



Synchronous truck-robot-drone last-mile delivery

Behrang Bootaki^a, Omar Abou Kasm^{a,*}, Joe Naoum-Sawaya^{b,c}

^a Department of Finance, Economics, and Decision Sciences, Algoma University, 1520 Queen Street E., Sault Ste. Marie, P6A 2G3, ON, Canada

^b IE University, Madrid, Spain

^c Ivey Business School, Western University, 1255 Western Rd, London, N6G 0N1, ON, Canada

ARTICLE INFO

Keywords:

Last-mile delivery
Truck-robot-drone delivery
Heuristics
Drone energy model

ABSTRACT

The continued growth of online shopping in recent years has placed greater demands on efficient last-mile delivery systems. Traditional truck parcel delivery faces challenges such as slow delivery in congested urban areas and high environmental impacts. These issues have inspired researchers and practitioners to develop novel multi-modal delivery systems based on the coordination between traditional delivery trucks and newer technologies such as delivery robots and drones. In this paper, we model a synchronous truck-robot-drone last-mile delivery system and propose a mixed integer program along with a heuristic solution. The proposed model is designed for a delivery truck equipped with a multi-visit drone and multiple single-delivery robots. The model accounts for multiple operational constraints, including drone energy endurance and the load capacities of both the drone and the robots. Computational experiments are conducted to demonstrate the efficiency of the proposed heuristic in solving real-sized instances. The impact of the proposed delivery system is evaluated using the 2021 Amazon Last Mile Routing Research Challenge dataset in Seattle. The case results show that replacing the truck with the proposed system can reduce travel times within customer zones by an average of 40% compared to the original routes. Time savings can reach up to 52% for larger deliveries, demonstrating the system's potential to improve efficiency and contribute to reduced environmental impacts in urban logistics.

1. Introduction

The transportation of goods from a central depot to customers' doorsteps, commonly known as the last-mile delivery, accounts for almost half of the total cost of the supply chain (Jacobs et al., 2019). The impact of last-mile delivery on the environment is also a major concern, as freight transport is a major source of carbon emissions within the transportation sector. For example, Amazon's US delivery emissions grew 75%, from 3.3 million tons of CO₂ in 2019 to 5.8 million tons in 2023. This increase is comparable to adding nearly half a million gas powered cars to the road for an entire year (Taft, 2024).

To address these challenges, researchers have been increasingly focused on developing more innovative models that incorporate robots and drones into last-mile delivery due to their smaller footprint on operational costs and the environment (Boccia et al., 2024; Yu and Puchinger, 2024). However, due to technological limitations, such as the limited payload capacity and travel range of drones and robots, the truck component remains an essential part of these novel hybrid delivery models. For example, UPS and Workhorse Group demonstrated a joint truck and drone delivery system called HorseFly, where a UPS truck is equipped with a single drone

* Corresponding author.

E-mail addresses: behrang.bootaki@algonau.ca (B. Bootaki), omar.aboukasm@algonau.ca (O. Abou Kasm), jnaoum-sawaya@ivey.ca (J. Naoum-Sawaya).

<https://doi.org/10.1016/j.trc.2026.105952>

Received 11 December 2025; Received in revised form 9 July 2026; Accepted 13 August 2026

Available online 24 August 2026

0968-090X/© 2026 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Table 1
Comparison between three types of delivery modes: trucks, drones, and robots.

Mode	Strengths	Limitations
Truck	-Can carry large parcels and multiple items. -Fast on highways, freeways, and main roads.	-Slow in high-traffic areas. -Longer operating times in congested areas due to limited parking. -Gas-based, CO₂-emitting, environmentally harmful. -Costlier due to operator-based operations.
Drone	-Very fast (up to 100 km/h). -Line-of-sight direct travel. -Unaffected by traffic congestion.	-Less safe than robots in congested areas with tall buildings. -Limited capacity (2–5 kg), unsuitable for heavy loads. -Energy-intensive, payload weight strongly affects range.
Robot	-Safer than drones, especially in urban environments. -Suitable for narrow streets and congested areas. -Can carry heavier payloads (10–100 kg) up to 50 km.	-Very slow. -Limited to short distances despite battery capacity. -Best for clustered customers in dense neighborhoods.

that performs autonomous package deliveries (UPS, 2017). The Mercedes-Benz Vision Van is another example of a joint truck-drone delivery system in which a truck is equipped with two drones, each with a payload capacity of two kilograms, for autonomous delivery within a radius of 10 km. In another demonstration, Mercedes-Benz and Starship Technologies developed a platform where a prototype Sprinter van carries eight Starship robots. Each robot navigates autonomously to the customer's doorstep and then returns to the van after completing the delivery (Mercedes-Benz Group, 2017).

Due to the much faster speed of drones compared to robots, truck-drone delivery models may appear more efficient than truck-robot models. However, due to physical limitations, it is not always possible to use multiple drones on a truck. Although using larger trucks, typically with higher environmental impacts, might allow the integration of bigger drones with multiple compartments, such solutions are not always feasible due to strict air traffic regulations in congested public areas or adverse weather conditions (Ostermeier et al., 2022). For this reason, a multi-modal delivery fleet can further enhance the flexibility of last-mile delivery systems, as each mode offers unique strengths that compensate for the limitations of the others. (see details in Table 1 Simoni et al., 2020; Reed et al., 2022; Heimfarth et al., 2022; Mahmoudi and Eshghi, 2022; Yu and Puchinger, 2024; Yu et al., 2024).

In this paper, we introduce a delivery model based on a truck, a single multi-visit drone, and multiple single-delivery robots. All robots and the drone are on board the truck at the beginning of the shift. Therefore, this model distinguishes itself from hub-based frameworks that rely on local depots for drone or robot deployment (Boysen et al., 2018; Schermer et al., 2020; Ostermeier et al., 2022; Heimfarth et al., 2022; Zhou et al., 2023; Morim et al., 2024; Lichau et al., 2025), thereby eliminating the hub-related infrastructure costs. Additionally, the proposed model operates synchronously, allowing the drone and robots to complete their assigned deliveries before rejoining the truck, at a location different from the launch node. This dynamic synchronization enhances truck utilization, contrasting with other hybrid models in which the truck remains idle at a parking location until the drones or robots return (Mercedes-Benz Group, 2017; Moeini and Salewski, 2019).

Fig. 1 presents a conceptual schematic of the proposed delivery truck. According to this platform, there is a dedicated section on the truck's rooftop for the drone, which is accessible from inside the truck for loading and battery swapping. There is also a separate rear section on the truck with multiple shelves for carrying packages, with space underneath to accommodate multiple robots. The truck driver can load the drone and robots from inside the truck. The drone is then launched from the rooftop and the robots are deployed from the side of the truck via a ramp.

The key decisions of the proposed model include: (i) how to allocate customers between the truck, robots, and drone; (ii) where to release the robots/drone along the truck's main route; (iii) where to collect the robots/drone during the truck's main route; and (iv) how to design the delivery routes for the truck, robots, and drone. The latest time when all vehicles return to the depot, defined as the delivery completion time, serves as the performance measure. The model is subject to operational constraints, such as (i) the drone's maximum energy capacity, maximum payload capacity, and maximum number of compartments; and (ii) the robots' maximum payload capacity and maximum traveling distance.

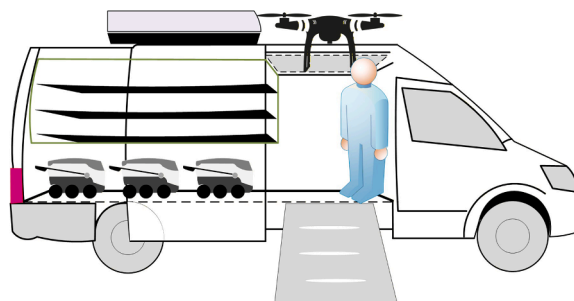


Fig. 1. Concept of a truck with a drone and multiple robots - a possible layout.

To efficiently solve the proposed problem, we develop a heuristic solution based on the route-first, split-second approach (Agatz et al., 2018; Ha et al., 2018; Hokama et al., 2025; Duan et al., 2025) and use the lazy drone property proposed by Hokama et al. (2025), which avoids full enumeration of all potential robot/drone delivery operations by searching only the promising ones. Our proposed method can efficiently handle real-world last-mile delivery problems with 50 to 200 customers, comparable in scale to typical Amazon operations, with computational times ranging from seconds to approximately two hours depending on instance size.

This paper makes the following contributions:

1. A synchronous truck-robot-drone last-mile delivery model is introduced for customers home delivery with the objective of minimizing the total delivery completion time.
2. A Mixed-Integer Linear Program (MILP) and a route-first, split-second heuristic are developed to model and solve this problem.
3. Computational experiments are conducted to show the efficiency of the proposed heuristic in solving large instances.
4. A real-world case study, based on Amazon's parcel delivery operations in the City of Seattle, USA, is conducted to illustrate the effectiveness of the proposed multi-modal delivery system.

The remainder of this paper is organized as follows: Section 2, reviews relevant recent research papers, and Sections 3 and 4 present the mathematical formulation of the problem and the proposed heuristic solution, respectively. Next, Sections 5 and 6 present computational experiments and a case study, respectively. Section 7 discusses the extensibility of the proposed model formulation and solution with respect to robot and drone configurations beyond those considered in the base model. Finally, Section 8 concludes the paper and outlines future directions.

2. Literature review

This section first presents an overview of truck-drone and truck-robot delivery models in last-mile logistics, then highlights the limited research on hybrid truck-robot-drone delivery.

2.1. Truck-drone delivery

Truck-drone last-mile delivery systems account for a large portion of the research on hybrid last-mile delivery models. The Flying Sidekick Traveling Salesman Problem (FSTSP), presented by Murray and Chu (2015), is a pioneering work in synchronous truck-drone delivery that integrates a drone into the classic Traveling Salesman Problem (TSP). Agatz et al. (2018) explored a Traveling Salesman Problem with Drone (TSP-D), where the drone and truck can visit the same node more than once. They proposed a route-first, cluster-second solution method, in which the routing stage generates an initial TSP tour and the clustering stage assigns customers to either the truck or the drone while preserving the node order of that tour. Ha et al. (2018) addressed the cost-minimization version of the TSP-D, aiming to reduce total transportation expenses by applying two algorithms: a local search-based method and a Greedy Randomized Adaptive Search Procedure (GRASP). Hokama et al. (2025) proposed a heuristic based on a route-first, split-second approach for solving the FSTSP. Their method constructs a graph of candidate truck-drone operations using a modified enumeration search strategy and then identifies the efficient combination of these operations by solving a shortest path problem over the resulting graph. Karak and Abdelghany (2019) considered the synchronous truck-drone systems with a multi-visit drone, i.e., drones that can carry multiple packages, and incorporated a maximum flight range for the drone and payload constraints. The multi-visit capability introduces challenges regarding flight ranges since the sequence of customers directly impacts the drone's energy consumption and thus its range, as the total payload weight varies depending on which customer is served first. Jeong et al. (2019) considered a hybrid truck-drone system with practical assumptions such as the impact of parcel weight on drone energy consumption and no-fly zones. Several papers, such as Cheng et al. (2020) and Poikonen and Golden (2020), presented different approaches to model drone energy consumption as a function of payload and integrated them within the optimization model. The inclusion of heterogeneous drones is another direction in this area. Wang et al. (2019) considered two classes of drones, including truck-carried and independently operating drones. Wen and Wu (2022) proposed a two-echelon system where a mothership drone (acting as an alternative carrier to a truck) deploys smaller drones that return to local depots upon task completion. Luo et al. (2024) considered multi-visit heterogeneous drones synchronizing with a single truck. Wang and Sheu (2019) and Gao et al. (2023) consider multiple trucks and multiple heterogeneous drones with the assumption of dedicated launch and retrieval locations distinct from customer nodes. Recent surveys by Luo et al. (2025) and Cengiz et al. (2025) provide comprehensive overviews of the literature on truck-drone modeling and solution methodologies.

2.2. Truck-robot delivery

Although robots are much slower than drones, they offer several advantages over drones, such as lower environmental and safety impacts, larger payload capacities, less energy intensive and more reliable deliveries, and lower manufacturing costs (Ostermeier et al., 2022). Therefore, hybrid delivery models based on trucks and delivery robots have also been presented in the literature. In terms of modeling, apart from certain drone-specific operational constraints, such as energy depletion modeling, no-fly zones, and Euclidean versus Manhattan distance measurements, many modeling and solution approaches used in truck-drone delivery can also be applied to truck-robot delivery (Yu et al., 2020, 2024).

The literature review on urban last mile truck-robot delivery is divided into two main directions: truck-borne and hub-based delivery models. In truck-borne delivery models, the truck functions as a 'mothership' that deploys and retrieves robots (Simoni et al.,

2020; Yu et al., 2020, 2024). Trucks loaded with packages transport robots to launching nodes, from which the robots serve customers before returning to the same or another node for retrieval. In hub-based truck-robot delivery, the truck visits a number of robot hubs holding the delivery robots, loads them, and proceeds to the next hub, while the robots perform the delivery process independently of the truck and then return to their original hub (Boysen et al., 2018; Bakach et al., 2021; Heimfarth et al., 2022; Liu et al., 2022; Ostermeier et al., 2023; Yu and Puchinger, 2024).

2.3. Truck-robot-drone delivery

Given the complementary advantages of robots and drones, a few papers in the literature have explored integrating drones and robots with trucks to improve last-mile delivery systems. Moeini and Salewski (2019) investigate the routing problem of a delivery truck carrying multiple drones and Autonomous Transportation Vehicles (ATVs). The truck visits a predefined set of grid points serving as parking spaces, each at most once. At each visited grid point, the drones and ATVs are deployed to deliver parcels to customers, and the truck waits for them to complete their deliveries and return. This strategy was originally introduced by Mercedes-Benz's Vans & Robots (Mercedes-Benz Group, 2017). The main drawback of this modeling approach is that the truck must remain idle while the robots and drones are performing deliveries. This disregards the main advantage of synchronization between the truck and other vehicles where the truck can continue visiting other customers while the other vehicles are performing deliveries in parallel, and they can subsequently rejoin at a later rendezvous node along the truck's path.

Schermer et al. (2020) investigate the drone-assisted traveling salesman problem with robot stations, where a single truck equipped with a drone serves all customers while potentially utilizing one or more robot stations. Morim et al. (2024) extend this work by introducing multiple trucks into the model where the trucks, with packages and drones but no robots on board, visit pre-established robot hubs, load the robots, and proceed to the next destination. The robots then carry out their deliveries independently of the truck's route and return to their respective hubs upon completing their tasks. Meanwhile, trucks and drones serve customers in tandem. Finally, and more recently, Mokhtari-Moghadam et al. (2025) present a multi-vehicle delivery framework involving trucks supported by drones and self-driving robots to serve customers either at home or through local depots. While the study provides valuable insights into coordinated truck-robot-drone operations, it simplifies an important practical aspect of drone operations by modeling energy usage with a fixed maximum flight time, which caps travel distance without accounting for payload-dependent energy consumption. In practice, a drone carrying a heavier payload depletes its battery faster, and ignoring this in routing decisions may lead to infeasible or unsafe operations.

