and when to release the robots) and vice versa.

Our work contributes by generalizing the truck-and-robot concept with a single truck and formulates the first Multi-Vehicle Truck-and-Robot Routing Problem (MVTR-RP) with robot depots. It is, therefore, a fundamental extension of the basic problem with only one truck and studies an entirely new area of problems. Prevailing solution approaches for the single-truck problem are not designed to solve this extended problem. Furthermore, existing approaches for the vehicle routing problem (VRP) do not provide an off-the-shelf solution for the MVTR-RP due to its specific problem setting. The MVTR-RP requires the scheduling of the robots from truck stops on top of the truck routing to possible robot depots and drop-off locations that are flexibly chosen as part of the decision problem. Tailored solution approaches are required to address the specific challenges when combining truck routing and robot scheduling within the MVTR-RP. We, therefore, develop a heuristic for the $\mathcal{NP}$-hard problem to simultaneously solve the routing of multiple vehicles while scheduling robot deliveries and defining the required truck stops at depots and drop-off locations. Our approach is based on advanced VRP approaches and proposes a novel neighborhood search algorithm – denoted as Set Improvement Neighborhood Search (SINS) – that improves a set of tours by optimally choosing a new set out of the neighborhoods of all incumbent tours. It creates and tests large pools of potential truck tours, from which the optimal set is chosen.

The remainder of this paper is organized as follows. We outline the delivery concept and develop the formal problem description in Section 2. Section 3 reviews related literature and Section 4 details the heuristic proposed. Section 5 analyses the numerical efficiency and develops managerial insights. We summarize our findings and outline potential future research areas in Section 6.

2. Problem description and formal model

This section details the underlying truck-and-robot concept. We first outline the logistical setup of both the single- and multi-vehicle concept in Section 2.1, and discuss the relation of the MVTR-RP to standard VRPs (Section 2.2). This builds the basis for introducing the formal model (Section 2.3). In our context, the term "vehicle" refers to goods and robots transporting truck,

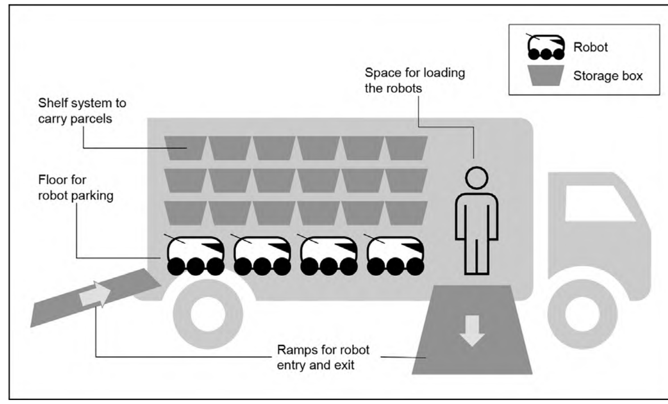

Fig. 1. Specialized truck for robot deliveries.

whereas the term "robots" refers to autonomously driving ground robots.

2.1. Technological and logistical setup of the truck-and-robot concept with robot depots

The basic truck-and-robot concept with a single truck. The truck-and-robot concept combines the use of autonomous delivery robots with specialized delivery trucks to launch robots in customer proximity for attended home delivery. The delivery trucks are specialized vans (see Fig. 1) that act as a mothership for the transportation of parcels and robots (Boysen et al., 2018b; Jennings & Figliozzi, 2019). The trucks used are pivotal for the entire fulfillment as they enhance delivery speed and flexibility. The faster trucks can cover larger distances between delivery requests in a short time, and the system can adapt to changing customer distributions and corresponding demand by integrating dedicated drop-off points as truck stops. This adds to the flexibility of the system and also reduces the number of robots and robot depots required. Robots can enter the truck via a ramp from the back, be loaded by the driver in the front part of the truck, and leave it via another ramp to the side. The truck's capacity is limited concerning robots and storage boxes. An example setup allows up to eight robots and 54 storage boxes (see, e.g., Vans, 2016). As these vehicles transport robots and parcels, they usually have a lower capacity than standard delivery vans. Various robot models are used (see e.g., Baum et al., 2019; Jaller et al., 2020), mainly differing in size and travel speed.

A key component of the truck-and-robot system is robot depots, small charging stations in the customer area operated by a robot provider, who offers delivery robots as a service (Grand View Research, 2020). The provider rents robots to multiple logistics service companies. Robots wait at these charging stations until they are needed by a logistics provider, who picks them up by truck and pays a time-dependent rental fee. Robots are homogeneous and can recharge in each depot.

Figure 2 illustrates the example of a tour with a single truck. The truck tour of the logistics provider starts at a goods warehouse where all parcels for delivery and the initial number of robots are loaded. The truck then visits dedicated locations to pick up and release robots within the delivery area. Additional potential truck stops where robots can be released are defined as drop-off points. In contrast to depots, robots can only be released at drop-off points, but no new robots can be loaded. The truck never waits for dropped-off robots but picks up new robots waiting at the robot depots. Once released, robots move autonomously on sidewalks at pedestrian speed and deliver parcels to customer doors. Customers are notified on arrival and receive their delivery by unlocking the

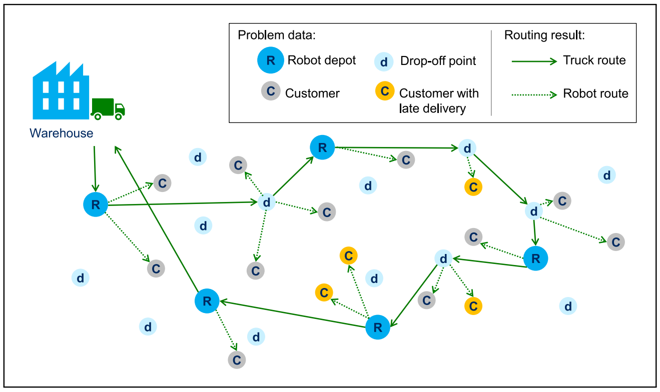

Fig. 2. Truck-and-robot tour with one truck (example).

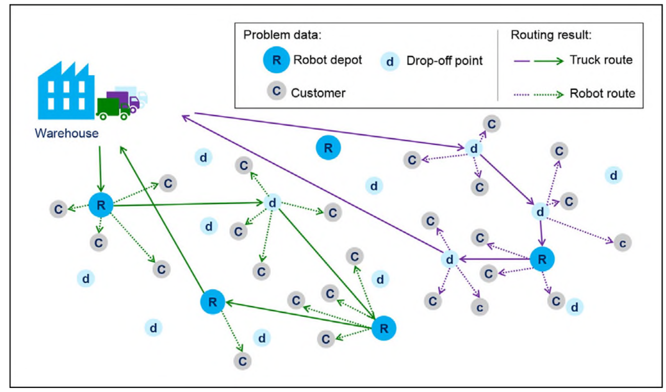

Fig. 3. Two separate truck-and-robot tours.

compartment after receiving a code. Since the customers must be present, they can choose a delivery time window. After a customer has retrieved the parcel, the robots return to the closest robot depot, from where the robot provider can rent it to another logistics company (not shown in Fig. 2 for better readability). Decision-relevant costs are the truck costs (i.e., travel times and distances from warehouse to robot depots, drop-off locations, and return to the warehouse) and robot travel times from the drop-off location to customers. The robot travelling from the customer back to the next robot depot is not decision relevant as the robots always approach the closest robot depot. The truck costs include time-based costs for the driver and distance-based travel costs. As robots are usually rented as a service from a robot provider and charged by usage time, a time-based fee applies for the actual travel time, i.e., the time from the drop-off until the return to the closest depot. The application of time windows imposes further cost considerations. If a robot arrives before the time window, it must wait for the customer. If it arrives after the time window, delay costs (as opportunity costs for reduced satisfaction or rebates on delivery fees) are incurred.

The routing of the truck is a central aspect of the truck-and-robot concept. Specifically, it needs to be decided which of the given robot depots and drop-off points are visited by the truck or not and in which sequence. This further includes possible multiple visits per location due to the given time window constraints and a corresponding release of robots at the same location at different times. In contrast to the travelling salesman problem (TSP), the number of truck stops and the stop locations (namely robot depots and drop-off points) are part of the decision problem. At the same time, the assignment of customers to truck stops needs to be considered, i.e., the drop-off location or robot depot as the starting point of each customer's delivery. This is also called *robot scheduling* and determines each robot's arrival and usage time. This is further constrained by robot availability at robot depots. We denote the problem of routing one truck with robots as Single-Vehicle Truck-and-Robot Routing Problem (SVTR-RP).

Multi-vehicle truck-and-robot routing. A single truck is only able to supply a given maximum number of customers, as its fulfillment capacity is limited. An increasing demand volume and the need to carry out deliveries simultaneously (given the time windows) require additional trucks to avoid late or failed deliveries. Figure 3 illustrates an example with two truck tours. Compared to Fig. 2, two trucks can serve more customers and avoid delays. The extension to the multi-vehicle problem, denoted as MVTR-RP, enables the simultaneous delivery to different customers by multiple tours. This increases the flexibility and the fulfillment capacity, but it also increases the problem complexity. In the MVTR-RP, multiple tours need to be determined, i.e., their start times, respective stops, and sequence. Multiple tours access the same resources (robot depots), and robot availability must be monitored. In addition, customers need to not only be *assigned* to truck stops (as in the basic SVTR-RP) but *clustered* (i.e., allocated) to delivery tours in the first place. This means a simultaneous decision is required on the (i) *assignment of customers to truck tours*, the (ii) *routing of multiple trucks* (i.e., selection of robot depots and drop-off locations as well as sequencing of the stops), and the (iii) *robot scheduling* (i.e., assignment of customers to stop locations of the tour, from where the respective robot starts). All three decisions are interdependent. For example, the reallocation of one customer to another tour impacts routing and robot scheduling. It is further not sufficient to allocate customers to the same tour based on similar locations and time windows since two customers far from each other with different deadlines could fit well on one tour.

To summarize, the underlying routing problem is specified by a set of customers with known demands and time windows that need to be supplied by robots launched by trucks. The trucks travel from a goods warehouse to robot depots and drop-off points. Each robot is released from a drop-off point, or robot depot, serves one customer, and returns then to the next robot depot. The objective is to minimize travel costs of trucks and robots and potential service-related costs (e.g., for delays) by assigning customers to vehicles, routing multiple truck tours and defining robot schedules that satisfy the entire demand while maintaining truck capacities and robot availability.

2.2. Classification of the MVTR-RP as VRP

The extension of the SVTR-RP with one truck (and hence a TSP) to an MVTR-RP turns our problem into a VRP, as customers need to be assigned to truck tours. There is rich literature on VRPs and corresponding exact, and (meta-)heuristic approaches to address the clustering and routing (i.e., cluster-first-route-second, route-first-cluster-second, integrated). For an overview of solution approaches for VRPs, we kindly refer to established reviews (e.g., Golden et al., 2008; Toth & Vigo, 2014). The effectiveness and efficiency of the solution approaches depend on the problem setting and decision scope. In general, the MVTR-RP belongs to the class of VRPs with time windows, including service times and truck and robot travel times. Apart from these general similarities, the MVTR-RP shows particularities that constitute its uniqueness as VRP variant. To begin with, there are obligatory (robot) deliveries to customers, but optional visits to robot depot and drop-off locations. This means that the number, sequence, and above all, locations of truck stops need to be determined. For example, a single tour may only visit one depot to supply ten customers, but it may as well visit five depots or drop-off points. These optional visits to robot depots and drop-off locations significantly complicate the assignment of cus-

Table 1
Notation of the MVTR-RP.

