

Joint optimization of customer location clustering and drone-based routing for last-mile deliveries

Mohamed Salama^a, Sharan Srinivas^{a,b,*}

^a Department of Industrial and Manufacturing Systems Engineering, University of Missouri, Columbia, MO 65211, United States

^b Department of Marketing, Trulaske College of Business, University of Missouri, Columbia, MO 65211, United States



ARTICLE INFO

Keywords:

Truck-drone routing
Last-mile delivery
Integrated optimization approach
Clustering delivery locations
Minimize total cost
Minimize delivery completion time

ABSTRACT

With growing consumer demand and expectations, companies are attempting to achieve cost-efficient and faster delivery operations. The integration of autonomous vehicles, such as drones, in the last-mile network design can curtail many operational challenges and provide a competitive advantage. This paper deals with the problem of delivering orders to a set of customer locations using multiple drones that operate in conjunction with a single truck. To take advantage of the drone fleet, the delivery tasks are parallelized by concurrently dispatching the drones from a truck parked at a focal point (ideal drone launch location) to the nearby customer locations. Hence, the key decisions to be optimized are the partitioning of delivery locations into small clusters, identifying a focal point per cluster, and routing the truck through all focal points such that the customer orders in each cluster are fulfilled either by a drone or truck. In contrast to prior studies that tackle this problem using multi-phase sequential procedures, this paper presents mathematical programming models to jointly optimize all the decisions involved. We also consider two policies for choosing a cluster focal point - (i) restricting it to one of the customer locations, and (ii) allowing it to be anywhere in the delivery area (i.e., a customer or non-customer location). Since the models considering unrestricted focal points are computationally expensive, an unsupervised machine learning-based heuristic algorithm is proposed to accelerate the solution time. Initially, we treat the problem as a single objective by independently minimizing either the total cost or delivery completion time. Subsequently, the two conflicting objectives are considered together for obtaining the set of best trade-off solutions. An extensive computational study is conducted to investigate the impacts of restricting the focal points, and the influence of adopting a joint optimization method instead of a sequential approach. Finally, several key insights are obtained to aid the logistics practitioners in decision making.

1. Introduction

In the era of online shopping, several components of e-commerce experience, such as marketing and transaction processing, have dramatically transformed. However, the dependency on delivery vans or trucks for the last-mile operation has remained unchanged for decades. This traditional process of delivering goods is not efficient in the long-run due to increasing traffic congestion and growing customer expectations (Tuerk, 2019; Fehr and Peers, 2019). Over 70% of the consumers across the globe are price sensitive and are reluctant to pay for faster delivery (Joerss et al., 2016). Yet, a majority expects reliable as well as quick delivery of goods. As a result, companies are competing to achieve cost-efficient and faster last-mile delivery operations. The integration of emerging

* Corresponding author.

E-mail address: SrinivasSh@missouri.edu (S. Srinivas).

technology, such as unmanned aerial vehicles or drones, in the last-mile network design can overcome these challenges and provide a competitive advantage. Owing to its relatively low operating cost and congestion-free aerial route, drones are a viable and attractive alternative for such operations (Raj and Sah, 2019). Besides, drones provide an environmentally friendly mode of delivery if low-carbon electricity sources are used to charge batteries (Stolaroff et al., 2018; Kirschstein, 2020).

Recently, the application of drones in delivery logistics has emerged prominently, especially after the initiatives of big corporations such as Amazon (2016), Google (X-Company, 2016), and DHL (2014). Due to rapid growth in Business-to-Consumer (B2C) transactions, where packages are smaller, drones are viable to deliver most customer orders (Guglielmo, 2013). Aside from e-commerce and retail logistics, drones are useful for other situations such as rescue operations (Bäckman et al., 2018), post-disaster assessment (Oruc and Kara, 2018), emergency management (Rabta et al., 2018), medical supply, and test sample transportation from remote locations (Katariya et al., 2018). Despite the potential benefits, drones have two major technical restrictions - (i) limited flight range or distance and (ii) restricted payload capacity (Otto et al., 2018). These two drawbacks impede the possibility of adopting a stand-alone drone shipment as an alternative to traditional truck delivery. Nevertheless, a drone-truck combination could be used for efficient delivery, where a truck carries a set of drones and customer orders to a central location that is within the flight range so that the drones can transport multiple low-weight orders simultaneously (Mourelo Fernandez et al., 2016; Chang and Lee, 2018). This would allow drones to ship most e-commerce orders as they are typically within the payload capacity. For instance, 86% of packages shipped by the e-commerce retailer, Amazon, weigh less than 3 kg (6.5 lb) (Guglielmo, 2013), which is less than the drone's common payload capacity (Otto et al., 2018). Thus, the necessity of using drones in delivery logistics is evident from different perspectives. However, most attempts by companies and research studies focus on using just a single drone in tandem with a truck (Jeong et al., 2019; Murray and Chu, 2015; Ha et al., 2018). Hence, the current challenge is to effectively operate multiple drones and a truck to achieve cheaper and quicker delivery. A new generation of vehicles, such as the Vision Van by Mercedes-Benz (Mercedes-Benz, 2018), has emerged recently for such truck-drone deliveries. Fig. 1 illustrates the top-view of a specially designed truck equipped with six drones on its roof for last-mile delivery tasks.

This paper deals with the problem of delivering orders to a set of customer locations using multiple drones that operate in conjunction with a single truck. To take advantage of the drone fleet, the delivery tasks are parallelized by concurrently dispatching the drones from a truck parked at a focal point (ideal drone launch location) to the nearby customer locations. Hence, the following key decisions must be optimized - (i) total drones to be employed, (ii) number of focal points (or truck stops) required and their locations, (iii) assignment of delivery locations to a focal point, and (iv) truck route that covers all the stops. Prior research tackles this problem by adopting a multi-phase sequential approach to minimize the total delivery completion time - partitioning a set of delivery locations into small clusters, identifying a focal point (truck stop and drone dispatch location) for each cluster, and then routing the truck through all focal points such that the customer orders in each cluster are fulfilled by drones (Mourelo Fernandez et al., 2016; Chang and Lee, 2018). Since these decisions are interrelated, such an approach may not optimize the overall delivery operations. Besides, these approaches assume a fixed set of drones per truck and allow each drone to execute at most one trip within a cluster. These gaps in the literature serve as a primary motivation for this research, which has the following key contributions:

- To the best of our knowledge, this work is the first to develop mathematical programming models for jointly optimizing the four key decisions mentioned above. In addition, several characteristics necessary for the real-life implementation of truck-drone delivery are integrated, such as payload-dependent flight range, drone capacity restriction, and situations requiring delivery only by a truck.
- Rather than restricting the drone-dispatch point to only a customer location (or depot) like most previous optimization-based approaches, our model allows the optimal truck stop to be anywhere in the delivery area.
- An unsupervised machine learning heuristic algorithm is proposed to obtain a good initial solution and accelerate the running time of the proposed optimization models.
- We develop two variants of the joint optimization problem - (i) minimizing total cost and (ii) minimizing delivery completion

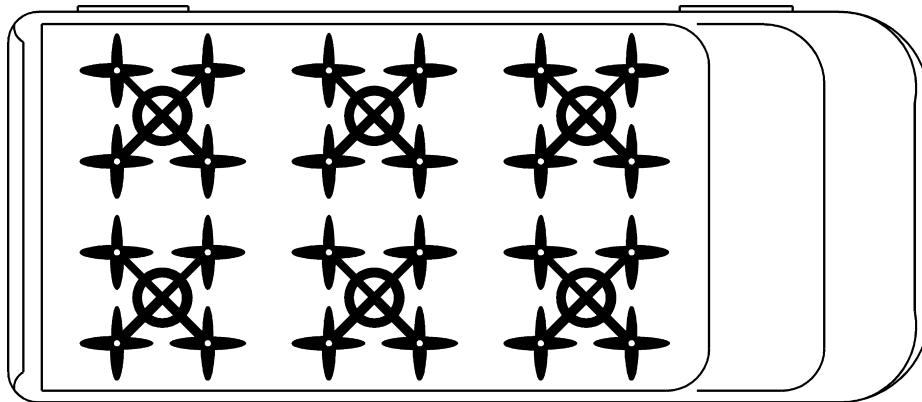


Fig. 1. A top-view of a truck carrying six drones on its roof for last-mile delivery tasks.

time. Further, the two objectives are considered together to obtain the best trade-off solutions.

The remainder of this paper is organized as follows. Section 2 provides a detailed review of the relevant literature. The problem under study is described in Section 3. In Section 4, the mathematical programming models for integrated optimization are introduced. Further, new strategies to accelerate the solution time of the computationally expensive optimization model are presented. In Section 5, an extensive set of potential scenarios is generated, and its computational results are discussed. Finally, Section 6 presents conclusions and directions for future research.

2. Literature review

The problem under study is analogous to the traditional truck and trailer routing problem (TTRP). In TTRP, some customers must be served only by a truck, while the remaining customers can be visited by a single truck or a truck-trailer combination (Lin et al., 2009; Derigs et al., 2013; Villegas et al., 2013). Akin to TTRP, we route two types of vehicles (a truck and drone instead of a truck and trailer) to different sets of customers (those who require truck-only delivery, and others who can be served either by a truck or drone). Nonetheless, our problem allows the truck and drone to operate independently, while a trailer cannot move without a truck in TTRP. On the other hand, our solution strategy is vaguely similar to the ring star problem (RSP), which considers a set of nodes (retailer locations) and a central depot with the goal of identifying a subset of retailers that can also act as small depots. The selected small depots are interconnected (ring structure) and supplied by the central depot, whereas the remaining retailers are served by the closest small depot (star topology) (Calvete et al., 2013). The RSP is similar to our work in the sense that both problems require the identification of focal points that serve neighboring locations. But, the RSP establishes several small depots in the network, while our problem locates truck stops. However, in contrast to our research, the RSP restricts each small depot to be one of the existing nodes (retailers). Besides, RSP focuses only on constructing a route connecting the focal nodes (small depots) and assigning the remaining nodes to exactly one of the focal nodes. Our research accomplishes the same, but additionally considers the delivery of orders from the focal point to its assigned nodes. Also, RSP assumes a single-vehicle, whereas our problem involves multiple drones working in tandem with a truck.

Research on employing drones in delivery operations has gained a lot of attention in recent years. Some studies considered direct drone delivery from distribution centers by using a network of recharging stations to overcome the flight range limitation (Hong et al., 2018). To circumvent the cost of establishing such stations, a truck collaborating with drones is considered in many other studies, in which a truck carries both drone and order until the customer location is within the drone flight range (Chang and Lee, 2018; Murray and Chu, 2015; Ha et al., 2018). Hence, the truck functions as a *moving depot*, as stated by Hong et al. (2018). The literature on coordinating the logistics operations using a truck-drone combination focused predominantly on extending classical routing problems - a variant of traveling salesman problem with drones (TSP-D) (Murray and Chu, 2015; Jeong et al., 2019; Agatz et al., 2018; Bouman et al., 2018; Ha et al., 2018; Mathew et al., 2015; Yurek and Ozmutlu, 2018), and a generalization of vehicle routing problems to include drones (VRP-D) (Poikonen et al., 2017; Schermer et al., 2018; Wang et al., 2017; Sacramento et al., 2019; Wang and Sheu, 2019).

In TSP-D, a set of customer locations are served either by a drone or truck. The truck carrying a single drone starts from, and returns to, a depot exactly once. Besides, a drone can only be launched from, and recovered at, customer locations. The truck can continue delivery after dispatch, as the drone may return to a site that is different from the original dispatch location. Prior research has adopted different approaches to solving the TSP-D. One of the earliest studies on the truck-drone routing was presented by Murray and Chu (2015), who developed a mixed integer linear programming (MILP) model along with simple heuristics to minimize the overall delivery completion time. They found the MILP model to be computationally expensive as it could not achieve optimality even after a 30-min runtime, whereas their heuristics were able to produce better solutions in less time. To overcome the drawback of not achieving optimality using the MILP model, subsequent works adopted different approaches for the TSP-D. Agatz et al. (2018) developed an integer programming model as well as several fast route-first cluster-second heuristics to solve the TSP-D. They achieved optimality for 12 customer instances within a two-hour run time, while their heuristics yielded near-optimal solutions quickly. Besides, their experimental analysis indicated a substantial reduction in delivery completion time when using truck-drone in tandem as opposed to truck-only delivery. However, they assumed drones to follow the road network, which diverges from previous works that consider Euclidean drone travel. To further expedite the time to optimality, Yurek and Ozmutlu (2018) proposed an iterative algorithm based on a decomposition approach for the TSP-D. Their model was able to achieve the optimal solution for 12 customer instances with an average solution time of 15 min. While all the above-mentioned research on TSP-D aimed at minimizing completion time, Ha et al. (2018) developed MILP formulation and two heuristics to minimize total operational costs. The authors found that minimizing cost has a significant improvement in drone utilization, but also increases the overall delivery completion time. Although this research above is similar to our work in the aspect of using truck and drone in tandem for delivery, there are many stark differences. In particular, all the aforementioned literature considers only a single drone, while our problem involves the use of multiple drones. In a recent study, Campbell et al. (2017) demonstrated the effectiveness of using multiple drones per truck in minimizing the operational cost. Moreover, our approach allows the drone dispatch points to be anywhere in the delivery area rather than restricting them to customer locations. While previous models on TSP-D aim to minimize either delivery completion time or operational costs, the current research considers both the objectives to unveil their independent impact and best trade-offs.