Table 2 provides an overview of hybrid last-mile delivery problems involving trucks, drones, and robots with focus on drone and robot modeling. The comparison summarizes key modeling characteristics, including the number of delivery resources, the number of customer visits allowed per drone or robot trip, and energy endurance modeling. Notably, the table reveals that payload-aware energy modeling has been investigated primarily for drones, although many studies still rely on simpler travel-distance or flight-time limits. Our study extends existing truck-robot-drone models by integrating multiple sidewalk robots within a synchronized truck-drone system for doorstep delivery, while accounting for drone energy consumption and multiple visits per trip. All sidekick vehicles are carried by the truck, which creates a fully self-contained system that eliminates the need for intermediate hubs and ensures direct customer service. Despite certain operational similarities between drones and robots, treating robots as a type of heterogeneous drone is not straightforward. Existing heterogeneous drone models typically involve vehicles with similar general characteristics that differ only in certain parameters such as payload capacity, battery capacity, and flight speed. In addition, the drone in our model requires tracking payload-dependent energy consumption dynamically across multiple customer visits per trip, while each robot is subject to a simpler distance-based endurance constraint with a single delivery per trip. Unifying these under a single framework would impose the complexity of drone energy modeling on robots, which do not require it. Beyond modeling, the substantial differences between the two modes also motivate dedicated solution strategies for each vehicle type.

3. Proposed truck-robot-drone delivery model

In this section, we provide a description of the proposed hybrid delivery system, called the Multi-Visit Flying Sidekick Traveling Salesman Problem with Multiple Robots (MV-FSTSP-MR) and present the mathematical programming formulation of the problem.

3.1. Problem description

Consider a directed graph $G = (V, E)$, where $V = \{0, 1, \dots, n\}$ and $E = \{(i, j) \mid i, j \in V, i \neq j\}$. The set V represents n customers and a central depot indexed by $\{0\}$. The set of all customers is denoted as $C = \{1, \dots, n\}$. A truck, with a single drone and R_{robot} robots onboard, initiates delivery operations from a central depot. At the beginning of the delivery shift, the drone and robots are released either from the central depot or kept on board the truck for release at a different location later. They can then be collected at various customer nodes along the truck route for battery swaps and reloading for another delivery trip. The truck can carry the parcels for all customers, while the drone can carry a few packages per trip, corresponding to the number of compartments it has. Each robot, by contrast, is limited to delivering a single package per trip. This assumption is based on realistic prototypes such as the Wingcopter 198 drone (Aria, 2021), which has three delivery compartments, and the Starship robot, which has a single-package compartment (Mercedes-Benz Group, 2017; Yu and Puchinger, 2024).

Fig. 2 shows the spatial-temporal coordination among the truck, robots, and drone. In this example, the truck, carrying a drone and multiple robots (e.g., eight), is deployed from the central depot at time zero and visits its first customer at time point t_1 . After

Table 2
Comparison of hybrid last-mile delivery problems.

Paper	D	R	D-V	R-V	D-E	R-E
Truck-Drone Models						
Murray and Chu (2015)	1	–	1	–	Max travel time	–
Agatz et al. (2018)	1	–	1	–	–	–
Ha et al. (2018)	1	–	1	–	Max travel time	–
Jeong et al. (2019)	m	–	1	–	Energy function	–
Karak and Abdelghany (2019)	m	–	m	–	Max flight range	–
Wang and Sheu (2019)	m	–	m	–	Max travel time	–
Wang et al. (2019)	m	–	m	–	Energy function	–
Cheng et al. (2020)	m	–	m	–	Energy function	–
Kitjacharoenchai et al. (2020)	m	–	m	–	Max travel time	–
Poikonen and Golden (2020)	m	–	m	–	Energy function	–
Luo et al. (2021)	m	–	m	–	Energy function	–
Masmoudi et al. (2022)	m	–	m	–	Energy function	–
Wu et al. (2022)	m	–	m	–	Energy function	–
Wen and Wu (2022)	m	–	m	–	Energy function	–
Gao et al. (2023)	m	–	m	–	Max travel time	–
Luo et al. (2024)	m	–	m	–	Energy function	–
Hokama et al. (2025)	1	–	1	–	–	–
Truck-Robot Models						
Boysen et al. (2018)	–	m	–	1	–	–
Simoni et al. (2020)	–	1	–	m	–	–
Yu et al. (2020)	–	m	–	m	–	Max travel time
Bakach et al. (2021)	–	m	–	1	–	Max travel time
Heimfarth et al. (2022)	–	m	–	1	–	–
Ostermeier et al. (2022)	–	m	–	1	–	–
Liu et al. (2022)	–	m	–	m	–	Max travel distance
Ostermeier et al. (2023)	–	m	–	1	–	–
Yu et al. (2024)	–	–	–	m	–	Distance-based
Truck-Robot-Drone Models						
Moeini and Salewski (2019)	m	m	1	1	Max travel time	Max travel time
Schermer et al. (2020)	1	m	1	1	Max travel time	Max travel time
Morim et al. (2024)	m	m	1	1	Max travel time	Max travel time
Mokhtari-Moghadam et al. (2025)	m	m	1	m	Max travel time	Max travel time
Our work	1	m	m	1	Energy function	Max travel time
D = Drone, R = Robot, m = multiple, D-V = Drone number of visits per trip, R-V = Robot number of visits per trip, D-E = Drone energy consumption, R-E = Robot energy consumption						

serving the first customer, some onboard vehicles are dispatched to their assigned customers, and the truck proceeds to its next stop. At time point t_2 , the truck visits and serves its second customer with no drone or robot interactions. At time point t_m , the truck visits its m th customer while retrieving all or part of the previously released vehicles. At this location, one of the following scenarios may occur: (i) all planned side vehicles have already arrived and are waiting. Robots wait without additional energy consumption, whereas the drone hovers and consumes energy until collected; or (ii) at least one side vehicle has not yet arrived, causing a delay for the truck. Once all vehicles arrive and the truck customer is served, the entire fleet departs simultaneously for their next destinations. When leaving the m th customer, the drone and robots may remain onboard or be re-dispatched. This collaboration continues until all customers are served and all vehicles return to the depot. The objective is to minimize the makespan ($C_{\max}$), defined as the time at which the last vehicle returns to the central depot, subject to operational constraints.

The remaining assumptions of this model are as follows:

- There are enough spare batteries carried by the truck for the drone and the robots to swap the depleting batteries when needed.
- Each customer node can be visited by the truck at most once.
- Both the drone and the robots are limited to a specific total payload weight regardless of the number of packages.
- The drone is further constrained by a battery endurance of θ (kWh), while the robots have a maximum travel time limit of γ (hours).
- The truck, drone, and robots travel at a constant velocity and therefore each edge (i, j) is associated with three different travel times for each mode of delivery.
- The service and setup times are factored into the travel times between nodes.

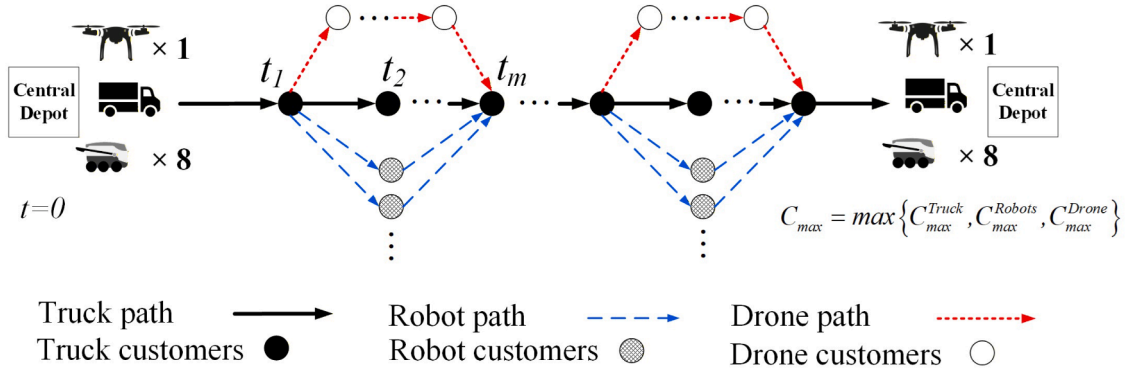


Fig. 2. Spatial-temporal coordination among truck, robots, and drone.

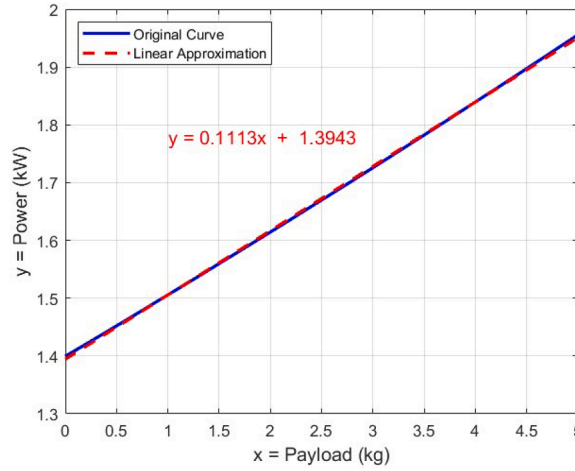


Fig. 3. Linear approximation fitted to the power consumption model for a Wingcopter 198 based on the drone's payload.

3.2. Drone energy modeling

We utilize Eq. (1), suggested by Dorling et al. (2016), to model the drone's energy usage during level flight and hovering.

$$P(q) = (W + b + q)^{\frac{3}{2}} \sqrt{\frac{g^3}{2\rho\zeta h}}, \quad (1)$$

where $P(q)$ represents the power (W) required for carrying a payload of q (kg) between two nodes. W is the weight of the frame of the drone (kg), b is the battery weight (kg), g is the gravitational acceleration (m/s^2), ρ is the air density (kg/m^3), ζ is the rotor disc area (m^2), and h is the number of rotors. Following Dorling et al. (2016), Eq. (1) can be replaced by the linear approximation in Eq. (2) using numerical quantification and linear regression.

$$P(q) = \alpha \times q + \beta, \quad (2)$$

where α and β are the slope (W/kg) and intercept (W) of the linear approximation. In our experiments, we use a Wingcopter 198 drone (Wingcopter, 2025), which is capable of carrying up to three packages per trip. With dimensions of $198 \times 152 \times 65$ cm ($W \times L \times H$), this drone can be accommodated on the roof of a full-size delivery truck, such as a Mercedes-Benz Sprinter or Ford Transit. For this 8-rotor drone with a payload capacity of 4.7 kg, a total drone weight of 20 kg (frame plus battery), a total energy capacity of $\theta = 5860.8$ kJ (two Li-ion 814 Wh batteries), $\rho = 1.204$ kg m^{-3} , $\zeta = 0.2$ m^2 , and $g = 9.80665$ m s^{-2} , Fig. 3 shows the nonlinear power curve from Eq. (1) alongside the approximated linear power formula from Eq. (2) for different values of the payload q varying from 0 to 5 kg. The estimated slope and intercept of the linear approximation are $\alpha = 0.1113$ kW/kg and $\beta = 1.3943$ kW, representing the power-to-weight proportional coefficient and the base power consumption, respectively.

3.3. Mathematical formulation

In this section, we develop an MILP model for the proposed MV-FSTSP-MR. We adopt a trip-based indexing approach to distinguish between different trips made by the drone or the robots (Luo et al., 2021). Table 3 summarizes all the notations used in this

Table 3

Notations.

Indices	Definition
$i, j, l \in V$	Index of nodes
$v \in \Omega$	Index of autonomous vehicle types
$k \in K_v$	Index of autonomous vehicle trips
Sets	Definition
$C = \{1, \dots, n\}$	Set of customer nodes
$K_v = \{1, \dots, k_{\max}^v\}$	Set of trip indices for each autonomous vehicle type v
$\Omega = \{drone, robot\}$	Set of autonomous vehicle types
$V = \{0, 1, \dots, n\}$	Set of all the nodes including the depot and customer nodes
$E = \{(i, j) i, j \in V, i \neq j\}$	Set of all the edges
Parameters	Definition
t_{ij}^G	Traveling time of $edge(i, j) \in E$ by the truck
t_{ijv}^U	Traveling time of $edge(i, j) \in E$ by an autonomous vehicle type v
w_i	Weight of the package at customer node i (kg)
θ	Energy capacity of drone (kWh)
α	Slope of the approximate line fitting the drone's power formula (W/kg)
β	Intercept of the approximate line fitting the drone's power formula (W)
γ	Maximum traveling time of robots (h)
Q_v	Maximum payload weight of an autonomous vehicle type v (kg)
R_v	Number of available autonomous vehicles of type v on board the truck at the central depot
max_visits	Maximum number of visits the drone can make on each trip
Decision Variables	Definition
z_i^G	1, if customer at node i is served by the truck; 0 otherwise
z_{ikv}^U	1, if customer at node i is served by the k th trip of the autonomous vehicle type v ; 0 otherwise
y_{ij}	1, if the truck travels $edge(i, j) \in E$; 0 otherwise
x_{ijkv}	1, if $edge(i, j) \in E$ is traveled by the k th trip of the autonomous vehicle type v ; 0 otherwise
h_{ikv}^L	1, if node i is used as a launching node for the k th trip of the autonomous vehicle type v
h_{ikv}^R	1, if node i is used as a retrieving node for the k th trip of the autonomous vehicle type v
r_{iv}	Number of the autonomous vehicles type v on board the truck when leaving node i
u_{iv}^U	Total payload of an autonomous vehicle type v when leaving node i
p_{iv}^U	Remaining energy of an autonomous vehicle type v when leaving node i
a_i^G	Arrival time of the truck at node i
d_i^G	Departure time of the truck from node i
a_{iv}^U	Arrival time of the autonomous vehicle of type v at node i
d_{iv}^U	Departure time of the autonomous vehicle of type v from node i

formulation. In our notations, we use the term “autonomous vehicles” to refer collectively to the drone and robots, regardless of their type. We also use the superscript “ G ” for the truck, and “ U ” for the drone and the robots. The proposed model is presented next.

