Engineering Applications of Artificial Intelligence 124 (2023) 106631

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Sustainable last mile parcel delivery and return service using drones

Nawin Yanpirat, Daniel F. Silva *, Alice E. Smith

Department of Industrial & Systems Engineering, Auburn University, 3301 Shelby Center, Auburn, AL 36849, USA

ARTICLE INFO

Keywords:
Flying sidekick
Sustainable last mile logistics
Drone logistics
Delivery and return
Traveling salesman problem
Variable neighborhood search

ABSTRACT

Improving last mile delivery, in terms of both efficiency and sustainability, has recently been enhanced using truck/drone tandems. However, nearly all these approaches focus only on deliveries, while returns are ignored. We propose a new model for integrating both delivery and returns in the combined operation of a truck and a drone, which we term the Flying Sidekick Traveling Salesman Problem integrating Deliveries and Returns with Multiple Payloads (FSTSP-DR-MP). This approach is more sustainable than truck-only deliveries as the drones operate through battery power with no emissions and can move more directly than vehicles along a road network. We formulate the problem as a mixed-integer linear program to minimize the total service time of the system, where the truck and the drone can perform both delivery and pickup of parcels during a single sortie. Because drones have a capacity of multiple parcels, each can visit more than one customer per dispatch, increasing drone utilization and responsiveness to customer expectations regarding return services, along with greater environmental improvements. Small-size cases are solved exactly using the MILP implemented in the CPLEX Python API. Since the MILP is not practical for realistically sized cases, we propose a meta-heuristic derived from Variable Neighborhood Search (VNS), which iteratively builds truck and drone routes. We show that this works well for up to 100 customers, a typical number in last-mile logistics. We assess the trade-offs between the ratio of returns to deliveries and drone capacity to provide managerial insights to the benefits of this approach for sustainable last mile logistics. Computational experiments show that integrating delivery and return truck-drone operations significantly reduces total service time and truck travel time compared to both traditional delivery schemes (single truck) and the well-known FSTSP drone schemes (one truck and one drone) up to 36.2% and 22.9%, respectively, by exploiting the nature of each customer (delivery or return) to increase the number of stops of each drone sortie. We also conduct a comparison with multiple drones have a single payload under similar scenarios. The results demonstrate that our model outperforms this alternative with an average improvement of 3.9% in total service time. Our approach addresses an important gap in the literature by accommodating returns, which are ubiquitous in last mile logistics, as well as providing for improved drone utilization, sustainability, and cost-effectiveness.

1. Introduction

Logistics for online retail has become a key component of business operations. Customers can easily purchase items online and receive them within a few days (or even quicker) and online sales are expected to become 21 percent of global retail sales by 2023 (Coppola, 2021). Furthermore, as a result of the coronavirus pandemic, since 2020, online shopping sales growth for most retailers has skyrocketed (Troise, 2021). Moreover, last mile delivery accounts for nearly 60 percent of the total transportation costs (Dolan, 2021), motivating logistics companies, along with online-retail companies, to seek improved methods for last mile service, such as autonomous ground vehicles (AGVs), semi-autonomous ground vehicles, droids, drones, bike and e-bike couriers (Joerss et al., 2016; Carracedo and Mostofi, 2022; D'hont et al., 2022). Of course, the adverse impacts of truck networks on the environment, principally through the use of fossil fuels and generation of emissions is well known and has increased with this increase in online purchasing. There has been much recent research on last mile delivery systems to improve operations and address sustainability aspects, including integrating delivery trucks with scooters and couriers on foot (Lin, 2011), truck-based autonomous robots (Boysen et al., 2018), pickup stations and autonomous vehicles (Ulmer and Streng, 2019), and using automatic delivery stations (Oliveira et al., 2017).

Drones, in particular, have ignited much recent attention. According to Rodrigues et al. (2022), using drones to deliver small parcels is more efficient in terms of costs and total service time than the conventional method of trucks only. They also find that using drones are more environmentally friendly since they produce less carbon, up to 94 percent less compared to trucks. Well-known companies such as Amazon, DHL, and Alibaba have conducted research into the potential use of drones

* Corresponding author.
E-mail address: silva@auburn.edu (D.F. Silva).

https://doi.org/10.1016/j.engappai.2023.106631
Received 22 November 2022; Received in revised form 17 May 2023; Accepted 8 June 2023
Available online 4 July 2023

for last mile delivery (Boysen et al., 2021). Though these are currently in the development phase, there are high expectations that drones will enhance last mile delivery services. Currently, prototype models from those companies can carry one parcel at a time and can perform only delivery services and are subject to the limitations of battery capacity and stationary launch and rendezvous locations.

The literature on combined truck and drone delivery was initiated by the now seminal work on the Flying Sidekick Traveling Salesman Problem (FSTSP) with a truck and a drone (Murray and Chu, 2015). The problem of routing a coordinated tandem of a truck and a drone has become known as the Traveling Salesman Problem with Drone (TSPD). This problem has been studied extensively and has been the focus of recent surveys by Macrina et al. (2020) and Viloria et al. (2021). Work has extended those ideas to multiple trucks and drones (Ham, 2018; Schermer et al., 2019a; Sacramento et al., 2019; Kitjacharoenchai et al., 2019), one truck and multiple drones with multiple payloads (Poikonen and Golden, 2020), and other variants such as a truck with multiple drones (Dell'Amico et al., 2021; Murray and Raj, 2020; Raj and Murray, 2020). However, none of these works have considered return services, although the need for parcel returns has also greatly increased with the rise of e-commerce.

According to Reagan (2019), online consumers return between 15 and 40 percent of ordered goods. However, most parcels are still returned in conventional ways, where customers send them from post offices or drop them off at designated locations. Even though returning products might be free of charge to customers, it still takes time and effort, and traveling expenses. And, of course, there is negative environmental impact with these separate trips to return purchases, particularly in the United States, where customers typically drive to the return locations.

In this paper, we extend FSTSP by coordinating a drone and a truck to perform return/pickup and delivery services, potentially in a single drone sortie, while incorporating multiple payloads. We consider one truck and one drone that can provide either return and/or delivery services for each customer. In other words, parcels are either delivered to a customer by truck or by drone (which is launched from the truck) while returns are collected from customers by truck or by drone. An additional novelty of our approach, absent from much of the truck-and-drone logistics literature, is the capacity of the drone to perform multiple stops, which is consistent with ongoing drone research on developing drones with greater payload/weight capacity and endurance (Cheng et al., 2019; Matheson, 2017). This feature allows for further exploitation of the differences between delivery and return customers. We propose a mathematical model and term it the Flying Sidekick Traveling Salesman Problem integrating Deliveries and Returns with Multiple Payloads (FSTSP-DR-MP).

The main difference between a standard FSTSP and our proposed model lies in the maximum number of possible stops for each drone sortie. For example, suppose that there are four customers and that the drone has a two-payload capacity. The maximum number of stops per sortie in the standard FSTSP model is two customers (Fig. 1(a)) because the model does not distinguish between pickup and delivery. However, the FSTSP-DR-MP model can make a maximum number of four stops (Fig. 1(b)), as long as it makes deliveries and picks ups as follows; Delivery → Delivery → Return → Return, or Delivery → Return → Delivery → Return. This yields greater potential service time savings than a standard FSTSP model and increases the utilization of the drone, and thus improves the sustainability of such a system. On the other hand, the standard FSTSP model does not have the payload management constraints to govern the sequence of customers on each drone route needed for our approach. These constraints are a key contribution of this paper, as without them the drone route could be infeasible, for example, by attempting to make a return followed by two deliveries on the same sortie and thus exceeding the drone's payload capacity (Fig. 1(c)).

In this paper, we make the following contributions to the sustainable logistics field.

- Considering return tasks to improve payload efficiency and sustainability while increasing customer satisfaction.
  Traditional TSPD/FSTSP does not use a drone's payload capacity to its full capability since pickup/return is not considered. The drone always rendezvouses with the truck with an empty payload. By considering return tasks, we can increase the drone utilization as well as improve customer satisfaction since return customers can be served in this model. There is no need for a customer to make a trip to submit the return at a postal center nor for a truck to make a repeat visit to the customer to pick up the return, thereby reducing overall negative environmental impact of parcel returns.
- Develop a VNS based meta-heuristic to solve instances of reasonable size.
  Because the mathematical model can only be solved exactly for quite small-size instances, we rely on a meta-heuristic and because of the strong neighborhood structure of our combinatorial problem, a variation of VNS is well suited.
- Exploring the benefits of distinguishing customer types: delivery and return customers.
  We utilize the drone by considering both delivery and return customers and conduct experiments to determine the advantage of such a model against the one that does not distinguish customer types. We also explore the effects of the proportion of return customers among total customers.
- Exploring the advantage of using a drone with multiple payloads over drones with a single payload.
  We conduct experiments to determine the advantage of having one drone with multiple payloads versus multiple drones with a single payload under the same circumstances.
- Exploring combined savings of considering multiple payloads and return customers.
  The FSTSP-DR-MP model is derived from the standard FSTSP and TSP models. We compare our solutions with these to assess the improved performance in terms of total service time and total truck travel time (a proxy for total emissions) savings when considering multiple payloads and return customers.
- Assessing performance of valid inequalities to improve the runtimes for the FSTSP model.
  Our model is an NP-hard problem since it derives from both FSTSP and TSP models. We introduce valid inequalities to the model to substantially reduce runtime of the mathematical model.

The remainder of this paper is organized as follows. The related literature is presented in Section 2. A formulation of the FSTSP-DR-MP model as a mixed integer linear program (MILP) is given in Section 3, while analysis of the optimal solutions is presented in Section 4. The FSTSP-DR-MP meta-heuristic approach for large-size instances and its performance are described in Sections 5 and 6, respectively. Finally, conclusions and future research implications are presented in Section 7.

2. Related literature

There is a wide body of literature for pickup and delivery problems which Parragh et al. (2008a,b) classifies into two main categories. The first is the vehicle routing problem with backhauls (VRPB) where goods/parcels are transported from the depot to customers and picked up from customers for return to the depot. The second main problem is the vehicle routing problem with pickups and deliveries (VRPPD), in which goods/passengers are transported between pickup and delivery locations. For more details on the class definitions and classification, including mathematical formulations and solution methods, we recommend Parragh et al. (2008a,b), Lahyani et al. (2015), Sathya and Muthukumaravel (2015), Koç and Laporte (2018), Castro et al. (2019) and Koç et al. (2020). As there are extensive applications, both vehicle routing problems with simultaneous delivery and pickup (VRPSDP) and

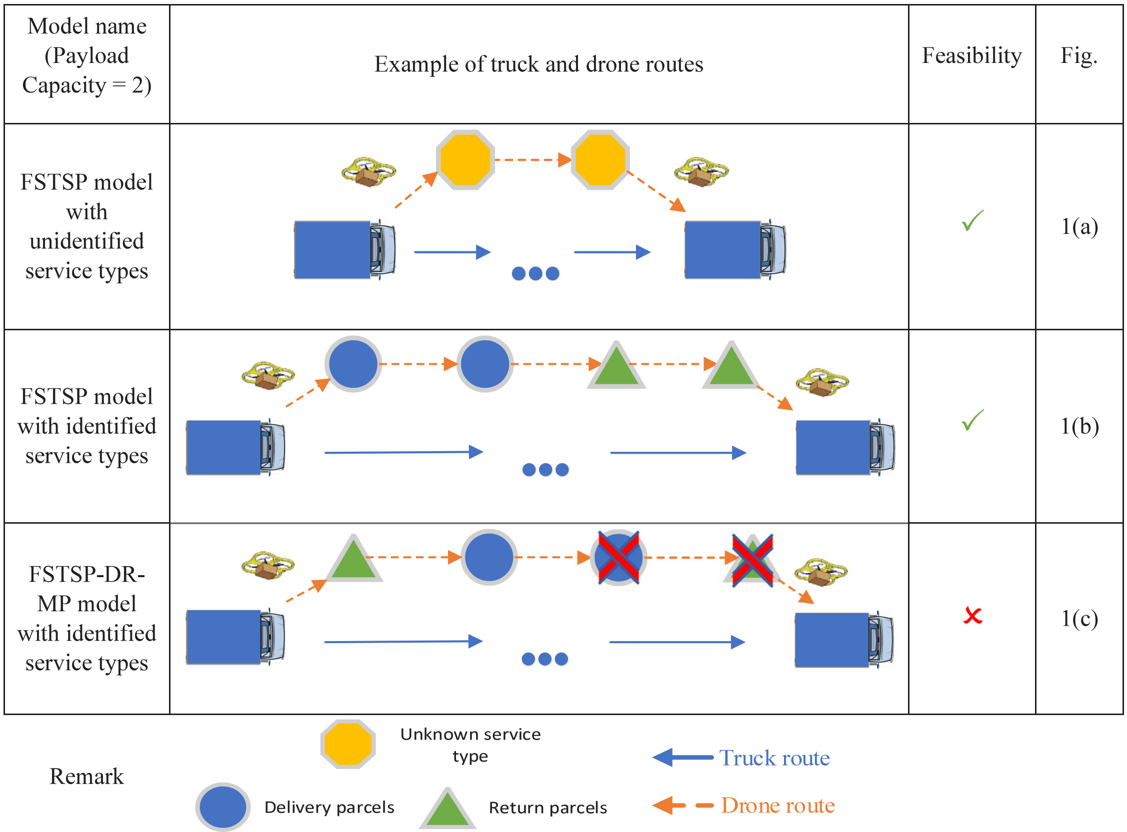

Fig. 1. FSTSP vs FSTSP-DR-MP with/without identifying service types.

traveling salesman problems with simultaneous delivery and pickup (TSPSDP) have attracted much research (Koç et al., 2020).