The VRP-D is an extension of the TSP-D, where a fleet of homogeneous trucks, each carrying a fixed number of drones, is used to supply a set of customers (Wang et al., 2017; Poikonen et al., 2017; Sacramento et al., 2019). Wang et al. (2017) introduced the VRP-D to minimize completion time and presented a theoretical study to compare the optimal objective value with and without using

drones. They considered different settings and concluded that integrating drones in the delivery process has substantial time savings. Particularly, they found two factors that impact the completion time - number of drones per truck and the drone speed (relative to the truck speed). However, the authors assumed rectilinear travel paths for both trucks and drones and unlimited drone battery-life. [Poikonen et al. \(2017\)](#) extended their work by relaxing these two assumptions. In addition, they related the VRP-D to the close-enough VRP (CEVRP), where a truck need not visit the actual location but achieve close proximity. In contrast to the previous two articles on VRP-D, [Sacramento et al. \(2019\)](#) presented a MILP model to minimize the total operational costs, while considering capacity and completion time constraints. The authors compared their formulation with the truck-only approach and concluded that using drones can achieve up to 30% cost savings. However, they assume that each truck is equipped with only a single drone. [Wang and Sheu \(2019\)](#) dealt with this limitation by formulating an arc-based model for VRP-D to minimize total fixed and operational costs. Their results show a 20% reduction in total cost, besides reducing the average delivery time by five minutes per customer. Nonetheless, they assume an unlimited supply of drones and restrict them to land only at a service hub or depot, but not at a customer location. The main commonality between our work and the VRP-D literature is the use of multiple drones for last-mile delivery operations. However, unlike this paper, all the previous work on VRP-D uses only a specific set of locations for launching and recovering drones.

Another category of hybrid drone-truck routing for optimizing last-mile delivery involves the coordination of multiple drones working in tandem with a single truck ([Karak and Abdelghany, 2019](#); [Mourelo Fernandez et al., 2016](#); [Chang and Lee, 2018](#); [Murray and Raj, 2020](#)). Recently, [Karak and Abdelghany \(2019\)](#) considered such an arrangement for minimizing the total operational cost of pick-up and drop-off services, where a set of customers are visited (for order pick-up and/or delivery) by drones that are deployed from a truck halted at a station. Their research shares the following aspects with our work. A single truck carrying multiple drones is departed from, and returned, to a depot. Along its route, it visits a number of truck stops (defined as stations in their study and cluster focal points in our work) and waits to dispatch and recollect drones. Yet, the truck stops are assumed to be known apriori by [Karak and Abdelghany \(2019\)](#), while our approach seeks to establish their optimal locations. Furthermore, we consider both truck and drone to be capable of serving a customer, whereas [Karak and Abdelghany \(2019\)](#) do not use the truck for delivering customer orders. On the other hand, we are aware of only two works that adopt a clustering-based method to minimize the completion time of last-mile deliveries with one truck and several identical drones ([Mourelo Fernandez et al., 2016](#); [Chang and Lee, 2018](#)). Both papers proposed a similar multi-phase sequential approach. First, the delivery locations are grouped into non-overlapping clusters. Subsequently, a truck stop per cluster is located for launching and recovering multiple drones that simultaneously serve the customers in that cluster. Finally, the best truck (carrying multiple drones and customer orders) route is established such that the truck starts from the depot, visit each truck stop exactly once, and then returns to the depot. Such an approach would also lower the operating cost compared to the truck-only delivery since the cost of using drones is significantly less than trucks ([Wang and Sheu, 2019](#)). In particular, [Mourelo Fernandez et al. \(2016\)](#) used an unsupervised machine learning approach, k -means algorithm, to partition the network into clusters, where the centroid of a cluster would serve as a truck stop for drone dispatch. Consequently, the authors developed a genetic algorithm to determine the best truck route covering all the cluster centroids. They compared the truck-drone operation with a truck-only delivery and concluded that a significant reduction in delivery completion time is possible only when employing two or more drones per truck. [Chang and Lee \(2018\)](#) improvised their approach by using the classical TSP model to find the shortest truck route through all the cluster stops. Besides, the authors shifted the cluster centroids (truck stops) towards the depot using a computationally intensive non-linear mathematical model to further reduce the delivery completion time. Even though the problem characteristics of these two works are near equivalent to our paper, we deviate from them and contribute to the literature in the following manner.

- We simultaneously optimize the clustering and routing decisions rather than using a sequential approach. Since these decisions are interrelated, a multi-phase method may not achieve the best possible outcome.
- Our work identifies the best truck stop location for each cluster instead of choosing it to be the cluster's centroid ([Mourelo Fernandez et al., 2016](#)) or shifting the centroid towards the depot ([Chang and Lee, 2018](#)).
- We allow the truck stops to be a customer or non-customer location.
- This research optimizes the drones required per truck, while previous similar work assumed the trucks to carry a fixed set of drones. This information would be useful for planning purposes, especially with the strategic decision on the number of drones to purchase.
- We allow the depot to be a potential drone dispatch point.
- The customer locations can be visited by both truck and drones in our approach rather than restricting them only to drones.
- Instead of assuming the drones to be capable of serving all customer locations irrespective of the order weight, we allow certain locations to be designated as truck-only delivery.
- As drone's flight range is payload-dependent in practice ([Jeong et al., 2019](#); [Mikrokopter, 2018](#); [Song et al., 2018](#)), we incorporate it as an attribute in our research instead of assuming constant flight range.
- We extend our formulation to allow multiple drone deliveries per cluster rather than restricting each drone to a single trip.
- We consider two different objective functions (operational cost and completion time) independently and simultaneously instead of optimizing just the delivery completion time.

3. Problem definition and assumptions

This research considers the delivery of goods from a depot (l_0) to N customer locations by using a truck and fleet of identical drones, where $\mathcal{L} = \{l_0, l_1, l_2, \dots, l_N\}$ denotes the set of delivery network vertices. The truck can dock up to G drones on its roof, where each drone has a restricted payload capacity. As a result, some customer orders require a truck-only delivery. Thus, a subset of customer locations can be served either by a drone or truck ($\mathcal{L}^D \subset \mathcal{L}$), whereas the remaining locations must be visited only by a truck ($\mathcal{L}^T = \mathcal{L} - \mathcal{L}^D$). The maximum drone flight range (F_l) is negatively associated with the outbound shipping weight, and therefore depends on the delivery location (Mikrokopter, 2018; Song et al., 2018; Jeong et al., 2019). Besides, F_l is set to null for orders exceeding drone payload capacity (i.e., $F_l = 0, \forall l \in \mathcal{L}^T$).

The delivery problem necessitates the customer locations to be partitioned into non-overlapping clusters, where each cluster has a focal point (or truck stop) defined by two-dimensional coordinates. To perform the delivery operations, the truck (carrying the drones), traveling at a speed of V^T , must start from the depot, visit every cluster stop exactly once to fulfill the orders within each cluster, and finally return to the same depot. If a cluster has more than one location assigned to it, then multiple drones are dispatched from the truck stop to parallelize shipping operations. The drone travels with a velocity V^D , unloads the package at the customer location in S_l time units, and then returns to the truck waiting at the same focal point.

Theoretically, the number of clusters established ranges from 1 to N . A single cluster indicates that the drones are directly dispatched from, and returned to, the depot, and therefore does not require a truck. On the other hand, a network with N clusters, where each cluster contains exactly one location, corresponds to a truck visiting each delivery location - a classical TSP which does not require any drones (Fig. 2(a)). Finally, if the number of clusters is between 1 and N , then it would require both truck and drones to cover all delivery locations (Fig. 2(b) and (c)). In this research, we consider the maximum number of allowable clusters (\hat{K}) as a parameter so that it can be controlled by the decision-maker based on operational needs. However, among the \hat{K} possible clusters, a cluster $k \in \mathcal{K}$ is formed if and only if it contains at least one delivery location.

Thus, given the set of delivery locations ($\mathcal{L} = \mathcal{L}^D \cup \mathcal{L}^T$), vehicles (a truck and up to G drones) along with their velocities (V^T and V^D) and maximum allowable clusters (\hat{K}), the objectives of our problem are to (i) independently minimize the total operational cost and completion time required to deliver all customer orders either by a drone or truck and (ii) obtain efficient trade-off solutions by simultaneously considering both the objectives. The proposed approach will optimize the objective by concurrently determining the following decisions - (i) total drones to deploy (g) given a capacity restriction of G drones, (ii) number of clusters to establish and their locations, (iii) delivery locations assigned to each truck stop, and (iv) the truck route. Besides, for each objective, we consider two policies for locating a truck stop - (i) restricting it to one of the customer locations (Fig. 2(b)), or (ii) allowing it to be anywhere in the delivery area (i.e., at a customer or a non-customer location), such as the illustrative example in Fig. 2(c). To formulate the optimization models for the problem under consideration, the following operating conditions are assumed.

- The drones are fully charged before leaving the truck.
- The drone launch sequence from a truck stop follows the farthest travel distance first policy within a cluster.
- The velocity of drone is independent of its payload.
- Drones carry only one delivery package at a time.
- Each drone makes at most one delivery per cluster.
- Drones travel between a cluster focal point (truck stop) and delivery location based on the Euclidean metric, while truck travels from one focal point to another on a rectilinear metric.
- The truck has sufficient capacity to accommodate all the customer orders. In the B2C delivery operations, where the vast majority of orders (more than 85%) weigh less than 5 lb, the trucks seldom run out of capacity as indicated by Sacramento et al. (2019).

4. Joint optimization of delivery locations clustering and truck-drone routing

To address the problem presented in the previous section, we propose two models to jointly optimize delivery locations clustering and truck-drone routing (JOCR) with the objective of minimizing the total cost, namely, fixed cost of drones and travel costs of truck and drones. The first model restricts each cluster focal point to coincide with a delivery location (min-cost JOCR-R), while the second allows cluster focal points to be anywhere in the delivery area (min-cost JOCR-U). Otherwise, both models share the same characteristics. An overview of the research methodology is provided in Fig. 3. A collection of input parameters are used to model the JOCR-R policy as an integer program (IP) and JOCR-U policy as a mixed integer nonlinear program (MINLP). Since the MINLP model is computationally intractable, it is approximated to a MILP model using several linearization procedures. In addition, novel acceleration techniques are used to reduce the search space and expedite the solution time of the MILP model. Besides, three variants are proposed to control the model characteristics - alternative objective functions, dealing with two conflicting objectives, and allowing multiple drone deliveries per cluster.

4.1. Optimization model for min-cost JOCR-R

In this section, the min-cost JOCR-R model is formulated using the following notations (indices and sets, parameters, and decision variables).