Index sets	
C	Set of customers, $k \in C$
D ($\hat{D}$)	Set of distinct drop-off points (including duplicates)
R ($\hat{R}$)	Set of distinct robot depots (including duplicates)
$\hat{L}$	Set of all (duplicate) locations from which robots can be started: $\hat{L} := \hat{D} \cup \hat{R}$
Ω ($\bar{\Omega}$)	Set of warehouse duplicates as start (end) position of each tour
I_a	Set of duplicate indices $i, i \in \hat{R}$, of one distinct robot depot $a, a \in R$
I_a^m	Set of elements $i \in I_a$ with $i \leq m$
Problem parameters	
d_k	Deadline for customer $k, k \in C$
Q (G)	Maximum robot (parcel volume) capacity of a truck
β_a	Initial amount of available robots in location $a, a \in R$
δ	Initial number of robots aboard a truck at start
ϵ	Length of time windows
η_k	Volume of parcels per customer k
ϑ_{ij}^v	Truck travel time from location i to location j with $i, j \in \hat{L} \cup \Omega \cup \bar{\Omega}$
ϑ_{ik}^r	Robot travel time from location $i, i \in \hat{L}$, to customer $k, k \in C$
λ_{ij}	Distance between locations i and j with $i, j \in \hat{L} \cup \Omega \cup \bar{\Omega}$
Cost parameters	
c^{late}	Cost of delays per time unit
c^{dist}	Cost of truck travel per distance unit
c^{veh} (c^{rob})	Cost of truck (robot) per time unit
Decision variables	
s_{ij}	Binary: 1, if a truck travels from location i to location j; 0 otherwise
t_i	Arrival time of a truck at location i
x_{ik}	Binary: 1, if customer k is supplied from location i; 0 otherwise
Auxiliary variables	
e_i	Number of robots taken out of depot location $i, i \in \hat{R}$
g_i	Volume of parcels aboard a truck when arriving at location i
q_i	Number of robots aboard a truck at departure from location i
l_k	Lateness (delay) time of delivery to customer k
w_k	Waiting time of robot at customer k

tomers to tours (and, as such, the robot deliveries), as every change in the truck tour also impacts the robot assignment. This then directly connects to the second difference. The same customer can be supplied from different drop-off locations and/or robot depots. As such, the assignment of a customer to a tour implies that a robot has to be scheduled to visit that customer (possibly from an already planned stop) but does not define a unique stop of the truck. Third, the MVTR-RP requires robot scheduling in addition to the truck routing decisions. This means that after each truck routing solution, the companion robot scheduling has to be solved and consequently requires an additional step in the solution finding. Finally, there is an unrestricted assignment of customers to trucks and robots, but capacity and robot availability needs to be ensured. We need to consider both the capacity of trucks as well as the availability of robots at depots. This again impacts the assignment decisions.

These differences constitute the genuine character of the MVTR-RP and require the development of tailored approaches to address the problem specifics presented. Same as for the SVTR-RP, where it is not possible to simply apply TSP solution approaches (see e.g., Boysen et al., 2018b), existing VRP approaches cannot be directly applied to the MVTR-RP.

2.3. Mathematical model of the multi-vehicle truck-and-robot routing problem

The distribution system introduced constitutes an entirely new problem. As such, we contribute by formalizing the decision problem at hand and formulating the mathematical model of the MVTR-RP. Table 1 summarizes the notation used.

Index sets. The MVTR-RP is based on the location sets of customers (C), robot depots (R) and drop-off points (D). In our application, robots may visit customers in the same neighborhood at different times (e.g., with an one-hour gap) due to the delivery time windows defined. Consequently, a truck tour potentially visits the same drop-off or depot locations multiple times to release robots for the respective deliveries. To enable multiple visits at each robot depot and drop-off point, we duplicate the elements in R and D, resulting in $\hat{R}$ and $\hat{D}$ as the corresponding sets of duplicates. All duplicate locations (i.e., duplicate robot depots and drop-off points) are summarized by the set $\hat{L} := \hat{D} \cup \hat{R}$. This also means that one or several trucks can retrieve robots from the same robot depot on multiple occasions. All trucks start at the same goods warehouse. As we need to track the number of trucks and their usage time, we also duplicate the warehouse location for each available truck. This results in a set of start and end locations (i.e., the warehouse duplicates) and is denoted by Ω and $\bar{\Omega}$, each containing one duplicate of the warehouse location $\omega \in \Omega, \bar{\omega} \in \bar{\Omega}$ for every available truck. Duplicates are needed as we use a two-index formulation, and in this way, all trucks can use the same location network $\hat{L}$, while individual tracking applies. Finally, to keep track of available robots in the unique robot depots, we define the set I_a of all duplicate locations i to the depot a (with $i \in \hat{L}, a \in R$), and the set I_a^m of indices in $i \in I_a$ with $i \leq m$ for a given number 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.

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

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

$$\min RC = c^{\text{veh}}\left(\sum_{\bar{\omega}\in\bar{\Omega}} t_{\bar{\omega}} - \sum_{\omega\in\Omega} t_{\omega}\right) + \sum_{i\in\hat{L}\cup\Omega}\sum_{j\in\hat{L}\cup\bar{\Omega}} c^{\text{dist}}\lambda_{ij}s_{ij} + \sum_{i\in\hat{L}}\sum_{k\in C} c^{\text{rob}}\vartheta^{r}_{ik}x_{ik} + \sum_{k\in C}(c^{\text{late}}l_k + c^{\text{rob}}w_k) \qquad (1)$$

subject to

$$\sum_{i\in\hat{L}} x_{ik} = 1 \qquad \forall k\in C \qquad (2)$$

$$x_{jk} \le \sum_{i\in\hat{L}\cup\Omega} s_{ij} \qquad \forall j\in\hat{L}, k\in C \qquad (3)$$

$$\sum_{j\in\hat{L}\cup\bar{\Omega}} s_{\omega j} = 1 \qquad \forall \omega\in\Omega \qquad (4)$$

$$\sum_{j\in\hat{L}\cup\Omega} s_{j\bar{\omega}} = 1 \qquad \forall \bar{\omega}\in\bar{\Omega} \qquad (5)$$

$$\sum_{i\in\hat{L}\cup\Omega} s_{ij} = \sum_{i\in\hat{L}\cup\bar{\Omega}} s_{ji} \qquad \forall j\in\hat{L} \qquad (6)$$

$$t_j \ge t_i + \vartheta^{v}_{ij} - M\cdot(1 - s_{ij}) \qquad \forall j\in\hat{L}\cup\bar{\Omega};\, i\in\hat{L}\cup\Omega \qquad (7)$$

$$l_k \ge t_i + \vartheta^{r}_{ik} - d_k - M\cdot(1 - x_{ik}) \qquad \forall k\in C;\, i\in\hat{L} \qquad (8)$$

$$w_k \ge d_k - t_i - \vartheta^{r}_{ik} - \epsilon - M\cdot(1 - x_{ik}) \qquad \forall k\in C;\, i\in\hat{L} \qquad (9)$$

$$q_{\omega} = \delta \qquad \forall \omega\in\Omega \qquad (10)$$

$$q_j \le q_i + e_j - \sum_{k\in C} x_{jk} + M\cdot(1 - s_{ij}) \qquad \forall i\in\hat{L}\cup\Omega;\, j\in\hat{R} \qquad (11)$$

$$q_j \le q_i - \sum_{k\in C} x_{jk} + M\cdot(1 - s_{ij}) \qquad \forall i\in\hat{L}\cup\Omega;\, j\in\hat{D} \qquad (12)$$

$$g_{\bar{\omega}} = 0 \qquad \forall \bar{\omega}\in\bar{\Omega} \qquad (13)$$

$$g_j \ge g_i + \sum_{k\in C}\eta_k x_{jk} - M\cdot(1 - s_{ji}) \qquad \forall i\in\hat{L}\cup\bar{\Omega};\, j\in\hat{L} \qquad (14)$$

$$t_i \le t_j \qquad \forall a\in R;\, i,j\in I_a : i\le j \qquad (15)$$

$$\sum_{h\in\hat{L}\cup\Omega} s_{hi} \ge \sum_{h\in\hat{L}\cup\Omega} s_{hj} \qquad \forall a\in R;\, i,j\in I_a : i\le j \qquad (16)$$

$$\beta_a - \sum_{i\in I^m_a} e_i \ge 0 \qquad \forall a\in R;\, m\in I_a \qquad (17)$$

$$s_{ij}\in\{0,1\} \qquad \forall i\in\hat{L}\cup\Omega,\, j\in\hat{L}\cup\bar{\Omega} : i\ne j \qquad (18)$$

$$s_{ii} = 0 \qquad \forall i\in\hat{L} \qquad (19)$$

$$x_{ik}\in\{0,1\} \qquad \forall i\in\hat{L};\, k\in C \qquad (20)$$

$$e_i\in\mathbb{Z} \qquad \forall i\in\hat{R} \qquad (21)$$

$$t_i \ge 0 \qquad \forall i\in\hat{L}\cup\Omega\cup\bar{\Omega} \qquad (22)$$

$$q_i\in[0,Q] \qquad \forall i\in\hat{L} \qquad (23)$$

$$g_i\in[0,G] \qquad \forall i\in\hat{L} \qquad (24)$$

$$l_k, w_k \ge 0 \qquad \forall k\in C \qquad (25)$$

The objective function (1) minimizes total routing costs (RC). The first term indicates the time-dependent usage costs across all trucks, i.e., the total time for each truck from departure to arrival at the warehouse. The second term sums up the total distance costs of all legs travelled by trucks. The third term considers the costs of robot travel from truck stops to customers. The last term adds the costs of robot waiting and delay times. Note that the robots' return time from a customer to the closest depot does not depend on the routing decisions and is, therefore, not decision-relevant. Constraints (2) ensure that every customer is served exactly once, while Constraints (3) ensure a robot starts only from stops visited by a truck. Constraints (4) and (5) ensure that only one truck starts from each start location and also returns to it. Trucks that are not required will stay at their start location, i.e., $s_{\omega\bar{\omega}} = 1$ and $t_{\omega} = t_{\bar{\omega}}$. Constraints (6) represent the flow constraint, stating that the truck leaves every location $j, j\in\hat{L}$ as often as it arrives there. Constraints (7) calculate the truck arrival times based on associated travel times. Note that these constraints ensure that every duplicate stop is visited only once and only by one truck. This means that a sufficient number of duplicates is needed for every stop (in the worst case this could be $|C|$). Constraints (8) and (9) calculate the delay and robot waiting time for each delivery. Equations (10) define the number of available robots aboard the truck at departure. Constraints (11) and (12) keep track of the robots aboard a truck after each stop, depending on whether the stop is a robot depot or a drop-off point. Constraints (13) and (14) ensure adherence to truck capacities by keeping track of the total quantity of parcels aboard the truck arriving at location j. This is done in a recursive manner by defining that the tour ends with an empty truck. Constraints (15) and (16) enforce (without loss of generality) that duplicates of the same location are visited in ascending order of their index. This fact is then used to determine the robots in each depot after every visit (left side of Constraints (17)). This limits robot availability at depots even if they are visited by different trucks. Finally, constraints (18)–(25) define the variable domains.