Our model falls into VRPB with the TSPSPD sub-class where all return parcels must return to the depot, however delivery tasks sometimes must be performed before return tasks when performed by the drone (to avoid exceeding drone capacity in each sortie). Regarding previous works on TSPSPD with drone(s), Ham (2018) proposes a model that considers multiple trucks and drones with multiple depots, with the goal of minimizing total service time. If the drones serve more than one customer, they must perform delivery before pickup, which is similar to our proposed concept in that it includes precedences. However, different from our approach, the drone serves customers by launching from and returning to depots with no interaction with trucks, thus lacking the coordination aspect which is integral to our model. Karak and Abdelghany (2019) propose a hybrid vehicle-drone routing model for return and delivery services. The model consists of one truck and multiple drones and is formulated as a mixed-integer program to minimize the vehicle-drone routing cost. However, they consider a truck as only a carrier of drones and packages, that is, the truck cannot have any direct interaction with customers. Moreover, the drones can only be launched at designated stations rather than from customer locations or the depot. Their model differs from ours in another respect, that is their objective function minimizes operation cost while our objective function minimizes service time, obviating the difficult task of assigning cost structures to the various components of the last mile delivery and return tasks. Lu et al. (2022) propose a vehicle-drone routing model for humanitarian logistics with pickup and delivery. The goal is to minimize total service time and maximize delivered parcels. They consider multiple trucks and drones where each drone can perform multiple visits in each sortie. However, the drone cannot perform pickup tasks. The model of Gacal et al. (2020) consists of one truck and one drone with backhauls. The goal is to minimize costs while serving all customers. However, they only solve small-size instances, 10 nodes, without proposing any alternative for large-size instances. Additionally, the drone in their work can only carry one parcel at a time, while we allow for multiple payloads; and they do not allow a customer to request both delivery and return services.

Incorporating trucks and drones is a topic that has drawn much recent attention from researchers. However, most of them are dominated by the TSP with drone(s) with delivery only. According to the surveys of Otto et al. (2018) and Macrina et al. (2020), models can be classified into two categories: the flying sidekick traveling salesman problem (FSTSP) and the parallel drone scheduling traveling salesman problem (PDSTSP); both consider delivery operations only. In light of our proposed model, only the FSTSP will be discussed. Seminal work in this area includes Ferrandez et al. (2016) and Arishi et al. (2022), who consider a truck which carried a single or multiple drones with drone-delivery clusters determined by using K-means clustering and deep reinforcement machine learning, respectively. Each drone delivers one parcel to a customer and then returns to the truck. On the other hand, Ha et al. (2018) modifies the FSTSP model by minimizing total operational costs. They propose two heuristics to solve the min-cost FSTSP. Agatz et al. (2018) propose a similar model, and the truck-drone routes are generated by the method of first-cluster, second-heuristic using a greedy partitioning heuristic and an exact partitioning algorithm. Chang and Lee (2018) propose multiple-trip drone delivery with the objective to determine the delivery routes for truck and drones to minimize the total service time. They employ a technique involving shifting the center of a cluster after applying a K-means clustering algorithm.

Gonzalez-R et al. (2020) consider a drone sortie that can serve more than one customer. Their model uses mixed integer programming and the solution is obtained by a branch-and-cut algorithm with a heuristic approach. Luo et al. (2021) also consider similar assumptions, but they include multiple drones in their model. Sacramento et al. (2019) and Lei et al. (2022) extend the FSTSP model from Murray and Chu (2015)

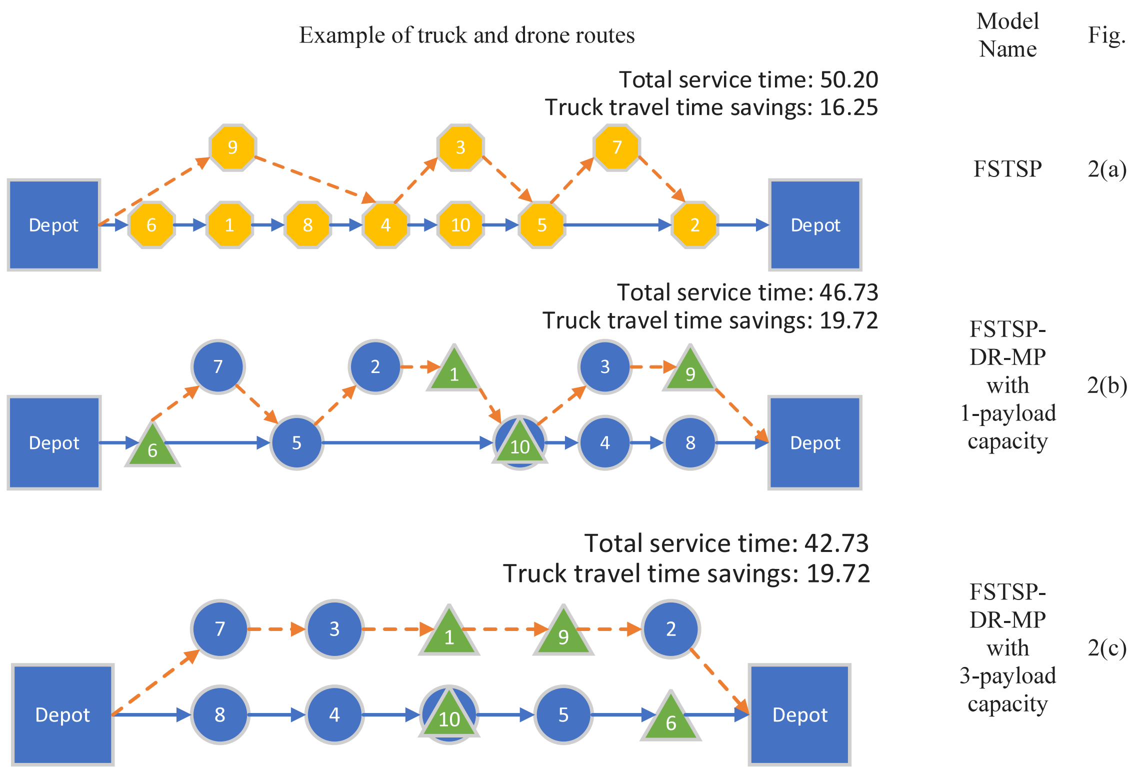

Fig. 2. Examples of truck-drone operation for the FSTSP-DR-MP.

by considering multiple capacitated trucks where minimizing cost is the objective function. Murray and Raj (2020) propose another model with one truck and multiple drones where minimizing total service time is the objective function. They add the following variants: (1) considering no truck driver's interaction with drones at the depot with the launching and rendezvous processes, and (2) automated launch and recovery systems for drones. Then, Raj and Murray (2020) extend the model by implementing the concept of queueing the drones in both the launch and retrieval phases by adding a scheduling problem which includes the effects of multiple drone speeds for different payload weights. Moshref-Javadi et al. (2020a) also propose a sidekick problem with one truck and multiple drones. They focus on reducing wait time of customers instead of minimizing total service time. They extend this work in Moshref-Javadi et al. (2020b) to allow the drones to rendezvous with the truck at same launch location. Thus, the truck can act as a mobile depot and launch the drones multiple times until all customers in the drone range are served. Also, Bruni et al. (2022) provide another approach to solve one truck with the multi-drone model from Moshref-Javadi et al. (2020b) by using a logic-based Benders decomposition method. Finally, Poikonen and Golden (2020) propose a k-multi-visit drone delivery model consisting of one truck and multiple drones. Each drone can perform multiple stops, but their paper does not differentiate between return and delivery customers, thereby ignoring precedences. Moreover, only drones can serve customers, whereas in our model both the truck and drone can interact with customers, which is more flexible, and we believe, more realistic. Note that all the truck and the drone operations in the aforementioned papers consider delivery only.

In summary, there is a serious research gap concerning the coordinated planning of parcel return and delivery employing drone-truck tandems. Yet, this situation is one that should be expected in many last mile delivery scenarios where both operational efficiency and environmental sustainability are targeted. Therefore, this new version of the problem that we formulate and solve herein is important and realistic.

3. Model formulation for FSTSP-DR-MP

Differentiating customers depending on whether they need delivery or return service enables the drone to potentially perform more stops during each sortie. Consider the 10-customer example shown in Fig. 2 below. The FSTSP model in Fig. 2(a) which does not combine delivery and return has an optimal total service of 50.20 min, while reduce tuck travel time by 16.25 min. Separating the customers by type, the drone now has the capability to perform return immediately after it delivers a parcel to another customer in one sortie, as illustrated in Fig. 2(b). The service time with the FSTSP-DR-MP model with a 1-payload capacity reduces the optimal total service time to 46.73 min and increases the savings for truck travel time to 19.72 min. By considering a 2-payload capacity drone as in Fig. 2(c), more customers can be served on the same flight, reducing the optimal total service time to 42.73 min, which is 9% lower than the standard FSTSP model while maintaining the same truck travel time savings, for this example.

With a greater payload capacity, there is more computational complexity since the algorithm must consider the feasibility of each sortie on sequences of return and delivery tasks to avoid exceeding payload capacity. Therefore, we must consider both battery life and drone payload capacity when planning routes.

3.1. Assumptions, model description, and notation

We make the following assumptions about the system.

- There is one truck, one drone, and one depot.
- The objective is to minimize the latest arrival time of the truck and drone at the depot after serving all customers (i.e., the makespan).
- The speeds of the truck and drone are known and constant.
- The drone flight endurance is known and fixed in terms of travel time.
- The truck has unlimited capacity while the drone can carry only P parcels at a time. Note that drone parcels are homogeneous, that is, interchangeable in weight, shape, and size.

- The drone can only serve a subset of customers due to considerations such as weight, security, and handling of certain parcels (for both return and delivery). Around 90% of parcels in a logistics system can be delivered/returned by drone (Guglielmo, 2013).
- All drone eligible customers can be served by either the truck or the drone.
- Some customers can request both delivery and return services.
- Each customer is only served once.
- The drone can be launched from and rendezvous with the truck at any customer location and at the depot.
- For each drone sortie, launch and rendezvous locations cannot be in the same location.
- The drone must depart before the truck at every launch location except the starting depot for safety purposes.
- If the drone arrives before the truck at the rendezvous location, it must hover in the air for safety purposes (thus expending battery), except if the rendezvous is at the depot.
- The drone uses the same amount of energy when it has one or more payloads or is empty and also whether it flies or hovers.
- The drone battery is replaced before each launch from the truck to ensure full range of flight on each mission.
- The drone must leave customer locations immediately after delivering or picking up parcels.