Indices and Sets

- $l, l' \in \mathcal{L}$ set of depot and customer (or delivery) locations, $\mathcal{L} = \{l_0, l_1, l_2, l_3, \dots, l_N\}$, where l_0 denotes the depot location
 $l \in \mathcal{L}^D$ subset of delivery locations that can be served by drones or truck, $\mathcal{L}^D \subset L$
 $l \in \mathcal{L}^T$ subset of delivery locations that must be served only by a truck, $\mathcal{L}^T \subset L$

Parameters

N	number of customer locations
\hat{K}	maximum allowable number of clusters
G	maximum number of drones that can be docked on a truck roof
C_1^D	fixed cost for employing a drone (in \$/drone)
C_2^D	travel cost of drone (in \$/mile)
C^T	travel cost of truck (in \$/mile)
(A_l, B_l)	coordinates of delivery location $l \in \mathcal{L}$
D_{ll}^E	Euclidean distance (in miles) between delivery locations $l' \in \mathcal{L}$ and $l \in \mathcal{L}$, where $D_{ll}^E = \sqrt{(A_{l'} - A_l)^2 + (B_{l'} - B_l)^2}$
D_{ll}^R	rectilinear distance (in miles) between delivery locations $l' \in \mathcal{L}$ and $l \in \mathcal{L}$, where $D_{ll}^R = A_{l'} - A_l + B_{l'} - B_l $
F_l	maximum flight range of a drone (in miles) serving delivery location $l \in \mathcal{L}$

Decision variables

g	number of drones carried by the truck
x_{ll}	1 if a delivery location $l \in \mathcal{L}$ acts as a focal point for its cluster, 0 otherwise
$x_{l'l}$	1 if a delivery location $l' \in \mathcal{L}$ is assigned to a focal point $l \in \mathcal{L}$, 0 otherwise
$y_{l'l}$	1 if truck travels from cluster focal point $l' \in \mathcal{L}$ to another focal point $l \in \mathcal{L}$, 0 otherwise
u_l	order in which cluster focal point $l \in \mathcal{L}$ is visited by the truck
\mathcal{C}_{JOCR}^R	total cost for the JOCR-R policy

The JOCR-R model is mathematically formulated as follows.

$$\text{Minimize } \mathcal{C}_{JOCR}^R = C_1^D g + C_2^D \sum_{l' \in \mathcal{L}} \sum_{l \in \mathcal{L}} x_{l'l} \times \left(2D_{l'l}^E \right) + C^T \sum_{l' \in \mathcal{L}} \sum_{l \in \mathcal{L}} y_{l'l} D_{l'l}^R \quad (1)$$

s.t.

$$\sum_{l \in \mathcal{L}} x_{ll} \leq \hat{K} \quad (2)$$

$$x_{l'l} \leq x_{ll} \quad \forall l', l \in \mathcal{L} \quad (3)$$

$$\sum_{l \in \mathcal{L}} x_{l'l} = 1 \quad \forall l' \in \mathcal{L} \quad (4)$$

$$x_{ll} = 1 \quad \forall l \in \mathcal{L} \ni l = \{l_0\} \quad (5)$$

$$\sum_{l' \in \mathcal{L}} x_{l'l} \leq g + 1 \quad \forall l \in \mathcal{L} \quad (6)$$

$$g \leq G \quad (7)$$

$$x_{l'l} D_{l'l}^E \leq F_{l'} \quad \forall l', l \in \mathcal{L} \quad (8)$$

$$\sum_{l' \in \mathcal{L}, l' \neq l} y_{l'l} = x_{ll} \quad \forall l \in \mathcal{L} \quad (9)$$

$$\sum_{l \in \mathcal{L}, l \neq l'} y_{l'l} = x_{l'l} \quad \forall l' \in \mathcal{L} \quad (10)$$

$$u_l - u_{l'} + (\hat{K} - 1)y_{ll'} + (\hat{K} - 3)y_{l'l} \leq (\hat{K} - 2) \quad \forall l, l' \in \mathcal{L} \setminus \{l_0\}, l' \neq l \quad (11)$$

$$x_{l'l}, y_{l'l} \in \{0, 1\} \quad \forall l', l \in \mathcal{L} \quad (12)$$

The objective function (1) minimizes the total cost of operating the truck and set of drones for the JOCR-R policy. Constraint (2) restricts the total truck stops to be capped by the maximum allowable clusters. Constraint (3) allows a delivery location $l' \in \mathcal{L}$ to be assigned to another location $l \in \mathcal{L}$ only if the latter serves as a focal point. Further, constraint (4) ensures that every delivery location $l' \in \mathcal{L}$ is assigned to exactly one cluster focal point location $l \in \mathcal{L}$. Constraint (5) forces the depot ($l_0 \in \mathcal{L}$) to be a focal point so that it can dispatch drones to nearby locations. In addition, this constraint also guarantees a truck visit to the depot (i.e., the final stop). Since a drone is assumed to make a single trip per cluster, constraint (6) ensures that the number of drone-supplied locations in each cluster cannot exceed the total drones carried by the truck (g). The truck has a capacity restriction of G drones on its roof, and this

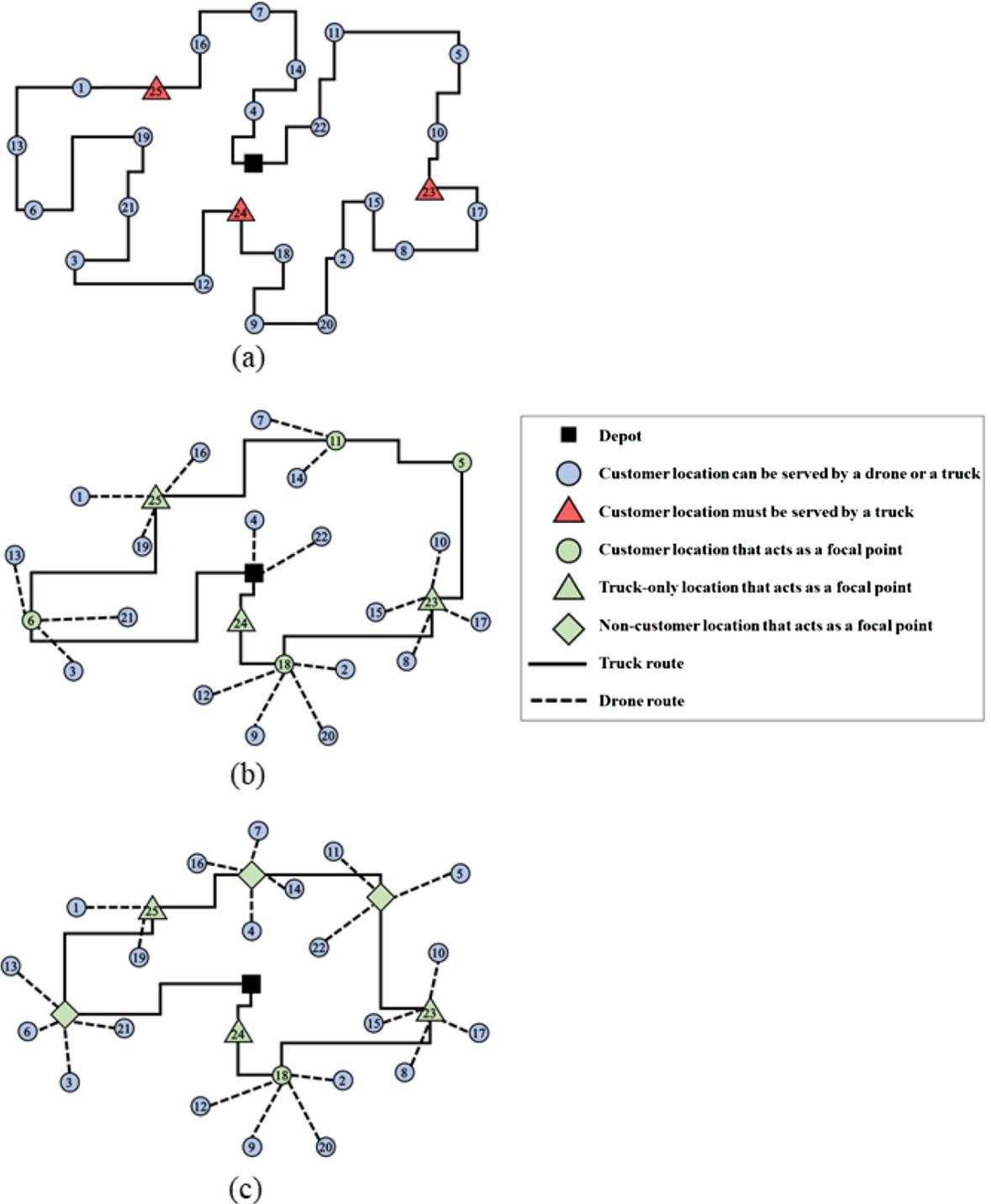


Fig. 2. Illustrative example of delivery using: (a) truck only; (b) truck with drones dispatched from delivery locations; (c) truck with drones can be dispatched from anywhere.

condition is guaranteed using constraint (7). Constraint (8) stipulates that a location served by a drone is assigned to a cluster focal point only if the distance between them is within the flight range. Constraints (9) and (10) specify the truck route by confining its stops to cluster focal points (i.e., when $x_{ll} = 1$) and limiting the number of truck visits to each focal point to one. In addition, constraint (11) eliminates sub-tours to ensure a single trip of the truck to visit all focal points before returning to the depot. Finally, the binary restriction on decision variables x_{ll} and y_{ll} are specified by constraint (12).

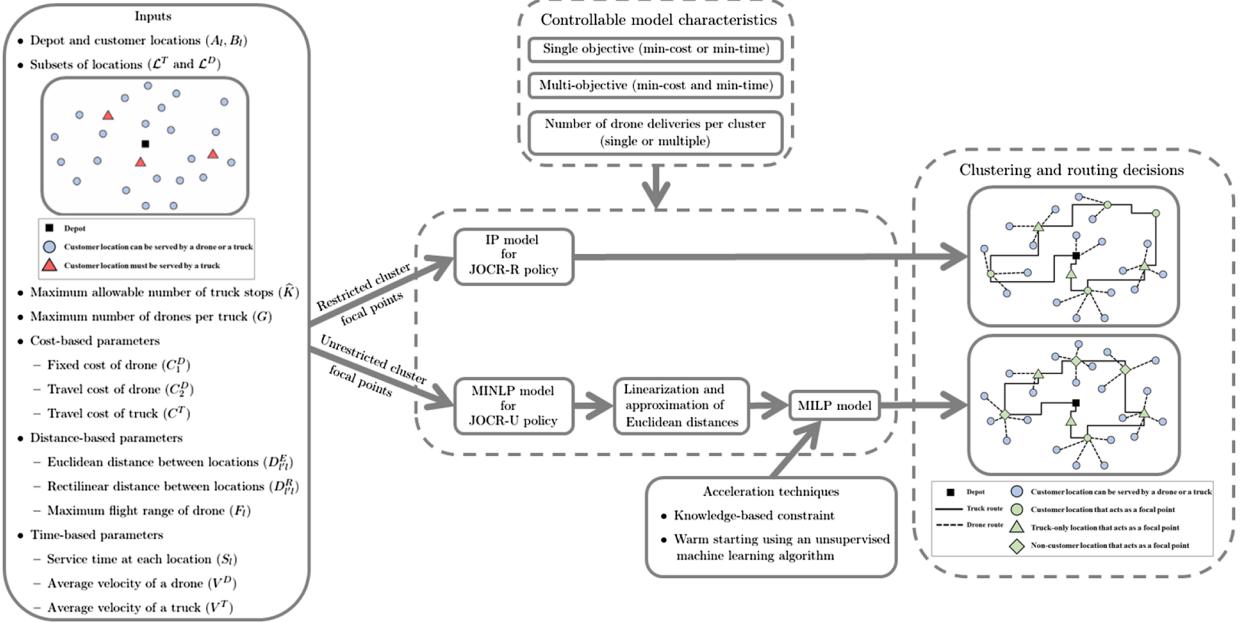


Fig. 3. Overview of research methodology for the proposed joint optimization approach.

4.2. Optimization model for min-cost JOCR-U

The min-cost JOCR-U model does not enforce the focal point to be a customer location. The problem is initially formulated as a MINLP model. Then, to provide a computationally efficient formulation, a linear approximation of the MINLP model is presented in Appendix A. The following new notations are considered to formulate the min-cost JOCR-U model.

Indices and Set

$k, k' \in \mathcal{K}$ set of possible clusters, $\mathcal{K} = \{k_0, k_1, k_2, k_3, \dots, k_{\hat{K}}\}$, where k_0 denotes the cluster with depot as its focal point

Decision Variables

(a_k, b_k)	coordinates of truck stop or cluster focal point $k \in \mathcal{K}$
d_{lk}^E	Euclidean distance (in miles) between delivery location $l \in \mathcal{L}$ and cluster focal point $k \in \mathcal{K}$
$d_{kk'}^R$	rectilinear distance (in miles) between cluster focal points $k \in \mathcal{K}$ and $k' \in \mathcal{K}$
x_{lk}	1 if a delivery location $l \in \mathcal{L}$ is assigned to cluster $k \in \mathcal{K}$, 0 otherwise
q_{lk}	1 if a delivery location $l \in \mathcal{L}$ is assigned to cluster $k \in \mathcal{K}$ and served by a drone, 0 otherwise
$y_{kk'}$	1 if truck travels from cluster focal point $k \in \mathcal{K}$ to another focal point $k' \in \mathcal{K}$, 0 otherwise
\mathcal{C}_{JOCR}^U	total cost for the JOCR-U policy

The MINLP model for min-cost JOCR-U is mathematically formulated as follows.

$$\text{Minimize } \mathcal{C}_{JOCR}^U = C_1^D g + C_2^D \sum_{l \in \mathcal{L}} \sum_{k \in \mathcal{K}} x_{lk} \times \left(2d_{lk}^E \right) + C^T \sum_{k \in \mathcal{K}} \sum_{k' \in \mathcal{K}} y_{kk'} d_{kk'}^R \quad (13)$$

s.t.