3. Related literature

Having derived the distribution system and the formal decision problem, this section reviews related literature. We refer to Boysen et al. (2021); Olsson et al. (2019); Savelsbergh & van Woensel (2016) for a detailed overview on current last-mile delivery concepts. The literature related to our problem can be classified into two streams. The first stream is clearly related to our setting and defined by 'truck-and-robot with depots' as a system in which the robots are transported aboard the truck, make a delivery and return to a robot depot. We first review the truck-and-robot literature (and related concepts) in Section 3.1. As this is an emerging area with only a small body of literature and none of the state-of-the-art publications consider multiple trucks, we extend our review to multi-vehicle problems in Section 3.2. This constitutes a second related stream and includes various other means of transportation, such as drones and cargo bikes that are combined with multiple trucks. Finally, we blend discussions, analyze them in the context of the MVTR-RP, and identify the research gap in Section 3.3.

3.1. Truck-and-robot concepts with depots and a single truck

Current publications for truck-and-robot systems with depots are based on the SVTR-RP, i.e., limited to one truck and without an assignment of customers to multiple tours. All approaches consider time windows or delivery deadlines. In a seminal paper, Boysen et al. (2018b) discuss such a basic truck-and-robot concept. The authors minimize the number of late deliveries and assume unlimited robot availability. Their solution approach is based on a local search procedure, and their analysis shows the potential benefits of the concept compared to standard truck delivery. Alfandari et al. (2022) build on this work by proposing alternative delay measures and a Branch-and-Benders-cut scheme for routing. Ostermeier et al. (2022) further extend the decision model by minimizing total delivery costs and incorporating limitations on robot availability. Their solution relies on a local search and solves instances with up to 125 customers. In numerical experiments, the system of one truck with robots reduces costs and emissions by more than 50% compared to normal truck delivery. Heimfarth et al. (2022) generalize the concept by including manual delivery by the truck driver. They apply a General Variable Neighborhood Search (GVNS) to minimize costs. This work considers the most general case of the SVTR-RP to date, and the proposed GVNS framework proved very efficient compared to other existing local search approaches. Exact solutions for truck-and-robot routing with one truck have only been obtained for very small instances or under simplifying assumptions (Alfandari et al., 2022), highlighting the problem's computational complexity.

Another concept based on robots relies on hubs (also called satellite locations) where goods arriving by truck can be stored. It requires more infrastructure and workforce since goods must be stored and robots loaded at the hubs when the truck is already gone. Bakach et al. (2021), for instance, propose a two-tier delivery system where a truck supplies local hubs in which goods are stored and loaded into robots. The robots then make pendulum tours for the customers. Poeting et al. (2019a,b); Sonneberg et al. (2019) and Bakach et al. (2022) are further examples of hub-and-robot settings. The key difference to our setting is that the robots are not transported on the truck but stay around one fixed hub. These variations reduce the complexity but induce practical disadvantages, such as long driving for robots and long waiting times for vehicles.

3.2. Related concepts with multiple trucks

To date, there is no approach to truck-and-robots with depots and multiple vehicles. We expand our literature analysis to related concepts that employ multiple vehicles and combine trucks with innovative transportation technologies. We analyze these works concerning our setting, focusing on multi-tour problems and the corresponding assignment of customers to tours. All these concepts share with our use case that trucks are combined with smaller vehicles that have a limited range, namely *robots*, *drones* or *cargo bikes*.

Robot-based concepts with multiple vehicles. A hub-and-robot approach is taken by Liu et al. (2020a), who consider delivery robots starting at and returning to fixed hubs. They assume larger robots that can carry more than one order. The customers are assigned to hubs based on k-means clustering. The remaining problem is a VRP for trucks supplying goods to the satellites and a VRP for the robots of each hub. Liu et al. (2020b) propose a similar concept, with the additional possibility of customers picking up their orders at a hub. Time windows are not considered in either of the publications.

The concept of multiple trucks with robot sidekicks is proposed by Chen et al. (2021b). It relies on robots transported by trucks and considers time windows but does not use robot depots. In this concept, the truck makes deliveries to customers, and robots make pendulum tours to other customers nearby in the meantime or while the truck waits for the return of the robots. The problem is solved with a cluster-first-route-second approach. Chen et al. (2021a) analyze the same problem and propose an ALNS for simultaneous clustering and routing. However, the savings potential is smaller compared to using robot depots. Chen et al. (2021b) report savings of 4 to 17% compared to normal truck deliveries, whereas Ostermeier et al. (2022) for example identified more than 50% savings compared to truck deliveries. Yu et al. (2022) consider the pick-up-and-delivery problem with several trucks and one robot aboard each truck. The robot can be launched by a truck, make several deliveries and then meet the same truck at a later stop, or the truck can wait at the stop where the robot was launched. An ALNS framework is used to minimize the weighted sum of travel distances but not total logistics costs (e.g., not including labor costs for waiting times). Based on this metric, in some scenarios, the system performs better than two benchmark approaches (hub-and-robot and a scenario with parallel tours by robots and trucks on their own). However, the reduction of emissions and costs would be much lower than in the MVTR case due to a very small number of robots and truck waiting time.

Truck-and-drone with multiple vehicles. A large body of truck-and-drone literature was published in recent years (see e.g., review of Otto et al., 2018). In these applications, a truck usually carries between one and four aerial drones (see Agatz et al., 2018; Moshref-Javadi et al., 2020; Murray & Chu, 2015; Murray & Raj, 2020 as examples for single-truck problems), which depart from the truck at a customer location to serve another customer and join the truck again at the start point or at another customer. There are no drone depots as in the truck-and-robot concept. The solution approaches for this particular application first solve a TSP (in the single truck case) or VRP (e.g., see Kitjacharoenchai et al., 2019) for the trucks and then select some customers for drone delivery. So far, only Li et al. (2020) consider multiple trucks and drones and deliveries with time windows. Their concept is based on a fleet of trucks delivering to customers directly and drones starting from the truck at the visit of customer locations or from the goods warehouse. Each drone can serve several customers per tour and then returns to its starting point, i.e., the truck waits for the drones to return. Dayarian et al. (2020) propose a fleet of trucks and drones, in which the latter supply further parcels to the trucks along their route as new orders are placed. This leads to a problem similar to a VRP, with the other decision on the meeting points for drone resupply. More recently, Luo et al. (2022) study a multi-truck routing problem with drones considering pickup and delivery operations. Each truck carries a single drone in their application, but each drone may visit multiple customers to fulfill pickup and delivery requests.

The truck-and-drone concepts differ significantly from robot concepts. Drone concepts consider a small and fixed number of drones that only serve some, not all of the customers (at most 50%, optimally around 25% Murray & Chu, 2015) and have to return to the truck. Robots move at pedestrian speed and return to depots instead of the trucks as otherwise tremendous waiting times apply. The main difference between the concepts is that in the robot concept presented, the number, sequence, and locations of stops (i.e., visited depots and drop-off locations) are not predefined but are part of the decision problem. This increases the complexity, especially when multiple vehicles are considered, as in the MVTR-RP. Therefore, existing routing approaches for truck-and-drone cannot be applied to the MVTR-RP, as they rely on the fact that the truck visits most customers on a truck-and-drone route. Drone depots are not considered. For a more detailed overview of drone delivery, we refer to Dayarian et al. (2020); Otto et al. (2018) and Macrina et al. (2020), and to Ostermeier et al. (2022) for an analysis of differences between truck-and-drone vs. truck-and-robot routing.

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

3.3. Research gap

The MVTR-RP constitutes an open research area as there is no approach in related truck-and-robot literature. Current literature on truck-and-robot routing only covers the single-vehicle problem. The distinctive features of the truck-and-robot problem complicate the direct transfer of available approaches, for example, from the classic VRP or truck-and-drone literature. The MVTR-RP has several specifics that differ from other delivery concepts, including VRPs. First, there are many potential locations for robot pickup and drop-off to be considered for truck routing. Second, these locations cannot store goods, which leads to synchronization between trucks and robots. Third, the truck and depots have limited capacity for two types of objects, goods and robots, and thus the decisions on stop locations are further restricted. Finally, deliveries must occur within time windows, as goods are retrieved from the robots manually in attended home deliveries. To summarize, none of the publications on related problems considers the routing of multiple trucks and the combination of coupled truck and robot movements, robot pickup along the tour and time windows. There is, consequently, a research gap when it comes to multi-vehicle settings.

4. Solution approach

The MVTR-RP generalizes the $\mathcal{NP}$-hard SVTR-RP. For the SVTR-RP, practically relevant problem sizes cannot be solved optimally in reasonable time with exact approaches (Heimfarth et al., 2022). An advanced heuristic for multi-vehicle truck routing and robot scheduling is needed to solve the problem efficiently. As we introduce a new decision problem, no reference approach is available that jointly includes the customer assignment to tours, the routing of trucks, and robot scheduling. Nevertheless, we base our solution approach on components of existing approaches for VRPs. In detail, we require an (initial) clustering of customers and an improvement phase. Clustering approaches for VRPs have been studied for decades, with the first approach provided by Gillett & Miller (1974). In our work, we derive a first clustering by constructing possible truck tours as candidates to which customers can be assigned to. This concept of template routes is an established approach to build a set of tour candidates in vehicle routing (see e.g., Kovacs et al., 2014; Li et al., 2005; Sungur et al., 2010; Tarantilis et al., 2012). Based on the template routes defined, an improvement is aspired. Due to the nature of the MVTR-RP, where small changes to the routes have a major impact on the overall solution, small neighborhoods for the search are a reasonable approach (e.g., assignment of a single customer to a different tour may have a significant impact on both truck stops required and robot scheduling). In this context, VNS approaches have been used successfully for many VRP variants (see e.g., Hansen & Mladenović, 2018; Hemmelmayr et al., 2009; Henke et al., 2015; Salhi et al., 2014) as it allows the exploitation of different neighborhoods. Following these insights from established VRP approaches and the analysis of differences to the MVTR-RP, we introduce a novel heuristic, the Truck-and-Robot Clustering and Routing (TRCR). Figure 4 outlines the complete heuristic.