The notation, sets, parameters, and decision variables used in the proposed mathematical model are presented as following.

Definition of sets

N:	All nodes
N_0:	All nodes except the final location
N_+:	All nodes except the starting location
C:	All customers
C':	All drone eligible customers
D:	All customers who request delivery service
D':	All drone eligible customers who request delivery service
R:	All customers who request return service
R':	All drone eligible customers who request return service

Definition of parameters

τ_{ij}:	Truck travel time from location i to j where $(i,j) \in N$, Manhattan distance
τ'_{ij}:	Drone travel time from location i to j where $(i,j) \in N$, Euclidean distance
S_R:	Drone retrieval time at any rendezvous location in N_+
S_L:	Drone launch time at any launch location in N_0
en:	Maximum endurance time for each drone sortie
P:	Maximum payloads that the drone can carry at a time
M_1:	A large number, greater than the largest possible service time
M_2:	An integer number greater than P
c:	The total number of customers

Definition of decision variables

x_{ij}:	Indicates whether the truck travels along arc $(i,j) \in N$	Binary
y_{ij}:	Indicates whether the drone travels along arc $(i,j) \in N$	Binary
z_{ik}:	Indicates whether the drone launches at location i and rendezvouses with the truck at location k where $(i,k) \in N$	Binary
t_i:	Departure time of the truck at location $i \in N$	Continuous
t'_i:	Departure time of the drone at location $i \in N$	Continuous
da_i:	Indicate whether the drone is ready to create a new sortie at location $i \in N$	Binary
cp_i:	Indicates how many times the drone performs return tasks up until customer $i \in N$ in the current drone sortie	Integer
cd_i:	Indicates how many times the drone performs delivery tasks up until customer $i \in N$ in the current drone sortie	Integer
cpd_i:	Indicates payload availability status at customer $i \in N$	Integer

Mathematical model

$$\textit{Objective} \qquad \text{Min } t_{c+1} \tag{1}$$

Flow balance constraints

$$\sum_{\substack{i\in N_0\\ i\neq j}} x_{ij} + \sum_{\substack{i\in N_0\\ i\neq j}} y_{ij} = 1 + \sum_{\substack{i\in N_0\\ i\neq j}} z_{ij} \quad ;\forall j \in C \tag{2}$$

$$\sum_{j\in D} x_{0,j} + \sum_{\substack{r\in R\\ r\notin D'}} x_{0r} + x_{0,c+1} = 1 \tag{3}$$

$$\sum_{j\in D} x_{j,c+1} + \sum_{\substack{r\in R\\ r\notin D'}} x_{r,c+1} + x_{0,c+1} = 1 \tag{4}$$

$$\sum_{\substack{i\in N_0\\ i\neq j}} x_{ij} = \sum_{\substack{k\in N_+\\ k\neq j}} x_{jk} \quad ;\forall j \in C \tag{5}$$

$$\sum_{\substack{j\in N_+\\ j\neq i}} y_{ij} \le 1 \quad ;\forall i \in N_0 \tag{6}$$

$$\sum_{\substack{i\in N_0\\ i\neq j}} y_{ij} \le 1 \quad ;\forall j \in N_+ \tag{7}$$

$$y_{ii} = 0 \quad ;\forall i \in N \tag{8}$$

$$\sum_{\substack{k\in N_+\\ k\neq i}} z_{ik} \le 1 \quad ;\forall i \in N_0 \tag{9}$$

$$\sum_{\substack{i\in N_0\\ i\neq k}} z_{ik} \le 1 \quad ;\forall k \in N_+ \tag{10}$$

$$\left(\sum_{\substack{j\in N_+\\ j\neq i}} x_{ij} + \sum_{\substack{l\in N_+\\ l\neq i}} y_{il}\right) - 1 \le \sum_{\substack{k\in N_+\\ k\neq i}} z_{ik} \qquad ;\forall i \in N_0 \tag{11}$$

$$\left(\sum_{\substack{i\in N_0\\ i\neq k}} z_{ik}\right) \le \sum_{\substack{j\in C\\ j\neq k}} y_{jk} \quad ;\forall k \in N_+ \tag{12}$$

$$\sum_{\substack{j\in N_0\\ j\neq k}} y_{jk} = \sum_{\substack{i\in N_0\\ i\neq k}} z_{ik} \quad ;\forall k \in \{N_+; k \notin C'\} \tag{13}$$

$$2z_{ik} \le \sum_{\substack{h\in N_0\\ h\neq i,k}} x_{hi} + \sum_{\substack{l\in C\\ l\neq k}} x_{lk} \quad ;\forall i \in C, k \in \{N_+ : k \neq i\} \tag{14}$$

$$z_{0,k} \le \sum_{\substack{h\in N_0\\ h\neq k}} x_{hk}; k \in N_+ \tag{15}$$

Drone availability constraints

$$2 - x_{ij} - \sum_{\substack{k\in N_+\\ i\neq k}} z_{ik} \geq da_j - \sum_{\substack{l\in N_0\\ l\neq j,i}} y_{lj} \quad ;\forall i \in N_0, j \in \{N_+ : j \neq i\} \tag{16}$$

$$da_j \leq da_i + \left(1 - x_{ij} + \sum_{\substack{l\in N_0\\ l\neq j}} z_{lj} + \sum_{\substack{k\in N_+\\ k\neq i,j}} y_{ik}\right) \quad ;\forall i \in N_0, j \in N_+ \tag{17}$$

$$da_j \geq da_i - \left(1 - x_{ij} + \sum_{\substack{l\in N_0\\ l\neq j}} z_{ljb} + \sum_{\substack{k\in N_+\\ k\neq i,j}} y_{ikb}\right) \quad ;\forall i \in N_0, j \in N_+ \tag{18}$$

$$\sum_{\substack{k\in N_+\\ i\neq k}} y_{ik} \leq da_i \quad ;\forall i \in N_0 \tag{19}$$

$$da_0 = 1 \tag{20}$$

$$z_{c+1,k} = 0 \quad ;k \in N \tag{21}$$

$$z_{ii} = 0 \quad ;\forall i \in N \tag{22}$$

Time constraints

$$t'_i \geq t_i - M_1\left(1 - \sum_{\substack{k\in N_+\\ k\neq i}} z_{ik}\right) \quad ;\forall i \in C \tag{23}$$

$$t'_i \leq t_i + M_1\left(1 - \sum_{\substack{k\in N_+\\ k\neq i}} z_{ik}\right) \quad ;\forall i \in C \tag{24}$$

$$t'_k \geq t_k - M_1\left(1 - \sum_{\substack{i\in N_0\\ i\neq k}} z_{ik}\right) \quad ;\forall k \in \{N_+ : k \neq c+1\} \tag{25}$$

$$t'_k \leq t_k + M_1\left(1 - \sum_{\substack{i\in N_0\\ i\neq k}} z_{ik}\right) \quad ;\forall k \in N_+ \tag{26}$$

$$t_k \geq t_h + \tau_{hk} + \sum_{\substack{i\in N_0\\ i\neq k}} z_{ik} S_R + \sum_{\substack{m\in N_+\\ m\neq k,h}} z_{km} S_L - M_1\left(1 - x_{hk}\right) \quad ;\forall h \in N_0, k \in \left\{N_+ : k \neq h\right\} \tag{27}$$

$$t'_k \geq t'_j + \tau'_{jk} + \sum_{\substack{i\in N_0\\ i\neq j,k}} z_{ik} S_R + \sum_{\substack{m\in N_+\\ m\neq j,k}} z_{km} S_L - M_1\left(1 - y_{jk}\right) \quad ;\forall j \in N_0, k \in \{N_+ : k \neq j\} \tag{28}$$

Payload management constraints

$$cd_j \geq cd_i + 1 - M_2\left(1 - y_{ij} + \sum_{\substack{l\in N_0\\ l\neq i,j}} z_{lj}\right) \quad ;\forall i \in N_0, j \in \left\{D' : j \neq i\right\} \tag{29}$$

$$cp_j \geq cp_i - M_2\left(1 - y_{ij} + \sum_{\substack{l\in N_0\\ l\neq i,j}} z_{lj}\right) \quad ;\forall i \in N_0, j \in \{D' : j \neq i\} \tag{30}$$

$$cp_r \geq cp_i + 1 - M_2\left(1 - y_{ir} + \sum_{\substack{l\in N_0\\ l\neq i,r}} z_{lr}\right) \quad ;\forall i \in N_0, r \in \{R' : r \neq i\} \tag{31}$$

$$cd_r \geq cd_i - M_2\left(1 - y_{ir} + \sum_{\substack{l\in N_0\\ l\neq i,r}} z_{lr}\right) \quad ;\forall i \in N_0, r \in \{R' : r \neq i\} \tag{32}$$

$$cd_k \leq M_2\left(1 - \sum_{\substack{i\in N_0\\ i\neq k}} z_{ik}\right) \quad ;\forall k \in N_+ \tag{33}$$

$$cp_i \leq M_2\left(1 - \sum_{\substack{i\in N_0\\ i\neq k}} z_{ik}\right) \quad ;\forall k \in N_+ \tag{34}$$

$$cpd_i \leq M_2\left(1 - \sum_{\substack{i\in N_0\\ i\neq k}} z_{ik}\right) \quad ;\forall k \in N_+ \tag{35}$$

$$cpd_i \geq cp_i - cd_i \quad ;\forall i \in C' \tag{36}$$

$$cpd_j \geq cpd_i - M_2\left(1 - y_{ij} + \sum_{\substack{l\in N_0\\ l\neq i,j}} z_{lj}\right) \quad ;\forall i \in C', j \in \{C' : j \neq i\} \tag{37}$$

$$M_2\left(P - cpd_j - cd_j\right) \geq y_{jk} - \sum_{\substack{i\in N_0\\ i\neq j,k}} z_{ik} \quad ;\forall j \in C', k \in \{D' : k \neq j\} \tag{38}$$

$$M_2\left(P - cp_j\right) \geq y_{jk} - \sum_{\substack{i\in N_0\\ i\neq j,k}} z_{ik} \quad ;\forall j \in C', k \in \{N_+ : k \neq j\} \tag{39}$$

Drone endurance constraint

$$t'_k - t'_i \leq en + M_1\left(1 - z_{ik}\right) + \sum_{\substack{m\in N_+\\ m\neq i,k}} z_{km} S_L \quad ;\forall i \in N_0, k \in \{N_+ : k \neq i\} \tag{40}$$

Range of decision variables

$$t_0 = 0 + \sum_{j\in N_+} z_{0,j} S_L \tag{41}$$

$$t'_0 \geq 0 + \sum_{j\in N_+} z_{0,j} S_L \tag{42}$$

$$da_i \in \{0,1\} \quad ;\forall i \in N \tag{43}$$

$$x_{ij} \in \{0,1\} \quad ;\forall i \in N_0, j \in \left\{N_+ : j \neq i\right\} \tag{44}$$

$$y_{ij} \in \{0,1\} \quad ;\forall i \in N_0, j \in \left\{N_+ : j \neq i\right\} \tag{45}$$

$$z_{ik} \in \{0,1\} \quad ;\forall i \in N_0, k \in \left\{N_+ : k \neq i\right\} \tag{46}$$

$$t_i \geq 0 \quad ;\forall i \in N \tag{47}$$

$$t'_i \geq 0 \quad ;\forall i \in N \tag{48}$$

$$0 \leq cd_i \leq P \quad ;\forall i \in N_0 \tag{49}$$

$$0 \leq cp_i \leq P \quad ;\forall i \in N_0 \tag{50}$$

$$0 \leq cpd_i \leq P \quad ;\forall i \in N_0 \tag{51}$$

3.2. Model definition

Since we focus on total service time, the objective is to minimize the final arrival time of the truck at the depot, shown in (1). Even though the arrival time of the drone does not appear in (1), it is bounded by constraints (25) and (26). Flow balance, drone availability, payload management, and time synchronization constraints are components of the solution approach.