Objective function

The objective function (3), restricted by Constraints (4)–(6), minimizes the delivery completion time of the model, which is defined as the latest arrival time of the vehicles to the central depot indexed by $\{0\}$.

$$\min C_{\max} \quad (3)$$

$$C_{\max} \geq a_{0, \text{drone}}^U \quad (4)$$

$$C_{\max} \geq a_{0, \text{robot}}^U \quad (5)$$

$$C_{\max} \geq a_0^G \quad (6)$$

Routing constraints

The following constraints handle the general requirements for customer assignments and truck routing, as well as the routing of autonomous delivery vehicles, including the drone and robots.

$$\sum_{v \in \Omega} \sum_{k \in K_v} z_{ikv}^U + z_i^G = 1, \quad \forall i \in C \quad (7)$$

$$z_i^G = \sum_{j \in V} y_{ji}, \quad \forall i \in C \quad (8)$$

$$\sum_{j \in V} y_{ij} = \sum_{j \in V} y_{ji} \leq 1, \quad \forall i \in V \quad (9)$$

$$\sum_{i \in C} y_{0i} = \sum_{i \in C} y_{i0} = 1 \quad (10)$$

$$\sum_{j \in V} x_{jikv} \geq z_{ikv}^U, \quad \forall i \in C, \forall k \in K_v, \forall v \in \Omega \quad (11)$$

$$z_{ikv}^U + z_{jkv}^U \geq 1 - M \times (1 - x_{ijkv}), \quad \forall (i, j) \in E, \forall k \in K_v, \forall v \in \Omega \quad (12)$$

$$\sum_{v \in \Omega} \sum_{k \in K_v} x_{ijkv} \leq 1, \quad \forall (i, j) \in E \quad (13)$$

$$\sum_{j \in V} x_{ijkv} \leq 1, \quad \forall i \in V, \forall k \in K_v, \forall v \in \Omega \quad (14)$$

$$\sum_{j \in V} x_{jikv} \leq 1, \quad \forall i \in V, \forall k \in K_v, \forall v \in \Omega \quad (15)$$

Constraint (7) ensures that each customer is served by only one type of vehicle and, if not served by the truck, is assigned to a single trip of an autonomous vehicle (drone or robot). Constraint (8) guarantees that if customer i is assigned to the truck, the truck must visit that customer via an edge ending at that specific node. Constraint (9) refers to the inbound and outbound routing requirements of the truck, while Constraint (10) ensures that the truck leaves the central depot and returns to it after the delivery process. Constraint (11) ensures that if a customer is assigned to a particular trip of an autonomous vehicle, there must be a corresponding travel from any node to that customer node using the same vehicle type and trip. Constraint (12) enforces that if there is travel between two customer nodes by a specific trip of an autonomous vehicle, at least one of these nodes must be served by that trip; in other words, both nodes cannot simultaneously act as launch or retrieval nodes when $x_{ijkv} = 1$. Constraint (13) ensures that each edge (i, j) is traversed by only one trip of a specific autonomous vehicle (either the drone or a robot, but not both). Constraints (14) and (15) together enforce that each trip of an autonomous vehicle, if used, must have exactly one starting node and one ending node.

Linking launch/retrieval nodes constraints

The following constraints serve as moderators to link the decision variables for the launch and retrieval nodes (h_{ikv}^L and h_{ikv}^R) with other variables in the model. These include customer assignments to the drone and robots (z_{ikv}^U), the truck routing variables (y_{ij}), and the drone and robot routing variables (x_{ijkv}).

$$\sum_{v \in \Omega} \sum_{k \in K_v} h_{ikv}^L \leq M \times \sum_{j \in V} y_{ji}, \quad \forall i \in V \quad (16)$$

$$\sum_{v \in \Omega} \sum_{k \in K_v} h_{ikv}^R \leq M \times \sum_{j \in V} y_{ij}, \quad \forall i \in V \quad (17)$$

$$h_{ikv}^R + \sum_{j \in V} x_{ijkv} = h_{ikv}^L + \sum_{j \in V} x_{jikv}, \quad \forall i \in V, \forall k \in K_v, v \in \Omega \quad (18)$$

$$h_{ikv}^L \geq 1 - M \times \left(2 - x_{ijkv} - z_{jkv}^U + z_{ikv}^U \right), \quad \forall i \in V, j \in C, i \neq j, k \in K_v, \forall v \in \Omega \quad (19)$$

$$h_{jkv}^R \geq 1 - M \times \left(2 - x_{ijkv} - z_{ikv}^U + z_{jkv}^U \right), \quad \forall i \in C, j \in V, i \neq j, \forall k \in K_v, \forall v \in \Omega \quad (20)$$

$$\sum_{i \in V} h_{ikv}^L = \sum_{i \in V} h_{ikv}^R \leq 1, \quad \forall k \in K_v, \forall v \in \Omega \quad (21)$$

Constraints (16) and (17) together ensure that if a particular node is visited by the truck, it can serve as a potential launch or retrieval node for the drone or robots. Constraint (18) ensures that if a node is designated as a retrieval point, there must be an inbound travel edge ending at that node for a specific trip of an autonomous vehicle. Likewise, if a node is designated as a launching point, there must be a corresponding outbound travel edge starting from that node to another node. Constraints (19) and (20) establish the conditions for a feasible launch and retrieval node for the autonomous vehicles, respectively. Specifically, a node i is considered as a launch point for a particular trip of the drone or a robot only if there is an outbound travel from node i to some customer j , where customer j is served by that trip, and node i itself is not served by the same trip. Constraint (21) makes sure that if a trip of an autonomous vehicle is launched at any node, it must eventually be retrieved.

Robot/drone balance constraints

The following constraints ensure that the number of autonomous vehicles of each type remains balanced throughout the truck's trip.

$$r_{0,v} + \sum_{j \in C} \sum_{k \in K_v} x_{0jkv} = R_v, \quad \forall v \in \Omega \quad (22)$$

$$r_{iv} + \sum_{\ell \in C} \sum_{k \in K_v} x_{\ell jkv} \geq r_{jv} + \sum_{\ell \in C} \sum_{k \in K_v} x_{j\ell kv} - M \times (1 - y_{ij}), \quad \forall i \in V, \forall j \in V / \{0\}, i \neq j, \forall v \in \Omega \quad (23)$$

$$r_{iv} + \sum_{j \in C} \sum_{k \in K_v} x_{j0kv} \geq R_v - M \times (1 - y_{i0}), \quad \forall i \in V, \forall v \in \Omega \quad (24)$$

Constraint (22) ensures that the total number of autonomous vehicles released from the central depot, combined with those remaining on board the truck, equals the total number of available vehicles. Similarly, Constraint (23) ensures that the number of autonomous vehicles of each type (drone or robot) remains balanced at all intermediate nodes visited by the truck. Constraint (24) ensures that by the end of the truck's tour, all autonomous vehicles of each type are gathered at the central depot, either on board the truck or retrieved directly at the depot.

Drone's payload and energy tracking constraints

The following constraints ensure that the drone's payload and its energy consumption is properly tracked throughout the delivery operations.

$$\sum_{i \in C} w_i \times z_{ikv}^U \leq Q_v, \quad \forall k \in K_v, \forall v \in \Omega \quad (25)$$

$$\sum_{i \in C} z_{ikv}^U \leq 1, \quad \forall k \in K_v, v = \text{robot} \quad (26)$$

$$\sum_{i \in C} z_{ikv}^U \leq \text{max_visits}, \quad \forall k \in K_v, v = \text{drone} \quad (27)$$

$$\sum_{i \in V} \sum_{j \in V} t_{ijv}^U \times x_{ijkv} \leq \gamma, \quad \forall k \in K_v, v = \text{robot} \quad (28)$$

$$w_{jv}^U \geq \sum_{\ell \in C} \left(w_\ell \times z_{\ell kv}^U \right) - w_j - M \times \left(3 - x_{ijkv} - z_{jkv}^U - h_{ikv}^L \right), \quad \forall i \in V, \forall j \in C, i \neq j, \forall k \in K_v, v = \text{drone} \quad (29)$$

$$w_{jv}^U \geq w_{iv}^U - w_j - M \times \left(3 - \sum_{k \in K_v} x_{ijkv} - \sum_{k \in K_v} z_{ikv}^U - \sum_{k \in K_v} z_{jkv}^U \right), \quad \forall i, j \in C, i \neq j, v = \text{drone} \quad (30)$$

$$p_{iv}^U \geq \theta - M \times \sum_{k \in K_v} z_{ikv}^U, \quad \forall i \in V, v = \text{drone} \quad (31)$$

$$p_{jv}^U \geq p_{iv}^U - t_{ijv}^U \times \left(\alpha \times (w_{jv}^U + w_j) + \beta \right) - M \times \left(3 - x_{ijkv} - h_{ikv}^L - z_{jkv}^U \right), \quad \forall (i, j) \in E, \forall k \in K_v, v = \text{drone} \quad (32)$$

$$p_{jv}^U \geq p_{iv}^U - t_{ijv}^U \times \left(\alpha \times w_{iv}^U + \beta \right) - M \times \left(3 - x_{ijkv} - z_{ikv}^U - z_{jkv}^U \right), \quad \forall (i, j) \in E, \forall k \in K_v, v = \text{drone} \quad (33)$$

$$p_{iv}^U \geq \beta \times t_{ijv}^U + \max \left\{ \beta \times \left(a_j^G - d_{iv}^U - t_{ijv}^U \right), 0 \right\} - M \times \left(2 - x_{ijkv} - h_{jkv}^R \right), \quad \forall i \in C, j \in V, i \neq j, \forall k \in K_v, v = \text{drone} \quad (34)$$

$$p_{jv}^U \geq p_{iv}^U + t_{ijv}^U \times \left(\alpha \times (w_{jv}^U + w_j) + \beta \right) - M \times \left(2 - x_{ijkv} - z_{jkv}^U \right), \quad \forall i \in V, \forall j \in C, \forall k \in K_v, v = \text{drone} \quad (35)$$

Constraint (25) enforces the payload capacity limits of the drone and robots. Constraints (26) and (27) ensure that a maximum of one customer is assigned to each robot trip and up to "max_visits" customers to each drone trip, respectively. Constraint (28) guarantees that the maximum travel endurance of the robots is respected. Since the drone's travel endurance depends on its payload, we must track the payload for each drone trip and calculate the corresponding energy consumption. Therefore, an additional set of constraints is introduced to enforce these requirements. In particular, Constraints (29) and (30) track the payload of the drone when departing from a launch node and a non-launch node, respectively. Constraint (31) ensures that the initial energy of the drone before departing any potential launch node is equal to its energy endurance. Constraints (32)–(34) track the remaining energy of the drone during a particular trip k in three stages: (i) when the drone is launched to its first assigned customer, (ii) when it flies between two consecutive drone-served customers, and (iii) when the drone travels from its last customer to the retrieval node. Additionally, Constraint (34) accounts for the drone hovering energy consumption while waiting to be collected by the truck upon its arrival. Constraint (35) links Constraints (31)–(34), ensuring that the decrement in the drone's energy levels is properly tracked throughout the different stages of its trip.

Time constraints

The following constraints track the arrival and departure times of all types of vehicles in the model.

$$a_j^G \geq d_i^G + t_{ij}^G - M \times (1 - y_{ij}), \quad \forall (i, j) \in E \quad (36)$$

$$d_i^G \geq a_i^G - M \times (1 - z_i^G), \quad \forall i \in C \quad (37)$$

$$d_i^G \geq a_{iv}^U - M \times (1 - z_i^G), \quad \forall i \in C, \forall v \in \Omega \quad (38)$$

$$a_{jv}^U \geq d_{iv}^U + t_{ijv}^U - M \times \left(1 - \sum_{k \in K_v} x_{ijkv} \right), \quad \forall (i, j) \in E, \forall v \in \Omega \quad (39)$$

$$d_{iv}^U \geq a_{iv}^U - M \times \left(1 - \sum_{k \in K_v} z_{ikv}^U \right), \quad \forall i \in C, \forall v \in \Omega \quad (40)$$

$$d_{iv}^U \geq d_i^G - M \times (1 - h_{ikv}^L), \quad \forall i \in V, \forall k \in K_v, \forall v \in \Omega \quad (41)$$

$$d_{iv}^U \leq d_i^G + M \times (1 - h_{ikv}^L), \quad \forall i \in V, \forall k \in K_v, \forall v \in \Omega \quad (42)$$

Constraint (36) ensures that the arrival time of the truck at node j equals the departure time of the truck from node i plus the travel time of the truck over edge (i, j) . Constraint (37) ensures that the departure time of the truck from a particular node is greater than its arrival time at that node. Constraint (38) ensures that the truck waits for any potential arriving autonomous vehicles (drone or robots) before departing from a particular node. Constraints (39) and (40) perform the same functions as Constraints (36) and (37), but for the autonomous vehicles. Constraints (41) and (42) together guarantee that all vehicles depart from the same node instantaneously.

Decision variables

Constraints (43)–(48) specify the types of the decision variables.

$$z_i^G \in \{0, 1\}, \quad \forall i \in C \quad (43)$$

$$x_{ijkv} \in \{0, 1\}, \quad \forall (i, j) \in E, \forall k \in K_v, \forall v \in \Omega \quad (44)$$

$$y_{ij} \in \{0, 1\}, \quad \forall (i, j) \in E \quad (45)$$

$$z_{ikv}^U, h_{ikv}^L, h_{ikv}^R \in \{0, 1\}, \quad \forall i \in V, \forall k \in K_v, \forall v \in \Omega \quad (46)$$

$$a_i^G, d_i^G \geq 0, \quad \forall i \in V \quad (47)$$

$$u_{iv}^U, p_{iv}^U, a_{iv}^U, d_{iv}^U \geq 0, r_{iv} \in \mathbb{Z}, \quad \forall i \in V, \forall v \in \Omega \quad (48)$$

Finally, to ensure that the solution space is not restricted, the parameter $k_{\max}^v$ should not be chosen too small. On the other hand, choosing a large value for $k_{\max}^v$ increases the model size and computational burden. To address this trade-off systematically, we utilize Eq. (49) to determine a tight upper bound on the number of trips for each type of autonomous vehicle (drone or robot). The proof is presented in Appendix A.

$$k_{\max}^v = n - \left\lceil \frac{n}{R_v + 1} \right\rceil. \quad (49)$$

4. Solution methodology

In this section, we present the details of the proposed solution methodology for efficiently solving the MV-FSTSP-MR problem. We first present the main heuristic, which includes three algorithms, and then present an iterative framework that can be utilized to find improved solutions.

4.1. Proposed heuristic solution

The proposed heuristic belongs to a class of methods named route-first, split-second heuristics, introduced by pioneering studies in FSTSP (Agatz et al., 2018; Ha et al., 2018; Hokama et al., 2025). One of the state-of-the-art heuristics in this category is the “Split Lazy Algorithm” (SLA) proposed by Hokama et al. (2025). The SLA was developed for the FSTSP with a single-visit drone and without operational constraints. In this paper, we propose a variation of SLA capable of handling a multi-visit drone, multiple robots, and operational constraints. The proposed heuristic is composed of three newly developed algorithms, represented by Algorithms 1–3, which are coordinated as shown in the flowchart in Fig. 4. Algorithm 1 is used for the design of the truck-drone delivery route. Algorithm 2, which is embedded in Algorithm 1, is used to design efficient drone paths given a list of drone-served customers and a specific pair of launch and retrieval nodes. Finally, Algorithm 3 is implemented to iteratively add robots to the previously created delivery routes, as long as there are robots on board that can improve the solution.

The heuristic solution starts with an initial TSP tour to serve all customers by the truck. In this step, the truck travel times (t_{ij}^G) are used to determine the most efficient truck tour. Next, Algorithm 1 is used to design the truck-drone delivery routes, while ensuring that the drone’s operational constraints are satisfied through efficient routing via Algorithm 2. Then, the truck tour is updated, and the drone-served customers are removed from it. We also ensure that the drone launch and retrieval nodes are never chosen for robot delivery in the subsequent steps. The generated truck-drone routes are then used as a starting point for adding robots using Algorithm 3 in an iterative manner to serve the remaining customers, as long as robots are available on board that can improve the solution. Each time a robot is added to the previously built routes, the truck tour, potential customers for robot delivery, and truck delays at retrieval nodes are updated accordingly.