The initialization generates a route for each available truck (i.e., the *template tours*). It allocates each customer to one of these initial truck routes (denoted as *clustering*), dependent on customers' locations and time windows (Section 4.1.1). The use of template tours has proved to be very effective as a basis for tour building and the subsequent improvement steps. The route of each customer cluster is then determined using a variable neighborhood descent (VND) framework (Section 4.1.2) that is based on Hansen & Mladenović (2018). The improvement phase (denoted as SINS), seeks improved routing and scheduling by changing the customer assignment to tours and truck routes. It also means that the clustering of customers to tours is improved by SINS. The principle of SINS is to improve a set of objects (tours in our case) by generating neighborhoods of all its elements and then searching for an improved set of elements from these neighborhoods. We consider small variations of existing tours (i.e., moving single customers between tours or add/remove single stops) to exploit possible neighborhoods and find improvements. SINS was designed to search these variations (neighborhoods) and find an optimal com-

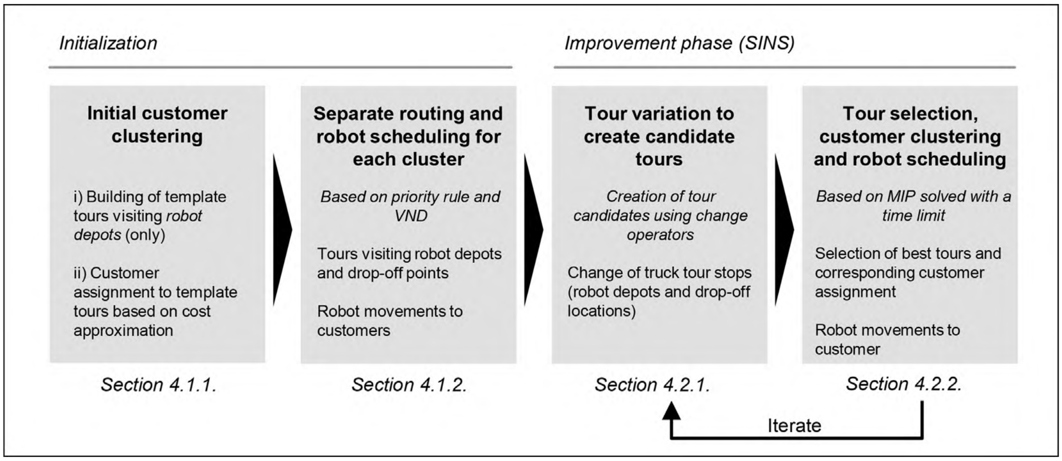

Fig. 4. Overview of the TRCR heuristic.

bination. Since the customer clustering to tours, truck routing, and robot scheduling are interdependent, SINS considers all of them in selecting a new set. We iteratively generate neighborhoods containing variations of each incumbent truck tour (Section 4.2.1) and solve an MIP to test whether a better set of truck-and-robot routes can be obtained by combining these variations (Section 4.2.2). As long as a better set can be found, this process is repeated. Otherwise, SINS terminates. SINS makes use of the fact that improvements can often be achieved by simultaneously changing two tours and reallocating customers between them. In the fourth step of Fig. 4, the clustering and truck routing (by choosing tours from a large pool) are decided simultaneously. As SINS uses an MIP in a metaheuristic fashion, it belongs to the growing field of matheuristics.

4.1. Initialization phase

The initialization provides a start solution consisting of multiple tours serving all customers. It first clusters customers to tours and then solves the routing and robot scheduling problem. Please note that the customer assignment to clusters is optimized again by SINS.

4.1.1. Initial customer clustering

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

(i) Template tour generation. The building of template tours is based on insights from typical truck-and-robot delivery tours found in our experiments: the first stop regularly lies outside a given range away from the start, the tour proceeds in an arc-shape, and finally returns to the start. Template tours only consider robot depots as these are essential for the supply of robots and the tour building. The tour generation is defined as follows. Candidate depots are selected that are between an inner and outer circle around the start position. The inner circle defines an area including a share of σ_1 robot depots, and the outer one an area including a share of σ_2 depots (see Fig. 5). All depots between these circles (see grey area) act as candidates for the template routes. Next, we select the convex hull of these candidate depots, i.e., only depots that form the convex hull are part of the template tours (see solid blue line in Fig. 5). Finally, we generate all possible tours that move along the convex hull and visit all of its depots exactly once. All depots

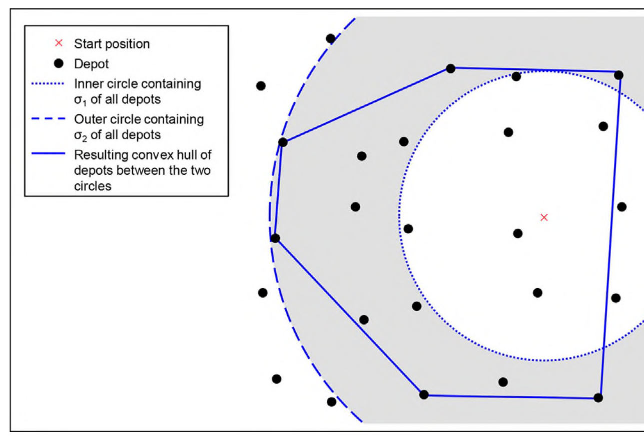

Fig. 5. Template tour generation.

on the convex hull are included in the tour and visited in their sequence as given by the convex hull, i.e., no shortcuts are allowed. This results in one template tour for every possible first depot and for each possible direction (clockwise/counterclockwise). This leads to $6 \times 2 = 12$ template tours for the example in Fig. 5. The generation of template tours can be applied for different shares of σ_1 and σ_2 to create a larger pool of templates. All tours created are then used as input for the customer clustering. Please note that the template tours are only used to cluster customers, not for actual routing.

(ii) Customer clustering. Customers have two characteristics related to routing: location and time windows. This means two customers with different locations and time windows may be assigned to the same tour if they fit into the tour sequence and arrival times. We have n clusters, each corresponding to one of the template tours. Inspired by the VRP clustering heuristic by Fisher & Jaikumar (1981), we solve an MIP to cluster customers for the subsequent routing based on a cost approximation for each customer-template tour combination. At this stage, we ignore robot availability and truck capacity. The notation for the clustering MIP is presented in Table 2.

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

Table 2
Notation for the initial customer clustering based on cost approximations.

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

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

$$\min \ F(X) = \sum_{k \in C} \sum_{\tau \in T} c_{k\tau} x_{k\tau} \tag{26}$$

subject to

$$\sum_{\tau \in T} x_{k\tau} = 1 \qquad \forall k \in C \tag{27}$$

$$x_{k\tau} \le z_\tau \qquad \forall k \in C, \tau \in T \tag{28}$$

$$\sum_{\tau \in T} z_\tau = n \tag{29}$$

$$x_{k\tau} \in \{0, 1\} \qquad \forall k \in C, \tau \in T \tag{30}$$

$$z_\tau \in \{0, 1\} \qquad \forall \tau \in T \tag{31}$$

The objective function (26) minimizes the total clustering costs of customer-tour combinations. Constraints (27) ensure that each customer is served once, (28) mandate that customers are served only via tours that are actually used, and (29) defines the number of template tours used as a basis for the improvement phase. Finally, (30) and (31) define the variable domains. The solutions of this step result in allocating each customer to one cluster. The customer clusters found are then used in the next step to determine actual routes for the customers in question.

4.1.2. Separate routing and robot scheduling for each cluster

Subsequent to the clustering, actual truck tours are determined by solving the routing problem for each customer cluster. As such, it resembles an SVTR-RP, and we build upon a known and efficient approach. We apply an adapted version of the heuristic proposed by Heimfarth et al. (2022) for the solution of an SVTR-RP. We use the priority rule “go to the location from which most robot deliveries can be started such that they reach customers in time” to generate an initial truck tour and a VND for improvement. After a truck tour is defined, the optimal robot schedule is determined. Here we apply a robot scheduling MIP, which is a special case of our “tour selection and customer clustering” MIP that will be introduced later in Section 4.2.2 when only one truck with unlimited parcel capacity is used. For the sake of streamlining the algorithmic description, we pick up on the robot scheduling approach later below when detailing the improvement phase.

After sequentially appending stops based on the priority rule mentioned, the resulting tour is used as a start solution for the VND framework, which sequentially tests the complete neighborhoods of the incumbent tour. If a better tour is found, it is accepted as the new incumbent tour, and the VND restarts from the first neighborhood. If no better tour is found, the VND accepts a random solution from the neighborhood with an initial probability of p, which is decreased by Δp in every iteration. This random acceptance of worse tours widens the search space in the early stage of the VND. If no new solution is accepted, the VND proceeds with the next neighborhood on the list. It terminates when all neighborhoods have been evaluated in a row without accepting a new solution. For a more detailed description of VND we refer to Hansen & Mladenović (2018). The neighborhoods evaluated within the VND are defined by the following operators (adapted from Heimfarth et al. (2022)) and evaluated in the order presented. Each neighborhood is limited to the m^{VND} cheapest tours (in terms of truck costs) to limit computational efforts.

1. **Remove drop-off point.** Removes a drop-off point from the current tour. Since truck distance is a main cost driver, this often leads to improvements.
2. **Remove depot.** Removes a depot. Depot removal may lead to non-feasible tours with respect to robot availability. In the event of non-feasibility, the closest depot is appended to the end of the tour.
3. **Add depot.** Adds a new depot to the existing tour. Additional depots can increase robot availability on parts of the tour and thus lead to better robot schedules at reduced costs.
4. **Add drop-off point.** Adds a new drop-off point to the tour. This may reduce robot usage or delays by bringing robots closer to the customers.
5. **Swap two stops.** Swaps two stops, i.e., both robot depots and drop-off locations. By swapping two stops, truck distance can be reduced or delays of deliveries starting at the later stop can be avoided.
6. **Relocate stop.** Shifts a single stop within a tour. Depending on whether the stop is shifted to an earlier or later point on the tour, the delays occurring at this stop or following stops can be reduced.

The VND results in n individual truck-and-robot routes serving each customer exactly once. As we generate each tour separately, the routes can still rely on the same robots in the same depots such that, in combination, this may lead to non-feasible solutions considering all tours. Nevertheless, the tours found to provide an efficient starting solution for the improvement phase. Feasibility is ensured in the next step (see Section 4.2.1).

4.2. Improvement phase (SINS)

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

4.2.1. Tour variation

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

- **Remove up to two stops.** Every possible tour resulting from removing one or two stops is added to the pool.

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

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

Before these pools can be passed to the next step of the algorithm to select the best combination of tours, we need to identify non-feasible tour combinations and exclude them. This is necessary as robot availability has been relaxed so far and may be violated. In the MVTR-RP, multiple tours access the same robot depots and respective robot availability, i.e., robot availability is shared between tours (see Eq. (17)). For a computationally efficient implementation of this constraint, we assume that robot depots can be refilled automatically (i.e., by transferring robots between depots) within a given refill time. No two tours may access the same robot depot during this time. We introduce the parameter ϑ^{f} to indicate the refill time, thus defining a waiting time until the next tour may access a given robot depot. Consequently, we forbid selecting two tours for the routing solution if these tours visit the same depot with less than ϑ^{f} time in between. Note that a very large ϑ^{f} ensures that depots are not shared between trucks at all and no refill is assumed. We exclude the combination of any two tours from the same pool (i.e., derived from the same incumbent tour) in the same way, as this proved inefficient. The rationale is that incumbent tours can be simultaneously adapted to make improvements, but choosing several similar tours proved unattractive.

However, excluding tour combinations may lead to overall non-feasibility when selecting tours in the next step, as it can happen that no two tours can be combined due to the use of the same robot depots. This may lead to insufficient robot depot visits to ensure sufficient robot availability. Therefore, we create another variant of each incumbent tour by sequentially replacing every depot included in another incumbent tour with the closest depot not included in any incumbent tour. This results in one additional variation for each of the n incumbent tours. This procedure is repeated until enough tours without robot depots used by multiple vehicles are available and, therefore, feasibility is ensured.

The result of this step is a set T of potential tours and a matrix $b_{\tau,\chi}$ defining whether any two tours τ and χ can be used at the same time.

4.2.2. *Optimal tour selection, customer clustering and robot scheduling*

This step selects the optimal set of tours and provides a feasible solution for the MVTR-RP based on the tours created within the tour variation step. This means it simultaneously defines the truck tours used, the customer clustering (i.e., from which truck each customer's robot starts) and the corresponding robot schedule (i.e., from which stop each customer's robot starts). Table 3 summarizes the notation of truck tour parameters and decision variables.