As in a standard TSP, where all customers must be served, constraint (2) forces all customers to be served only once by either the truck or the drone. Constraints (3) and (4) force the truck to leave and enter the depot. Constraint (5) enforces flow balance for the truck at each customer location. The truck departure time at each location is governed by constraint (27).

Constraints (6) and (7) allow the drone to enter and leave each node only once, while constraint (8) prevents the drone from forming a small sub-tour with only one node. Constraints (9) and (10) allow only one drone launch or rendezvous at any customer location as well as at the depot. Furthermore, to be certain that the truck will be at the rendezvous locations to recover the drone, constraints (14) and (15) force the truck to arrive at those locations. Constraints (11) and (12) force the drone to leave and arrive at the truck once the drone route is assigned by z_{ik} where $i \in N_0$ and $k \in N_+$. These also govern the route of the drone that arrives and leaves at customer locations during the drone flight. Since not all customers can be served by drone, constraint (13) ensures that the drone only serves drone-eligible customers. When the drone is launched from the truck, constraints (23) and (24) force the truck and drone departure times from that location to be equal. Likewise, when the drone rendezvouses with the truck, constraints (25) and (26) force the departure times of the truck and drone to be equal. We also prevent the truck from launching the drone without having it by employing constraint (19) when $da_i = 0$. Hence, with constraint (20), we set $da_0 = 1$ to allow the truck to launch the drone immediately at the starting depot. If the drone is launched at node i, constraint (16) sets $da_j = 0$, for the immediate successor node of the truck to prevent the truck from launching the drone unless the drone rendezvouses with the truck at node j. We also retain the da_i value if there is no drone launch or rendezvous with the truck, using constraints (17) and (18). Also, we enforce constraint (21) to prevent the drone from launching again after it reaches the depot, and constraint (22) to prevent each drone sortie from allowing launch and rendezvous locations at the same location. The drone departure time at each customer location is governed by constraint (28) for delivery and return tasks along with launch and retrieve times (S_L and S_R) when the interaction between the truck and drone occurs.

Since multiple payloads are allowed, we must keep track of the current state of the payloads on each drone sortie to avoid infeasible solutions. We use the integer variables cp_i and cd_i to track the number of return and delivery tasks during each sortie, respectively, in constraints (29)–(32). We force the drone to rendezvous with the truck after the drone performs a return that fills the payload capacity limit with constraint (38). We also force the drone to rendezvous with the truck once a certain amount of delivery and return tasks has been completed with variable cpd_i using constraint (39). Integer variable cpd_i represents the fact that payload capacity is exhausted, that is, whenever cp_i is greater than cd_i, the drone will have less payloads for delivery tasks in that sortie, as in constraint (36). Constraint (37) prevents cpd_i from decreasing throughout each sortie. To illustrate the purpose of cpd_i, let us assume that the drone has a two-payload capacity. This drone could perform up to four stops since we could perform Delivery → Delivery → Return → Return. In this sortie, cpd_i is always zero at every customer since the number of return tasks never exceeds the number of delivery tasks. However, if the first stop of a drone sortie is a return task, then $cpd_i = 1$ at this customer, meaning that one unit of capacity of this drone has been used. Therefore, the drone has only one empty payload slot left to perform additional return and delivery tasks before returning to the truck. Also, constraints (33)–(35) reset cp_i, cd_i, and cpd_i to zero when the drone meets the truck at location $i \in N_+$. Constraint (40) allows the drone to be launched only when the planned travel time of the drone is less or equal to en. Constraints (41) and (42) set t_i and t'_i to 0 if the truck does not launch a drone before leaving the depot. Otherwise, constraint (41) sets t_i equal to S_L, and constraint (42) sets t'_i greater or equal to S_L since the drone does not need to leave the depot at the same time as the truck leaves. Constraints (43)–(51) define the range of the decision variables.

3.3. Valid inequalities

Valid inequalities for VRPD problems are found in Luo et al. (2021), Bruni et al. (2022), Tamke and Buscher (2021), and others which prove to be useful on countering the usual issues that arise in terms of runtimes for truck-drone coordination MILPs. Hence, in this paper, we introduce the following valid inequalities, based on those works.

$$t_{c+1} \geq \sum_{i \in N_0} \Big(\sum_{j \in N_+} x_{ij}\tau_{ij} + (S_R + S_L) \sum_{k \in N_+} z_{ik} \Big) \tag{52}$$

$$t_{c+1} \geq \sum_{i \in N_0} \Big(\sum_{j \in N_+} y_{ij}\tau'_{ij} + (S_R + S_L) \sum_{k \in N_+} z_{ik} \Big) \tag{53}$$

$$\sum_{k \in N_+} \sum_{i \in N_0} z_{ik} \leq \left\lceil \frac{c}{2} \right\rceil \tag{54}$$

Constraints (52) and (53) provide valid lower bounds. Constraint (52) represents the tour completion time assuming the truck never needs to wait for the drone, while (53) assumes that all drone routes are connected without relying on the truck as the carrier from the predecessor rendezvous location to its next launch location. This amounts to pretending that the drone never needs to hover and wait for the truck to arrive at a rendezvous location. Constraint (54) provides an upper bound for the total number of rendezvous.

4. Analysis of solutions from the MILP

To test the performance of the FSTSP-DR-MP model on improving sustainability in last mile logistics, we performed numerical experiments varying two factors: (1) percentage of return customers and (2) payload capacity. Like other drone-assisted logistics problems, the FSTSP-DR-MP model is NP-hard. Therefore, experiments cannot be conducted using the MILP with a realistic number of customers. Hence, in this section, we will concentrate on 10 small-size instances, each consisting of 10 customers, where we can obtain optimal solutions with the MILP. In Sections 5 and 6 we discuss the meta-heuristic approach which can readily solve large-size instances of up to 100 customers. Even though 10 customers do not represent a realistic number, these experiments identify which factors are statistically significant to be considered in the VNS approach with large-size instances. All experiments are conducted on a Windows 10 desktop PC with an Intel Xeon W-2295 3.0 GHz and 128 GB Ram. For software, we use the CPLEX API version 12.8 in Python version 3.6.

4.1. Considering percent of return customers and payload capacity

We performed experiments with 10 customers using data sets from Murray and Chu (2015) and classifying the customers into return or delivery. We also derive parameter values from Murray and Chu (2015) by setting the truck and drone speeds at 25 MPH and 35 MPH, respectively; drone endurance (en) is set to 40 min. The fixed times for launching (S_L) and retrieving (S_R) packages from the drone are 1 min each. On average, 85 percent of customers are eligible to receive services by the drone (C') (as in Murray and Chu (2015)). To test the percentage of return customers against the time savings, we compare solutions with 0%, 20%, and 40% return customers. Note that drone eligibility for each customer does not alter with changes in the percentage of return customers. We also tested the drone payload capacity at three levels which are 1, 2, and 3 payloads. To assess the performance of this model, we compare the results with the standard TSP and FSTSP solutions as presented in Table 1. The results reveal percentage savings in the total service time of FSTSP-DR-MP with respect to TSP of 22.6 to 32.5% and of 6.0 to 18.3% with respect to FSTSP, on average, when at least one return customer is included. We obtain optimal solutions for every scenario with an average runtime of 5 min. This implies that differentiating return service has the potential to reduce total service time compared to treating all services as deliveries. Moreover, we can identify the trend of reducing total truck travel time when integrating the drone and considering return services with respect to TSP and FSTSP of 29.9 to 36.2% and 6.2 to 22.9%, on average, respectively. Such improvements translate into sustainability advantages as trips are consolidated and the drone is handling more of the packages.

To analyze these results, we performed analysis of variance (ANOVA) to test the effects of payload capacity and percentage of returns on total service time and total truck travel time savings, as in Tables 2 and 3. The analysis of variance demonstrates that both factors are statistically significant at the 0.05 level of significance with *p-values* approaching zero. However, the F value of payload capacity is far larger, indicating this is the more significant factor. The interaction plots presented in Figs. 3 and 4 demonstrate a positive interaction effect between payload capacity and percentage of returns on both total service time and total truck travel time savings. However, as the levels

Table 1
Results of the percent return customers and payload capacity experiment compared to standard truck only system and drone-truck system that does not consider returns.

Payload capacity	Percentage of return customers	Percentage of total service time savings of FSTSP-DR-MP over:				Percentage of total truck travel saving of FSTSP-DR-MP over:			
		Standard TSP		FSTSP		Standard TSP		FSTSP	
		Mean	Max	Mean	Max	Mean	Max	Mean	Max
1	0%	17.9%	32.2%	0.0%	0.0%	26.9%	46.5%	0.0%	0.0%
	20%	23.2%	37.6%	6.5%	14.9%	31.4%	45.5%	6.2%	18.9%
	40%	22.6%	42.3%	6.0%	21.3%	29.9%	48.6%	15.2%	31.3%
2	0%	28.6%	42.3%	13.2%	22.1%	34.6%	48.6%	10.5%	24.3%
	20%	29.9%	45.1%	14.9%	22.8%	34.8%	49.6%	21.0%	32.6%
	40%	30.4%	45.5%	15.5%	26.7%	35.7%	57.1%	22.3%	36.7%
3	0%	29.9%	45.5%	14.9%	22.8%	35.8%	57.1%	12.5%	25.8%
	20%	31.5%	46.6%	16.9%	26.7%	35.0%	49.4%	21.2%	31.3%
	40%	32.5%	54.3%	18.3%	32.6%	36.2%	57.1%	22.9%	36.7%

Table 2
Analysis of variance based on total service time savings.

Source	DF	*F*-value	*P*-value
Model	53	82.86	0.000
Linear	13	322.73	0.000
Data	9	364.06	0.000
Return customers	2	39.19	0.000
Payloads	2	420.30	0.000
2-way interactions	40	4.90	0.000
Data*Return customers	18	2.51	0.009
Data*Payloads	18	6.99	0.000
Payloads*Return customers	4	6.26	0.001
Error	36		
Total	89		

Table 3
Analysis of variance based on total truck travel time savings.

Source	DF	*F*-value	*P*-value
Model	53	32.39	0.000
Linear	13	124.19	0.000
Data	9	165.70	0.000
Return customers	2	3.38	0.045
Payloads	2	58.19	0.000
2-way interactions	40	2.55	0.003
Data*Return customers	18	1.45	0.169
Data*Payloads	18	3.50	0.001
Payloads*Return customers	4	3.25	0.023
Error	36		
Total	89		

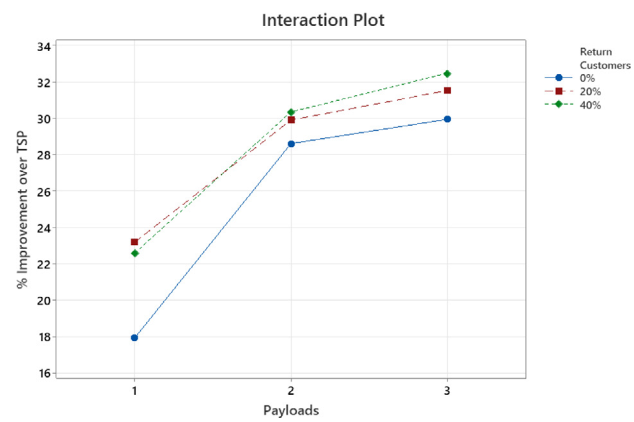

Fig. 3. Interaction plot for percentage improvement over TSP based on total service time savings.