Algorithm 1: truck-drone routing

Algorithm 1 is developed to handle multi-visit drone delivery with the drone’s operational constraints. Considering an initial TSP tour τ , a list of drone-served customers is specified starting from the j th customer, with potential retrieval nodes beginning from the $(j + m + 1)$ th customer in τ . This allows all the customer indices $j, j + 1, \dots, j + m$ to be considered as a set of customers to be served by the drone through multiple visits. Here, the parameter m denotes the allowed number of visits, which is searched from 1 to $\max_visits$ in an external loop to find the most efficient number of visits for the drone, considering the locations of the customers, the locations of the launch and retrieval nodes, and the weight of the payload. Algorithm 1 iteratively constructs a graph G' , which is used to build the final truck-drone routes by finding the shortest path over its edges. Each edge (i, j) in this graph, also called an “operation,” represents a truck path and a drone path, each serving a dedicated set of customers. Fig. 5 illustrates an example of a joint truck-drone delivery with two operations, starting at node 0 (depot) and terminating at node 0. Operation 1 starts at node 0 and ends at node 4, with a drone path serving customers 1 and 3, and a truck path serving customer 2. Operation 2 continues the delivery process by starting at node 4 and ending at node 0, with a drone path serving customers 5 and 7, and a truck path serving customer 6. In this toy example, the shortest path over graph G' includes two edges (operations) (0, 4) and (4, 0). However, in each iteration of Algorithm 1, graph G' is updated whenever a new operation is discovered or an existing operation (i, j) with a different truck-drone allocation and lower completion time is found.

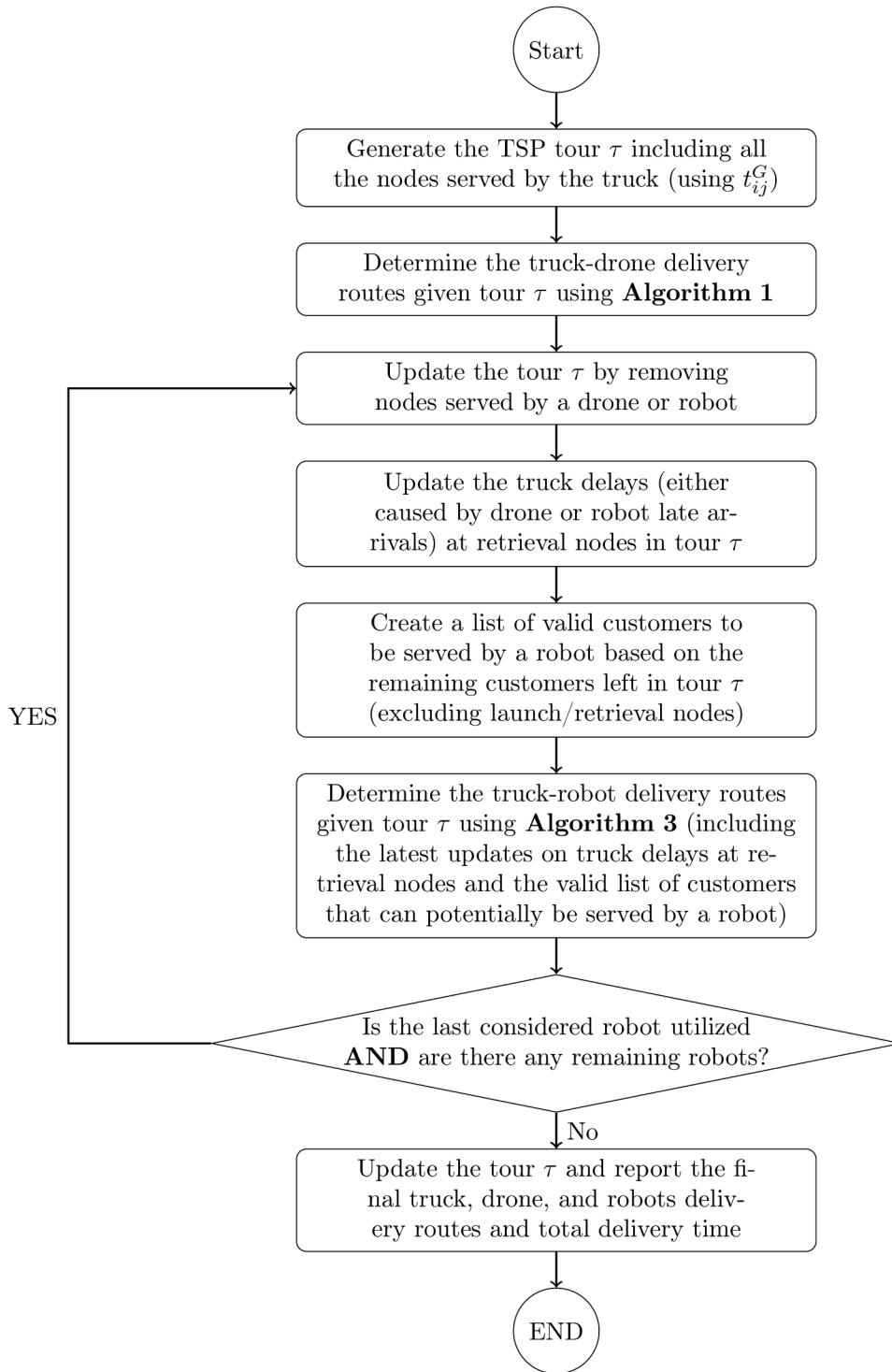


Fig. 4. Flowchart of the truck-drone-robot routing procedure.

Algorithm 1: Heuristic for τ -MV-FSTSP with drone constraints.**1. Initialization**

Construct initial graph G' by adding edges (i, j) from tour τ with traveling times t_{ij}^G

2. Graph construction

for $m = 1$ **to** $\max_visits$ **do**

for each node $j \in \tau \setminus depot(0)$ **do**

$k_{\max} \leftarrow depot$

for $i = j - 1$ **down to** 0 **do**

$k \leftarrow j + m + 1$

 Compute truck time for operation (i, k) along τ

 Compute drone time for operation (i, k) serving $\{j, \dots, j + m\}$ along τ using [Algorithm 2](#)

$time(i, k) \leftarrow \max\{\text{truck time, drone time}\}$

if operation (i, k) is missing or improves G' **then**

 Add or update operation (i, k)

end

if drone faster then

 break

else

for $k : j + m + 1 < k \leq k_{\max}$ **do**

 Compute truck time for operation (i, k) along τ

 Compute drone time for operation (i, k) serving $\{j, \dots, j + m\}$ along τ using [Algorithm 2](#)

$time(i, k) \leftarrow \max\{\text{truck time, drone time}\}$

if operation (i, k) is missing or improves G' **then**

 Add or update operation (i, k)

end

if drone faster then

$k_{\max} \leftarrow k - 1$; break

end

end

end

end

end

end

3. Solution extraction

Solve shortest path on G' (Dijkstra)

Recover truck and drone routes

Output: Routes and total completion time

Algorithm 2: drone routing

[Algorithm 2](#) determines the best possible sequence of customers to be visited by the drone. This sequence avoids creating unnecessary delays for the truck and ensures that the least-energy drone path is chosen when there is no delay associated with the path. Furthermore, this algorithm ensures that all operational constraints regarding drone energy consumption and payload capacity limitations are met. This algorithm takes the launching node, the retrieval node, the truck travel time for the proposed operation, and the list of customers allocated to the drone as inputs. To find the best drone path, all permutations of the drone customers are generated. Each candidate path is then evaluated in terms of both flight time and energy usage. If a path is feasible in terms of energy, it is further evaluated based on the delay it causes for the truck at the retrieval node. Among all candidate drone paths that result in no delay for the truck, the one with the least energy usage is selected. If all paths result in delays for the truck, the one with the least delay is chosen, regardless of energy usage. This approach ensures that the drone routing aligns with the main objective function of the problem, namely minimizing the makespan, while also promoting energy efficiency when it does not affect the truck waiting time.

Algorithm 3: robot allocations

[Algorithm 3](#) is developed to include the operational constraints related to the robots and to assign multiple robots to the pre-constructed truck-drone routes from [Algorithm 1](#) in an iterative manner. This algorithm has a structure similar to that of [Algorithm 1](#), except that it serves only one customer and does not include the energy efficiency consideration. This will reduce the complexity of the algorithm significantly by eliminating the need for an outer loop to determine the efficient number of customers for each candidate

Algorithm 2: Drone path selection with feasibility checks.**Initialization**

Given launch node l , retrieval node r , truck time in operation (l, r) , and drone customers C

Generate $\mathcal{A} = \{(l, \sigma, r) : \sigma \in \mathcal{P}(C)\}$, where $\mathcal{P}(C)$ denotes the set of all permutations of C

$S \leftarrow \emptyset$

for each path $\pi \in \mathcal{A}$ **do**

if $\sum_{j \in \pi \setminus \{l, r\}} w_j > Q_{\text{drone}}$ **then**
 | continue

end

 Compute drone time in path π , denoted as T_π

 Compute drone energy usage in path π , denoted as E_π

if $E_\pi > \theta$ **then**

 | continue

end

 Compute truck delay $\delta_\pi \leftarrow \max\{0, T_\pi - \text{truck time}\}$

$S \leftarrow S \cup \{(\delta_\pi, T_\pi, E_\pi, \pi)\}$

end

if $S \neq \emptyset$ **then**

if $\exists \pi : \delta_\pi = 0$ **then**

 | Select π^* with minimum E_π

else

 | Select π^* with minimum δ_π

end

 Set best drone time $\leftarrow T_{\pi^*}$, best drone path $\leftarrow \pi^*$

else

 Set best drone time $\leftarrow \infty$, best drone path $\leftarrow \emptyset$

end

Output: Best drone time and path

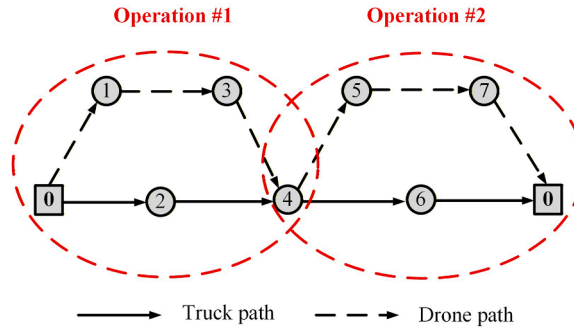


Fig. 5. Illustration of two operations in a synchronous truck-drone delivery.

drone path, and a dedicated sub-procedure (Algorithm 2) to identify the best drone path between any two rendezvous nodes by jointly considering energy consumption and travel time. To ensure that the multi-robot feature in Algorithm 3 functions correctly, the truck's delays at retrieval nodes are incorporated into the travel times of the corresponding edges. The proposed heuristic solution is polynomially bounded, as proven in Theorem 1.

Theorem 1. *The proposed heuristic is polynomially bounded with worst-case complexity of $O(n^3)$ (proof in Appendix B).*

In summary, the proposed heuristic follows a two-stage design: the drone is dispatched first due to its speed and multi-compartment advantage, while robots are assigned only when they contribute to reducing the overall completion time. The heuristic leverages the lazy drone property from Hokama et al. (2025) to avoid full enumeration of truck-drone operations. Algorithm 2 selects the best feasible drone path by jointly considering energy consumption, payload weight, and travel time, and Algorithm 3 assigns robots while enforcing constraints such as maximum travel distance and single delivery per trip.

Algorithm 3: Heuristic for τ -TSP-R with robot constraints.**1. Initialization**

Construct initial graph G' from tour τ with traveling times t_{ij}^G plus any truck delays

Let $\mathcal{J} \subseteq \tau \setminus \{0\}$ be robot-eligible customers

2. Graph construction

for each $j \in \mathcal{J}$ **do**

$k_{\max} \leftarrow \text{depot}$

for $i = j - 1$ **down to** 0 **do**

$k \leftarrow j + 1$

 Compute robot time for path (i, j, k)

if feasible then

 Compute truck time for operation (i, k) along τ

$\text{time}(i, k) \leftarrow \max\{\text{robot time, truck time}\}$

if operation (i, k) is missing or improves G' **then**

 Add or update operation (i, k)

end

if robot faster then

 break

else

for $k : j + 1 < k \leq k_{\max}$ **do**

 Compute robot time for path (i, j, k)

if feasible then

 Compute truck time for operation (i, k) along τ

$\text{time}(i, k) \leftarrow \max\{\text{robot time, truck time}\}$

if operation (i, k) is missing or improves G' **then**

 Add or update operation (i, k)

end

if robot faster then

$k_{\max} \leftarrow k - 1$; break

end

end

end

end

end

end

end

3. Solution extraction

Solve shortest path on G' (Dijkstra)

Recover truck and robot routes

Output: Routes and total completion time

4.2. Improvement framework

In this section, we present a complementary stage to the proposed heuristic, aimed at improving the quality of the final delivery routes by using variations of the initial TSP tour. Specifically, the proposed heuristic is run more than once, each time slightly tweaking the initial TSP tour, to see if any alternative tour might yield better solutions.

Algorithm 4 provides the details on the proposed improvement framework. We use a tabu search (Glover, 1990) to search over different tours that are given to the proposed heuristic as the initial tour τ . Originally, a TSP tour for truck-only delivery was specified to start the heuristic; however, by searching the neighbors of the TSP tour and swapping the sequence of customers in the original truck-only tour, we aim to obtain better final truck, robot, and drone routes. The improvement framework increases the complexity of the solution process in two aspects: first, it analyzes multiple neighbors of the current solution, each of which needs to be assessed by the heuristic; and second, this neighborhood search is repeated through multiple iterations. Generally, the use of the improvement framework is recommended when the heuristic can solve the problem instance within a reasonably short time (a few seconds to minutes), allowing multiple iterations to be run efficiently. It is also suitable when parallel computing resources are available to evaluate several neighboring solutions simultaneously, or when there is no strict time limit on the improvement stage.

Algorithm 4: Improvement framework.**1. Initialization**Set hyperparameters (P , tabu tenure, max_it, time budget, np)Generate an initial TSP tour τ visiting all customers with the truckEvaluate the solution and completion time of tour τ as (S, T) using heuristicSet $(\tau^*, S^*, T^*) \leftarrow (\tau, S, T)$ **2. Tabu search****for** $i = 1$ **to** max_it **do** **if** time budget violated or no improvement for np iterations **then**

break

end Generate P neighbors τ_p from the current tour τ via random swaps Evaluate the solution and completion time of tours τ_p as (S_p, T_p) using heuristic solution Sort tours in ascending order of T_p **for** $k = 1$ **to** P **do** **if** $T_k < T^*$ **or** τ_k not tabu **then** Update $(\tau, S, T) \leftarrow (\tau_k, S_k, T_k)$ Update (τ^*, S^*, T^*) if improved

Update tabu list based on tabu tenure size

break

end **end****end****Output:** (τ^*, S^*, T^*) **Table 4**

Summary of parameters used for computational analysis.

Parameter	Value / Description
Truck speed	45 km/h
Drone speed	60 km/h
Robot speed	12 km/h
Robot maximum travel time (γ)	0.5 h
Robot / drone capacity (Q_r)	10 kg / 5 kg
Robot / drone max visits per trip	1 / 3
Number of robots / drones (R_r)	4 (small), 8 (medium/large) / 1
Drone energy capacity (θ)	5860.8 kJ
Drone power model parameters	$\alpha = 0.1113$ kW/kg, $\beta = 1.3943$ kW

5. Computational analysis

We implement our MV-FSTSP-MR model for small- to medium-sized instances on the standard Solomon dataset (Solomon, 1987). We perform two modifications to this dataset to make it more applicable to a drone-assisted last-mile delivery setting: (i) we scale down the customer demand weights by a factor of 10, bringing them into a range of 0.1 to 5 kg, to accommodate drone integration; (ii) we divide the coordinates in the dataset by 10, to transform the customer locations into a moderate service area of 100 km² (i.e., 10 km $\times$ 10 km). Travel times between nodes for the truck and robots are calculated based on the Manhattan distance. For the drone, travel times are calculated based on the Euclidean distance between nodes. For instances larger than 100 customers, which are not covered by the Solomon dataset, the instances are generated using the same logic, i.e., coordinates within a 10 km $\times$ 10 km plane and package weights in the range [0.1 kg, 5 kg]. Table 4 summarizes the parameters used in Section 5.