Since a set of potential truck tours τ is given, drop-off (D) and robot depot locations (R) do not need to be duplicated, and $L := D \cup R$ is the set of all locations potentially reachable by truck. A truck tour τ is defined by a tuple Y_τ, where $y_\tau(u)$ is the location of the uth stop, $y_\tau(u) \in L$. From the set of all potential tours, we want to select a maximum of n tours and allocate all customers to the stops of these tours in a cost-minimal manner. For each tour $\tau \in T$, we pre-calculate the arrival times $\psi_{\tau u}$ at its stops u based on the truck travel times. With known robot travel times and customer deadlines, we can then calculate the robot travel and the waiting and delay costs $c^{T}_{k\tau u}$ of serving a customer k from stop u on tour τ as shown in Eq. (32). Its first term represents the robot costs for travelling from the truck stop to the customer and waiting at the customer if needed. The second term adds the cost of a potential delay in the delivery. Again, as the robot always returns to the closest depot, the return costs are not decision-relevant.

$$c^{T}_{k\tau u} := c^{rob}(\vartheta^{r}_{y_\tau(u)k} + (d_k - \epsilon - \psi_{\tau u} - \vartheta^{r}_{y_\tau(u)k})^{+}) + c^{late}(\psi_{\tau u} + \vartheta^{r}_{y(u)k} - d_k)^{+} \qquad \forall u \in U, k \in C \tag{32}$$

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

$$\min \ F(X) = \sum_{k\in C}\sum_{\tau\in T}\sum_{u\in U_\tau} c_{k\tau u}\cdot x_{k\tau u} + \sum_{\tau\in T} c^{f}_{\tau} z_\tau \tag{33}$$

subject to

$$\sum_{\tau\in T}\sum_{u\in U_\tau} x_{k\tau u} = 1 \qquad \forall k \in C \tag{34}$$

$$\sum_{k\in C}\sum_{u\in U_\tau} \eta_k x_{k\tau u} \le G\cdot z_\tau \qquad \forall \tau \in T \tag{35}$$

$$z_\tau + z_\chi \le 1 \qquad \forall \tau, \chi \in T : b_{\tau\chi} = 1 \tag{36}$$

$$\sum_{\tau\in T} z_\tau \le n \tag{37}$$

$$\beta_{a\tau u} \le \beta_{a\tau u-1} + q_{\tau u-1} - q_{\tau u} - \sum_{k\in C} x_{k\tau u} \qquad \forall a \in L, \tau \in T, u \in U_\tau : a = y_t(u) \tag{38}$$

$$\beta_{a\tau u} = \beta_{a\tau u-1} \qquad \forall a \in R, \tau \in T, u \in U_\tau : a \neq y_t(u) \tag{39}$$

$$q_{\tau 0} = \delta \qquad \forall \tau \in T \tag{40}$$

$$\beta_{a\tau 0} = \beta_a \qquad \forall a \in R, \tau \in T \tag{41}$$

$$\beta_{a\tau u} = 0 \qquad \forall a \in D, \tau \in T, u \in U_\tau \tag{42}$$

$$x_{k\tau u} \in \{0,1\} \qquad \forall k \in C, \tau \in T, u \in U_\tau \tag{43}$$

$$z_\tau \in \{0,1\} \qquad \forall \tau \in T \tag{44}$$

Table 3
Notation for the optimal customer clustering to given tours.

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

$$\beta_{a\tau u} \ge 0 \qquad \forall a \in R, \tau \in T, u \in U_\tau \tag{45}$$

$$0 \le q_{\tau u} \le Q \qquad \forall \tau \in T, u \in U_\tau \tag{46}$$

The objective function (33) minimizes the sum of the costs of robot travel, waiting and potential delay and the costs of all truck tours selected. Constraints (34) ensure that every customer is supplied by exactly one robot and Constraints (35) that robots are only started from tours used, not exceeding the goods capacity of each truck. Constraints (36) allow the definition of tour combinations excluded, and (37) limit the total number of tours. Constraints (38) and (39) keep track of the robots available for a tour in each location and on the truck, depending on whether the location is visited (i.e., $a = y_\tau(u)$) or not. Due to Constraints (36), no coupling of robot availability between tours is required. The variable $\beta_{a\tau u}$ is only needed since the same tour could visit a robot depot several times. Equations (40) state the initial number of robots aboard the trucks, and (41) define the same for each robot depot. Constraints (42) ensure robots cannot be stored at drop-off points. Finally, Constraints (43)–(46) define the variable domains.

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

5. Numerical studies

This section completes numerical studies to obtain insights into the computational performance of the TRCR and managerial implications related to the MVTR-RP. We describe the parameter setting applied in our experiments in Section 5.1. Section 5.2 investigates the performance of TRCR compared to benchmark approaches. We further provide managerial insights using sensitivity analyses on depot refill times, time window distributions, and fleet sizes (Section 5.3).

5.1. Instances, parameter setting and test bed

We apply a representative data set for urban deliveries within a 4×4 kilometer delivery area. Munich (Germany) was selected as a representative example of a larger European city, and the delivery area resembles the northern half of the Munich city center. In the default data set, we assume a customer set $|C| = 50$, and randomly select 50 building locations in the delivery area obtained from OpenStreetMap Foundation (2019). Up to $n = 3$ trucks are available for this delivery area. $|R| = 25$ depots are first distributed in an equidistant manner and then slightly shifted by a random distance between 0 and 500 meters in south–north and east–west directions. $|D| = 48$ drop-off points are distributed by a random uniform distribution. The warehouse is randomly selected from the set of depots and drop-off locations. In our problem context, we consider a planning horizon of up to three hours, depending on the subsequent setting of delivery deadlines. For the determination of customer deadlines we assume a random-uniform distribution. The interval for the deadlines is defined as $[t^{\mathrm{M}} \cdot \rho_{\min}, t^{\mathrm{M}} \cdot \rho_{\max}]$, where t^{M} is the time needed to travel from the starting point to the furthest customer by truck. The parameters are set to $\rho_{\min} = 3$ and $\rho_{\max} = 6$ in the default case. The initial number of robots is set to $\beta_a = 0.08 \cdot |C|$ in every depot $a, a \in R$ and the depot refill time to $\vartheta^{\mathrm{f}} = 15$ minutes. Similarly, the truck's capacity and the initial number of loaded robots is set ($Q = \delta = 0.08 \cdot |C|$). The parcel volume per customer is $\eta = 1$, and the truck's parcel capacity is $G = 100$. The truck capacity reflects that both robots and parcels are transported together, and less space is available compared to standard trucks carrying up to 200 parcels. The average speed is 30 kilometers per hour for the truck and 5 kilometers per hour for the robots. A handling time per stop of 40 seconds is added to the resulting travel times. Following the costs empirically derived by Ostermeier et al. (2022), we assume the cost rates of $c^{\mathrm{dist}} = 0.20$ €/km and $c^{\mathrm{veh}} = 30$ €/h for the truck, $c^{\mathrm{late}} = 5$ € for delays and $c^{\mathrm{rob}} = 1.0$ €/h for robot use. Please note that we also apply the robot costs to the return time from the customer to the closest depot to enable a fair comparison of total costs. As explained above, these costs are not decision relevant (as known a priori) but contribute to the total costs.

Within the TRCR, the applied values of σ_1 and σ_2 (share of depots defining circle areas) in the *initial truck tour generation* are chosen as $[0.0, 0.3]$ and $[0.3, 0.8]$. We set the number of points considered as potential stops between two depots in the start heuristic to $m^{\mathrm{e}} = 5$. The VND is parameterized with the probability of accepting a worse solution $p = 0.5$, its decrease per cycle $\Delta p = 0.001$ and the maximum neighborhood size $m^{\mathrm{VND}} = 20$. The maximum number of tours for each operator used in the *tour variation* step is set to $m^{\mathrm{VAR}} = 20$, and the time limit m^{MIP} for solving the *optimal tour selection, customer assignment and robot scheduling* to 1 minute at the start and 10 minutes after the increase. The

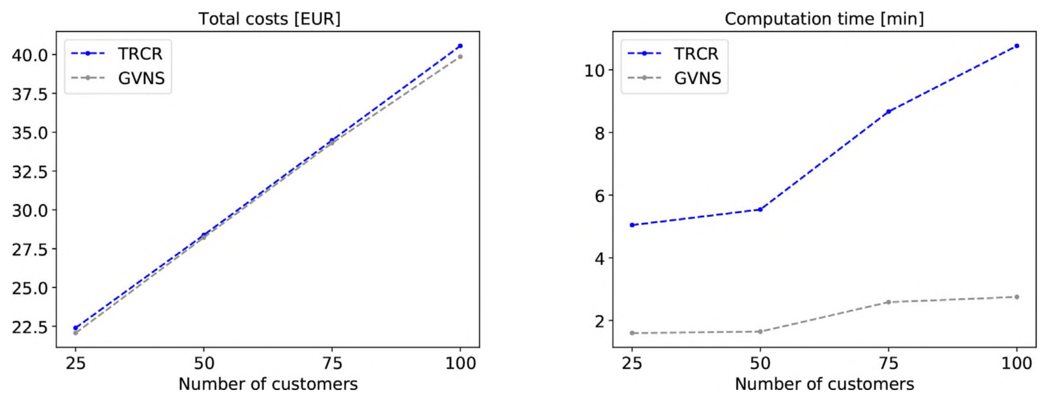

Fig. 6. Comparison of TRCR vs. GVNS benchmark for SVTR-RP, average of 20 instances.

default setting of the base scenario is highlighted in the following charts with a bold x-label. We generate 20 instances for each set of parameters, each with different locations and deadlines. Henceforth, each data point shown represents the average across 20 instances.

Our approach was implemented in Python (version 3.6.5) with Gurobi (version 8.0.1) as MIP solver and executed on a 64-bit PC with an Intel Core i7-8650U CPU (4×1.9 gigahertz), 16 gigabyte RAM, and Windows 10 Enterprise.

5.2. Computational efficiency of the TRCR-heuristics compared to alternative approaches

The MVTR-RP constitutes an $\mathcal{NP}$-hard problem. Even for the simpler SVTR-RP no exact solutions could be developed so far (see Heimfarth et al., 2022), and no comparison with exact solutions is available for relevant problem sizes. For example, the optimization of an MVTR-RP instance with six customers, six drop-off locations, and four robot depots (even without generating duplicate locations to further reduce the computation effort) was terminated with an optimality gap of 70% after 10 hours of computation. Reducing further the number of locations may enable optimal solutions but results in problem settings without any informative value for the solution quality. The lack of exact solutions and existing heuristics (due to the novelty of the problem) requires to refer to special cases of our problem. We, therefore, apply the GVNS framework for the SVTR-RP by Heimfarth et al. (2022) as a benchmark for single truck problems in the first comparison. This benchmark approach (denoted as *GVNS*) is most suitable as it addresses total logistics costs and outperforms other truck-and-robot routing approaches.

Moreover, we develop an alternative solution approach as a benchmark for the MVTR-RP with the GVNS at the center. We extend the GVNS of Heimfarth et al. (2022) with a customer clustering to tours to enable a comparison for the MVTR-RP. For the customer clustering, we solve a VRP MIP (see Appendix A), as this represents a good approximation of actual customer clusters due to the optimization concerning delivery times and distances covered. We relax the time window constraint to deadlines within the MIP to reduce computation times. We further limit the search to 60 minutes. The resulting clusters (i.e., customers served by the same vehicle in the VRP solution) are then input to the GVNS for subsequent truck-and-robot routing of each cluster. We denote this benchmark approach with *MIP&GVNS*. Compared to the integrated approach of TRCR with the simultaneous routing of multiple trucks and robot scheduling, the MIP&GVNS can be described as a cluster-first-route-second approach. We also tested the VRP clustering heuristic by Fisher & Jaikumar (1981) as an alternative, followed again by the GVNS. The resulting solution quality is, however, worse than for the MIP&GVNS. Please note that MIP&GVNS does not prevent the different trucks from using the same robots at depots and as such, provides a simplified search in favor of the benchmark approach.