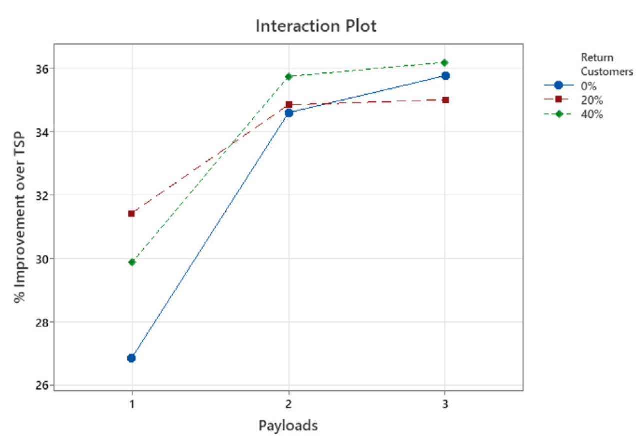

Fig. 4. Interaction plot for percentage improvement over TSP based on total truck travel time savings.

of both factors increase, there are diminishing returns. Note that the interaction between payload capacity and percentage of returns has a stronger effect on total truck travel time savings than on total service time savings, as expected. With more customers being served by the drone as both factors increase, the truck has fewer customers to visit in its route, which is not the case for total service time savings. Although the drone serves more customers as both factors increases, the drone travel time is still a part of total service time savings.

4.2. Performance of multiple payloads vs. multiple drones

We performed another experiment to test the performance differences between having multiple drones with a single payload versus one drone with multiple payloads. Note that we use the multi-visit traveling salesman problem with multi-drones (MTSP-MD) by Luo et al. (2021) for multiple drones with a single payload while altering their assumptions to match with ours as given in Section 3.1 and the parameters in Section 4.1. To compare, we use the FSTSP-DR-MP model with a payload capacity of 2, while the MTSP-MD consists of 2 drones with a single payload. Additional launch and rendezvous times (replace batteries and load/unload parcels) must be accounted for when the truck driver launches each drone. As a result, a single drone with multiple payloads can outperform multiple drones with a single payload by 3.9% on average, based on the results in Table 4.

4.3. Performance of valid inequalities

To assess the performance of the valid inequalities detailed in Section 3.3, we performed an experiment with the same assumptions as in Section 3.1 and parameters as in Section 4.1. We test instances with 40% return customers with a drone payload capacity of 3, setting a computational time limit of 24 h. The results in Appendix A, Table A.1

Table 4
Results of assessing performance of FSTSP-RP-MP vs. MTSP-MD.

Percentage of return customers	Average objective value for FSTSP-DR-MP	Average objective value for MTSP-MD	Performance improvement
0%	49.00	52.41	7.2%
20%	48.06	49.24	2.4%
40%	47.76	48.95	2.2%
		Average	3.9%

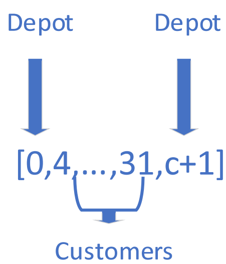

Fig. 5. Encoding of a truck solution.

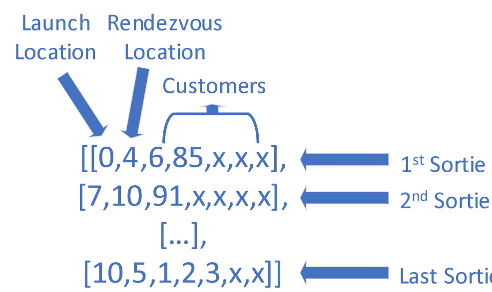

Fig. 6. Encoding of drone solutions.

clearly show that the valid inequalities assist the model to find the optimal solutions and proof of optimality within 5 min for most instances of 10 customers tested. Without valid inequalities, the upper bound for every instance reaches the optimal value; however, the lower bound does not improve, hence the model cannot offer proof of optimality for any instance within 24 h. This shows that the implemented valid inequalities are effective at reducing runtime for every instance tested, with slightly less impact on instance 10.

5. FSTSP-DR-MP meta-heuristic

The MILP formulation for FSTSP-DR-MP presented in Section 3 cannot solve realistically-sized instances within a reasonable computational time. To elaborate, the number of constraints and variables increase drastically with the number of customer locations, as seen in Appendix A, Table A.2. Thus, we devise a meta-heuristic for FSTSP-DR-MP, a variant the Variable Neighborhood Search (VNS). VNS is a well-known meta-heuristic for combinatorial problems with a strong neighborhood structure, as found in Freitas and Penna (2020), Schermer et al. (2019), Molina et al. (2020), and de Armas et al. (2015). A solution (s) to the FSTSP-DR-MP comprises two components: a truck solution (or a truck route), and drone solution (or a set of drone routes). We define the neighborhood of a given solution as any other solution that shares the same truck route. Therefore, the neighborhood includes all possible drone routes for a given truck route. After the meta-heuristic changes the neighborhood by creating a new truck solution, a local search meta-heuristic proceeds to explore different combinations of drone routes.

To replicate the decision variables from the MILP model, we designed a meta-heuristic that involves generating truck and drone routes. In the case of the truck route, we utilized a list data structure that captures the truck route, as illustrated in Fig. 5. The list has several columns, with the first column denoting the starting location, i.e., the depot, and the subsequent columns representing the intermediate locations that the truck must visit. The last column of the list represents the depot again, completing the Hamiltonian route. For drone routes, we use an array structure as in Fig. 6. Each row represents a drone sortie while the first and second columns represents launch and rendezvous locations, respectively. The remaining columns represents the sequence of customers of each drone sortie.

Initially, when only the truck route is generated, we can calculate its objective value by summing the travel times of each adjacent location in the truck solution. However, when the meta-heuristic generates a drone sortie, we must take into account the difference between the arrival times of the truck and the drone from the launch point to the rendezvous location. This information is used to determine which vehicle arrives last and when the drone will be ready to launch again. This is crucial for specifying the departure time of the next rendezvous location, as the drone must leave before the truck leave for safety purposes, as mentioned in Section 3.1.

We would like to emphasize that we only consider feasible solutions in our approach. For details on how to ensure the feasibility of drone routes, please refer to Section 5.1. Note that we do not need to ensure the feasibility for the truck route since its encoding (Fig. 5) inherently handles all subtours. The overall meta-heuristic is illustrated in Algorithm 1 with the subordinate heuristic presented in Algorithm 2.

The search starts by creating an empty set W for storing difficult-to-assign customers in later steps, and randomly separating customers into truck and drone groups ($DroneGroup$) with a ratio of φ_T customers in the truck group and φ_D customers in the drone group, as in line 2. Note that a customer cannot be in both groups, and all customers in the drone group must be drone eligible customers. We expect customers located further away from the depot to be served by the drone instead of the truck. Therefore, we rank all customers based on their distances from the depot and give the furthest 30% of drone eligible customers 50% more likelihood of being chosen for the drone group ($DroneGroup$).

After determining the customers in the truck and drone groups, we find a TSP solution (s) for all customers in the truck group using Concorde, a well-known TSP solver (Applegate et al., 1998). Then, we explore the neighborhood using move operators (line 3) by removing a customer uniformly at random from $DroneGroup$ to be inserted to a drone route to gain new solution (s) as shown in Algorithm 2. Once $DroneGroup$ becomes an empty set, it checks for remaining customers in W (line 4). If there are no customers in W, the algorithm proceeds to line 7 to compare the current solution (s) with the best-known solution (s^*). Otherwise, the meta-heuristic inserts the remaining customers from W in FIFO order, one by one, to the truck route wherever the insertion results in the minimum increase in total service time before proceeding to line 7.

The algorithm repeats the process again from line 2 until one of the termination criteria is met, which are (1) The number of iterations exceeds β iterations, or (2) There is no improvement on the best-known solution within $\frac{\beta}{4}$ iterations. If the termination criteria are met, the VNS proceeds to line 11 where it performs a mathematical optimization with an exact method (CPLEX) for the drone routes of the best-known solution. We set the truck route (x_{ij}), launch and rendezvous locations (z_{ik}), and departure time for the drone at each launch and rendezvous

Algorithm 1. FSTSP-DR-MP Main

Require: customers (C, C'); cost metrices (τ, τ'); drone endurance (en); launch and retrieval times (S_L, S_R); maximum payloads (P); total move operators (k_{max}); range of candidates for launch and rendezvous locations (ω); maximum iterations (β); difficult-to-assign customers (W); ratio of drone and truck customers (φ_D, φ_T)

```
1   repeat
2       s, DroneGroup ← ChangeNeighborhood(C, C′, τ, φ_D, φ_T)
3       s, W ← ExploreNeighborhood(s, W, τ, τ′, en, S_L, S_R, P, k_max, ω, DroneGroup)
4       if W ≠ ∅
5           s ← Insert customers from W to the truck route
6       end if
7       if f(s) ≤ f(s*) then
8           s* ← s
9       end if
10  until stop
11  s* ← ExactMethod(s*, τ′, en, P)
12  return s*
```

Algorithm 2. Explore Neighborhood

```
1   for all j ∈ DroneGroup do
2       if there is no existing drone route then
3           s ← Create a new drone route with customer j
4           if the heuristic fails to create a drone route with customer j then
5               Relocate customer j to W
6           end if
7       else
8           k ← 1
9           while k ≤ k_max do
10              if k == 1 then
11                  s′ ← Insert customer j to an existing drone route
12                  if the insertion is successful then
13                      s ← s′
14                      Continue
15                  end if
16              else if k == 2 then
17                  s′ ← Create a new drone route with customer j
18                  If the creating a drone route is successful then
19                      s ← s′
20                      Continue
21                  else
22                      Relocate customer j to W
23                  end if
24              else if k == 3 then
25                  for all i ∈ W do
26                      s′ ← Insert customer i to an existing drone route with relaxed rules
27                      if the insertion is successful then
28                          s ← s′
29                          Continue
30                      end if
31                  end for
32              end if
33              k ← k + 1
34          end while
35      end if
36  end for
37  return s
```

locations (t_i' and t_k') to have the same values of the best-known solution then solve the MILP presented in Section 3 for the remaining variables. Note that the customers on the truck route cannot be relocated; only drone customers can. The goal is to create a new solution by letting drone customers swap locations between drone routes and potentially reduce the total service time of the system. Then, the meta-heuristic terminates and returns the best solution (s^*).

It should be noted that the algorithm abandons the current solution (s) and starts a new iteration immediately whenever a new objective value ($f(s)$) from a recently accepted neighborhood is worse than the best-known solution ($f(s^*)$), in lines 2, 3 and 5.

The $ExploreNeighborhood()$ function is described in Algorithm 2. There are three move operators for assigning a candidate customer $j \in DroneGroup$ to be served by the drone. These are standard insertion ($k = 1$), creating a new drone route ($k = 2$), and insertion with relaxed rules ($k = 3$). Note that the order of move operators matters, and shuffling move operators is not required since inserting a customer into an existing drone route has less impact towards total service time than creating a new drone route. The first time this function is called, a drone route does not exist. So, only one option is available, which is creating a drone route with customer j (line 3). To create a drone route, the heuristic uses the selected customer location and includes the ω nearest locations on the truck route (including the depot) to the list of candidates for potential launch and rendezvous locations. (The number of such candidates, ω, is a settable parameter. A higher ω means more candidates are considered but requires more computation time.) The heuristic calculates the drone travel time from the candidate launch/rendezvous locations and the selected customer to create a new drone route. In line 3, the heuristic only accepts drone routes where the endurance constraint is not violated and the truck travel time from the launch location to the rendezvous location is less than or equal to the drone travel time from the launch location to the rendezvous location (to avoid having the truck waits for the drone). If the new drone route does not meet the criteria, the heuristic moves customer j to set W (line 4).