Computational experiments are performed on a DELL laptop with an Intel(R) Core (TM) i7-12700H CPU, 2.30 GHz, and 16 GB of RAM. The proposed MV-FSTSP-MR model is programmed using the GAMSpy library with the CPLEX 22.1 solver in a Python environment. The proposed heuristic solutions are programmed in Python. The solution time limit is set to one hour. For the MILP solver, this means that the solver terminates after one hour of computation, whereas for the heuristic's improvement framework, it means that once the elapsed time surpasses the budget, subsequent iterations are not executed. The hyperparameter settings for the improvement framework in this experiment are as follows: number of neighbors ($P = 20$), tabu tenure ($tt = 5$), maximum iterations (max_it = 100), and tolerance for no improvement ($np = 10$). These parameters were selected to be large enough to allow adequate exploration within the one hour.

Table 5
Results of small-sized problems.

Instance	CPLEX			Heuristic		Heuristic*		
	Cost (min)	Time (s)	Optimality Gap (%)	Cost (min)	Time (s)	Cost (min)	Time (s)	Relative impr. (%)
6-1	16.3	5.0	0	17.0	0.0	16.3	0.8	4.2
6-2	19.6	4.8	0	20.3	0.0	19.6	0.9	3.3
6-3	20.1	9.9	0	20.5	0.0	20.1	0.7	1.9
6-4	26.7	4.7	0	28.6	0.0	26.7	0.6	6.9
6-5	16.0	9.2	0	16.0	0.0	16.0	0.8	0.0
Avg.		6.7	0		0.0		0.7	3.3
7-1	17.1	69.1	0	17.1	0.0	17.1	1.1	0.0
7-2	20.5	68.0	0	21.8	0.0	21.4	1.2	1.9
7-3	19.5	91.1	0	20.0	0.0	19.5	0.7	2.6
7-4	20.8	115.1	0	22.1	0.0	20.8	0.8	5.5
7-5	21.9	79.8	0	24.0	0.0	21.9	1.3	9.0
Avg.		84.6	0		0.0		1.1	3.8
8-1	23.4	766.4	0	24.0	0.0	23.4	1.7	2.7
8-2	16.0	1637.3	0	17.3	0.0	16.9	3.3	2.3
8-3	25.0	537.0	0	27.1	0.0	25.2	1.2	6.9
8-4	20.5	997.0	0	21.1	0.0	20.5	1.1	2.6
8-5	18.0	910.7	0	18.3	0.0	18.0	1.7	1.8
Avg.		969.7	0		0.0		1.8	3.3
9-1	22.6	3600.0	88.1	23.1	0.0	22.6	1.5	2.3
9-2	21.1	3600.0	60.3	21.5	0.0	21.3	1.5	0.7
9-3	26.9	3600.0	38.1	30.1	0.0	26.9	3.0	10.6
9-4	24.0	3600.0	77.8	24.5	0.0	23.2	1.3	5.4
9-5	23.9	3600.0	56.0	23.9	0.0	23.9	2.8	0.0
Avg.		3600.0	64.1		0.0		2.0	3.8
12-1	42.1	3600.0	92.4	29.6	0.0	29.0	2.5	2.0
12-2	34.9	3600.0	96.2	26.4	0.0	24.3	6.7	8.2
12-3	38.9	3600.0	92.5	34.8	0.0	31.5	5.0	9.3
12-4	32.5	3600.0	95.9	29.4	0.0	28.5	2.6	3.0
12-5	41.1	3600.0	93.8	26.4	0.0	26.4	2.9	0.0
Avg.		3600.0	94.2		0.0		4.0	4.5

5.1. Small instances

In this section, we provide benchmarks with the MILP solver for small instances. To generate the evaluation dataset for the CPLEX solver, we randomly select $n = 6, 7, 8, 9$, and 12 customers from the 100 available customer nodes in the RC101 instance of the Solomon dataset. The reason for not choosing instances other than RC101 is that the Solomon dataset was originally designed for Vehicle Routing Problems with time windows, and all other instances in different groups mostly share the same customer coordinates, differing only in their time windows. Five different samples for each problem size are solved. Table 5 presents the results, including the average values for five instances with the same number of customers on a small scale. The results include the performance of the MILP solver, the proposed heuristic, its improved version, and the percentage of improvement that the improved version provides over the original heuristic. The level of improvement is calculated as $(cost(Heuristic) - cost(Heuristic^*)) / cost(Heuristic) \times 100$.

As shown in the results presented in Table 5, for those instances where the MILP solver managed to solve to optimality, the difference in delivery completion time between the optimal solution and the one identified by the heuristic is less than one minute. Regarding the improved version, it can obtain almost the same or better solutions in comparison with CPLEX. For a small problem with only $n = 12$ customers, CPLEX reaches the time limit and the proposed heuristic is able to identify better solutions in negligible time (under 50 milliseconds). The improvement framework can also provide, on average, a 3.7% improvement in comparison with the original heuristic for small-sized problems.

5.2. Medium instances

In this section, we present the performance of the proposed heuristic and its improved version for larger problems. Due to the computational intractability of the MILP solver for this class of instances, we evaluate the proposed truck-robot-drone delivery system by comparing heuristic solutions with a truck-only baseline (i.e., a TSP tour). This comparison highlights the potential time savings of the proposed approach in delivering all packages compared to using a single delivery truck. To generate the evaluation dataset for this category, we select $n = 20, 30, 40, 50, 60$, and 70 random customers from the RC101 set. Five different samples are generated for each problem size, and the average results are reported.

The results shown in Table 6 include the TSP cost, heuristic cost, heuristic cost savings (in minutes), computational time for the heuristic, improved heuristic cost, improved heuristic cost savings (in minutes), computational time for improved heuristic, and the

Table 6
Results of medium-sized problems.

Instance	Truck-only (TSP) cost (min)	Heuristic			Heuristic*			
		Cost (min)	Cost saving (min)	Time (s)	Cost (min)	Cost saving (min)	Time (s)	Relative impr. (%)
20-1	57.6	37.5	20.1	0.1	34.8	22.8	15.4	7.26
20-2	56.5	38.1	18.4	0.1	34.0	22.6	23.8	10.88
20-3	57.6	37.8	19.8	0.1	36.4	21.2	23.1	3.65
20-4	56.3	31.3	24.9	0.1	31.3	24.9	15.3	0.0
20-5	56.0	40.1	15.9	0.0	37.1	18.9	18.5	7.52
Avg.			19.8	0.1		22.1	19.2	5.86
30-1	63.2	41.4	21.8	0.2	38.5	24.7	91.0	7.01
30-2	70.4	42.5	27.9	0.3	41.0	29.5	95.4	3.69
30-3	68.5	46.9	21.7	0.3	46.3	22.2	83.3	1.24
30-4	67.2	37.8	29.4	0.2	37.6	29.6	49.9	0.45
30-5	70.7	45.0	25.7	0.2	43.9	26.7	98.3	2.31
Avg.			25.3	0.2		26.5	83.6	2.94
40-1	82.1	51.6	30.5	1.4	50.2	31.9	187.1	2.62
40-2	72.8	47.2	25.6	1.8	47.2	25.6	242.2	0.0
40-3	80.5	53.2	27.3	0.9	51.5	29.0	301.0	3.20
40-4	77.6	45.6	32.0	1.0	45.6	32.0	216.8	0.0
40-5	71.7	49.8	21.9	1.0	48.7	23.0	272.6	2.19
Avg.			27.5	1.2		28.3	243.9	1.60
50-1	77.3	54.9	22.5	1.8	53.7	23.6	640.2	2.17
50-2	87.2	58.3	28.9	3.5	56.1	31.1	1258.8	3.76
50-3	72.8	51.2	21.6	2.3	50.0	22.8	751.5	2.46
50-4	75.5	52.9	22.6	3.0	51.5	24.0	661.6	2.67
50-5	84.5	53.8	30.7	2.0	53.2	31.3	623.2	1.15
Avg.			25.3	2.5		26.6	787.1	2.44
60-1	92.0	60.4	31.6	4.7	59.8	32.2	1552.2	1.06
60-2	94.9	67.1	27.9	6.3	63.1	31.9	2898.2	5.96
60-3	90.7	64.5	26.2	4.9	63.5	27.1	1782.7	1.41
60-4	92.8	64.0	28.8	6.7	64.0	28.8	1394.0	0.0
60-5	96.3	58.7	37.6	6.7	58.7	37.6	1481.1	0.0
Avg.			30.4	5.9		31.5	1821.6	1.69
70-1	103.2	68.9	34.3	10.1	68.3	34.9	3636.3	0.86
70-2	100.8	65.5	35.3	9.9	64.2	36.6	3743.2	1.94
70-3	97.3	65.8	31.6	10.9	64.7	32.6	3664.2	1.58
70-4	95.7	63.3	32.4	11.0	62.5	33.3	2810.3	1.34
70-5	101.3	65.3	36.1	11.1	64.8	36.5	3060.2	0.72
Avg.			33.9	10.6		34.8	3382.9	1.29
RC101	119.2	76.1	43.1	54.3	75.3	43.9	3926.9	1.05

improvement gained from the improved heuristic over the original one. The cost savings are calculated based on the difference in total delivery time between serving all customers with the truck-only (TSP) tour and serving them using the proposed multi-modal delivery system. The results show that on average, the proposed multi-modal delivery system can achieve approximately half an hour in delivery time savings compared to a truck-only delivery system for this range of instances. The average computational time for the heuristic solution increases from 0.1 s to 10.6 s, while for the improved version, it increases from 19.2 s to 3382.9 s. Beyond time complexity, the improvement obtained from the improved version declines from 5.86% to 1.29%, indicating that as instance size grows, the choice of starting solution (other than the TSP tour) becomes less significant. Fig. 6 demonstrates how the computational time of the improved heuristic increases while only marginal improvements are obtained as the size of the problem increases, making it not recommended for large instances. Finally, the results show that using the drone and robots in addition to the truck saves on average 43 min of delivery time compared to a classic truck-only delivery, highlighting the value of the hybrid last-mile delivery systems.

5.3. Large instances

In this section, we generate larger problems with $n = 100, 150, 200$, and 250 to demonstrate the applicability of the proposed heuristic for cases with a very large number of parcels. Due to the solution time complexity for the instances of this size, we only use the heuristic solution to solve the instances. To measure the quality of the solutions, we calculate the time-saving gains in comparison to a truck-only TSP tour delivery, as we did in Section 5.2. Table 7 summarizes the average results over five random instances of each category. For such large instances, the computational time of the heuristic can become significantly large as the problem size increases. For example, instances of $n = 250$ require the heuristic more than two hours to solve the problem. The cost-saving increases from 44.7 min to 79.1 min when the number of nodes increases from $n = 100$ to $n = 250$.

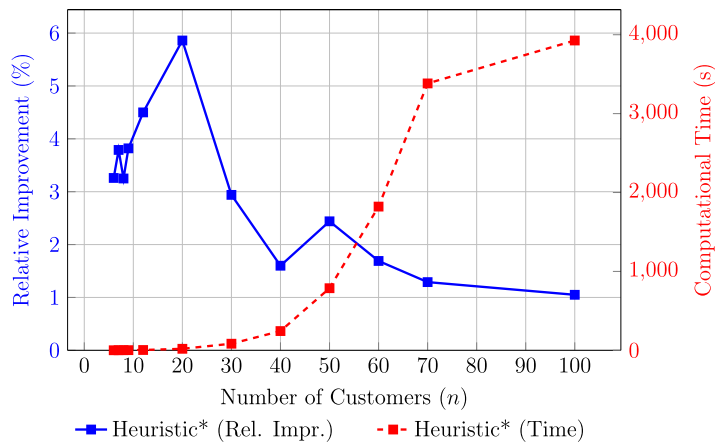


Fig. 6. Relative improvements vs. computational times for improvement framework on small- to medium-sized MV-FSTSP-MR problems.

Table 7

Average results of heuristic solution for large-sized problems.

n	Cost Saving (min)	Cost Saving (%)	Heuristic time (s)	Heuristic SD (s)	TSP time (s)	Alg. 1 time (s)	Alg. 3 time (s)	Alg. 3 iterations
100	44.7	31.7	87.5	9.8	0.7	85.5	1.3	3.6
150	59.5	32.3	652.7	62.3	1.8	645.8	5.1	5.0
200	69.7	33.9	2693.5	181.1	2.6	2666.0	24.9	4.6
250	79.1	32.8	9245.2	369.5	5.8	9185.8	53.6	6.6

To better understand the computational behavior of the proposed heuristic on large instances, we have also recorded the elapsed time for each component of the heuristic solution in this table. Specifically, we measured the running time of the initial TSP construction, [Algorithms 1](#), and [3](#), as well as the number of iterations [Algorithm 3](#) was executed in each instance. The findings show that most of the computational effort is concentrated in [Algorithm 1](#), whereas [Algorithm 3](#) requires relatively little time despite being executed multiple times. This confirms that [Algorithm 1](#) constitutes the primary computational bottleneck of the proposed approach, largely due to the drone's multi-visit capability.

6. Model implications and insights: Amazon case study

In this section, we conduct experiments on an Amazon order dataset from Seattle ([Merchán et al., 2024](#)) to evaluate the performance of the proposed multi-modal delivery system and solution in large-scale, real-world settings. Using the original Amazon truck driver tours as benchmarks, we assess the delivery time savings achieved by replacing a conventional truck with our modified truck.

6.1. Instances

Due to the large volume of data, we restrict our experiments to high-quality route IDs from the training dataset of Seattle, USA. As shown in [Fig. 7](#), the distribution of the proposed delivery instances mainly fall between 50 and 200 customers. Therefore, we categorize all instances into three groups and randomly choose 10 route IDs from each group. Through this process, a total of 30 route IDs were chosen for the case study experiments, as detailed in [Table 8](#). Because the central depots in the Amazon dataset are typically located far from the customer zones, we separate the travel time between the depot and the service area from the travel within the zone, as shown in this table. Travel between the depot and the customer zone usually occurs on highways, whereas in-zone travel takes place on congested local streets. This distinction enables a more accurate analysis of the proposed multi-modal delivery system versus a conventional Amazon truck, due to substantial depot-to-zone commute times, which on average account for 25% of the total travel times (39%, 21%, and 15% for Categories 1–3, respectively).

The truck travel times between nodes are already provided in the dataset. To estimate robot travel times, we scale the truck travel times by the ratio of truck velocity to robot velocity. Drone travel times are computed using the Haversine distance between nodes, based on their latitude and longitude, divided by the average drone velocity ([Jiang et al., 2025](#)). The same vehicle velocities as in [Section 5](#) are used in this case study. Since the dataset does not include package weights (only dimensions), we randomly generate payload weights between 0.1–5 kg, ensuring that approximately 86% of the values fall below 2.3 kg (5 pounds) to follow the Amazon statistics ([Guglielmo, 2013](#)).