$$\sum_{k \in \mathcal{K}} x_{lk} = 1 \quad \forall l \in \mathcal{L} \quad (14)$$

$$d_{lk}^E = \sqrt{(A_l - a_k)^2 + (B_l - b_k)^2} \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (15)$$

$$x_{lk} d_{lk}^E \leq F_l q_{lk} \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (16)$$

$$\sum_{l \in \mathcal{L}} q_{lk} \leq g \quad \forall k \in \mathcal{K} \quad (17)$$

$$g \leq G \quad (18)$$

$$\sum_{k' \in \mathcal{K}, k \neq k'} y_{kk'} = 1 \quad \forall k \in \mathcal{K} \quad (19)$$

$$\sum_{k \in \mathcal{K}, k \neq k'} y_{kk'} = 1 \quad \forall k' \in \mathcal{K} \quad (20)$$

$$u_k - u_{k'} + (\hat{K} - 1)y_{kk'} + (\hat{K} - 3)y_{k'k} \leq (\hat{K} - 2) \quad \forall k, k' \in \mathcal{K} \setminus \{k_o\}, k \neq k' \quad (21)$$

$$d_{kk'}^R = |a_k - a_{k'}| + |b_k - b_{k'}| \quad \forall k, k' \in \mathcal{K} \quad (22)$$

$$(a_k, b_k) = (A_{l_o}, B_{l_o}) \quad \forall k \in \mathcal{K} \exists k = \{k_o\} \quad (23)$$

$$x_{lk} \in \{0, 1\} \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (24)$$

$$y_{kk'} \in \{0, 1\} \quad \forall k, k' \in \mathcal{K} \quad (25)$$

The objective function (13) minimizes the total cost for the JOCR-U policy. Constraint (14) ensures that every delivery location $l \in \mathcal{L}$ is assigned to one and only one cluster focal point $k \in \mathcal{K}$. Eq. (15) determines the drone travel distance between a delivery location $l \in \mathcal{L}$ and its focal point $k \in \mathcal{K}$. Further, constraint (16) ensures that this flight distance to be within the maximum drone flight range for location $l \in \mathcal{L}$. Also, constraint (16) forces delivery location $l \in \mathcal{L}$ to overlap with its cluster focal point $k \in \mathcal{K}$ (i.e., $d_{lk}^E = 0$) only when that location is not served by a drone ($q_{lk} = 0$). Besides guaranteeing truck visits to locations $l \in \mathcal{L}$, our formulation also allows them to act as cluster focal points for drone dispatch to potential nearby locations (based on constraints (15) and (16)). Constraint (17) limits the drone-served locations per cluster to the total drones carried by the truck (g), which is, in turn, governed by the capacity constraint (18).

With respect to truck routing, constraints (19) and (20) stipulate that each cluster focal point $k \in \mathcal{K}$ has exactly one inbound and one outbound visit by the truck. Constraint (21) is for sub-tour elimination, which ensures that the truck returns to the depot only after visiting all the truck stops. Eq. (22) computes the rectilinear travel distance for a truck between two cluster focal points k and k' . In particular, the coordinates of cluster focal points (truck stops/drone dispatch points) are determined concurrently by Eqs. (15) and (22), which yield the optimum combination of distances to be traveled by drones and trucks, such that it leads to the minimum cost. Eq. (23) assigns the cluster focal point $\{k_o\} \in \mathcal{K}$ to the depot with coordinates (A_{l_o}, B_{l_o}) to ensure a truck visit to the depot, in addition to allowing delivery locations to be assigned to the depot like any other cluster focal point. Finally, the binary restrictions on the decision variables are specified using constraints (24) and (25). Thus far, we have a MINLP model that is computationally complex to solve as the objective function (13) and constraints (15), (16), and (22) are non-linear. To achieve optimal solutions in a reasonable time, we linearize the model in Appendix A.

Proposition 1. For the same parameters, if there exists a feasible solution for the min-cost JOCR-R model, then the optimal total cost of the JOCR-U (\mathcal{C}_{JOCR-U}^*) is less than or equal to that of JOCR-R (\mathcal{C}_{JOCR-R}^*).

Proof. Let \mathcal{S}^R and \mathcal{S}^U denote the set of all feasible solutions for the min-cost JOCR-R and min-cost JOCR-U models, respectively. Then, the feasible solutions in \mathcal{S}^R satisfy the constraints of both JOCR-R and JOCR-U model. In addition, there may be other feasible solutions for JOCR-U, where focal points are located at non-customer locations. Thus,

$$\mathcal{S}^R \subset \mathcal{S}^U$$

Consider a feasible solution X^* that achieves the minimum total cost of JOCR-U, \mathcal{C}_{JOCR-U}^* . By definition, $X^* \in \mathcal{S}^U$, but may not always be in \mathcal{S}^R . If $X^* \in \mathcal{S}^R$, then

$$\mathcal{C}_{JOCR-U}^* = \mathcal{C}_{JOCR-R}^*$$

However, if $X^* \notin \mathcal{S}^R$, then

$$\mathcal{C}_{JOCR-U}^* < \mathcal{C}_{JOCR-R}^*$$

Thus, $\mathcal{C}_{JOCR}^U \leq \mathcal{C}_{JOCR}^R$. \square

4.3. Variant 1: Alternative objective function for JOCR-R and JOCR-U

While minimizing total operational cost is an important objective from a company's perspective, faster delivery is seen as a key priority to customers. Therefore, the objective of minimizing the delivery completion time is commonly adopted in the literature on last-mile deliveries (Murray and Chu, 2015; Yurek and Ozmutlu, 2018; Jeong et al., 2019; Wang et al., 2017). The proposed min-cost JOCR-R and min-cost JOCR-U models can be easily adapted to minimize the delivery completion time. We introduce the following additional notations to represent the alternative objective function.

S_l	service time at a delivery location $l \in \mathcal{L}$
V^D	average velocity (in mph) of the drone
V^T	average velocity (in mph) of the truck
\mathcal{T}_{JOCR}^R	delivery completion time for the JOCR-R policy
\mathcal{T}_{JOCR}^U	delivery completion time for the JOCR-U policy

The min-time JOCR-R and min-time JOCR-U can be formulated by replacing objective function (1) and (13) with Eqs. (26) and (27), respectively. The non-linear terms in these objective functions can be linearized as illustrated in Appendix B.

$$\text{Minimize} \quad \mathcal{T}_{JOCR}^R = \sum_{l \in \mathcal{L}} \max_{l' \in \mathcal{L}} \left(x_{ll} \times \left(2D_{ll}^E/V^D + S_{l'} \right) \right) + \sum_{l' \in \mathcal{L}} \sum_{l \in \mathcal{L}} y_{ll'} \times D_{ll'}^R/V^T \quad (26)$$

$$\text{Minimize} \quad \mathcal{T}_{JOCR}^U = \sum_{k \in \mathcal{K}} \max_{l \in \mathcal{L}} \left(x_{lk} \times \left(2d_{lk}^E/V^D + S_l \right) \right) + \sum_{k \in \mathcal{K}} \sum_{k' \in \mathcal{K}} y_{kk'} \times d_{kk'}^R/V^T \quad (27)$$

Corollary 1. For the same parameters, let \mathcal{T}_{JOCR-U}^* and \mathcal{T}_{JOCR-R}^* be the optimal delivery completion time for JOCR-U and JOCR-R, respectively. Then, $\mathcal{T}_{JOCR-U}^* \leq \mathcal{T}_{JOCR-R}^*$

4.4. Variant 2: Dealing with multiple conflicting objectives

Minimizing total cost and delivery completion time are two conflicting objectives for the problem under study. For instance, if minimizing delivery completion time is the sole objective, the JOCR models may utilize all available drones to exploit simultaneous order fulfillment. While this may shorten the overall completion time, it increases the fixed cost of using drones. Thus, given the conflicting nature, it is essential to obtain the set of best trade-off or Pareto optimal solutions, in which an improvement in one objective is not possible without degrading the other. In this research, we use the ϵ -constraint method, one of the best-known techniques to handle multi-objective problems, to obtain the Pareto-optimal solutions (Salazar-Aguilar et al., 2011). The method is based on optimizing one of the objectives (e.g., minimizing total cost), while the other is bounded from above by an additional constraint. A straightforward application can be the minimization of total cost while maintaining a desired completion time (e.g., shift hours of driver). The following model solves the min-cost JOCR-R model after adding constraint (28), where ϵ is the upper bound on the delivery completion time.

$$\begin{aligned} \text{Minimize} \quad & \mathcal{C}_{JOCR}^R \\ \text{s. t. constraints (2 – 12)} \quad & \\ & \mathcal{T}_{JOCR}^R \leq \epsilon \end{aligned} \quad (28)$$

Similarly, the min-cost JOCR-U model can also be formulated to incorporate the objective of delivery completion time. It is to be noted that every solution obtained using the ϵ -constraint method is a Pareto-optimal. Thus, the model can be solved for diverse values of ϵ to provide the practitioner with a broad set of Pareto-optimal solutions.

4.5. Variant 3: Allowing multiple drone deliveries per cluster

The JOCR models discussed in the previous sections assume the drones to make at most one trip per cluster. However, the formulation may be relaxed to allow multiple deliveries by a drone within a cluster. The potential benefit of such a setting is that drone utilization can be maximized, and fewer drones can fulfill the delivery tasks. However, the problem of minimizing total cost by considering multiple drone deliveries should impose an upper limit on the delivery completion time (i.e., $\mathcal{T}_{JOCR}^U \leq \epsilon$). Otherwise, the model will often provide a trivial solution of choosing a single drone (i.e., $g = 1$) to complete the delivery tasks (see Proposition 2), where the same drone makes all the trips in a cluster.

Proposition 2. If $\widehat{K} < N$, $C_1^D > 0$, and multiple drone deliveries per cluster is allowed, then the optimal solution for the min-cost models (JOCR-R and JOCR-U) will always employ a single drone ($g^* = 1$).

Proof. Let g^* , D^* , and T^* be the number of drones required, total drone flight distance, and total truck travel distance to achieve the minimum total cost. If $\widehat{K} < N$, then the truck cannot serve all the customers (i.e., $g^* > 0$). Since multiple drone deliveries per cluster is allowed, g^* does not impact D^* and T^* , and therefore will not affect the total travel costs of drones and truck. However, the total operational cost for the min-cost models will increase by C_1^D for every unit of drone employed, as $C_1^D > 0$. To avoid this additional cost, g^* must be as minimum as possible. Thus, the optimal solution is to operate a single drone ($g^* = 1$). \square

To accommodate multiple drone deliveries in the JOCR-R models, we introduce $\mathcal{I} = \{i_1, i_2, \dots, i_G\}$ to represent the set of drones that can be carried by a truck. In addition, two binary variables are also required. Let $z_{li} \in \{0, 1\}$ equal one if a delivery location $l \in \mathcal{L}$ is served by drone $i \in \mathcal{I}$, and $e_{li} \in \{0, 1\}$ equal one if drone $i \in \mathcal{I}$ is employed in cluster with focal point $l \in \mathcal{L}$. The following modifications must be incorporated in the min-cost JOCR-R model to relax the assumption of single drone delivery per truck. Constraint (29) should be added to ensure that a location $l \in \mathcal{L}$ is served by one and only one drone, if it is not a focal point. Constraint (30) is incorporated to stipulate that a drone $i \in \mathcal{I}$ is employed in cluster $l \in \mathcal{L}$ if at least one location in the cluster is served by the drone. The required number of drones is now determined by constraint (31) instead of constraint (6). Finally, the maximum time spent at a cluster $l \in \mathcal{L}$ is redefined to be the maximum time taken by a drone $i \in \mathcal{I}$, using constraints (32)–(34) instead of constraint (55), where d_{li}^E and s_{li} respectively denote the Euclidean distance and service time, for a location $l \in \mathcal{L}$ assigned to cluster $l \in \mathcal{L}$ and served by drone $i \in \mathcal{I}$.

$$x_{ll} + \sum_{i \in \mathcal{I}} z_{li} = 1 \quad \forall l \in \mathcal{L} \quad (29)$$

$$e_{li} \geq x_{l'i} + z_{l'i} - 1 \quad \forall l, l' \in \mathcal{L}, i \in \mathcal{I} \quad (30)$$

$$\sum_{i \in \mathcal{I}} e_{li} \leq g \quad \forall l \in \mathcal{L} \quad (31)$$

$$t_l \geq \sum_{l' \in \mathcal{L}} \left(2d_{l'l}^E / V^D + s_{l'l} \right) \quad \forall l \in \mathcal{L}, i \in \mathcal{I} \quad (32)$$

$$d_{l'l}^E \geq D_{l'l}^E - M(1 - x_{l'l}) - M(1 - z_{l'l}) \quad \forall l', l \in \mathcal{L}, i \in \mathcal{I} \quad (33)$$

$$s_{l'l} \geq S_{l'l} - M(1 - x_{l'l}) - M(1 - z_{l'l}) \quad \forall l', l \in \mathcal{L}, i \in \mathcal{I} \quad (34)$$

Similarly, the reformulation of min-cost JOCR-U to accommodate multiple drone deliveries is as follows. Constraint (35) should be added to ensure that a drone $i \in \mathcal{I}$ is assigned to location $l \in \mathcal{L}$, if that location is selected to be served by a drone. Constraint (36) must be incorporated to control the deployment of drone $i \in \mathcal{I}$ in cluster $k \in \mathcal{K}$ using binary variable e_{ki} , where $e_{ki} \in \{0, 1\}$ is one if drone $i \in \mathcal{I}$ is used in cluster $k \in \mathcal{K}$. Similar to constraints (31)–(34), constraint (37) replaces constraint (17), while inequalities (38)–(40) substitutes for constraint (56).