5.1. First neighborhood search

If at least one drone route exists, the heuristic proceeds to line 8. Starting with $k = 1$, the heuristic selects a drone route, following the

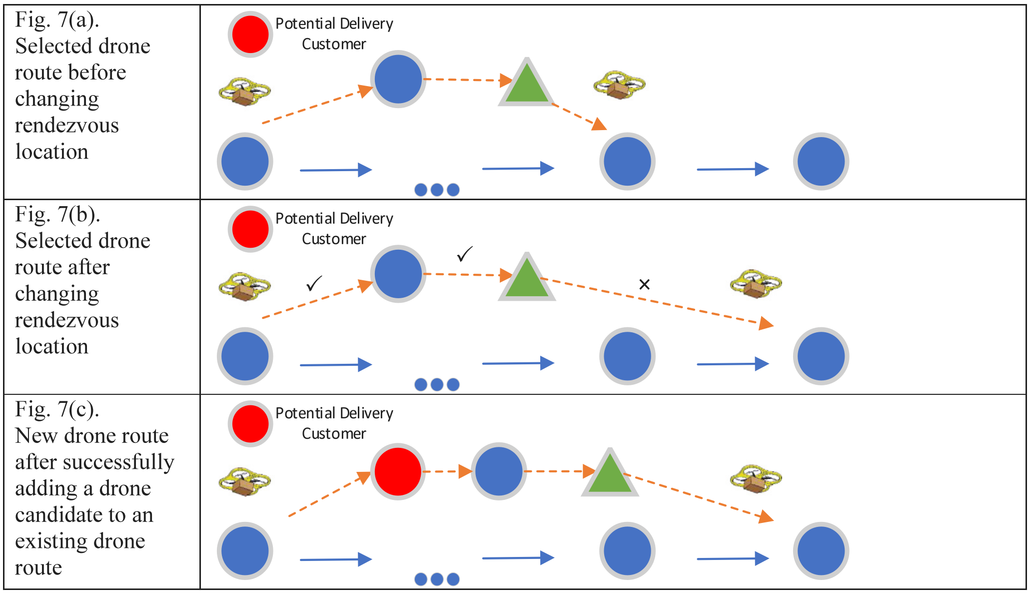

Fig. 7. Example of adding a potential delivery customer to an existing drone route, neighborhood type 1.

order that the drone routes were created and skipping those at full capacity. In the selected drone route, if the rendezvous location is not a launch location of another drone route, the current rendezvous location is moved to an adjacent customer in the truck route, as in Fig. 7(a). The purpose of the move is to allow for additional truck travel time to the rendezvous, so that the new drone travel time will be lower than the new truck travel time from launch to rendezvous after adding customer j to the drone route. Then, the algorithm checks for feasible insertion locations within this drone route, as illustrated in Fig. 7(b). Note that checkmarks represent feasible insertions according to the payload management constraints. Then, the heuristic tests inserting customer j in all feasible positions (considering endurance and payload capacity) along the drone route and selects the position that increases the drone travel time the least, as in Fig. 7(c). The new drone travel time must still be less than the new truck travel time from the launch location to the rendezvous location (line 11). If this criterion is not met, the rendezvous location maintains its original spot and repeats this step with the next drone route (in the order that the drone routes were created) until customer j is added to a drone route or all routes are exhausted without a feasible insertion. If customer j has been included on an existing drone route (line 12), the solution is updated. Otherwise, k is increased (line 33).

5.2. Second neighborhood search

If $k = 2$ (line 16), a new drone route with customer j is created following the procedure detailed above (line 3), except that the acceptance criteria are relaxed to accept an insertion as long as the drone does not violate the endurance constraint, and the truck travel time to the rendezvous does not exceed the drone travel time by a ratio of $\frac{Dronespeed}{Truckspeed}$. This means a small amount of truck wait time is acceptable. If customer j is successfully added to an existing drone route without violating acceptance criteria (line 18), the heuristic proceeds to update the solution. Otherwise, since customer j was not added to an existing route nor to a newly established drone route, it moves customer j to W (difficult to schedule customers) (line 22) and continues to increase the k value (line 33).

5.3. Third neighborhood search

Finally, if $k = 3$ (line 24) the algorithm attempts to insert customers in W to existing drone routes, selecting customers by FIFO order and following the same procedure as in Section 5.1. This process gives customers that could not be added before to a drone route a final opportunity. Here, the acceptance criteria are relaxed to accept the move when the drone does not violate the endurance constraint nor the payload constraints, and the truck travel time to the rendezvous does not exceed the drone travel time by a maximum of a ratio of $\frac{Dronespeed}{Truckspeed}$, as in line 11. Note that if there has been no change in drone routes since the previous attempt in line 24, the algorithm will only process the most recent customer from W to save runtime.

5.4. Time complexity

We estimate the time complexity of our algorithm by analyzing each step's time complexity as follows. Starting with Algorithm 1, changing the neighborhood (line 2) has a time complexity $O(|C|)$. Then, the meta-heuristic moves to line 3 and explores the neighborhood using the heuristic in Algorithm 2, which consists of three neighborhood searches. The first neighborhood search in Algorithm 2 has a time complexity of $O(|C|P)$. The second neighborhood search has a time complexity of $O(|C|\omega)$, while the third neighborhood search has a time complexity of $O(W)$. After exploring the neighborhoods, the heuristic returns solutions to Algorithm 1 and verifies if there are remaining customers in W. If so, it proceeds to line 5, which has time complexity $O(W)$. Then, the meta-heuristic repeats the whole process again starting from line 2 of Algorithm 1 until the termination criteria is met, i.e., reaching maximum number of iterations (β) or no improvement over certain number of iterations ($\beta/4$).

Therefore, we can estimate the overall time complexity of this meta-heuristic as $O(\beta|C|) + O(|C|P\beta) + O(|C|\omega\beta) + O(W\beta) + O(W\beta)$. Because $|W| < |C|$, we can omit the $|W|$ from the equation and rephase the overall time complexity as $O(\beta|C|(P + \omega))$. Note that the exact method in line 11 in Algorithm 1 is exponential with problem size.

6. FSTSP-DR-MP meta-heuristic experimentation

In this section, computational experience is reported. First, we compare the meta-heuristic with the exact method on small-size instances to assess performance. Second, we investigate the effect of randomness in the meta-heuristic. Third, we assess performance on large-size instances. Fourth, we consider the best drone payload capacity for large-size instances. During preliminary experiments, we extensively tested various combinations of settable parameters for the

Table 5
Results for 10 customers averaged over 10 instances.

Payload capacity	Percentage of return customers	MILP average runtime (min)	FSTSP-DR-MP meta-heuristic average runtime (min)	Percentage of solution overages compared with MILP solutions
1	0%	2.4	2.7	4.7%
	20%	2.8	9.5	4.0%
	40%	1.2	2.9	4.5%
2	0%	2.1	3.0	10.7%
	20%	2.3	10.9	5.1%
	40%	2.9	2.8	4.5%
3	0%	2.3	3.7	5.5%
	20%	4.6	8.9	1.2%
	40%	18.9	3.3	2.8%
			Average	4.8%

meta-heuristic algorithm and identified the most suitable ones, as follows. We set ω to 4 and β to 16,000. These settings allow the algorithm to balance exploring potential launch and rendezvous candidates and to obtain a sufficient number of solutions with the maximum number of iterations. We also set φ_T from 30 to 80% and φ_D from 20 to 70%, respectively. We use this ratio because some customers must be on the truck route since the drone needs launch and rendezvous locations to create drone routes. All meta-heuristic experiments are coded in Python 3.6 and run on a Windows 10 Desktop PC with Intel Xeon W-2295 3.0 GHz and 128 GB Ram.

6.1. FSTSP-DR-MP meta-heuristic vs MILP model

The meta-heuristic is compared computationally with the MILP model for 10 instances of 10 customers with varying parameters for each instance. Table 5 shows that the VNS obtains solutions that are on average 4.8% worse than the corresponding MILP solution. There is a high percentage of solution overages when there are no return customers as expected since each drone sortie can hold fewer drone customers. On the other hand, we can identify a low percentage of solution overages, as low as 1.2%, when return customers and multiple payloads are considered. These results prove that the VNS works as intended for solving the FSTSP-DR-MP model where return customers and multiple payloads are considered. The VNS uses an average runtime of 5 min per instance which is the similar to the CPLEX runtime. However, based on the results, the longest runtimes for CPLEX and the meta-heuristic are 3 h and 18 min, respectively.

To assess the performance of the proposed meta-heuristic algorithm against the exact method, a goodness-of-fit testing was conducted. The detailed results can be found in Appendix A, Table A.3. For the computed χ^2 value of 24.559 for 89 degrees of freedom, the observed probability value (p-value) is greater than 0.995 (Walpole and Myers, 1978). Therefore, statistically, the effectiveness of the proposed meta-heuristic is not significantly different than the exact method.

6.2. Randomness of the FSTSP-DR-MP meta-heuristic

To investigate the effects of random seed on the meta-heuristic, we consider instances of 50 customers, a relatively large size. All assumptions and parameters remain the same as in Section 3.1 and Section 4.1, respectively, except that the percentage of return customers is set to 40% and a 3-payload capacity is used. Choosing these two parameters at their maximum levels enables more potential savings. Results can be found in Appendix A, Table A.4. The average mean absolute percentage error over all instances is 1.8%. Therefore, we can conclude that the FSTSP-DR-MP VNS is quite consistent relative to seed.

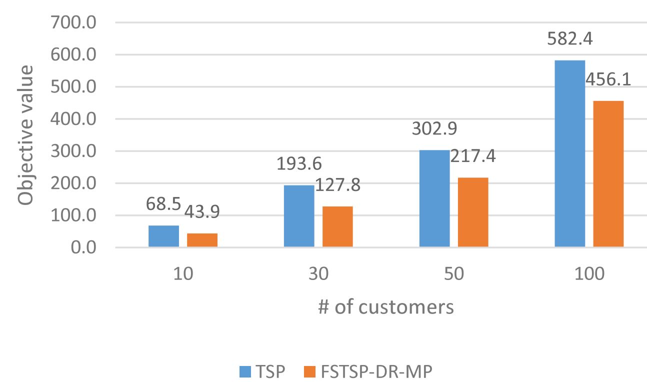

Fig. 8. Average total truck travel time of the pure TSP solution vs the FSTSP-DR-MP meta-heuristic with a 3-payload capacity and return ratio of 40%.

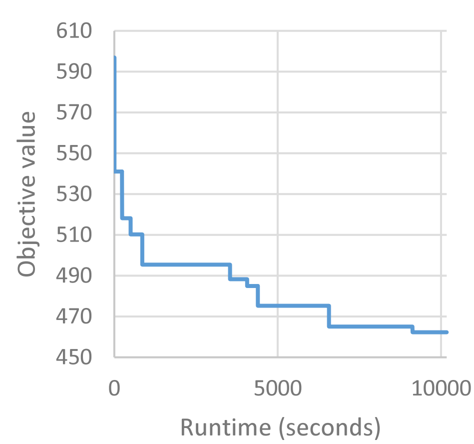

Fig. 9. Total service time progression against runtime of the FSTSP-DR-MP meta-heuristic.

6.3. Scale-up test of the FSTSP-DR-MP meta-heuristic

The number of customers was gradually increased to 100, to mirror the typical sizes of last mile delivery. Trucks from FedEx and UPS tend to perform around 100 stops per day (Kuo, 2018). All instances have the same customer density, and we consider 10 instances for each number of customers. It should be noted that all parameters are the same as in Section 6.1 except the drone endurance (en) has been increased to 60 min to match the problem size. For more information on data sets, please see Appendix B. We also compare our solutions with the pure TSP solution to evaluate the total service time savings varying with instance sizes. However, we cannot compare our solutions with the basic FSTSP model due to its limitation in solving large-size instances.

Table 6
Results of scale-up experiments using for a return ratio of 40%.

Number of customers	Average total service time savings of FSTSP-DR-MP over TSP		Average runtime (min)
	Percentage	Saving amount (min)	
10	30.3%	20.5	5.3
30	29.4%	57.0	20.0
50	22.0%	66.6	54.8
100	14.7%	85.8	193.4

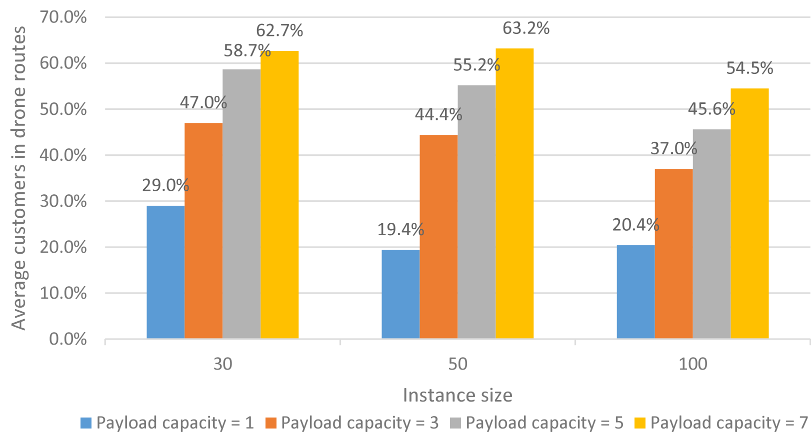

Fig. 10. Average number of customers served by drones of varying instance sizes and payload capacity.