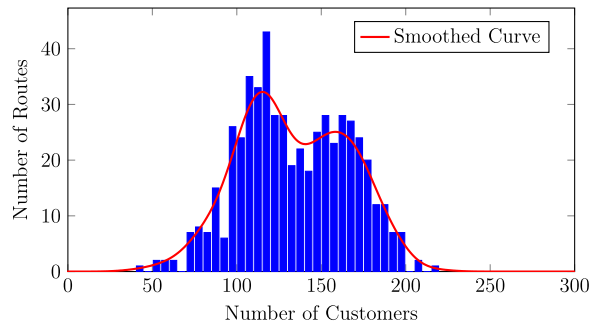


Fig. 7. Histogram of number of customers in Amazon delivery routes under study.

Table 8

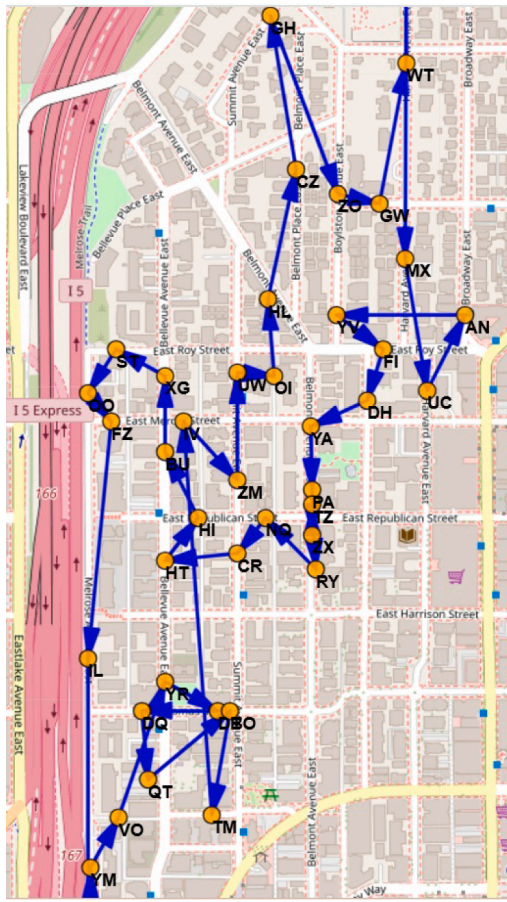
Instances used for Amazon case study.

Category	No.	# Customers	Total Travel Time (min)	Time to Zone (min)	Time to Depot (min)	Zone Travel Time (min)
#1 $n \in [50, 100]$	1	75	153.73	39.56	39.58	74.59
	2	83	151.31	41.25	36.24	73.82
	3	94	170.26	38.62	36.54	95.10
	4	97	119.25	17.18	20.07	82.00
	5	95	142.66	17.30	21.04	104.32
	6	98	141.16	19.62	19.96	101.58
	7	89	129.78	22.99	25.53	81.26
	8	85	107.39	22.02	22.96	62.41
	9	93	143.00	24.46	24.66	93.88
	10	64	85.74	20.97	18.73	46.04
#2 $n \in [100, 150]$	11	140	150.81	15.55	4.15	131.11
	12	126	143.17	14.76	6.22	122.19
	13	123	168.72	11.43	10.48	146.81
	14	135	158.42	14.54	12.00	131.88
	15	149	213.01	20.78	18.41	173.82
	16	134	178.75	10.70	10.33	157.72
	17	139	203.96	17.94	20.02	166.00
	18	147	199.22	11.42	13.29	174.51
	19	127	207.19	46.71	42.23	118.25
	20	134	184.37	41.64	40.48	102.25
#3 $n \in [150, 200]$	21	173	193.97	9.99	10.11	173.87
	22	176	177.81	10.70	7.42	159.69
	23	153	198.69	17.60	15.52	165.57
	24	180	220.20	18.07	13.15	188.98
	25	180	178.32	7.42	8.72	162.18
	26	181	253.14	31.79	32.04	189.31
	27	152	205.12	20.39	13.08	171.65
	28	191	235.42	20.65	18.94	195.83
	29	165	204.75	18.46	18.34	167.95
	30	199	235.22	20.47	20.42	194.33

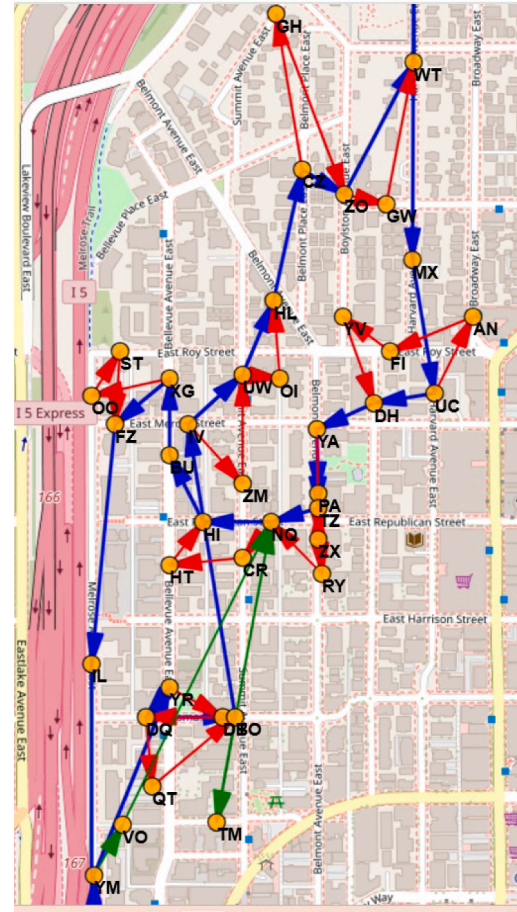
6.2. An illustrative example

Fig. 8 illustrates the original delivery route taken by an Amazon driver versus the route suggested by the proposed multi-modal system to serve the customers in a sample instance with 40 customers, referenced as “RouteID_f82c3715-3bd1-4a55-bbd3-a94e7e494dc7”, focusing on the portion of the map corresponding to the customer zone. The truck travels from the depot to the first customer (“YM”), serves all customers within the neighborhood (zone), and returns to the depot after the last customer (“IL”). In the original route, the truck spends 14.79 min traveling from the depot to the first customer, 32.16 min moving between customers within the neighborhood, and 15.17 min returning to the depot, for a total of 62.13 min. By implementing the proposed modified truck with robots and drone, the total delivery time is reduced to 47.67 min, and the zone travel time decreases from 32.16 min to 17.71 min, representing a 44.9% reduction. This shows significant efficiency gains in the congested customer area while maintaining full service coverage.

Another insight highlights how the drone and robots assist the truck differently during delivery. Due to its higher speed, the drone performs several rendezvous throughout the truck tour, whereas the robots exhibit a “release-early and catch-late” behavior. As seen in Fig. 8b, two robots contribute to the delivery operations. The first robot is released at the first customer visited by the truck in the neighborhood, i.e., customer “YM,” serves customer “VO,” and is collected at node “NQ” (near the end of the truck’s tour within



(a) Original Amazon truck delivery



(b) Proposed truck-robot-drone delivery

Fig. 8. Original Amazon truck driver route vs. the proposed multi-modal routes for an Amazon sample instance with 40 customers. Blue: truck route; red: drone routes; green: robot routes.

the neighborhood). The second robot is released near the beginning of the truck tour at node “BO,” serves customer “TM,” and is collected near the end of the truck tour at customer node “NQ.” Although the truck had eight robots onboard, only two were deployed due to the neighborhood’s small size, indicating a limited robot contribution for this scale.

6.3. Evaluation of the Amazon instances

In this section, we solve all the instances listed in Table 8 using the truck, robot, and drone specifications described in Section 5. The truck is assumed to be equipped with eight onboard robots, and the drone can perform a maximum of three visits per trip. Table 9 summarizes the results of the heuristic solution for solving the case instances. This table reports: (i) the total delivery completion time in minutes when using the proposed modified truck instead of the Amazon truck, (ii) the zone travel time in minutes, (iii) the overall delivery time reduction in minutes, (iv) the relative time reduction in both overall time and travel time within the zone, (v) the number of robots actually used out of eight available robots on board, (vi) the contribution of different vehicle types in the delivery process, and (vii) the computational time of the proposed heuristic in obtaining the results.

According to the results, on average a delivery time reduction of 51.2 min is achieved, which comes entirely from the time the truck spends inside the neighborhood. Spending 51.2 min less within the customer zone can bring significant benefits to the neighborhood in terms of reduced traffic congestion, pollution, noise, and operational costs. In relative measures, the average Amazon delivery time is reduced by 29.3% and 39.5% for overall and zone travel times, respectively. Additionally, for larger deliveries, time savings can reach up to 43.1% and 52.3% for overall and zone travel times, respectively, potentially translating into substantial reductions in environmental impact in urban logistics. On average, 3.8 robots were used during the delivery process in this case study, and the number of robots employed never exceeded seven. Therefore, we can conclude that a normal-sized delivery truck, such as a Mercedes-Benz Sprinter equipped with eight Starship Robots, would suffice for the Amazon case study. Fig. 9 also shows the computational time of the heuristic solution for the Amazon case instances as a polynomial function of the number of customers.

Table 9
Amazon case study results for proposed truck-robot-drone delivery with eight robots on board.

Instance	Total Completion Time (min)	Zone Travel Time (min)	Overall Time Reduction (min)	Overall Time Reduction (%)	Zone Time Reduction (%)	Robot Count	# Customers Served by Truck	# Customers Served by Drone	# Customers Served by Robot	Comput. Time (s)
1	123.38	44.25	30.35	19.7	40.7	6	43	25	7	49.01
2	118.03	40.54	33.28	22.0	45.1	3	49	29	5	75.82
3	136.81	61.64	33.45	19.6	35.2	4	57	30	7	136.75
4	82.09	44.83	37.16	31.2	45.3	7	54	33	10	139.41
5	92.10	53.76	50.56	35.4	48.5	2	59	32	4	136.96
6	88.02	48.44	53.14	37.6	52.3	3	49	41	8	152.97
7	98.20	49.68	31.58	24.3	38.9	3	50	34	5	103.83
8	78.98	34.00	28.41	26.5	45.5	3	46	35	4	69.61
9	94.57	45.44	48.43	33.9	51.6	2	51	39	3	115.41
10	66.06	26.35	19.68	23.0	42.8	3	33	28	3	20.94
11	101.27	81.57	49.54	32.8	37.8	3	83	49	8	852.82
12	84.59	63.61	58.58	40.9	47.9	5	64	49	13	530.16
13	120.61	98.70	48.11	28.5	32.8	5	78	30	15	508.18
14	90.22	63.68	68.20	43.1	51.7	3	73	55	7	736.33
15	152.05	112.86	60.96	28.6	35.1	5	84	44	21	1310.62
16	130.19	109.17	48.56	27.2	30.8	5	79	35	20	787.19
17	151.35	113.39	52.61	25.8	31.7	4	88	36	15	921.70
18	153.14	128.44	46.08	23.1	26.4	2	105	31	11	1195.75
19	165.70	76.77	41.49	20.0	35.1	2	79	42	6	527.96
20	140.72	58.59	43.65	23.7	42.7	5	65	58	11	649.71
21	111.14	91.04	82.83	42.7	47.6	7	87	67	19	2350.79
22	115.62	97.49	62.19	35.0	39.0	6	99	55	22	2644.80
23	144.69	111.58	54.00	27.2	32.6	5	96	40	17	1465.42
24	150.69	119.47	69.51	31.6	36.8	4	101	61	18	2916.68
25	120.94	104.81	57.38	32.2	35.4	2	119	54	7	3156.30
26	181.97	118.14	71.17	28.1	37.6	3	114	57	10	3326.47
27	151.84	118.36	53.28	26.0	31.0	3	97	41	14	1546.74
28	162.41	122.81	73.01	31.0	37.3	4	109	60	22	4108.81
29	160.63	123.84	44.12	21.5	26.3	3	106	45	14	2209.70
30	151.37	110.49	83.85	35.6	43.1	3	111	72	16	5245.58

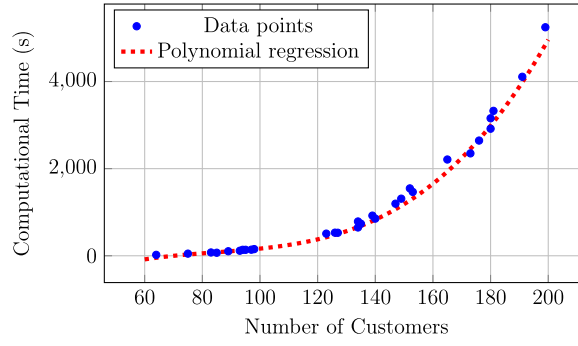


Fig. 9. Polynomial growth of heuristic solution runtime for Amazon case study.

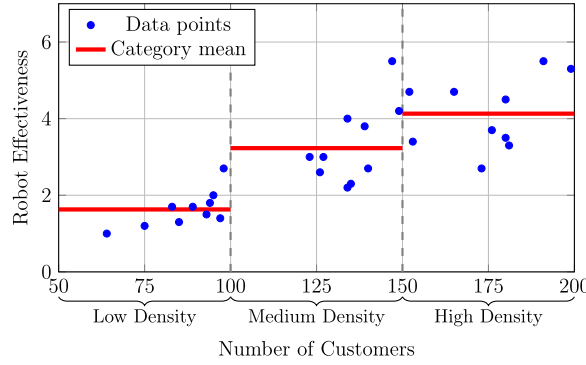


Fig. 10. Robot effectiveness across different customer density levels in the Amazon case study.

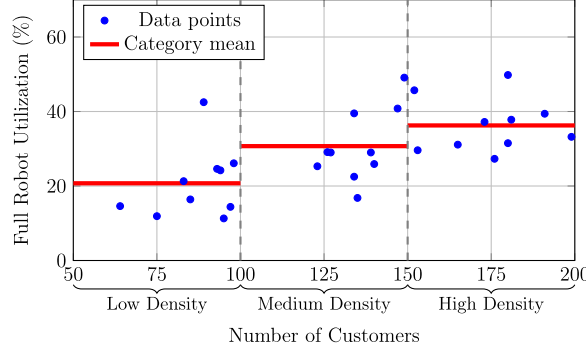


Fig. 11. Robot utilization across different customer density levels in the Amazon case study.

Since the service area remains nearly identical across the Amazon cases while the number of customers varies, the instance categories in Table 8 can also be interpreted as three customer density categories: low, medium, and high. To demonstrate the impact of customer density on robot performance, we define two main measures. First, “robot effectiveness” is defined as the ratio of customers served by robots to the number of robots deployed. Second, “robot utilization” is defined as the fraction of total travel time by all robots relative to the zone travel time, divided by the number of robots deployed. According to Figs. 10 and 11, both measures generally increase with customer density, indicating that robots remain active for a larger fraction of their operational time in denser environments. These observations highlight the improved robot operational performance in denser urban areas.