$$\sum_{i \in \mathcal{I}} z_{li} \geq \sum_{k \in \mathcal{K}} q_{lk} \quad \forall l \in \mathcal{L} \quad (35)$$

$$e_{ki} \geq x_{lk} + z_{li} - 1 \quad \forall l \in \mathcal{L}, k \in \mathcal{K}, i \in \mathcal{I} \quad (36)$$

$$\sum_{i \in \mathcal{I}} e_{ki} \leq g \quad \forall k \in \mathcal{K} \quad (37)$$

$$t_k \geq \sum_{l \in \mathcal{L}} \left(2d_{lki}^E / V^D + s_{lki} \right) \quad \forall k \in \mathcal{K}, i \in \mathcal{I} \quad (38)$$

$$d_{lki}^E \geq D_{lki}^E - M(1 - z_{li}) \quad \forall l \in \mathcal{L}, k \in \mathcal{K}, i \in \mathcal{I} \quad (39)$$

$$s_{lki} \geq S_{lki} - M(1 - x_{lk}) - M(1 - z_{li}) \quad \forall l \in \mathcal{L}, k \in \mathcal{K}, i \in \mathcal{I} \quad (40)$$

4.6. Choosing the maximum allowable clusters (\hat{K}) for JOCR-R and JOCR-U

The maximum allowable clusters (\hat{K}) affects the number of truck stops and computational time required to solve the model. From a theoretical perspective, if a single delivery per drone in a cluster is assumed, then the minimum value for \hat{K} is $\lceil \frac{N}{G} \rceil$, which enforces one cluster and truck stop for every G delivery locations. However, if the customer locations are dispersed, then setting \hat{K} to the minimum value may give an infeasible solution. This is because the minimum value of \hat{K} may not be achievable without violating some of the constraints, such as the drone's flight range restriction. On the other hand, the maximum value of \hat{K} can be as high as N clusters (each location can be a focal point by itself), but it increases the problem size (or search space) considerably, thereby requiring unreasonable computational time. Thus, it is recommended to set \hat{K} to a certain threshold over the minimum value (i.e., $\lceil \frac{N}{G} \rceil + \kappa \leq N$), where κ is the threshold parameter. Upon solving the optimization model, the number of established clusters could be less than or equal to \hat{K} . In the case of equality, it indicates that a higher number of truck stops (i.e., clusters) may improve the objective function at the expense of higher computational time, thereby requiring a trade-off between them. On the other hand, if the model requires fewer clusters than \hat{K} to optimize the objective, then the chosen value is sufficient.

4.7. Accelerating solution time for min-cost and min-time JOCR-U

Allowing focal points to be anywhere on the delivery plane increases the search space substantially. Therefore, large problem instances may become computationally intractable. In this section, we present two problem-specific strategies to accelerate the solution time for JOCR-U.

4.7.1. Adding a knowledge-based constraint (KBC) to reduce search space

The drone flight range, coordinates of all the delivery locations, and the distances between them are known apriori. Therefore, based on this information, it is possible to establish the area (or all feasible coordinates) in which a potential cluster focal point will be located for delivery location l . Further, if the area of the prospective cluster focal points that can serve location l does not overlap with the area of potential focal points that can serve location l' , then l and l' cannot be assigned to the same cluster. We can inject this prior knowledge in the MILP model and accelerate the solution time by adding constraint (41), where parameter $Q_{ll'}$ is 1 if delivery locations l and l' can be served together within the same cluster and 0 otherwise.

$$(1 - Q_{ll'})(x_{lk} + x_{l'k}) \leq 1 \quad \forall l, l' \in \mathcal{L}, k \in \mathcal{K}, l < l' \quad (41)$$

Note that the value of the binary parameter $Q_{ll'}$ should be set to one if the condition in inequality (42) is satisfied and 0 otherwise, where $D_{ll'}^E = \sqrt{(A_l - A_{l'})^2 + (B_l - B_{l'})^2}$.

$$F_l + F_{l'} \geq D_{ll'}^E \quad \forall l, l' \in \mathcal{L} \quad (42)$$

4.7.2. Warm starting (WS) using a heuristic algorithm

Another strategy to accelerate the MILP solution time is by using a warm-start (WS) procedure, where we initiate the optimization problem with a good feasible solution. Prior research that considered a similar problem used an unsupervised machine learning algorithm, iterative k -means clustering (see [Appendix C](#)), to effectively cluster the delivery locations and obtain the focal point of each cluster ([Mourelo Fernandez et al., 2016; Chang and Lee, 2018](#)). However, unlike our study, their heuristics assumed all the delivery locations to be accessible by drones. Therefore, we propose a heuristic that modifies the clusters to accommodate all types of customer locations and then uses it as a feasible solution to warm-start the MILP model. The proposed heuristic is detailed in [Algorithm 1](#) and adopts a three-step procedure. First, the locations that can be served by drones (\mathcal{L}^D) are clustered, and their focal points are obtained by using the iterative k -means clustering. In the second phase, the algorithm aims to move each cluster focal point to the nearest delivery location that is served only by a truck (\mathcal{L}^T), while taking into account the flight range restrictions of drones. Finally, the optimal truck route via the modified focal points is obtained using the standard TSP model.

Algorithm 1. Unsupervised machine learning heuristic for warm-starting

```

1: Inputs: set of delivery locations that can be served by a drone/truck ( $\mathcal{L}^D$ ), locations that must be served by a truck ( $\mathcal{L}^T$ ), delivery coordinates ( $A_l, B_l$ ) of all
   locations, drone flight range ( $F_l, \forall l \in \mathcal{L}$ ), total drones can be carried by a truck ( $G$ )
2: Obtain the cluster focal points for locations that can be served by truck or drone and the locations assigned to each cluster focal point using the iterative  $k$ -means
   algorithm in Appendix C
3: for each location  $l \in \mathcal{L}^T$  do
4:   for each cluster focal point  $k \in \mathcal{K}$  do
5:     Compute the distance between cluster focal point  $k$  and location  $l$  ( $\delta_{lk}$ )
      
$$\delta_{lk} = \sqrt{(A_l - a_k)^2 + (B_l - b_k)^2}$$

6:   end for
7: end for
8: Establish a set consisting of distances between every cluster focal point and truck-only location
9:   
$$\mathcal{D} = \{\delta_{lk} \mid l \in \mathcal{L}^T, k \in \mathcal{K}\}$$

10: while  $\mathcal{D} \neq \emptyset$  do
11:   Determine  $(\hat{l}, \hat{k}) = \underset{l \in \mathcal{L}^T, k \in \mathcal{K}}{\operatorname{argmin}} \{\delta_{lk}\}$ 
12:   for each location  $l \in \mathcal{L}^D$  do
13:     if location  $l$  is assigned to cluster  $\hat{k}$  (i.e.,  $x_{l\hat{k}} = 1$ ) then
14:       Compute distance between  $\hat{l}$  and location  $l$  ( $\Delta_{\hat{l},l}$ )
15:       
$$\Delta_{\hat{l},l} = \sqrt{(A_{\hat{l}} - A_l)^2 + (B_{\hat{l}} - B_l)^2}$$

16:     end if
17:   end for
18:   if  $\Delta_{\hat{l},l} \leq R_l \{ \forall l' \mid l \in \mathcal{L}^D, x_{l'l} = 1 \}$  then
19:     Set coordinates of focal point  $\hat{k}$  as the coordinates  $\hat{l}$ ,  $(a_{\hat{k}}, b_{\hat{k}}) \leftarrow (A_{\hat{l}}, B_{\hat{l}})$ 
20:     Remove  $\hat{l}$  from being considered as cluster focal point,  $\mathcal{L}^T \leftarrow \mathcal{L}^T \setminus \{\hat{l}\}$ 
21:   end if
22:   Remove  $\delta_{lk}$  from set  $\mathcal{D}$  (i.e.,  $\mathcal{D} \leftarrow \mathcal{D} \setminus \{\delta_{lk}\}$ )
23: end while
24: Compute the rectilinear distance among all the cluster focal points established
25: Obtain the optimal truck route covering all the focal points by using the rectilinear distance as inputs and solving the standard TSP model
26: return focal points of clusters ( $a_k, b_k$ ), optimal truck route between them, and assignments of delivery locations to clusters ( $x_{lk} \forall l \in \mathcal{L}, k \in \mathcal{K}$ )

```

5. Computational study

In this section, we conduct extensive numerical analysis to evaluate the effectiveness of the proposed JOCR models. Numerous test instances of different sizes are generated to (i) assess the impact of unrestricted focal points on total cost and delivery completion time as opposed to restricted truck stops, (ii) compare the performance of our joint optimization approach with a sequential heuristic method proposed in the literature ([Chang and Lee, 2018](#)), (iii) analyze the impact of changing the maximum allowable number of truck stops, and (iv) explore the performance of JOCR-U under different warm-start strategies. A sensitivity analysis is conducted to ascertain the influence of critical parameters on the performance measures. Also, we illustrate the possibility of obtaining Pareto-optimal solutions (cost and time trade-off) in the case of conflicting objectives. Finally, we investigate the impact of allowing multiple

drone deliveries per cluster. The proposed IP and MILP models have been developed using General Algebraic Modeling System (GAMS 24.5.6) and solved using CPLEX 12.8 optimizer, while the heuristic algorithms were coded and solved using Python programming language. Further, all the computational instances were executed on a computer with Intel Core-i7 @ 3.9 GHz processor and 8 GB RAM.

5.1. Experimental setup of test instances

We create test instances by varying the number of customer locations (N) from 20 to 35 in increments of 5. To ensure a robust evaluation, ten replications are generated for each test instance. Besides, each instance requires 10% of the total customers to be served only by a truck ($\xi = 10\%$). Therefore, if N is divisible by 5 and not 10, then 50% of the replications will have $(N + 5) \times \xi$ truck-only locations and the remaining replications will require $(N - 5) \times \xi$ truck-only locations to ensure ξ to be 10% on average across all replications. The coordinates of the delivery locations are considered to be randomly distributed within an area of 30×30 miles². For all instances, the threshold parameter (κ), which controls the maximum allowable number of clusters (\hat{K}), is fixed to three. Consistent with the parameters in recent literature, the average velocity of both drone (V^D) and truck (V^T) is set as 25 mph (Wang and Sheu, 2019; Ha et al., 2018), and the maximum drone flight range (F_l) is restricted to 10 miles (for all delivery locations that can be served by drones) (Gross, 2013; Sacramento et al., 2019). The truck can accommodate up to six drones on its roof (i.e., $G = 6$), and every drone carried by truck is assumed to incur a fixed cost (C_1^D) of \$3. On the other hand, the operating cost of drone (C_2^D) is set at \$0.15 per mile, while the truck operating cost (C^T) is set at \$1.25 per mile (Campbell et al., 2017). The average service time at each location (S_l) is assumed to be one minute.

5.2. Evaluation of test instances

To evaluate the test instances and understand the impact of each objective, we solve the JOCR-R and JOCR-U models by considering a single objective (i.e., independently minimizing cost and completion time). In addition, to benchmark our JOCR-U model, we consider an approach developed by Chang and Lee (2018) for a very similar problem. Their method involves the use of a sequential heuristic for clustering and routing decisions with unrestricted focal point locations (referred to as SHCR-U in this paper), and includes three sequential steps - (i) partitioning delivery locations that can be served by drones into non-overlapping clusters using iterative k-means clustering (see Appendix C), (ii) optimizing cluster focal points by shifting the centroids of clusters, obtained from the first step, towards the depot, and (iii) routing the truck via all cluster focal points by using the standard TSP model. Since we force 10% of the customers to be served by a truck, their locations are included in the truck routing step of the heuristic.

Table 1 provides a comparison of the average total cost and delivery completion time across 10 replications for the JOCR-R, JOCR-U, and SHCR-U models. In addition, the table also summarizes the percentage difference in cost ($\mathcal{C}_{R-U}^{gap} = \frac{\mathcal{C}_{JOCR}^U - \mathcal{C}_{JOCR}^U}{\mathcal{C}_{JOCR}^U} \times 100\%$)

and time ($\mathcal{T}_{R-U}^{gap} = \frac{\mathcal{T}_{JOCR}^U - \mathcal{T}_{JOCR}^U}{\mathcal{T}_{JOCR}^U} \times 100\%$) between the two focal point policies - restricted and unrestricted. Likewise, it also gives the

percentage change in cost ($\mathcal{C}_{SHCR-JOCR}^{gap} = \frac{\mathcal{C}_{JOCR}^U - \mathcal{C}_{SHCR}^U}{\mathcal{C}_{SHCR}^U} \times 100\%$) and time ($\mathcal{T}_{SHCR-JOCR}^{gap} = \frac{\mathcal{T}_{JOCR}^U - \mathcal{T}_{SHCR}^U}{\mathcal{T}_{SHCR}^U} \times 100\%$) between the joint optimization approach and sequential heuristic proposed in the literature.