Results are in Table 6 and Fig. 8; note that the meta-heuristic can solve instances of 100 customers within a reasonable runtime. This is workable in a typical last mile delivery operation which plans routes during an overnight or early morning preparation phase. While the service time savings increases with instance size, the percent of total service time savings and total truck travel time of the FSTSP-DR-MP over the TSP solution tends to decrease. This is consistent with Murray and Raj (2020)'s results on large-size instances but still results in large operational savings (not to mention the associated environmental benefits). As expected, runtime goes up linearly as customers increase by an order of magnitude from 10 to 100, as discussed in Section 5. However, if time is a serious constraint, the VNS can be terminated early to meet that constraint and still return a very good solution (see typical runtime progression in Fig. 9).

6.4. Payload capacity experiments with large-size instances

We investigated the effect of payload capacity on the percentage of customers served by drones for large-size instances. All assumptions and parameters are the same as Section 6.3 except the payload capacity. The results are shown in Figs. 10 and 11. For every instance size (number of customers), the greater the payload capacity, the higher the average percentage of customers that rely on drones, as expected. We can clearly identify the increase of the drone utilization and decrease in total service time due to the increasing payload capacity. However, the results show that even with a payload capacity of 7, the number of customers served by drones does not reach its maximum indicating further possible savings with larger drone capacities than are currently available. However, as payload capacity is increased, the average number of customers served by drones increases at a marginally diminishing rate. Because a drone does not use its additional payload capacity on every drone sortie, the total time savings of the system trends at a diminishing rate as well.

7. Conclusions and future research

As online purchases proliferate, so do the adverse environmental impacts of home or office delivery. A more sustainable option is to use efficient electrical vehicles such as drones. Drones have the additional advantages of being autonomous and being able to travel in direct, traffic free routes. When integrated with traditional delivery vehicles such as trucks or vans, the resulting system is both practical and more sustainable. Our approach adds the important but heretofore neglected aspect of returns. Returns are very common with online shopping (even more than with in person shopping) and they are problematic for customers and expensive for sellers. We propose a new model to integrate delivery and return operations, combining trucks and drones with multiple payload capacity, termed the Flying Sidekick Traveling Salesman Problem Integrating Deliveries and Returns with Multiple Payloads (FSTSP-DR-MP). This significantly expands the current last mile drone delivery literature by modeling and solving a mathematically complex but also very practical situation which combines both delivery and return customers.

We formulate this model as a mixed-integer programming model based on Murray and Chu (2015)'s FSTSP model with additional constraints to handle parcel return and delivery-return operations using drones, as well as drones with multiple payloads. While the model can be solved exactly for very small-size instances, we turn to meta-heuristics, namely the Variable Search Neighborhood or VNS paradigm, to solve situations of realistic size. This meta-heuristic can obtain solutions with a mean optimality gap of just 4.8% on 10-customer instances. Moreover, it is not very sensitive to randomness and can handle realistic problem sizes of up to 100 customers quite quickly.

We measure the performance of our MILP in terms of the percentage of total service time savings and total truck travel time, using both the standard TSP and the iconic FSTSP as benchmarks. Total time savings is highly correlated with total truck travel time, and this, in turn, is highly correlated with emissions and use of fossil fuels. The number of return customers and payload capacity of the drone affect the reduction in total truck operation time and total service time of the FSTSP-DR-MP when compared to these baselines, as expected. The results generated from the numerical experiments on 10-customer instances show that the FSTSP-DR-MP can reduce emissions from diesel (or gasoline) engines up to 36.2% and 22.9% when compared to TSP and FSTSP, respectively. Moreover, this occurs with an impressive

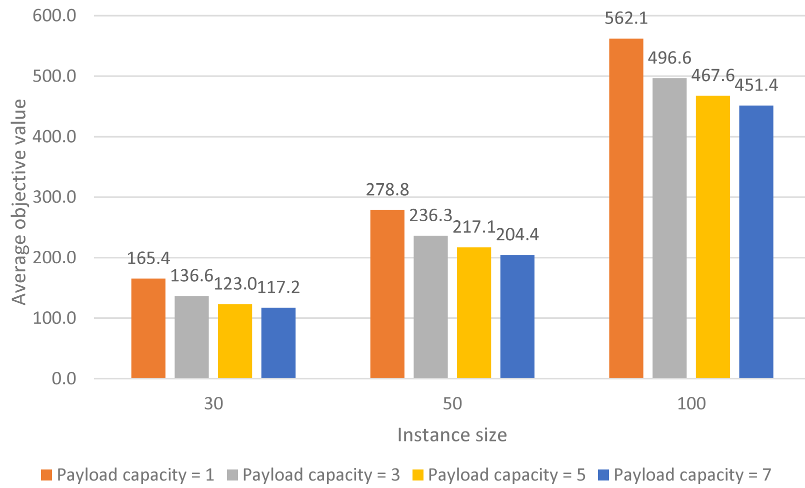

Fig. 11. Average total service time of varying instance sizes and payload capacity.

reduction of overall service times, up to 32.5% and 18.3%, respectively, compared to the TSP and FSTSP.

For future research, considering multiple trucks and drones is a natural extension. This increases the capability of improving total service time savings compared to the standard TSP and FSTSP. With multiple drones, we may need to consider drone scheduling management, such as air traffic, to avoid midair collision or congestion. Moreover, considering varying speeds for trucks and drones would enhance detailed planning. Not only could this potentially conserve fuel or battery life, but it might also maintain or even improve service time savings. Finally, considering additional objective(s) such as emissions or cost can be useful. We used time as the objective, because it is an absolute direct measure, whereas estimating cost or emissions are quite situation specific. As time is highly correlated with distance traveled by the truck and thus the negative environmental impact, we have also optimized implicitly for sustainability. But an explicit approach could be undertaken with the quantification of the environmental factors for the various delivery vehicles.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix A. Experimental results

See Tables A.1–A.4.

Table A.1
Results of an experiment on assessing the performance of valid inequalities.

Instance no.	With valid inequalities			Without valid inequalities		
	Optimal solution	Runtime (min)	Gap	Best-known solution	Runtime (h)	Gap
1	59.62	1.99	0%	59.62	24 h	100%
2	60.89	0.90	0%	60.89	24 h	100%
3	31.12	0.66	0%	31.12	24 h	73%
4	31.52	0.73	0%	31.52	24 h	100%
5	42.73	0.64	0%	42.73	24 h	80%
6	55.53	0.87	0%	55.53	24 h	72%
7	32.98	0.31	0%	32.98	24 h	74%
8	53.30	1.18	0%	53.30	24 h	64%
9	41.42	0.62	0%	41.42	24 h	80%
10	54.70	180.92	0%	54.70	24 h	100%

Table A.2
Number of variables and constraints based on total customers.

Total customers	Number of binary variables	Number of integer variables	Number of constraints
10 customers	576	48	2,513
30 customers	4,096	128	39,334
50 customers	10,816	208	157,949
100 customers	41,616	408	1,132,409

Table A.3
Experimental results on total service times by approach used in goodness-of-fit test.

Return customers	Payload capacity	Instance	CPLEX	Meta-heuristic	Computed χ^2-value
0%	1	1	70.19	75.69	0.430
		2	70.17	71.85	0.040
		3	42.90	44.26	0.043
		4	43.41	46.87	0.276
		5	50.20	50.47	0.001
		6	63.00	65.79	0.123
		7	48.93	56.60	1.201
		8	62.87	62.89	0.000
		9	48.31	48.31	0.000
		10	61.41	64.36	0.142
	2	1	62.57	68.10	0.488
		2	60.88	71.84	1.972
		3	34.36	40.13	0.970
		4	33.83	40.22	1.207
		5	46.73	46.73	0.000
		6	59.53	59.56	0.000
		7	41.61	50.61	1.946
		8	54.41	60.17	0.610
		9	41.41	41.42	0.000
		10	54.69	61.36	0.814
	3	1	60.57	66.10	0.504
		2	60.88	65.63	0.370
		3	34.36	34.72	0.004
		4	33.52	39.44	1.045
		5	44.73	44.73	0.000
		6	57.53	59.56	0.072
		7	39.35	40.60	0.039
		8	53.29	53.30	0.000
		9	41.41	41.42	0.000
		10	54.70	61.57	0.865
20%	1	1	66.44	68.20	0.047
		2	70.17	70.90	0.007
		3	40.73	42.18	0.052
		4	40.76	41.42	0.011
		5	46.73	47.00	0.002
		6	59.53	61.80	0.086
		7	45.00	47.01	0.089
		8	54.42	57.78	0.208
		9	46.60	46.60	0.000
		10	54.70	63.79	1.512
	2	1	60.58	60.58	0.000
		2	60.89	67.70	0.762
		3	34.36	36.60	0.146
		4	33.52	38.27	0.672
		5	44.73	44.73	0.000
		6	57.53	57.80	0.001
		7	39.62	39.62	0.000
		8	53.30	57.71	0.366
		9	41.42	41.42	0.000
		10	54.70	60.38	0.592
	3	1	59.62	59.93	0.002
		2	60.89	65.63	0.369
		3	31.83	31.83	0.000
		4	31.83	31.83	0.000
		5	42.73	42.73	0.000
		6	55.53	57.56	0.074
		7	38.50	38.50	0.000
		8	53.30	53.30	0.000
		9	41.42	41.42	0.000
		10	54.70	54.70	0.000

(*continued on next page*)

Appendix B. Data sets

We have provided relevant parameters and customer locations used in our experiments at the following link, https://github.com/winnerdd/DataSets.git. Please note that we could not share parameters and customer locations on 10-customer instances since we used data from the literature and have not received authorization from the owner of that data to share it. However, we have included a Python file for generating similar 10-customer instances in the given link.

Table A.3 (*continued*).

Return customers	Payload capacity	Instance	CPLEX	Meta-heuristic	Computed χ^2-value
40%	1	1	66.44	68.20	0.047
		2	70.17	71.85	0.040
		3	39.52	41.42	0.091
		4	39.45	41.42	0.098
		5	46.73	47.13	0.003
		6	59.53	59.82	0.001
		7	41.62	51.26	2.232
		8	60.50	61.93	0.033
		9	46.60	46.60	0.000
		10	59.61	61.36	0.051
	2	1	59.62	60.05	0.003
		2	60.89	70.66	1.569
		3	34.29	36.59	0.153
		4	31.83	31.83	0.000
		5	44.73	44.73	0.000
		6	57.53	58.76	0.026
		7	39.35	39.62	0.002
		8	53.30	57.71	0.366
		9	41.42	41.42	0.000
		10	54.70	60.20	0.554
	3	1	59.62	59.62	0.000
		2	60.89	65.63	0.369
		3	31.12	31.12	0.000
		4	31.52	31.83	0.003
		5	42.73	42.73	0.000
		6	55.53	57.56	0.074
		7	32.98	32.98	0.000
		8	53.30	57.55	0.340
		9	41.42	41.42	0.000
		10	54.70	59.02	0.342
				$\sum \chi^2$	24.559

Table A.4
Results of seed experiments of the FSTSP-DR-MP meta-heuristic.