Special case of a single truck. We set the number of available trucks to $n = 1$ to compare TRCR directly to the benchmark approach of Heimfarth et al. (2022). Figure 6 shows the comparison for different instance sizes. In this special case, TRCR comes close to the solution quality of the GVNS of Heimfarth et al. (2022), which has been developed specifically for such settings. Our approach leads to costs that are only 1% higher on average. This is an acceptable gap as TRCR is designed to solve the MVTR-RP with a larger pool of tours for several available vehicles. The GVNS benchmark, on the other hand, is tailored to a single truck and consequently uses tailored operators to improve the SVTR-RP. The computation time is two to three times (up to 8 minutes) higher, as TRCR also solves the *tour selection, customer clustering and robot scheduling* MIP. However, this is not needed in the case of a single truck. In summary, this validates TRCR's ability to find good solutions compared to state-of-the-art approaches for the SVTR-RP

Performance comparison for multiple trucks. The routing of multiple trucks is at the core of the MVTR-RP. Figure 7 shows the computation times and logistical performance obtained from our TRCR (simultaneous approach) and the benchmark denoted as MIP&GVNS (cluster-first-route-second approach) for different numbers of customers. The TRCR is 23–60% faster, especially since the clustering MIP always reaches its time limit and then requires additional time for the routing. TRCR computation times are generally at a level acceptable for practical use, as the tours can be planned during picking time in the warehouse. In additional experiments, we show that TRCR is able to solve instances with up to 200 customers in around 60 minutes. Moreover, TRCR is able to reduce total costs by 18–24%. The cluster-first-route-second benchmark results in the use of too many trucks and a suboptimal clustering of customers resulting in longer delays and truck distances. Since the benchmark does not prevent trucks from accessing the same depots, an average of 3 depots are visited by more than one truck. In comparison, TRCR ensures the minimum refill time between two visits of the same depot and thus results in only 0.5 depots visited by more than one truck on average. Consequently, TRCR leads to a Pareto improvement compared to the cluster-first-route-second benchmark. Furthermore, the average objective value reduction in the improvement phase of TRCR is 18%. This highlights the effectiveness of both our start heuristic and improvement phase. TRCR's runtime is dominated by the MIP solved for tour selection, customer clustering and robot scheduling in the improvement phase, which accounts for more than 99% of the runtime. A detailed comparison of the two approaches for individual instances is provided in Appendix B.

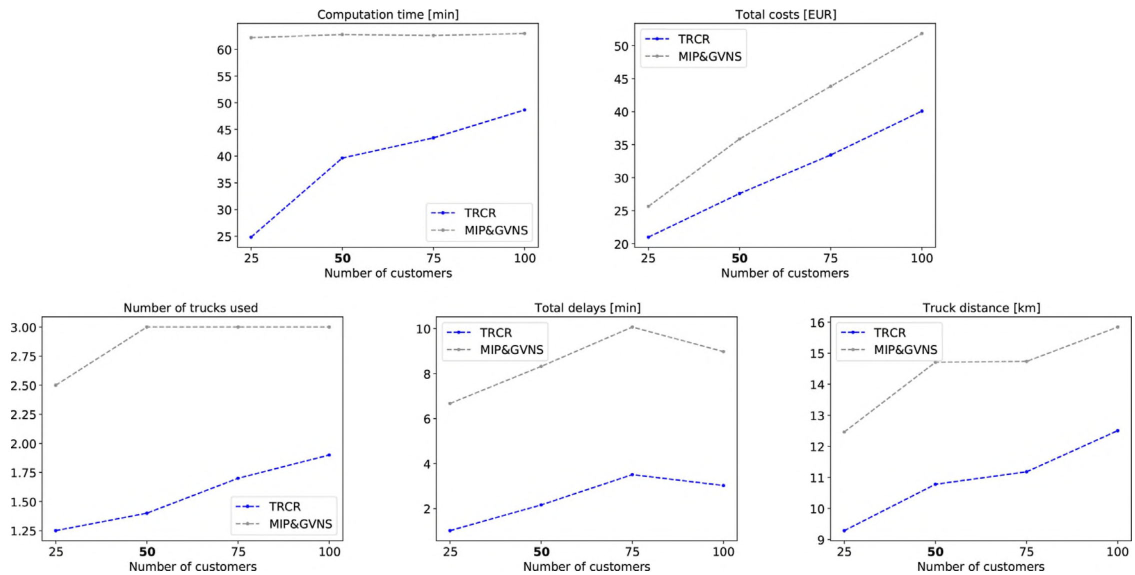

Fig. 7. Comparison of TRCR vs. benchmark for MVTR-RP, average of 20 instances.

Table 4
TRCR vs. benchmark for different spatial customer distributions, average of 20 instances.

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

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

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

5.3. Managerial insights to system performance

Having shown the algorithm's efficiency, we now analyze the impact of hierarchical and strategic managerial decisions entered into the MVTR-RP as input parameters and compare the MVTR-RP to conventional truck deliveries without robots. First, the assumption of refilling depots is analyzed, as this is a crucial assumption in the truck-and-robot concept and our solution approach. Next, the time window distribution is varied since time windows are expected to have a strong influence on customer clustering. Finally, we detail the benefits of additional trucks (and thus one further advantage of TRCR) in the case of tight time windows. To highlight the overall attractiveness of truck-and-robot systems, we henceforth compare the performance to traditional truck deliveries (i.e., all customers are supplied by trucks). We apply the MIP (see Appendix A) for this purpose, and we report the best-known solution (labeled *VRP*) and the lower bound of the costs (labeled as *VRP LB*) after 60 minutes of computation.

Robot depot refill assumptions and comparison with truck delivery. Robot availability at robot depots is limited. The time required to refill a depot with new robots after a truck visit could have a significant impact on solution quality if visiting a single depot several times is beneficial. We analyze the impact of this time on a truck-and-robot system with 25 (default case as described above) and 12 depots, respectively. Note that reducing the number of depots also leads to fewer available robots. The two scenarios are denoted TRCR-12 and TRCR-25. Figure 9 shows that there is hardly any impact on costs and other performance metrics with varying refill times for solutions obtained with TRCR. In particular, even if the time is set to 0 (i.e., two trucks arriving at the depot at the same time can access the depot's full robot availability), this does not lead to significant improvements. This means that the assumption in our solution approach (which prevents two trucks visiting the

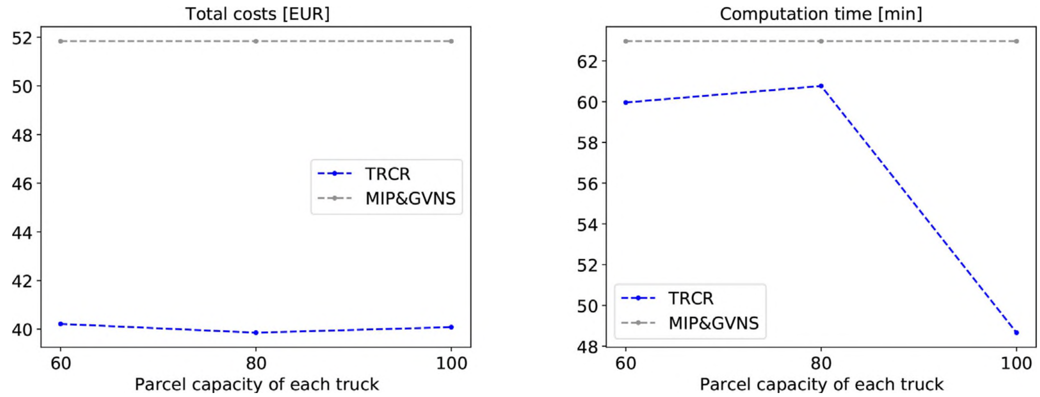

Fig. 8. TRCR vs. benchmark for varying truck capacity and 100 customers, average of 20 instances.

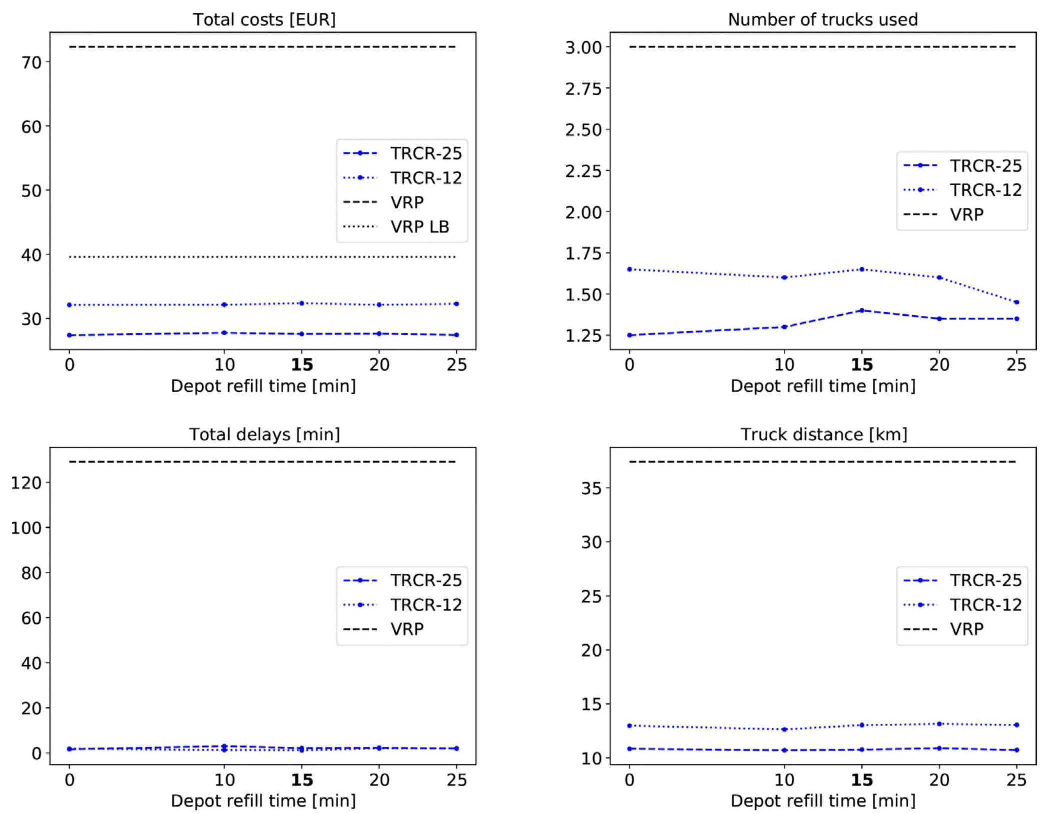

Fig. 9. System performance for different depot refill times, average of 20 instances.

same depot within the depot refill time) does not worsen the results. It further shows no benefit in ensuring an immediate refill of visited depots in practice. Reducing the number of depots from 25 to 12 leads to longer truck distances travelled and a cost increase of 17% on average. Comparison with the conventional truck-only delivery reveals cost savings of 62% and a truck mileage reduction of 71% due to the truck-and-robot concept. This corresponds to a 71% reduction in local emissions if diesel trucks are used. The MIP is terminated after 60 minutes, resulting in a 45% average MIP gap. However, even compared to the lower bound, TRCR has a cost savings potential of 30%.