It is evident from Table 1 that the joint optimization approach outperforms the sequential heuristic method for all the test instances evaluated. The proposed JOCR-U model achieves an average cost reduction of 10%-13% and time savings of 12%-15% over the SHCR-U approach. Besides, allowing unrestricted focal points would achieve substantial savings with respect to cost (\mathcal{C}_{R-U}^{gap}) and time (\mathcal{T}_{R-U}^{gap}) when compared to restricted truck stop locations, which validates Proposition 1 and Corollary 1. Thus, the results demonstrate the dominance of the JOCR-U model, thereby highlighting the importance of adopting unrestricted focal points and joint optimization approach.

Moreover, the benefit of adopting the unrestricted focal point policy is higher when the number of delivery locations per trip is fewer. This may be because of low customer density for smaller N , where the locations are likely to be spatially dispersed, which, in

Table 1
Average minimum total cost and minimum total completion time of test instances.

N	Total cost					Delivery completion time				
	\mathcal{C}_{JOCR}^R	\mathcal{C}_{SHCR}^U	\mathcal{C}_{JOCR}^U	$\mathcal{C}_{SHCR-JOCR}^{gap}$	\mathcal{C}_{R-U}^{gap}	\mathcal{T}_{JOCR}^R	\mathcal{T}_{SHCR}^U	\mathcal{T}_{JOCR}^U	$\mathcal{T}_{SHCR-JOCR}^{gap}$	\mathcal{T}_{R-U}^{gap}
20	135.3	122.2	106.2	-13.1%	-21.5%	5.8	5.4	4.6	-14.8%	-20.7%
25	143.4	138.6	120.0	-13.4%	-16.3%	6.5	6.6	5.5	-15.4%	-15.4%
30	161.6	159.4	142.7	-10.5%	-11.7%	7.2	7.3	6.3	-12.5%	-12.5%
35	170.5	169.2	151.0	-10.8%	-11.4%	7.5	7.8	6.6	-14.3%	-12.0%

Table 2Average cost and time gaps between focal point polices for $N = 20$ and different delivery areas.

Delivery area	Total cost			Delivery completion time		
	\mathcal{C}_{JOCR}^R	\mathcal{C}_{JOCR}^U	\mathcal{C}_{R-U}^{gap}	\mathcal{T}_{JOCR}^R	\mathcal{T}_{JOCR}^U	\mathcal{T}_{R-U}^{gap}
30 × 30	135.3	106.2	-21.5%	5.8	4.6	-20.7%
25 × 25	110.7	93.8	-15.3%	4.8	4.1	-14.6%
20 × 20	85.9	77.8	-9.4%	3.8	3.4	-10.5%
15 × 15	63.9	61.3	-4.1%	2.7	2.6	-3.7%

turn, makes a customer location an unattractive (or inefficient) focal point. Conversely, as N increases, the customer density also escalates and subsequently improves the likelihood of coinciding the best focal point coordinate with a delivery location. To validate our findings, the impact of customer density is examined by solving the min-cost and min-time models for four different densities, which are generated by varying the delivery area and fixing N to 20. Table 2 presents the average values of minimum cost and time across 10 replications for different customer densities and focal point policies. It is evident that both \mathcal{C}_{R-U}^{gap} and \mathcal{T}_{R-U}^{gap} decreases with increasing customer density, and is less than 5% when the delivery area is 15×15 miles² with 20 customer locations. Thus, as the spatial density of delivery locations increases, the solutions obtained by both JOCR models become more comparable.

5.2.1. Effect of varying maximum allowable truck stops

In our experimental setup discussed in Section 5.1, we empirically set the value of the threshold parameter (κ) as 3 to control the number of allowable clusters (\hat{K}). This value was sufficient in many cases evaluated, but in certain replications, the number of focal points established was equal to \hat{K} . In this section, we assess the impact of changing the κ for one such replication. Table 3 shows the objective value and CPU time for a replication with 35 locations at different κ values. It can be observed that the rate of improvement in the objective function decreases gradually before plateauing as κ is increased. In particular, when κ exceeds 3, the total cost for JOCR-R and JOCR-U improves marginally (<1.5%), while completion time for both the models remains unchanged. However, in all cases, the running time of the mathematical models increases exponentially for each unit rise in κ . Hence, if the parameter \hat{K} is fully utilized in the best solution, then increasing it may achieve a better outcome, but only at the expense of significantly higher computational time. Thus, the decision-maker must choose a reasonable value of κ by taking into account the time available for planning the last-mile routing.

5.2.2. Computational time and effect of warm-starting

The proposed JOCR-U model could become computationally expensive for large instances as the search space increases considerably. In this section, the computational time required for both the proposed models are studied, along with the effect of acceleration techniques (KBC, WS, and KBC + WS) on JOCR-U running times. Table 4 compiles these results and also provides the time taken for the linearized JOCR-U to achieve 5%, 1%, and 0% gap to the *best solution* (may not be optimal due to linear approximation of the Euclidean distance). Note that if a WS acceleration technique is used, then the computational time also includes the time required to obtain the initial solution. As expected, the running times increase exponentially for both the models as N increments.

Nevertheless, the JOCR-R model is extremely fast since the running time is always in the order of seconds for all the evaluated instances. Unsurprisingly, the JOCR-U takes a longer computational time to achieve the best solution. However, all the proposed acceleration techniques accomplish a considerable reduction in the computational time as opposed to no acceleration, especially the combination of KBC and WS achieves up to 24% faster solution time for large instances. Another interesting finding is that the JOCR-U model can achieve 5% gap to the *best solution* within a few seconds for all the test instances, and an even higher resolution of 1% gap in about 10 min. Besides, the solution obtained in this reasonable amount of time is still substantially better than the JOCR-R and

Table 3Objective value and CPU time of an instance at different threshold values (κ) for both proposed JOCR models.

		$\kappa = 1$		$\kappa = 2$		$\kappa = 3$		$\kappa = 4$		$\kappa = 5$	
		Objective Value	CPU time								
min-cost	JOCR-R	183.2	1	177.0	3	170.3	6	168.5	18	167.8	66
	JOCR-U	167.4	107	159.2	874	150.8	15,486	148.6	54,143	147.9	>100,000
min-time	JOCR-R	8.0	8	7.8	68	7.4	144	7.4	206	7.4	473
	JOCR-U	7.0	74	6.7	324	6.4	8,117	6.4	34,167	6.4	>100,000

Table 4
Average computational time (in seconds) of the proposed JOCR mathematical models for both objective functions.

N	Without acceleration	JOCR-R				JOCR-U				Until the best solution			
		Until the optimal solution		Until 5% gap to best solution		Until 1% gap to best solution		Until the best solution		Without KBC and WS acceleration		With KBC and WS acceleration	
		Without acceleration	With KBC and WS	With KBC	With WS	Without acceleration	With KBC	With WS	With KBC and WS	Without acceleration	With KBC	With WS	With KBC and WS
min-cost	20 <1	<1	<1	25	21	18	16	54	49	46	43	46	43
	25 1	17	15	68	59	55	49	97	87	84	80	84	80
	30 4	91	81	73	23	211	205	182	876	791	784	784	733
	35 9	163	117	94	944	748	726	679	19,837	17,502	17,410	17,410	14,952
min-time	20 <1	<1	<1	8	5	3	2	9	7	6	4	54	46
	25 5	18	16	13	39	32	29	27	63	61	51	51	46
	30 47	39	31	28	24	88	72	66	161	139	133	133	122
	35 118	138	98	93	87	867	595	577	545	8,437	7,701	7,678	7,678

Table 5
Experimental factor settings.

Experimental Factors	Symbol	Levels	Settings
Number of delivery locations	N	2	20, 50
Percent of delivery locations that must be served by a truck	ξ	2	0%, 20%
Velocity of drone (in mph)	V^D	2	15, 35
Flight range of drone (in miles)	F_l	2	7.5, 12.5

SHCR-U models. Since delivery operations are characterized by fast rhythm, the ability of our models to provide high-quality solutions within a short time period could be a benefit to logistics practitioners in real-life situations.

5.3. Sensitivity analysis

In order to test the performance of the proposed JOCR models beyond the existing instances and identify the impact/sensitivity of the key parameters, we evaluate different cases by exploring additional scenarios. The analysis of previous test instances indicates that both min-cost and min-time objectives are substantially affected by the number of customer locations (N). In addition, the proportion of truck-only locations (ξ) and drone characteristics, such as drone velocity (V^D) and flight range (F_l), can also affect the system performance measures. To understand the influence of these four parameters, we consider two levels for each of them (low and high values) as shown in [Table 5](#), and generate a total of 16 different scenarios (i.e., 2^4 possible combinations using the two levels of the four parameters). Besides, we assume each scenario to be independent of each other. For example, one case may assess the use of short-range low-velocity drones to serve a set of locations, while another scenario may investigate the effect of long-range high-speed drones on the performance measures.

For each scenario, ten replications are generated and solved using the proposed JOCR models and the SHCR approach resulting in a total of 960 cases (16 scenarios \times 10 replications \times 2 objectives \times 3 models). [Table 6](#) summarizes the results of the 16 scenarios for all models under consideration. Also, similar to [Table 1](#), the impact of using a joint optimization approach over sequential heuristic ($\mathcal{C}_{SHCR-JOCR}^{gap}$ and $\mathcal{T}_{SHCR-JOCR}^{gap}$), and the savings achieved by unrestricted focal point policy over restricted truck stops (\mathcal{C}_{R-U}^{gap} and \mathcal{T}_{R-U}^{gap}) are reported.

For all the scenarios under consideration, the JOCR-U outperforms both the SHCR-U and JOCR-R. On average, using the JOCR-U approach rather than the sequential heuristic, saves about 12% and 15% in total costs and delivery completion time, respectively. Further, unrestricted focal point policy yields an average reduction of 14% in min-cost and 11% in min-time as opposed to restricting the focal points to customer-only locations. Besides, the best performance is achieved when all packages can be delivered by drones ($\xi = 0\%$) that have long-range ($F_l = 12.5$) and high-speed ($V^D = 35$). For serving 50 locations with such a setting, adopting our JOCR-U approach instead of the SHCR-U method provides an average reduction of 21.6% and 32.1% in min-cost and min-time, respectively.

[Table 7](#) summarizes the 16 scenarios based on the low and high levels of each of the four factors. The following insights can be derived from the impact of each factor on the performance measures.

Table 6
Average min-cost and min-time of 10 replications for 16 scenarios.

Scenario	N	ξ	V^D	F_l	Minimum total cost				Minimum delivery completion time					
					\mathcal{C}_{JOCR}^R	\mathcal{C}_{SHCR}^U	\mathcal{C}_{JOCR}^U	$\mathcal{C}_{SHCR-JOCR}^{gap}$	\mathcal{C}_{R-U}^{gap}	\mathcal{T}_{JOCR}^R	\mathcal{T}_{SHCR}^U	\mathcal{T}_{JOCR}^U		
1	20	0%	15	7.5	140.9	122.9	109.1	-11.2%	-22.6%	7.3	7.0	6.2	-11.4%	-15.1%
2	20	0%	15	12.5	117.0	98.3	84.4	-14.1%	-27.9%	6.4	6.9	5.8	-15.9%	-9.4%
3	20	0%	35	7.5	140.9	122.9	109.1	-11.2%	-22.6%	5.4	4.9	4.3	-12.2%	-20.4%
4	20	0%	35	12.5	117.0	98.3	84.4	-14.1%	-27.9%	4.4	4.5	3.3	-26.7%	-25.0%
5	20	20%	15	7.5	155.6	152.2	141.5	-7.0%	-9.1%	8.0	7.4	7.3	-1.4%	-8.8%
6	20	20%	15	12.5	143.6	149.0	132.8	-10.9%	-7.5%	7.3	7.3	6.9	-5.5%	-5.5%
7	20	20%	35	7.5	155.6	152.2	141.5	-7.0%	-9.1%	5.9	5.7	5.3	-7.0%	-10.2%
8	20	20%	35	12.5	143.6	149.0	132.8	-10.9%	-7.5%	5.3	5.6	5.0	-10.7%	-5.7%
9	50	0%	15	7.5	179.9	176.5	151.6	-14.1%	-15.7%	10.1	11.0	8.9	-19.1%	-11.9%
10	50	0%	15	12.5	161.5	171.1	134.2	-21.6%	-16.9%	9.1	10.9	8.3	-23.9%	-8.8%
11	50	0%	35	7.5	179.9	176.5	151.6	-14.1%	-15.7%	7.0	7.9	5.9	-25.3%	-15.7%
12	50	0%	35	12.5	161.5	171.1	134.2	-21.6%	-16.9%	6.3	7.8	5.3	-32.1%	-15.9%
13	50	20%	15	7.5	217.3	219.8	205.4	-6.6%	-5.5%	11.9	11.8	11.1	-5.9%	-6.7%
14	50	20%	15	12.5	211.5	223.5	199.6	-10.7%	-5.6%	11.5	11.7	10.9	-6.8%	-5.2%
15	50	20%	35	7.5	217.3	219.8	205.4	-6.6%	-5.5%	8.3	9.1	7.9	-13.2%	-4.8%
16	50	20%	35	12.5	211.5	223.5	199.6	-10.7%	-5.6%	8.1	9.0	7.8	-13.3%	-3.7%

Table 7

Impact of changing each of the four factors on min-cost/min-time and performance of proposed solution approaches.