Instance	Mean absolute percentage error (from mean of seeds)						
	Seed = 11	Seed = 23	Seed = 58	Seed = 88	Seed = 90	Range	Average
1	1.7%	1.1%	2.9%	2.2%	1.1%	1.1-2.9%	1.8%
2	1.7%	1.7%	0.6%	1.8%	2.2%	0.6-2.2%	1.6%
3	0.1%	0.6%	0.1%	0.3%	0.7%	0.1-0.7%	0.4%
4	4.6%	3.4%	1.0%	0.6%	0.7%	0.6-4.6%	2.0%
5	1.2%	0.3%	0.4%	2.1%	0.7%	0.3-2.1%	1.0%
6	0.6%	2.8%	0.8%	3.0%	0.2%	0.2-3.0%	1.5%
7	1.2%	0.3%	1.5%	1.1%	1.2%	0.3-1.5%	1.1%
8	0.2%	0.7%	2.6%	0.6%	1.2%	0.2-2.6%	1.1%
9	2.1%	2.7%	4.0%	4.6%	0.5%	0.5-4.6%	2.8%
10	0.9%	0.4%	5.4%	13.2%	5.2%	0.4-13.2%	5.0%
						Overall average	1.8%

References

Agatz, N., Bouman, P., Schmidt, M., 2018. Optimization approaches for the traveling salesman problem with drone. Transp. Sci. 52, 965–981.

Applegate, D., Bixby, R., Cook, W., Chvátal, V., 1998. On the solution of traveling salesman problems. Doc. Math. 3, 645–656.

Arishi, A., Krishnan, K., Arishi, M., 2022. Machine learning approach for truck-drones based last mile delivery in the era of industry 4.0. Eng. Appl. Artif. Intell. 116, 105439.

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

Boysen, N., Schwerdfeger, S., Weidinger, F., 2018. Scheduling last mile deliveries with truck-based autonomous robots. European J. Oper. Res. 271, 1085–1099.

Bruni, M.E., Khodaparasti, S., Moshref-Javadi, M., 2022. A logic-based benders decomposition method for the multi-trip traveling repairman problem with drones. Comput. Oper. Res. 145, 105845. http://dx.doi.org/10.1016/j.cor.2022.105845.

Carracedo, D., Mostofi, H., 2022. Electric cargo bikes in urban areas: a new mobility option for private transportation. Transp. Res. Interdiscip. Perspect. 16, 100705. http://dx.doi.org/10.1016/j.trip.2022.100705.

Castro, M.P., Cire, A.A., Beck, J.C., 2019. An MDD-based Lagrangian approach to the multicommodity pickup-and-delivery TSP. INFORMS J. Comput. 32 (2), 263–278.

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

Cheng, D., Charles, A.C., Srigrarom, S., Hesse, H., 2019. Morphing concept for multirotor UAVs enabling stability augmentation and multiple-parcel delivery. In: AIAA Scitech 2019 Forum. http://dx.doi.org/10.2514/6.2019-1063.

Coppola, D., 2021. Worldwide e-Commerce Share of Retail Sales 2015–2024. Statista, https://www.statista.com/statistics/534123/e-commerce-share-of-retail-sales-worldwide/?_ga=2.255428744.1043030823.1622629800-2138888546.1622629800. (Accessed 10 November 2022).

de Armas, J., Lalla-Ruiz, E., Expósito-Izquierdo, C., Landa-Silva, D., Melián-Batista, B., 2015. A hybrid GRASP-VNS for ship routing and scheduling problem with discretized time windows. Eng. Appl. Artif. Intell. 45, 350–360.

Dell'Amico, M., Montemanni, R., Novellani, S., 2021. Modeling the flying sidekick traveling salesman problem with multiple drones. Networks 78 (3), 303–327.

D'hont, J., Juwet, M., Demeester, E., Slaets, P., 2022. Development of an electric tricycle for service companies and last-mile parcel delivery. Transp. Probl. 17 (2), 175–187.

Dolan, S., 2021. The Challenges of Last Mile Delivery Logistics and the Tech Solutions Cutting Costs in the Final Mile. Business Insider, https://www.businessinsider.com/last-mile-delivery-shipping-explained. (Accessed 10 November 2022).

Ferrandez, S.M., Harbison, T., Weber, T., Sturges, R., Rich, R., 2016. Optimization of a truck-drone in tandem delivery network using K-means and genetic algorithm. J. Ind. Eng. Manag. 9, 374–388.

Freitas, J.C., Penna, P.H., 2020. A variable neighborhood search for flying sidekick traveling salesman problem. Int. Trans. Oper. Res. 27, 267–290.

Gacal, J.B., Urera, M.Q., Cruz, D.E., 2020. Flying sidekick traveling salesman problem with pick-up and delivery and drone energy optimization. In: 2020 IEEE International Conference on Industrial Engineering and Engineering Management. pp. 1167–1171.

Gonzalez-R, P.L., Canca, D., Andrade-Pineda, J.L., Calle, M., Leon-Blanco, J.M., 2020. Truck-drone team logistics: A heuristic approach to multi-drop route planning. Transp. Res. C 114, 657–680.

Guglielmo, C., 2013. Turns Out Amazon, Touting Drone Delivery, Does Sell Lots of Products That Weigh Less Than 5 Pounds. Forbes, 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 10 November 2022).

Ha, Q.M., Deville, H., Pham, Q.D., Hà, M.H., 2018. On the min-cost traveling salesman problem with drone. Transp. Res. C 86, 597–621.

Ham, A.M., 2018. Integrated scheduling of *m*-truck, m-drone, and *m*-depot constrained by time-window, drop-pickup, and *m*-visit using constraint programming. Transp. Res. C 91, 1–14.

Joerss, M., Schröder, J., Neuhaus, F., Klink, C., Mann, F., 2016. Parcel delivery: The future of last mile. Travel Transp. Logist. September.

Karak, A., Abdelghany, K., 2019. The hybrid vehicle-drone routing problem for pick-up and delivery services. Transp. Res. C 102, 427–449.

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

Koç, Ç., Laporte, G., 2018. Vehicle routing with backhauls: Review and research perspectives. Comput. Oper. Res. 91, 79–91.

Koç, Ç., Laporte, G., Tükenmez, I., 2020. A review of vehicle routing with simultaneous pickup and delivery. Comput. Oper. Res. 122, 104987. http://dx.doi.org/10.1016/j.cor.2020.104987.

Kuo, M., 2018. How do couriers like FedEx plan their routes? Routific Blog, https://blog.routific.com/blog/how-do-experts-like-fedex-plan-delivery-routes. (Accessed 20 March 2023).

Lahyani, R., Khemakhem, M., Semet, F., 2015. Rich vehicle routing problems: From a taxonomy to a definition. European J. Oper. Res. 241, 1–14.

Lei, D., Cui, Z., Li, M., 2022. A dynamical artificial bee colony for vehicle routing problem with drones. Eng. Appl. Artif. Intell. 107, 104510. http://dx.doi.org/10.1016/j.engappai.2021.104510.

Lin, C.K.Y., 2011. A vehicle routing problem with pickup and delivery time windows, and coordination of transportable resources. Comput. Oper. Res. 38, 1596–1609.

Lu, Y., Yang, C., Yang, J., 2022. A multi-objective humanitarian pickup and delivery vehicle routing problem with drones. Ann. Oper. Res. 319, 291–353.

Luo, Z., Poon, M., Zhang, Z., Liu, Z., Lim, A., 2021. The multi-visit traveling salesman problem with multi-drones. Transp. Res. C 128, 103172. http://dx.doi.org/10.1016/j.trc.2021.103172.

Macrina, G., Di Puglia Pugliese, L., Guerriero, F., Laporte, G., 2020. Drone-aided routing: A literature review. Transp. Res. C 120, 102762. http://dx.doi.org/10.1016/j.trc.2020.102762.

Matheson, R., 2017. Hybrid Drones Carry Heavier Payloads for Greater Distances. MIT News Office, https://news.mit.edu/2017/hybrid-drones-carry-heavier-payloads-greater-distances-0804. (Accessed 10 November 2022).

Molina, J.C., Salmeron, J.L., Eguia, I., Racero, J., 2020. The heterogeneous vehicle routing problem with time windows and a limited number of resources. Eng. Appl. Artif. Intell. 94, 103745. http://dx.doi.org/10.1016/j.engappai.2020.103745.

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

Moshref-Javadi, M., Lee, S., Winkenbach, M., 2020b. Design and evaluation of a multi-trip delivery model with truck and drones. Transp. Res. 136, 101887. http://dx.doi.org/10.1016/j.tre.2020.101887.

Murray, C.C., Chu, A.G., 2015. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. Transp. Res. C 54, 86–109.

Murray, C.C., Raj, R., 2020. The multiple flying sidekicks traveling salesman problem: Parcel. Transp. Res. C 110, 368–398.

Oliveira, L.K., Morganti, E., Dablanc, L., Oliveira, R.L.M., 2017. Analysis of the potential demand of automated delivery stations for e-commerce deliveries in Belo Horizonte, Brazil. Res. Transp. Econ. 65, 34–43.

Otto, A., Agatz, N., Campbell, J., Golden, B., Pesch, E., 2018. Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. Networks 41, 1–458.

Parragh, S.N., Doerner, K.F., Hartl, R.F., 2008a. A survey on pickup and delivery problems part I: Transportation between customers and depot. J. Betriebswirtschaft 58, 21–51.

Parragh, S.N., Doerner, K.F., Hartl, R.F., 2008b. A survey on pickup and delivery problems part II: Transportation between pickup and delivery locations. J. Betriebswirtschaft 58, 81–117.

Poikonen, S., Golden, B., 2020. Multi-visit drone routing problem. Comput. Oper. Res. 113, 104802. http://dx.doi.org/10.1016/j.cor.2019.104802.

Raj, R., Murray, C.C., 2020. The multiple flying sidekicks traveling salesman problem with variable drone speeds. Transp. Res. C 120, 102813. http://dx.doi.org/10.1016/j.trc.2020.102813.

Reagan, C., 2019. That Sweater You Don't Like Is a Trillion-Dollar Problem for Retailers. CNBC, https://www.cnbc.com/2019/01/10/growing-online-sales-means-more-returns-and-trash-for-landfills.html. (Accessed 10 November 2022).

Rodrigues, T.A., Patrikar, J., Oliveira, N.L., Matthews, H.S., Scherer, S., Samaras, C., 2022. Drone flight data reveal energy and greenhouse gas emissions savings for very small package delivery. Patterns 3, 100569. http://dx.doi.org/10.1016/j.patter.2022.100569.

Sacramento, D., Pisinger, D., Ropke, S., 2019. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. Transp. Res. C 102, 289–315.

Sathya, N., Muthukumaravel, A., 2015. A review of the optimization algorithms on traveling salesman problem. Indian J. Sci. Technol. 8, http://dx.doi.org/10.17485/ijst/2015/v8i1/84652.

Schermer, D., Moeini, M., Wendt, O., 2019. A hybrid VNS/Tabu search algorithm for solving the vehicle routing problem with drones and en route operations. Comput. Oper. Res. 109, 134–158.

Schermer, D., Moeini, M., Wendt, O., 2019a. A matheuristic for the vehicle routing problem with drones and its variants. Transp. Res. C 106, 166–204.

Tamke, F., Buscher, U., 2021. A branch-and-cut algorithm for the vehicle routing problem with drones. Transp. Res. B 144, 174–203.

Troise, D., 2021. Retail's online future grows brighter after pandemic bump. AP News. https://apnews.com/article/technology-shopping-coronavirus-p{and}emic-online-shopping-e-commerce-40008ad7099cad4d737c3ee7d156d56a. (Accessed 10 November 2022).

Ulmer, M.W., Streng, S., 2019. Same-day delivery with pickup stations and autonomous vehicles. Comput. Oper. Res. 108, 1–19.

Viloria, D.R., Solano-Charris, E.L., Muñoz-Villamizar, A., Montoya-Torres, J.R., 2021. Unmanned aerial vehicles/drones in vehicle routing problems: a literature review. Int. Trans. Oper. Res. 28, 1626–1657.

Walpole, R.E., Myers, R.H., 1978. Probability and Statistics for Engineers and Scientist, second ed. Macmillan Publishing Co. Inc.