6.4. Scenario analysis

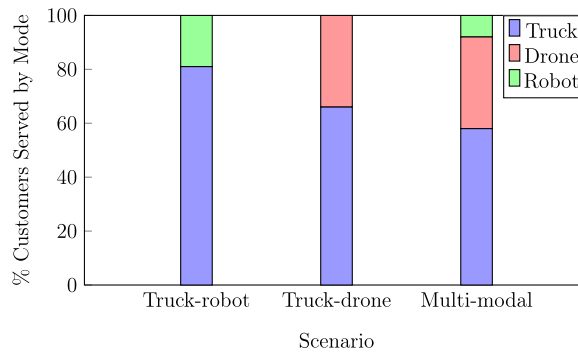
In this section, we aim to answer the following questions:

1. How will multi-modality help in improving the flexibility of the Amazon delivery? For that we compare the results from the proposed modified truck (one multi-visit drone and eight delivery robots) in Section 6.3 to a case with two extreme scenarios with either zero robots on board (truck-drone model) or a scenario with no drone on board (truck-robot model) (see Section 6.4.1).

Table 10

Delivery time savings gained by multi-modal delivery versus scenarios without robots or drone.

Scenario	Overall Time Reduction (min)	Overall Time Reduction (%)	Zone Time Reduction (%)	Robot Count	% Customers Served by Truck	% Customers Served by Drone	% Customers Served by Robot
Truck-robot	19.6	11.2%	15.0%	4.17	81%	–	19%
Truck-drone	39.4	22.8%	31.1%	–	66%	34%	–
Multi-modal	51.2	29.3%	39.5%	3.83	58%	34%	8%

**Fig. 12.** Average contribution of truck, drone and robots in different scenarios in Amazon case study.

- How will the effect of the limited-drone-delivery assumption, whether arising from predictable or unexpected real-time environmental conditions common in congested urban areas, impact the performance of the proposed multi-modal delivery system, and to what extent can robot integration compensate for the drone limitations? In this case, by randomly removing a portion of customers from drone service, we assess how this limitation can be mitigated through the use of onboard robots (see [Section 6.4.2](#)).
- How does the number of robots onboard the truck affect delivery performance in the proposed multi-modal delivery system? (see [Section 6.4.3](#)).

6.4.1. Impact of the multi-modality

In this analysis, we solve all 30 instances in [Table 8](#) with two different configurations to demonstrate the impact of multi-modality in this study. First, we remove the drone component from the proposed modified truck and keep the eight robots on board the truck. In another configuration, we remove all the robots on board from the proposed modified truck and turn it into a truck-drone delivery model. The drone here still has the capability of performing a maximum of three deliveries per trip. The results of these two extreme conditions are then compared with those of [Table 9](#), and the average results are recorded in [Table 10](#).

According to the results, the drone has the same contribution in a multi-modal scenario as in a truck-drone scenario. This is expected because the heuristic solution first allocates customers between the truck and drone, and then assigns the remaining customers to the robots. However, in the truck-robot scenario, the analysis shows that the robots' contribution increased more than twofold, rising from 8% to 19% in comparison to the multi-modal scenario. The contributions of the different vehicles are illustrated in [Fig. 12](#).

Although the truck-robot delivery model possesses the most active robot fleet compared to other configurations, the average delivery time savings have decreased from 51.2 min in the multi-modal delivery and 39.4 min in the truck-drone delivery to 19.6 min, which represents nearly a 50% reduction. Therefore, in situations with no drone-access limitations, the multi-modal delivery provides the highest efficiency, the truck-drone model holds the second-best configuration, and the truck-robot model results in the least reduction in delivery time. In a more detailed comparison, the multi-modal delivery achieves 11.8 min more savings compared to the truck-drone model, while the truck-drone model achieves 19.8 min more savings compared to the truck-robot model. Although the drone plays a critical role in the flexibility of the proposed modified truck, integration of the robots provide an additional contribution of 11.8 min in delivery time savings, 6.5% in relative overall delivery time reduction, and an extra 8.4% in zone delivery time reduction compared to the truck-drone delivery model. The diagram of the relative delivery time savings for different scenarios is plotted in [Fig. 13](#), providing a clearer picture of incremental delivery time savings from a truck with multi robots, to a truck with a single drone, and up to a truck with both a drone and multiple robots on board.

6.4.2. Limited drone delivery access

To analyze the robot contribution in the proposed multi-modal delivery system, we design a scenario in which a portion of customers is restricted from drone delivery. This is a common assumption, as drones are the most vulnerable component in last-mile delivery to environmental impacts such as weather changes or no-fly-zone regulations. In this analysis, we randomly exclude 20% of the customers from drone delivery and then solve the instances again under two scenarios: zero robots on board and eight robots on board. This process is repeated over ten independent replications in each instance, each with a different randomly selected subset of customers. The results in [Table 11](#) correspond to the average performance across these replications over all instances. This table also provides a comparison with the scenario where the drone has full accessibility to all customers, as previously shown in [Table 10](#).

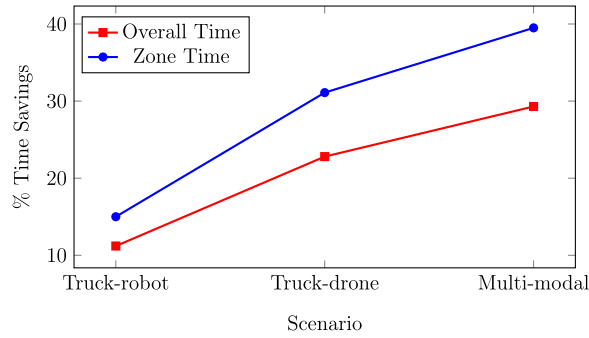


Fig. 13. Average delivery time savings for different scenarios in the Amazon case study.

Table 11
Results for limited drone access.

	$R_{\text{robot}} = 8$		$R_{\text{robot}} = 0$	
	Full accessibility	20% impact	Full accessibility	20% impact
Overall Time Reduction (min)	51.2	45.7	39.4	29.9
Overall Time Reduction (%)	29.3	26.5	22.8	19.0
Zone Time Reduction (%)	39.5	34.9	31.1	26.7
Number of Robots Used	3.8	3.8	–	–
Robot Contribution (%)	8.3	10.0	–	–

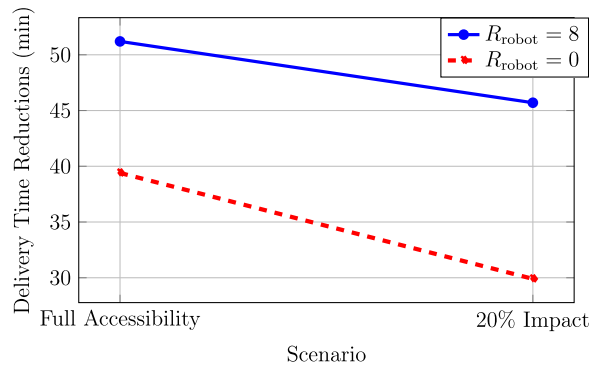


Fig. 14. Impact of 20% drone inaccessibility for two robot scenarios.

According to the results, when a delivery truck with both a drone and robots is impacted by 20% drone inaccessibility, its overall average time saving decreases from 51.2 min to 45.7 min, showing a performance decline of 5.5 min (10.7%). However, when a similar truck equipped only with a drone (zero robots on board) is affected by this scenario, the overall average time saving decreases from 39.4 min to 29.9 min, indicating a more significant decline of 9.5 min (24.1%). This difference in performance deterioration is clearly illustrated in Fig. 14. Therefore, in practice, the proposed multi-modal delivery system is about twice as flexible to environmental conditions that affect drone performance as a comparable delivery system with a truck and only a drone. Another important finding is that this impact is mitigated without changing the number of operational robots, but with an increase in their contribution from an average of 8.3% to 10.0%.

6.4.3. Impact of the number of robots

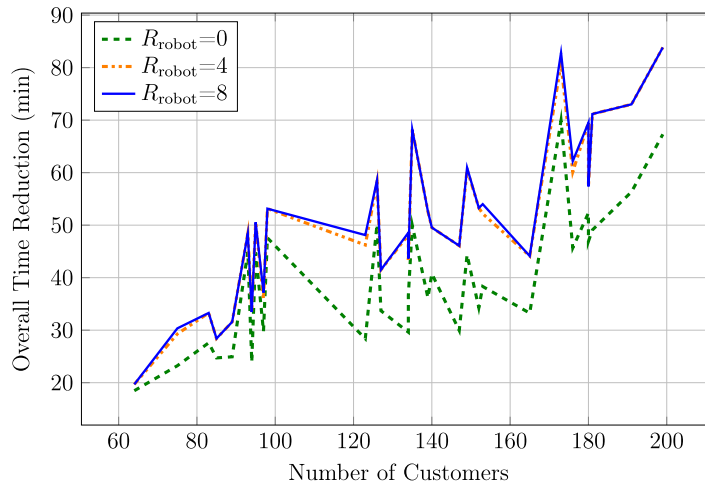
In this analysis, we evaluate the impact of varying the number of robots onboard the truck across all 30 Amazon instances, while keeping the number of drones fixed at one. The number of drones is fixed at one, as a multi-compartment drone capable of visiting multiple customers in a single trip (e.g., Wingcopter 198) is practically limited to one per truck. In contrast, multiple small sidewalk robots can be economically deployed onboard a single truck (e.g., up to eight Starship robots on a Mercedes-Benz Sprinter) at a much lower cost (Condcliffe, 2018; Plümmer, 2021; Drone Launch Academy, 2026). We consider three scenarios: $R_{\text{robot}} = 0$, $R_{\text{robot}} = 4$, and $R_{\text{robot}} = 8$, and the average results for each scenario are summarized in Table 12.

The results reveal a marginal difference in performance between the eight-robot and four-robot scenarios, suggesting that four robots onboard is sufficient for this Amazon case study. This finding offers valuable guidance to logistics operators regarding the level of investment needed in the robot fleet. Fig. 15 shows the overall time reductions for each instance across all three scenarios,

Table 12

Average results for different number of robots on board.

	$R_{\text{robot}} = 8$	$R_{\text{robot}} = 4$	$R_{\text{robot}} = 0$
Overall Time Reduction (min)	51.2	50.8	39.4
Overall Time Reduction (%)	29.3	29.0	22.8
Zone Time Reduction (%)	39.5	39.2	31.1
Number of Robots Used	3.8	3.3	0

**Fig. 15.** Overall time reductions for different numbers of robots on board.

revealing an increasing trend in time reductions as problem size grows. The gap between the dashed green line ($R_{\text{robot}} = 0$) and the solid blue line ($R_{\text{robot}} = 8$) widens with instance size, highlighting the growing benefit of deploying robots for larger last-mile delivery problems. Meanwhile, the dashed-dotted orange line ($R_{\text{robot}} = 4$) nearly overlaps with the eight-robot scenario across all instances, confirming the marginal difference between the two configurations.

7. Modeling extensions

This section discusses the extensibility of the proposed MV-FSTSP-MR framework and solution regarding multi-compartment robots and multi-drone fleet extensions.

7.1. Multi-compartment robot extension

The current formulation of the MV-FSTSP-MR is motivated by existing commercially deployed sidewalk robot platforms (e.g., Starship and Amazon Scout), which are typically designed with a single-compartment structure due to size and weight limitations when being transported onboard a truck (Yu and Puchinger, 2024). Accordingly, the model assumes that each robot carries at most one parcel per deployment cycle. However, future robotic systems may incorporate multi-compartment designs. In such cases, the proposed MILP formulation can be naturally extended with minimal modifications. In particular, the right-hand side of Constraint (26) can be generalized from one to the number of available compartments per robot, thereby allowing each robot to serve multiple customers within a single deployment while preserving model consistency.

Similarly, the heuristic framework can be extended by modifying Algorithm 3 to allow multiple customer services within a single robot trip. This can be achieved by constructing each robot route as a shortest path from the launch node to the retrieval node while visiting multiple customers, which is similar to Algorithm 1 but without the drone-specific energy-efficiency considerations.

7.2. Multi-drone fleet extension

A multi-compartment drone capable of visiting multiple customers in a single trip (e.g., Wingcopter 198) is relatively large and is practically limited to one per truck. However, the proposed MILP model can be easily adapted to cases with multiple drones of this type by simply increasing the parameter R_{drone} in Constraints (22) and (24) from one to the desired quantity. From a heuristic perspective, the proposed Algorithm 1 can be extended in a manner similar to Algorithm 3 to include multiple vehicles of the same type by adding them iteratively, tracking truck delays at the rendezvous nodes, and introducing additional drones as long as they contribute to reducing the overall completion time.

8. Conclusion

This paper presents a mixed-integer program to model the last-mile delivery problem using a delivery truck equipped with a multi-visit drone and multiple robots, referred to as MV-FSTSP-MR. In our model, we assume that all vehicle types work synchronously to deliver products to customers' doorsteps. Due to the inefficiency of commercial solvers, we develop a novel heuristic based on the principles of the route-first, split-second approach. We also propose an iterative improvement framework that incorporates the heuristic to search for better solutions.

The results of a computational study using a benchmark dataset in a 100 km² urban service area show that the proposed heuristic is effective across small, medium, and large problem sizes. For small instances, it achieves near-optimal performance, with delivery completion times within one minute of the global optimum, while requiring only fractions of a second to a few seconds of computation. For large instances with 100–250 customers, it provides approximately one hour of delivery time savings, corresponding to a 32.7% reduction relative to a conventional truck-only (TSP) system, with computation times ranging from a few minutes to two hours and a half. The improved version of the heuristic further enhances solution quality for small and medium instances within reasonable time. However, it becomes less attractive for large problems, as the relative improvement decreases while the required computational time increases.

To demonstrate the practicality of the proposed multi-modal delivery system and the heuristic solution in a realistic setting, we used historical delivery data from an Amazon case study in Seattle, USA. In this case study, the customer size ranges from 50 to 200. Analysis of the case study indicates that implementing the proposed truck-robot-drone model can reduce delivery times per shift by an average of 51 min, with potential savings of up to 84 min for larger delivery sizes. In relative terms, the average Amazon travel time within the customer zones is reduced by 40%, with the potential to decrease by up to 52%. Due to the slower speeds of sidewalk robots, the drone handles a larger portion of deliveries compared to the robots. However, there is clear value in integrating all three vehicle types. First, including robots improved average travel time by 8.4% compared to a model with only the truck and multi-visit drone. Second, there may be locations that are difficult for the drone to access, such as downtown areas with high-rise buildings or regions with no-fly zone restrictions. Finally, drones are more vulnerable to environmental factors than delivery robots. As shown in the case study, under adverse conditions that limit drone access to certain areas, the proposed truck-drone-robot system experiences only a minor performance decline compared to a truck relying solely on a drone. We also analyzed the impact of the number of robots onboard the truck on delivery performance. Results show that completion time decreases as the number of robots increases; however, the marginal benefit diminishes beyond a certain point due to operational constraints. Notably, four robots achieved nearly the same average performance as the original configuration of eight in the Amazon instances, suggesting that doubling the robot fleet does not always justify the additional investment.