Factor	Level	Minimum total cost					Minimum total completion time				
		\mathcal{C}_{JOCR}^R	\mathcal{C}_{SHCR}^U	\mathcal{C}_{JOCR}^U	$\mathcal{C}_{SHCR-JOCR}^{gap}$	\mathcal{C}_{R-U}^{gap}	\mathcal{T}_{JOCR}^R	\mathcal{T}_{SHCR}^U	\mathcal{T}_{JOCR}^U	$\mathcal{T}_{SHCR-JOCR}^{gap}$	\mathcal{T}_{R-U}^{gap}
N	20	139.3	130.6	117.0	-10.8%	-16.8%	6.3	6.2	5.5	-11.4%	-12.5%
	50	192.6	197.7	172.7	-13.2%	-10.9%	9.0	9.9	8.3	-17.5%	-9.1%
ξ	0%	149.8	142.2	119.8	-15.3%	-20.8%	7.0	7.6	6.0	-20.8%	-15.3%
	20%	182.0	186.1	169.8	-8.8%	-6.9%	8.3	8.5	7.8	-8.0%	-6.3%
V^D	15	165.9	164.2	144.8	-12.0%	-13.8%	9.0	9.3	8.2	-11.2%	-8.9%
	35	165.9	164.2	144.8	-12.0%	-13.8%	6.3	6.8	5.6	-17.6%	-12.7%
F_l	7.5	173.4	167.9	151.9	-9.7%	-13.2%	8.0	8.1	7.1	-11.9%	-11.7%
	12.5	158.4	160.5	137.8	-14.3%	-14.5%	7.3	8.0	6.7	-16.9%	-9.9%

- **Number of locations (N):** As expected, the total cost and delivery completion time become higher if more customers (N) are involved. However, it is to note that when N is increased by 150% (from 20 to 50 locations), the delivery completion time increased only by about 47% for both JOCR-R and JOCR-U policies. Similarly, the best total costs for restricted and unrestricted JOCR increased approximately by 38% and 48%, respectively. In other words, cost and time do not change proportionally with N , but increase at a slower rate. Hence, practitioners should include many orders per truck trip whenever possible. Another advantage is that the performance (total cost and completion time) of JOCR-U relative to SHCR-U is further enhanced for larger values of N .
- **Proportion of truck-only locations (ξ):** Requiring 20% of the delivery locations to be served by trucks instead of 0% has a greater negative impact on the JOCR-U policy (41% and 29% increase in cost and time, respectively) as opposed to JOCR-R policy (21% and 18% increase in cost and time, respectively). The JOCR-R policy is less affected because it already applies the restriction of customer-only focal points by default. Although the JOCR-U policy has superior performance over JOCR-R and SHCR-U in all the cases, the improvement achieved is relatively lower when ξ is increased to 20%. Thus, to achieve faster and cheaper delivery, it is ideal to limit the proportion of truck-only locations to a smaller value. This may be achieved by investing in drones with higher payload capacity so that it can deliver a heavier package instead of a truck.
- **Velocity of drone (V^D):** Increasing the drone velocity reduces the delivery completion time by about 30% for both JOCR-R and JOCR-U policies. Besides, the min-time JOCR-U model achieves greater improvement over the other two models with faster drones. Therefore, if the company is keen on achieving faster delivery, then investing in high-speed drones is recommended.
- **Flight range of drone (F_l):** Operating drones with longer flight range (12.5 miles instead of 7.5 miles) reduces the total cost and completion time by 9% and 7%, respectively, for both JOCR-R and JOCR-U models. Moreover, the percent improvement in cost and time achieved by the JOCR-U model over SHCR-U is considerably greater for long-range drones. However, the purchasing cost of such drones are likely to be more expensive than short-range drones. Therefore, decision-makers should consider a trade-off between the additional purchasing cost of long-range drones and their benefits.

5.4. Influence of objective functions on clustering and routing decisions

In this section, we visually illustrate the influence of objective functions on clustering and routing decisions. A test case with 25 locations is chosen, as shown in Fig. 4, in which the last three customer locations must be served only by a truck, while the remaining 22 locations can be served either by a drone or a truck. Fig. 4(a) and (b) represent the best way to cluster the locations and route the truck for the min-cost and min-time JOCR-R models, respectively. Likewise, Fig. 4(c) and (d) illustrate the best plan for minimizing total cost and completion time in the JOCR-U models, respectively. The min-cost JOCR-R model yields a total cost of \$127.4, and the corresponding delivery completion time for this solution is 6.7 h. On the other hand, the min-time JOCR-R completes the delivery in 5.7 h but results in a higher total cost of \$143.8. A similar trend is observed for the JOCR-U models as well, where the min-cost solution results in lower cost and higher completion time when compared to the min-time solution. Thus, solely optimizing one objective may worsen the performance of the other criterion. If the decision-maker is interested in considering more than one objective, then a multi-criteria optimization approach (as discussed in Section 4.4) is beneficial to obtain the best trade-off solution.

In both JOCR-R and JOCR-U models, the objective function plays a key role in deciding the number of drones carried by truck. The solutions of min-time objective (Fig. 4(b) and (d)) utilizes the entire fleet of drones and deploys them only at certain truck stops to achieve faster delivery completion time. Conversely, the solutions obtained using min-cost objective (Fig. 4(a) and (c)) uses fewer drones and dispatches them at every truck stop, and therefore completes the delivery operation with lower cost but longer completion time. In addition, the min-cost objective for JOCR-R and JOCR-U results in shorter drone travel distance compared to the min-time objective. To achieve this, one of the key strategies adopted by the min-cost models is to serve more locations by a truck compared to the min-time models. Besides routing, the focal point location also plays a key role in minimizing the drone travel distance, especially for unrestricted focal point policy. For example, consider locations 12 and 17 in Fig. 4(c) and (d). The minimum drone distance for the min-cost model is achieved by making these locations almost along a straight line with their focal point (location 19). On the

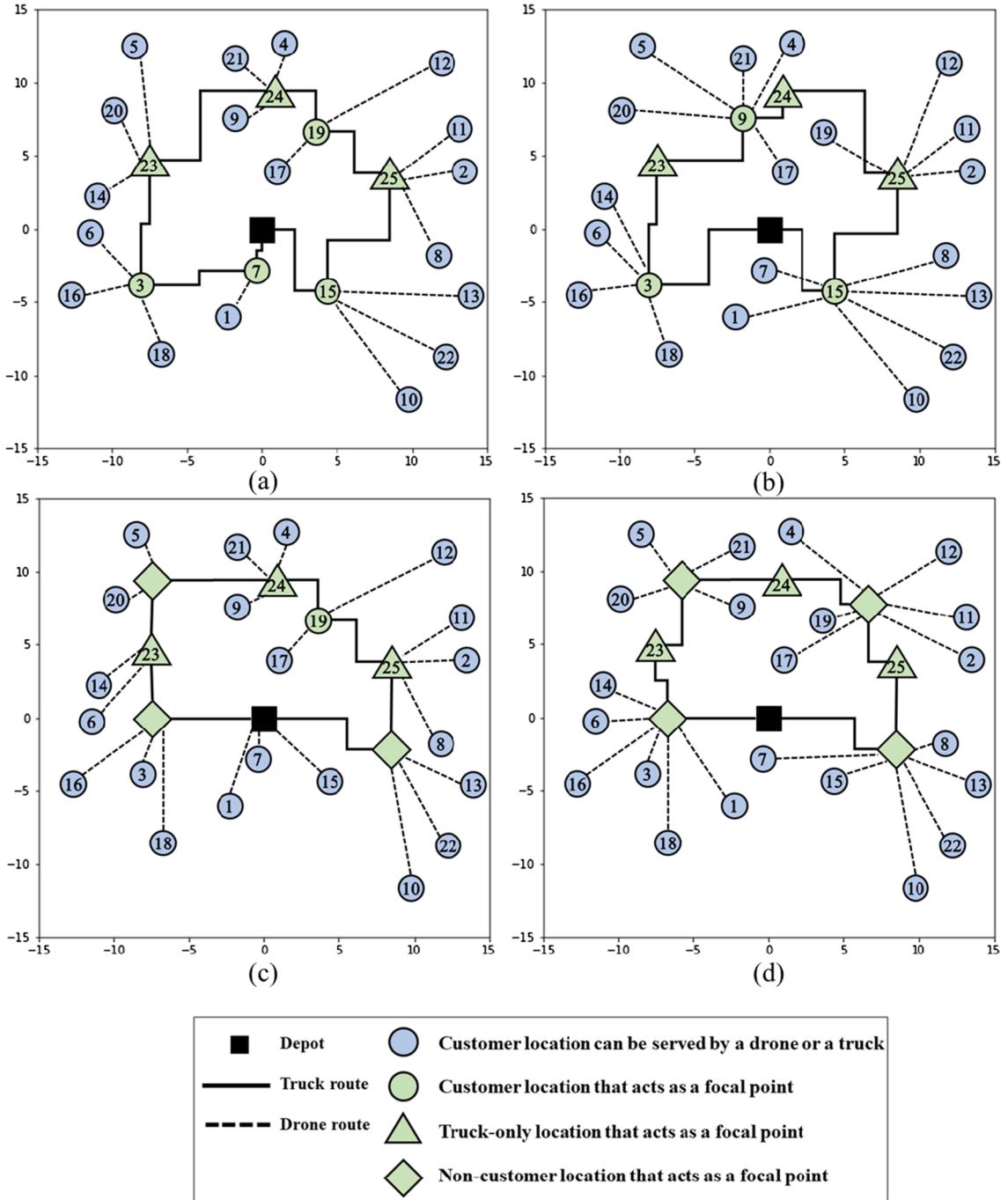


Fig. 4. Solutions of an example using the proposed models: (a) min-cost JOCR-R ($\mathcal{C}_{JOCR}^R = 127.4$ and $\mathcal{T}_{JOCR}^R = 6.7$); (b) min-time JOCR-R ($\mathcal{C}_{JOCR}^R = 143.8$ and $\mathcal{T}_{JOCR}^R = 5.7$); (c) min-cost JOCR-U ($\mathcal{C}_{JOCR}^U = 108.6$ and $\mathcal{T}_{JOCR}^U = 6.9$); (d) min-time JOCR-U ($\mathcal{C}_{JOCR}^U = 127.1$ and $\mathcal{T}_{JOCR}^U = 4.9$).

other hand, the min-time model (in Fig. 4 (d)) minimizes the maximum drone flight distance by locating a focal point to be approximately halfway between these two locations. Finally, both the objectives aim to lower the truck travel distance, and this is also influenced by the location of truck stops. For instance, in Fig. 4 (c), the non-customer focal points to the north and the south of location 23 are optimized to share the same truck path, as moving them any further towards the left would increase the distance traveled by truck.

Aside from these differences, Fig. 4 also highlights some of the unique features of the proposed model. First, the truck-only locations function as cluster focal points whenever it can improve the objective value (e.g., location 25 in Fig. 4 (a), (b), and (c)). Likewise, the depot is also used as a focal point if it is beneficial (e.g., to locations 1, 7, and 15 in Fig. 4 (c)). Finally, the JOCR-U

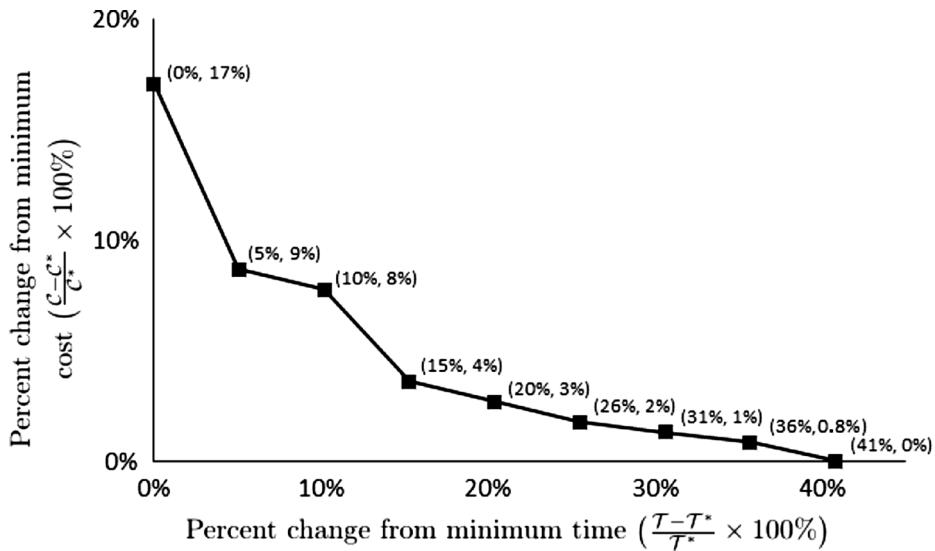


Fig. 5. Conflicting min-cost and min-time objectives of an example for JOCR-R and JOCR-U.

models use the customer location as focal points when appropriate and choose a non-customer location only when it improves the objective value (Fig. 4 (c) and (d)). All of these demonstrate the holistic approach of the proposed JOCR models to achieve the best clustering and routing decisions and also provide insights for the development of novel heuristics in future research.