Time window structure. The key feature of TRCR is the clustering of customers. The clustering is based on customer locations and time windows as they define the differences between any two customers. Besides the influence of locations (see Section 5.2), we therefore further assess the impact of time windows that are expected to have a strong impact on costs and logistical performance (see e.g., Heimfarth et al. (2022)). First, the effect of earlier and later time windows is assessed. Next, customers in the same region are offered similar time windows.

(i) *Early vs. late time windows*. When designing a last-mile delivery system, one crucial question is how fast deliveries can reach the customer. We, therefore, test the system's performance with earlier or later time windows, i.e., different values of the deadline factor interval $[\rho_{\min}, \rho_{\max}]$. Figure 10 summarizes results for a $\rho_{\min}$ of 1 (earlier), 3 (default) and 5 (later). With $\rho_{\max} = \rho_{\min} + 2$ the span of all deadlines remains constant. Comparing the results of TRCR for $\rho_{\min} = 3$ vs. 5, we see that additional trucks can ensure that earlier deadlines are met at scant additional costs. On the other hand, if time windows are too early ($\rho_{\min} = 1$ instead of 3), even increasing the number of trucks used by 61% cannot prevent a 19-fold increase in delays and further leads to a 35% longer distance travelled by trucks. Together, these

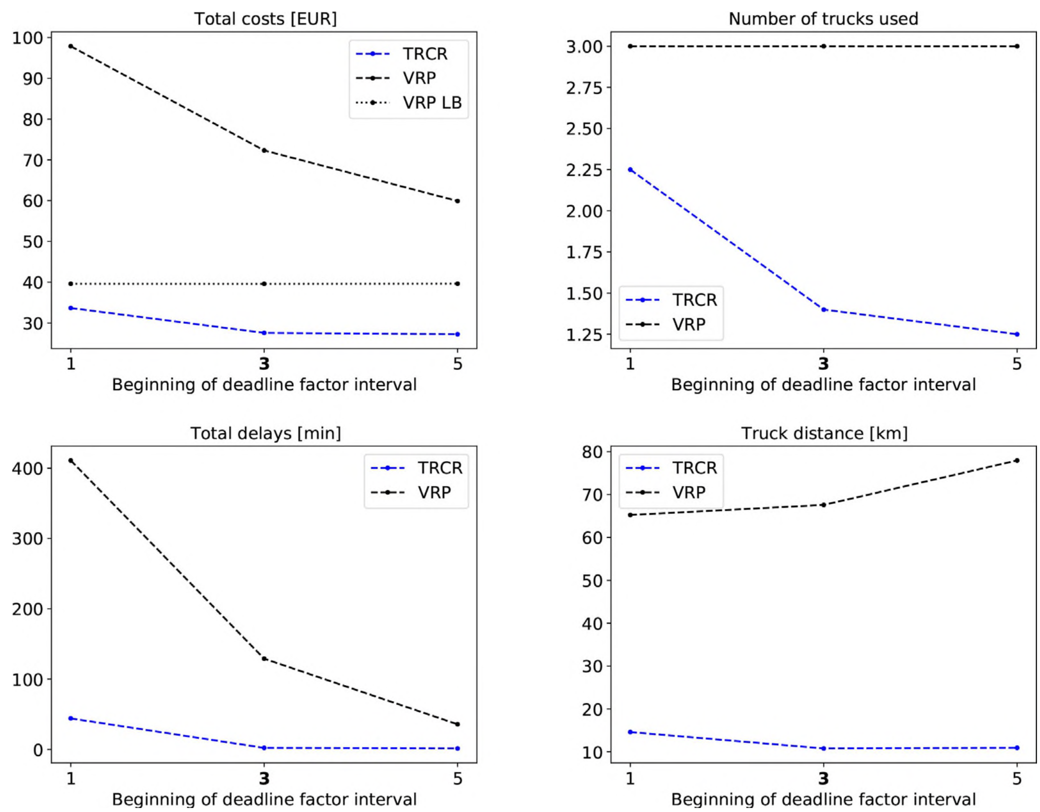

Fig. 10. System performance for different deadline factor intervals.

effects increase total costs by 22% (between 6 and 35% per instance). There is slight variance in the number of trucks used: for $\rho_{\min} = 1$; this number is 2 or 3 in all solutions; for $\rho_{\min} = 3$ or 5, only 1 or 2 trucks are used. This shows that a truck-and-robot system must be designed specifically to meet the lead times promised. In times of high demand, offering later time windows to customers can relieve the system. The scenario with truck deliveries (again denoted as *VRP*) benefits from later deadlines as well, as it can reduce its high level of delays. However, it does so by further increasing truck distance. The cost savings of using the truck-and-robot system remain high, at 54% with the later deadlines.

(ii) *Location-based time windows*. Assigning similar time windows to adjacent customers could potentially enable shorter truck tours and reduce robot waiting time and delays. We, therefore, test two policies for making such an assignment. Each policy splits the customers into two equal groups and assigns each group to one-half of the deadline factor interval. The first half of all deadlines are in one region and the second half in another. The first policy (denoted *distance*) splits the customers based on their distance from the warehouse and assigns the closer customers to the earlier deadlines. The second policy (denoted *angle*) finds a straight line through the warehouse that separates the customers into two equal groups and randomly assigns one to the earlier and one to the later half of the deadlines. We compare the TRCR results for these scenarios with the default case of entirely random deadlines. Both zone types lead to a decrease in delays (angle −68% and distance −85%). Interestingly, assigning zones based on distance does so while at the same time reducing the number of trucks used (−4%) and their total distance travelled (−4%). This shows that the same truck can first serve the closer customers with early deadlines and then take care of the customers who are further away with later deadlines. In the *angle* case, more trucks (7%) are needed, and they cover a larger distance (5%). Consequently, this time window distribution does not seem to facilitate efficient truck tours. The total costs in this scenario remain unchanged. Only the *distance* scenario enables a 2% cost reduction in total, thus leading to a Pareto improvement (all metrics considered are improved). It is, therefore, a favorable policy to offer customers a time window based on their distance from the warehouse.

Table 5
Performance change of TRCR for different available truck fleets, average of 20 instances.

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

Fleet size. One key question when implementing a truck-and-robot system is how many trucks will be needed. This is also necessary to plan the shifts. We, therefore, investigate the effect of the number of available trucks on the system's performance. The instances used were generated with a deadline factor interval of $[\rho_{\min}, \rho_{\max}] = [1, 3]$. This leads to the best solutions using two or three trucks in the default case of three available trucks. Table 5 shows the results with a varying number of available trucks. Our derived cost function for trucks includes a distance-based fee for trucks. This means that the fixed costs for a truck are transferred to mileage costs. With a total lifetime mileage, the total costs of a truck (incl. variable and fixed costs) can be calculated per distance unit (see Hübner & Ostermeier, 2019 for a similar approach). TRCR tends to use more trucks whenever they become available even if this does not reduce total costs: these are reduced by 24% when a second truck is added (between 11 and 42% per instance), primarily due to a 74% reduction in delays. The second truck is used in all instances. After that, additionally available trucks do not affect total costs, as the reduction in delays is outweighed by the increase in truck distance. When a third truck becomes available, it is only used in 30% of the instances, and a fourth truck only in 20%. Even in those cases, the benefits of using more trucks are marginal. This shows the necessity to employ more than one truck in the truck-and-robot concept and that TRCR can help to size the truck fleet for a given demand scenario optimally. In the VRP case (not shown in Table 5), all available trucks are used to reduce the high delays. Consequently, fewer available trucks result in a high-cost increase driven by delays.

6. Conclusion

Truck-and-robot systems will contribute to reducing costs, emissions and traffic congestion caused by last-mile delivery. Multiple trucks combined with robots are needed to enable large-scale applications and short lead times. Since state-of-the-art literature is limited to single-truck problems (e.g., Alfandari et al., 2022; Boysen et al., 2018a; Ostermeier et al., 2022), we present an extension to truck-and-robot routing by allowing the use of multiple trucks. The resulting *multi-vehicle truck-and-robot routing problem* (MVTR-RP) requires the simultaneous vehicle routing of multiple tours and the scheduling of robot movements. To simultaneously solve the routing and robot scheduling problem, we propose a novel heuristic, the Truck-and-Robot Clustering and Routing (TRCR) heuristic. The approach further relies on tailored start heuristics, problem-specific neighborhood operators and estimates leading to solvable MIPs. Our numerical experiments show that this simultaneous solution is 23 to 60% faster and yields 18 to 24% better solutions than a benchmark based on a cluster-first-route-second approach. The solution approach yields robust results overall, demonstrating the advantages of the truck-and-robot concept compared to various benchmarks and alternative delivery concepts. Further numerical studies show that using multiple trucks instead of one for the same delivery area and time windows reduces delays by about 75% on average, only slightly increases total truck distance by about 15%, and hence results in total cost improvements of about 20–25%. This highlights the benefits of multiple trucks. We were able to further identify cost reductions by 62% and truck emissions by 71% compared to conventional truck delivery. This improvement potential further increases with more challenging time windows. Sizing the truck fleet, defining the time windows customers can choose, and the total number of available robot depots are identified as key decisions for the hierarchical planning of truck-and-robot operations.

As we deal with an innovative delivery concept, there are still many promising opportunities for future research. One could build on our approach to develop exact solution approaches. Furthermore, the SINS approach of optimally choosing a set of tours from the neighborhoods of all incumbent tours has proven effective and could be transferred to other problems for which the solution is a set of objects (e.g., defining heterogeneous vehicle fleets, purchasing machinery, prioritizing maintenance tasks, defining insurance policies). Further modifications to the truck-and-robot concept and the instances applied can also be investigated, such as alternative delivery modes (by a driver, by drone etc.) and different spatial situations. Our results show the attractiveness of truck-and-robot in a city, but the system's high flexibility could make it adaptable to more rural settings, too. This could require the introduction of additional delivery modes. The concept can be further extended to pickup and delivery operations, as customer returns are becoming increasingly important for logistics operations. A heterogeneous fleet of trucks with varying capacities and range limitations should be assessed. This would be particularly relevant if an existing truck fleet is successively replaced with electric vehicles. It would require additional decisions and constraints in the tour selection step. So far, we leverage existing VRP approaches for developing the TRCR, namely a VND and template tours. Different solution approaches could be adapted to solve the MVTR-RP to enable a comparison of different approaches. Machine learning methods seem a promising way forward because of the complex interdependencies of the various problem aspects involved.

Acknowledgments

Our gratitude belongs to the Editor-in-Chief Ruud Teunter, and the three anonymous reviewers that provided constructive feedback to improve our paper.