Finally, future work could focus on designing exact algorithms to solve the problem, such as branch-and-price, which may benefit greatly from a good starting point potentially provided by the heuristic proposed in this work. More spatially-aware approaches, such as region partitioning methods, may also improve solution quality for larger instances and are worth investigating. In terms of modeling aspects, extensions to the proposed framework including multi-compartment robot routing and multi-drone fleet configurations are discussed in Section 7. Further extensions may consider multiple trucks with multiple drones and robots, where drones and robots are not restricted to a single truck and can assist any truck in the system. Additionally, future work could incorporate uncertainties into the model, such as travel time variability or drone and robot malfunctions. Another useful direction is to compare the proposed system with hub-based delivery architectures in real-world scenarios to provide a comprehensive analysis of their cost-effectiveness. Finally, the proposed framework could also be extended by incorporating urban delivery regulations, such as time-restricted parking, curbside stop duration limits, and spatial constraints on robot and drone deployment at rendezvous nodes.

CRedit authorship contribution statement

Behrang Bootaki: Writing – original draft, Validation, Software, Methodology, Conceptualization; **Omar Abou Kasm:** Writing – review & editing, Validation, Supervision, Methodology, Funding acquisition, Conceptualization; **Joe Naoum-Sawaya:** Writing – review & editing, Validation, Supervision, Methodology, Conceptualization.

Acknowledgments

This research was supported by the Academic Strategic Initiative (ASI) Fund at Algoma University. Joe Naoum-Sawaya is supported by the [Natural Sciences and Engineering Research Council of Canada](#) (NSERC) Discovery Grant [RGPIN-2024-04176](#).

Data availability

Data will be made available on request.

Appendix A. Maximum number of trips for autonomous vehicles

Proposition. Given n customers to be served by a truck and R_v autonomous vehicles of type v (robots or drones) synchronized with the truck, the maximum number of trips required for each vehicle type is

$$k_{\max}^v = n - \left\lfloor \frac{n}{R_v + 1} \right\rfloor. \quad (\text{A.1})$$

Equivalently, writing $n = (R_v + 1)q + r$ with $q = \left\lfloor \frac{n}{R_v + 1} \right\rfloor$ and $0 \leq r < R_v + 1$:

$$k_{\max}^v = \begin{cases} R_v q & r = 0 \\ R_v q + (r - 1) & r > 0 \end{cases} \quad (\text{A.2})$$

Proof. Each autonomous vehicle trip must depart from and return to a truck-served node. Hence every group of customers served in a single synchronized stop contains exactly one truck-served customer. With n customers partitioned into groups of at most $R_v + 1$, there are at least $\left\lfloor \frac{n}{R_v + 1} \right\rfloor$ such groups, so the truck must serve at least $\left\lfloor \frac{n}{R_v + 1} \right\rfloor$ customers. Autonomous vehicles serve the remaining customers, so the number of trips cannot exceed:

$$n - \left\lfloor \frac{n}{R_v + 1} \right\rfloor. \quad (\text{A.3})$$

Writing $n = (R_v + 1)q + r$:

Case 1: $r = 0$. Partition the n customers into q groups of $R_v + 1$. Assign one customer per group to the truck and the remaining R_v to autonomous vehicles. This yields

$$k_{\max}^v = R_v q \quad (\text{A.4})$$

robot trips, achieving the upper bound.

Case 2: $r > 0$. Form q full groups of $R_v + 1$, yielding $R_v q$ robot trips, and one partial group of r customers. Assign one of the r customers to the truck and the remaining $r - 1$ to autonomous vehicles, yielding $r - 1$ additional trips. The total is

$$k_{\max}^v = R_v q + (r - 1), \quad (\text{A.5})$$

which again achieves the upper bound. $\square$

Appendix B. Proof of Theorem 1

Proof. Algorithm 1 extends the SLA to handle multi-customer drone deliveries. For each drone batch of size at most $m = 3$, the heuristic evaluates all $m!$ possible delivery orders to determine the optimal sequence. Since m is bounded, the drone permutation evaluation contributes only a constant factor, $O(m!) = O(1)$. Given the SLA has a worst-case complexity of $O(n^3)$ (Hokama et al., 2025), where n is the number of customers, Algorithm 1 then has complexity of $O(n^3 \cdot m!) = O(n^3 \cdot O(1)) = O(n^3)$. Algorithm 3, which extends the SLA to handle multiple robots, is run R_{robot} times, where R_{robot} represents the number of robots on board. Since R_{robot} is bounded (at most eight), this again contributes only by a constant factor, $O(1)$, leaving its overall complexity $O(n^3)$. Finally, because the two algorithms are executed sequentially, the total worst-case complexity is $O(n^3) + O(n^3) = O(n^3)$. $\square$

References

- Agatz, N., Bouman, P., Schmidt, M., 2018. Optimization approaches for the traveling salesman problem with drone. *Transp. Sci.* 52 (4), 965–981.
- Aria, A., 2021. Wingcopter debuts a triple-drop drone to create 'logistical highways in the sky'. <https://techcrunch.com/2021/04/27/wingcopter-debuts-a-triple-drop-drone-to-create-logistical-highways-in-the-sky/>. Accessed: July 8, 2026.
- Bakach, I., Campbell, A.M., Ehmk, J.F., 2021. A two-tier urban delivery network with robot-based deliveries. *Networks* 78 (4), 461–483.
- Boccia, M., Mancuso, A., Masone, A., Sterle, C., 2024. Exact and heuristic approaches for the truck-drone team logistics problem. *Transp. Res. C Emerg. Technol.* 165, 104691.
- Boysen, N., Schwerdfeger, S., Weidinger, F., 2018. Scheduling last-mile deliveries with truck-based autonomous robots. *Eur. J. Oper. Res.* 271 (3), 1085–1099.
- Cengiz, E., Yilmaz, C., Kahraman, H.T., 2025. A literature review of drone-truck routing problems: challenges and future research directions. *RAIRO-Oper. Res.* 59 (5), 3169–3205.
- Cheng, C., Adulyasak, Y., Rousseau, L.-M., 2020. Drone routing with energy function: formulation and exact algorithm. *Transp. Res. B: Methodol.* 139, 364–387.
- Condliffe, J., 2018. Why sidewalk delivery robots still need safety drivers. <https://www.technologyreview.com/2018/01/30/145935/why-sidewalk-delivery-robots-still-need-safety-drivers-too/>.
- Dorling, K., Heinrichs, J., Messier, G.G., Magierowski, S., 2016. Vehicle routing problems for drone delivery. *IEEE Trans. Syst. Man Cybern. Syst.* 47 (1), 70–85.
- Drone Launch Academy, 2026. How much is a drone? costs for different types & models. https://dronelaunchacademy.com/resources/how-much-is-a-drone-costs-for-different-types-models/#elementor-toc_heading-anchor-17.
- Duan, J., Luo, H., Wang, G., 2025. Approaches to the truck-drone routing problem: a systematic review. *Swarm Evol. Comput.* 92, 101825.
- Gao, J., Zhen, L., Wang, S., 2023. Multi-trucks-and-drones cooperative pickup and delivery problem. *Transp. Res. C Emerg. Technol.* 157, 104407.
- Glover, F., 1990. Tabu search: a tutorial. *Interfaces* 20 (4), 74–94.
- Guglielmo, C., 2013. Turns out Amazon, touting drone delivery, does sell lots of products that weigh less than 5 pounds. <https://www.forbes.com/sites/connieguglielmo/2013/12/02/turns-out-amazon-touting-drone-delivery-does-sell-lots-of-products-that-weigh-less-than-5-pounds/>. Accessed: July 8, 2026.
- Ha, Q.M., Deville, Y., Pham, Q.D., Hà, M.H., 2018. On the min-cost traveling salesman problem with drone. *Transp. Res. C Emerg. Technol.* 86, 597–621.
- Heimfarth, A., Ostermeier, M., Hübner, A., 2022. A mixed truck and robot delivery approach for the daily supply of customers. *Eur. J. Oper. Res.* 303 (1), 401–421.
- Hokama, P. H. D.B., Lintzmayer, C.N., San Felice, M.C., 2025. A faster heuristic for the traveling salesman problem with drone. *Optim. Lett.* 19 (4), 771–791.
- Jacobs, K., Warner, S., Rietra, M., Mazza, L., Buvat, J., Khadikar, A., Cherian, S., Khemka, Y., 2019. The last-mile delivery challenge. <https://www.capgemini.com/insights/research-library/the-last-mile-delivery-challenge/>. Accessed: July 8, 2026.
- Jeong, H.Y., Song, B.D., Lee, S., 2019. Truck-drone hybrid delivery routing: payload-energy dependency and no-fly zones. *Int. J. Prod. Econ.* 214, 220–233.
- Jiang, Y., Liu, M., Jia, X., Xue, Q., 2025. The multi-visit vehicle routing problem with multiple heterogeneous drones. *Transp. Res. C Emerg. Technol.* 172, 105026.
- Karak, A., Abdelghany, K., 2019. The hybrid vehicle-drone routing problem for pick-up and delivery services. *Transp. Res. C Emerg. Technol.* 102, 427–449.
- Kitjacharoenchai, P., Min, B.-C., Lee, S., 2020. Two echelon vehicle routing problem with drones in last mile delivery. *Int. J. Prod. Econ.* 225, 107598.
- Lichau, S., Sadykov, R., François, J., Dupas, R., 2025. A branch-cut-and-price approach for the two-echelon vehicle routing problem with drones. *Comput. Oper. Res.* 173, 106869.
- Liu, D., Kaisar, E.L., Yang, Y., Yan, P., 2022. Physical internet-enabled e-grocery delivery network: a load-dependent two-echelon vehicle routing problem with mixed vehicles. *Int. J. Prod. Econ.* 254, 108632.

- Luo, H., Duan, J., Wang, G., 2025. Mathematical models for truck-drone routing problem: literature review. *Appl. Math. Model.*, 116074.
- Luo, Y., Deng, X., Zhang, W., Ke, Y., Wan, S., Qian, Y., 2024. Collaborative intelligent delivery with one truck and multiple heterogeneous drones in COVID-19 pandemic environment. *IEEE Trans. Intell. Transp. Syst.* 25 (7), 7907–7920.
- Luo, Z., Poon, M., Zhang, Z., Liu, Z., Lim, A., 2021. The multi-visit traveling salesman problem with multi-drones. *Transp. Res. C Emerg. Technol.* 128, 103172.
- Mahmoudi, B., Eshghi, K., 2022. Energy-constrained multi-visit TSP with multiple drones considering non-customer rendezvous locations. *Expert Syst. Appl.* 210, 118479.
- Masmoudi, M.A., Mancini, S., Baldacci, R., Kuo, Y.-H., 2022. Vehicle routing problems with drones equipped with multi-package payload compartments. *Transp. Res. E Logist. Transp. Rev.* 164, 102757.
- Mercedes-Benz Group, 2017. Mercedes benz vans & robots. <https://www.youtube.com/watch?v=iDOKrgOnjiE>. Accessed: July 8, 2026.
- Merchán, D., Arora, J., Pachon, J., Konduri, K., Winkenbach, M., Parks, S., Noszek, J., 2024. 2021 Amazon last mile routing research challenge: data set. *Transp. Sci.* 58 (1), 8–11.
- Moeini, M., Salewski, H., 2019. A genetic algorithm for solving the truck-drone-ATV routing problem. In: *World Congress on Global Optimization*. Springer, pp. 1023–1032.
- Mokhtari-Moghadam, A., Salhi, A., Yang, X., Nguyen, T.T., Pourhejazy, P., 2025. A multi-objective approach for the integrated planning of drone and robot assisted truck operations in last-mile delivery. *Expert Syst. Appl.* 269, 126434.
- Morim, A., Campuzano, G., Amorim, P., Mes, M., Lalla-Ruiz, E., 2024. The drone-assisted vehicle routing problem with robot stations. *Expert Syst. Appl.* 238, 121741.
- Murray, C.C., Chu, A.G., 2015. The flying sidekick traveling salesman problem: optimization of drone-assisted parcel delivery. *Transp. Res. C Emerg. Technol.* 54, 86–109.
- Ostermeier, M., Heimfarth, A., Hübner, A., 2022. Cost-optimal truck-and-robot routing for last-mile delivery. *Networks* 79 (3), 364–389.
- Ostermeier, M., Heimfarth, A., Hübner, A., 2023. The multi-vehicle truck-and-robot routing problem for last-mile delivery. *Eur. J. Oper. Res.* 310 (2), 680–697.
- Plümmer, T., 2021. German start-up creates a delivery drone capable of toting three separate packages. <https://www.washingtonpost.com/technology/2021/04/28/drone-delivery-triple-drop/>.
- Poikonen, S., Golden, B., 2020. Multi-visit drone routing problem. *Comput. Oper. Res.* 113, 104802.
- Reed, S., Campbell, A.M., Thomas, B.W., 2022. The value of autonomous vehicles for last-mile deliveries in urban environments. *Manag. Sci.* 68 (1), 280–299.
- Schermer, D., Moeini, M., Wendt, O., 2020. The drone-assisted traveling salesman problem with robot stations. In: *HICSS*, pp. 1–10.
- Simoni, M.D., Kutanoglu, E., Claudel, C.G., 2020. Optimization and analysis of a robot-assisted last mile delivery system. *Transp. Res. E Logist. Transp. Rev.* 142, 102049.
- Solomon, M.M., 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper. Res.* 35 (2), 254–265.
- Taft, M., 2024. Amazon's shipping and delivery emissions just keep going up. <https://www.wired.com/story/amazons-shipping-and-delivery-emissions-just-keep-going-up/>. Accessed: July 8, 2026.
- UPS, 2017. Ups tests residential delivery via drone launched from atop package car. <https://metroairportnews.com/ups-tests-residential-delivery-via-drone-launched-atop-package-car>. Accessed: July 8, 2026.
- Wang, D., Hu, P., Du, J., Zhou, P., Deng, T., Hu, M., 2019. Routing and scheduling for hybrid truck-drone collaborative parcel delivery with independent and truck-carried drones. *IEEE Internet Things J.* 6 (6), 10483–10495.
- Wang, Z., Sheu, J.-B., 2019. Vehicle routing problem with drones. *Transp. Res. Part B Methodol.* 122, 350–364.
- Wen, X., Wu, G., 2022. Heterogeneous multi-drone routing problem for parcel delivery. *Transp. Res. C Emerg. Technol.* 141, 103763.
- Wingcopter, 2025. Wingcopter 198. <https://wingcopter.com/wingcopter-198>. Accessed: July 8, 2026.
- Wu, G., Mao, N., Luo, Q., Xu, B., Shi, J., Suganthan, P.N., 2022. Collaborative truck-drone routing for contactless parcel delivery during the epidemic. *IEEE Trans. Intell. Transp. Syst.* 23 (12), 25077–25091.
- Yu, S., Puchinger, J., 2024. Collaborative truck-robot deliveries: challenges, models, and methods. *Ann. Oper. Res.*, 1–38. <https://link.springer.com/article/10.1007/s10479-024-06127-w>.
- Yu, S., Puchinger, J., Sun, S., 2020. Two-echelon urban deliveries using autonomous vehicles. *Transp. Res. E Logist. Transp. Rev.* 141, 102018.
- Yu, S., Puchinger, J., Sun, S., 2024. Electric van-based robot deliveries with en-route charging. *Eur. J. Oper. Res.* 317 (3), 806–826.
- Zhou, H., Qin, H., Cheng, C., Rousseau, L.-M., 2023. An exact algorithm for the two-echelon vehicle routing problem with drones. *Transp. Res. Part B Methodol.* 168, 124–150.