Appendix A. An MIP model for customer clustering of the benchmark approach

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

We introduce the set of available vehicles F. The binary decision variable s_{fij} is 1 if vehicle f travels from location i to location j, and 0 otherwise. The auxiliary decision variable t_k denotes the arrival time at customer k and t_f^T the total tour time of vehicle f. This leads to the objective function (A.1), which sums up the cost of truck distance, truck time and delays. Constraints (A.2) ensure that every customer is visited exactly once. Constraints (A.3) keep track of the earliest possible arrival times at customers. Constraints (A.4) derive the delay from the arrival time. (A.5) defines the total operating time of each truck. (A.6) and (A.7) establish flow constraints for the trucks at every stop. Constraints (A.8) limit the truck capacity, and Constraints (A.9) to (A.12) define the solution space.

$$\min \sum_{f\in F}\sum_{i\in C\cup\{\omega\}}\sum_{j\in C\cup\{\omega\}} c^{\text{dist}}\lambda_{ij}s_{fij} + \sum_{f\in F} c^{\text{veh}}t_f^T + \sum_{k\in C} c^{\text{late}} l_k \tag{A.1}$$

subject to

$$\sum_{i\in C\cup\{\omega\}}\sum_{f\in F} s_{fik} = 1 \qquad \forall k\in C \tag{A.2}$$

$$t_j \geq t_i + \vartheta_{ij}^t - M\cdot(1-s_{fij}) \qquad \forall i\in C\cup\{\omega\},\, j\in C,\, f\in F \tag{A.3}$$

$$l_k \geq t_k - d_k \qquad \forall k\in C \tag{A.4}$$

$$t_f^T \geq t_k + \vartheta_{k\omega}^t - M\cdot(1-s_{fk\omega}) \qquad \forall k\in C,\, f\in F \tag{A.5}$$

$$\sum_{i\in C\cup\{\omega\}} s_{fik} = \sum_{i\in C\cup\{\omega\}} s_{fki} \qquad \forall k\in C,\, f\in F \tag{A.6}$$

$$\sum_{k\in C} s_{f\omega k} \leq 1 \qquad \forall f\in F \tag{A.7}$$

Table B.6
TRCR vs. MIP&GVNS performance for the 20 instances with 50 customers (negative number indicates reduction due to TRCR).

Instance	Cost difference [%]	Runtime difference [%]
0	−17	−63
1	−16	13
2	−26	−80
3	−24	−48
4	−21	−31
5	−26	3
6	−19	−10
7	−31	−69
8	−15	−58
9	−26	−70
10	−27	14
11	−23	−79
12	−23	−43
13	−27	52
14	−19	−87
15	−27	−4
16	−24	2
17	−21	−41
18	−24	−61
19	−25	−77
Average	−23	−37

$$\sum_{i \in C \cup \{\omega\}} \sum_{k \in C} \eta_k s_{fik} \le G \qquad \forall f \in F \tag{A.8}$$

$$s_{fij} \in \{0, 1\} \qquad \forall i, j \in C \cup \{\omega\}, f \in F \tag{A.9}$$

$$t_\omega = 0 \tag{A.10}$$

$$t_k \ge 0; \;\; l_k \ge 0 \qquad \forall k \in C \tag{A.11}$$

$$t_f^T \ge 0 \qquad \forall f \in F \tag{A.12}$$

Appendix B. Detailed computation results per instance

Table B.6 compares TRCR to MIP&GVNS for every single instance of our default data set with 50 customers. Note that the cost advantage of TRCR is quite stable, while computation times show a stronger variance.

References

Agatz, N., Bouman, P., & Schmidt, M. (2018). Optimization approaches for the traveling salesman problem with drone. *Transportation Science, 52*(4), 965–981.
Alfandari, L., Ljubić, I., & de Melo da Silva, M. (2022). A tailored Benders decomposition approach for last-mile delivery with autonomous robots. *European Journal of Operational Research, 299*, 510–525.
Anderluh, A., Hemmelmayr, V. C., Nolz, P. C., et al., (2017). Synchronizing vans and cargo bikes in a city distribution network. *Central European Journal of Operations Research, 25*(2), 345–376.
Arslan, A. M., Agatz, N., Kroon, L., Zuidwijk, R., et al., (2019). Crowdsourced delivery—A dynamic pickup and delivery problem with ad hoc drivers. *Transportation Science, 53*(1), 222–235.
Bakach, I., Campbell, A. M., & Ehmke, J. F. (2021). A two–tier urban delivery network with robot–based deliveries. *Networks, 75*(4), 467–483.
Bakach, I., Campbell, A. M., & Ehmke, J. F. (2022). Robot-based last-mile deliveries with pedestrian zones. *Frontiers in Future Transportation, 2*.
Baum, L., Assmann, T., Strubelt, H., et al., (2019). State of the art - automated micro-vehicles for urban logistics. *IFAC-PapersOnLine, 52*(13), 2455–2462.
Bennett, T. (2020). Autonomous last mile delivery robots are gaining traction, slowly. https://www.sharedmobility.news/autonomous-last-mile-delivery-robots-are-gaining-traction-slowly/.
Boysen, N., Briskorn, D., Fedtke, S., & Schwerdfeger, S. (2018a). Drone delivery from trucks: Drone scheduling for given truck routes. *Networks, 72*(4), 506–527.
Boysen, N., Fedtke, S., Schwerdfeger, S., et al., (2021). Last-mile delivery concepts: A survey from an operational research perspective. *OR Spectrum*, (43), 1–58.
Boysen, N., Schwerdfeger, S., & Weidinger, F. (2018b). Scheduling last-mile deliveries with truck-based autonomous robots. *European Journal of Operational Research, 271*(3), 1085–1099.
Buldeo Rai, H., Verlinde, S., & Macharis, C. (2019). The "next day, free delivery" myth unravelled. *International Journal of Retail & Distribution Management, 47*(1), 39–54.
Chen, C., Demir, E., & Huang, Y. (2021a). An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and delivery robots. *European Journal of Operational Research, 294*(3), 1164–1180.
Chen, C., Demir, E., Huang, Y., & Qiu, R. (2021b). The adoption of self-driving delivery robots in last mile logistics. *Transportation Research Part E: Logistics and Transportation Review, 146*, 102214.
Daimler (2019). Vans & robots. https://www.daimler.com/innovation/specials/future-transportation-vans/paketbote-2-0.html.
Dayarian, I., Savelsbergh, M., & Clarke, J.-P. (2020). Same-day delivery with drone resupply. *Transportation Science, 54*(1), 229–249..
Fisher, M. L., & Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks, 11*, 109–124.
Gillett, B. E., & Miller, L. R. (1974). A heuristic algorithm for the vehicle-dispatch problem. *Operations Research, 22*(2), 340–349.
Golden, B., Raghavan, S., & Wasil, E. A. (2008). The vehicle routing problem: Latest advances and new challenges. *Operations research/computer science interfaces series*. Springer.
Grand View Research (2020). Autonomous last mile delivery market size, share & trends analysis report. https://www.grandviewresearch.com/industry-analysis/autonomous-last-mile-delivery-market.
Hansen, P., & Mladenović, N. (2018). Variable neighborhood search. In R. Martí, P. M. Pardalos, & M. G. C. Resende (Eds.), *Handbook of heuristics: vol. 328* (pp. 759–787). Cham: Springer International Publishing.
van Heeswijk, W. J. A., Mes, M. R. K., Schutten, J. M. J., et al., (2019). The delivery dispatching problem with time windows for urban consolidation centers. *Transportation Science, 53*(1), 203–221.
Heimfarth, A., Ostermeier, M., & Hübner, A. (2022). A mixed truck and robot delivery approach for the daily supply of customers. *European Journal of Operational Research, 303*, 401–421.
Hemmelmayr, V. C., Doerner, K. F., & Hartl, R. F. (2009). A variable neighborhood search heuristic for periodic routing problems. *European Journal of Operational Research, 195*, 791–802.
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., Holzapfel, A., Kuhn, H., & Obermair, E. (2019). Distribution in omnichannel grocery retailing: An analysis of concepts realized. In S. Gallino, & A. Moreno (Eds.), *Operations in an omnichannel world*. In *Springer series in supply chain management* (pp. 283–310). Cham: Springer Nature Switzerland AG.
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.
Hübner, A., & Ostermeier, M. (2019). A multi-compartment vehicle routing problem with loading and unloading costs. *Transportation Science, 53*(1), 282–300.
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 and Logistics Management, 46*(6/7), 543–561.
Jaller, M., Otero-Palencia, C., Pahwa, A., et al., (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. A., & Brunese, P. A. (2019). Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. *Computers & Industrial Engineering, 129*, 14–30.
Kiwibot (2020). Company website. https://www.kiwibot.com/.
Kovacs, A. A., Parragh, S. N., Hartl, R. F., et al., (2014). A template-based adaptive large neighborhood search for the consistent vehicle routing problem. *Networks, 63*(1), 60–81.
Li, F., Golden, B., & Wasil, E. (2005). Very large-scale vehicle routing: New test problems, algorithms, and results. *Computers & Operations Research, 32*(5), 1165–1179.
Li, H., Wang, H., Chen, J., & Bai, M. (2020). Two-echelon vehicle routing problem with time windows and mobile satellites. *Transportation Research Part B: Methodological, 138*, 179–201.
Liu, D., Deng, Z., Mao, X., Yang, Y., & Kaisar, E. I. (2020a). Two-echelon vehicle-routing problem: Optimization of autonomous delivery vehicle-assisted e-grocery distribution. *IEEE Access, 8*, 108705–108719.
Liu, D., Deng, Z., Zhang, W., Wang, Y., & Kaisar, E. I. (2020b). Design of sustainable urban electronic grocery distribution network. *Alexandria Engineering Journal, 60*(1), 145–157.
Luo, Z., Gu, R., Poon, M., Liu, Z., & Lim, A. (2022). A last-mile drone-assisted one-to-one pickup and delivery problem with multi-visit drone trips. *Computers & Operations Research, 148*, 106015.
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. https://www.marble.io/.

McKinsey & Company (2021). Efficient and sustainable last-mile logistics: Lessons from Japan. https://www.mckinsey.com/industries/travel-logistics-and-infrastructure/our-insights/efficient-and-sustainable-last-mile-logistics-lessons-from-japan?cid=eml-web.

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

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

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

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

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

OpenStreetMap Foundation (2019). Main page. 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. (2022). Cost-optimal truck-and-robot routing for last-mile delivery. *Networks, 79*(3), 364–389.

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 N. Mustafee, K.-H. G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, & Y. J. Son (Eds.), *Proceedings of the 2019 winter simulation conference*. Piscataway, NJ, United States: IEEE.

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

Salhi, S., Imran, A., & Wassan, N. A. (2014). The multi-depot vehicle routing problem with heterogeneous vehicle fleet: Formulation and a variable neighborhood search implementation. *Computers & Operations Research, 52*, 315–325.

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

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 T. Bui (Ed.), *Proceedings of the 52nd Hawaii international conference on system sciences*. Maui, Hawaii: University of Hawaii at Manoa.

Starship (2019). Company website. https://www.starship.xyz/.

Statista (2022). U.S. consumers: Online grocery shopping - statistics & facts. https://www.statista.com/statistics/257516/us-retail-e-commerce-sales-cagr-by-product-category/.

Sungur, I., Ren, Y., Ordóñez, F., Dessouky, M., & Zhong, H. (2010). A model and algorithm for the courier delivery problem with uncertainty. *Transportation Science, 44*(2), 193–205.

Tarantilis, C. D., Stavropoulou, F., & Repoussis, P. P. (2012). A template-based tabu search algorithm for the consistent vehicle routing problem. *Expert Systems with Applications, 39*(4), 4233–4239.

Toth, P., & Vigo, D. (2014). Vehicle routing: Problems, methods, and applications. *MOS-SIAM series on optimization* (2nd ed.). Society for Industrial and Applied Mathematics.

Vans, M.-B. (2016). Vans & robots: Efficient delivery with the mothership concept. URL https://youtu.be/yUMOLzCsifs.

Yu, S., Puchinger, J., Sun, S., et al., (2022). Van-based robot hybrid pickup and delivery routing problem. *European Journal of Operational Research, 298*, 894–914.