5.5. Obtaining Pareto optimal solutions

In this section, we illustrate the ability of our proposed models to handle multiple objectives and obtain Pareto-optimal solutions. We choose the same instance illustrated in Fig. 4 as it clearly depicts the conflicting nature of the two objectives. For example, the JOCR-U model can achieve the lowest cost (C^*) of \$108.6 for the instance depicted in Fig. 4, but only at the expense of increasing the delivery duration by two hours from the fastest achievable time (T^*) of 4.9 h. However, lowering the completion time would increase the total costs. Therefore, we use the ϵ -constraint method to obtain a set of best trade-off solutions. Fig. 5 presents the best compromise solutions for the chosen example, where the min-cost JOCR-U model is solved repeatedly after incorporating the completion time constraint and changing its limit from 4.9 to 6.9 in increments of 0.25 h.

The results demonstrate that focusing solely on minimizing total cost increases the corresponding completion time by 41% from T^* . On the other hand, focusing only on reducing completion time worsens the total cost by 17% when compared to C^* . According to the decision maker's priorities, a single objective can be adopted while sacrificing the potential reduction in the other, or a trade-off can be considered between the two objectives. For example, it can be observed from Fig. 5 that investing 9% more than C^* would help us complete the delivery 36% faster than the worst-case situation or only 5% slower than the best-case result of 4.9 h. Thus, this approach would enable practitioners to deal with two objectives simultaneously and facilitate decision-making.

5.6. Impact of multiple drone deliveries per cluster

Similar to prior relevant literature, all our previous analysis considered a drone to make at most one delivery in a cluster (Mourelo Ferrandez et al., 2016; Chang and Lee, 2018). In this section, we investigate the impact of allowing multiple drone deliveries per cluster on the number of drones required, drone utilization, and total cost. The drone utilization is the proportion of the average drone flight time in a cluster to the total time spent in that cluster. As stated in Section 4.5, the delivery time is restricted to a user-

Table 8

Average total cost, number of drones, and utilization percentage assuming a drone performs a single delivery within a cluster vs. multiple deliveries.

N	Number of drones				Utilization %				Minimum total cost			
	Single delivery		Multiple deliveries		Single delivery		Multiple deliveries		Single delivery		Multiple deliveries	
	JOCR-R	JOCR-U	JOCR-R	JOCR-U	JOCR-R	JOCR-U	JOCR-R	JOCR-U	JOCR-R	JOCR-U	JOCR-R	JOCR-U
20	3.1	3.3	2.5	2.8	51%	59%	63%	70%	135.3	106.2	132.1	103.6
25	3.1	3.5	2.3	3.1	52%	57%	70%	64%	143.4	120.0	141.9	117.7
30	3.6	4	3.0	3.7	56%	58%	67%	63%	161.6	142.7	158.7	141.3
35	4.1	4.6	3.1	3.5	52%	53%	68%	69%	170.5	151.0	166.0	145.8

specified upper limit. Otherwise, the model will often select only one drone to lower the costs as shown in [Proposition 2](#). To ensure an equitable comparison, the time constraint in the multi-trip model is limited to the corresponding completion time yielded by min-cost JOCR models.

[Table 8](#) compiles the results comparing single and multiple drone trips per cluster. Allowing the same drone to visit many customer locations in a cluster decreases the total number of drones employed by 22% and 15% on average for JOCR-R and JOCR-U, respectively. As a result of using fewer drones, up to 4% lower cost is achieved by the multi-trip model while ensuring a delivery completion time that is similar to the situation involving one drone trip per cluster. Most importantly, the drone utilization becomes considerably higher (an average increase of 27% for JOCR-R and 18% for JOCR-U) when multiple drone deliveries are allowed within a cluster. In summary, the variant of allowing more than one drone delivery per cluster has proven to be beneficial for both the policies. Hence, practitioners must consider multiple drone delivery per cluster, especially when adopting the policy of restricted focal points.

6. Conclusions and future research

This paper considers the problem of last-mile delivery using multiple drones and a single truck. We adopt the approach of grouping customer locations into non-overlapping clusters and routing the truck via each cluster's focal point to facilitate simultaneous drone deliveries in that cluster. Unlike the common sequential approaches in the literature, this paper presents a new integrated method for clustering and routing decisions. We propose mathematical programming models to solve the problem for two different policies - (i) restricting truck stops to a customer-only location (JOCR-R) and (ii) allowing truck stops to be anywhere in the delivery region (JOCR-U). Besides, we formulate our models to consider the two common objectives in the literature, namely, minimizing total costs and minimizing delivery completion time. Thus, we provide the flexibility to treat the problem as a single objective to optimize one of the performance measures or handle it as a multi-objective to achieve the best compromise solutions. Furthermore, we formulate a variant of our proposed models to allow multiple drone deliveries in a cluster. In addition, we introduce a knowledge-based constraint and machine learning-based heuristic to accelerate the JOCR-U models.

An extensive numerical analysis is conducted for the proposed models and benchmarked with a recently introduced sequential heuristic approach in the literature. Solving the test instances independently for the two objectives revealed that the proposed joint optimization approach outperforms the sequential heuristic method for all the cases. Besides, allowing the focal points to be anywhere in the delivery region instead of restricting it to a customer location provides substantial savings with respect to cost and delivery completion time. In addition, a sensitivity analysis of key model parameters is also conducted to establish their independent effects on the performance measures. Also, the ϵ -constraint method achieves the best compromise between the two objectives. Finally, we found that allowing a drone to make multiple deliveries in a cluster utilizes the drone fleet efficiently. Moreover, such a setting requires fewer drones while achieving a completion time that is similar to the case of a single trip per drone within a cluster. Overall, numerous insights drawn from our analysis could aid the practitioners and researchers in the effective routing of one truck and multiple drones.

In the era of supply chain 4.0, the models developed in this research could enable industries to meet the rapidly growing requirements of consumers and improve the last-mile delivery operations. While our model considered the truck to wait at the focal points to receive the dispatched drones, future research could examine the situation of coordinating the truck and drones so that it can be dispatched and collected at different focal points or along the truck route. Moreover, our approach for single-truck multi-drone routing can be extended to multi-truck multi-drone, in which the coordination between the fleet of vehicles would be more challenging. With respect to methodology, future research could consider the development of efficient heuristic or meta-heuristic approaches that could solve large instances quickly and accurately.

Appendix A. Linearization of JOCR-U MINLP model

In this section, we linearize the objective function and non-linear constraints of the proposed MINLP model. The following additional notations are introduced to avoid non-linearity and reformulate the problem under study as a MILP model.

Parameters

P	number of planes for finding approximate Euclidean distance
θ	rotation angle between two consecutive planes
M	large positive number

Decision variables

d_{lk}^{x-E}	horizontal axis (or x-axis) length of Euclidean distance between delivery location $l \in \mathcal{L}$ and cluster focal point $k \in \mathcal{K}$ (i.e., $d_{lk}^{x-E} = A_l - a_k $)
d_{lk}^{y-E}	vertical axis (or y-axis) length of Euclidean distance between delivery location $l \in \mathcal{L}$ and cluster focal point $k \in \mathcal{K}$ (i.e., $d_{lk}^{y-E} = B_l - b_k $)
$d_{kk'}^{x-R}$	horizontal axis (or x-axis) length of rectilinear distance between cluster focal points k and $k' \in \mathcal{K}$ (i.e., $d_{kk'}^{x-R} = a_k - a_{k'} $)
$d_{kk'}^{y-R}$	vertical axis (or y-axis) length of rectilinear distance between cluster focal points k and $k' \in \mathcal{K}$ (i.e., $d_{kk'}^{y-R} = b_k - b_{k'} $)

There are many endeavors in the literature to find a good linear approximation to the Euclidean distance ([Hale et al., 2012](#);

O'Muirgheasa et al., 2013). In this research, we adopt the recent linearization technique developed by Xie et al. (2018) to linearize constraint (15) as it has shown to produce a highly accurate approximation of the Euclidean distance metric (i.e., an error of less than –0.01% can be achieved). Their technique computes the Euclidean distance between two locations using x -axis and y -axis components and slope angle between the two locations. While the x -axis and y -axis components can be depicted using linear constraints in a mathematical model, the slope angle is challenging to represent linearly. Therefore, different values of slope angles, between 0 and $\frac{\pi}{2}$, are examined. Each of the examined slope angles represents one of the predefined P planes, while a fixed rotation angle (θ) is imposed between every two consecutive planes. In this research, we choose six planes ($P = 6$) and a rotation angle (θ) of 0.2831 radian to achieve <1% error in approximating Euclidean distance (Xie et al., 2018).

Thus, the non-linearity in Eq. (15) can be avoided by adopting a two-step reformulation. First, the horizontal and vertical axis components of the Euclidean distance is represented with a set of equivalent linear constraints (43)–(46). Then, the Euclidean distance is redefined in terms of the x -axis and y -axis components in constraint (47).

$$d_{lk}^{x-E} \geq A_l - a_k \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (43)$$

$$d_{lk}^{x-E} \geq a_k - A_l \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (44)$$

$$d_{lk}^{y-E} \geq B_l - b_k \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (45)$$

$$d_{lk}^{y-E} \geq b_k - B_l \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (46)$$

$$d_{lk}^E \geq d_{lk}^{x-E} \cos(p\theta) + d_{lk}^{y-E} \sin(p\theta) \quad \forall l \in \mathcal{L}, k \in \mathcal{K}, p = 0, 1, 2, \dots, P-1 \quad (47)$$

Likewise, the equivalent linearization of Eq. (22) is achieved by linear constraints (48)–(51).

$$d_{kk'}^R \geq a_k - a_{k'} \quad \forall k, k' \in \mathcal{K} \quad (48)$$

$$d_{kk'}^R \geq a_{k'} - a_k \quad \forall k, k' \in \mathcal{K} \quad (49)$$

$$d_{kk'}^{y-R} \geq b_k - b_{k'} \quad \forall k, k' \in \mathcal{K} \quad (50)$$

$$d_{kk'}^{y-R} \geq b_{k'} - b_k \quad \forall k, k' \in \mathcal{K} \quad (51)$$

Constraint (16) is also non-linear as it involves the multiplication of two decision variables ($x_{lk} \times d_{lk}^D$). The purpose of constraint (16) is to ensure that the left hand side becomes zero if the delivery location $l \in \mathcal{L}$ is not assigned to the focal point $k \in \mathcal{K}$, and exactly equal to the distance between them if it is assigned. The same characteristics can be achieved without non-linearity by modifying constraint (47) as constraint (52). As a result, the non-linear constraint (16) can now be rewritten as linear constraint (53). Similarly, the second term of the objective function can be rewritten as ($C_2^D \sum_{l \in \mathcal{L}} \sum_{k \in \mathcal{K}} 2d_{lk}^E$).

$$d_{lk}^E \geq d_{lk}^{x-E} \cos(p\theta) + d_{lk}^{y-E} \sin(p\theta) - M(1 - x_{lk}) \quad \forall l \in \mathcal{L}, k \in \mathcal{K}, p = 0, 1, 2, \dots, P-1 \quad (52)$$

$$d_{lk}^E \leq F_l q_{lk} \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (53)$$

The third term of the objective function (13) is analogous to the objective function of the standard TSP model. However, in our approach, both $y_{kk'}$ and $d_{kk'}^R$ are decision variables. The non-linearity due to the multiplication of these two variables could be circumvented by fixing the binary variable $y_{kk'}$. Since the locations of cluster focal points (i.e., their coordinates) are not determined yet, any feasible truck route sequence could be chosen without affecting the quality of results. For example, Eq. (54) can be used to provide a feasible truck route, which gives a route: $k_0 \rightarrow k_1 \rightarrow k_2 \rightarrow \dots \rightarrow k_{|\mathcal{K}|} \rightarrow k_0$. In addition to overcoming non-linearity, using a fixed sequence of the truck route as an input leads to the exclusion of constraints (19)–(21), thereby decreasing the problem complexity.

$$y_{kk'} = 1 \quad \forall k, k' \in \mathcal{K}: (k = k' - 1) \vee (k = |\mathcal{K}| \wedge k' = 0) \quad (54)$$

Appendix B. Linearization of min-time JOCR models

The objective function of the min-time JOCR models (Eqs. (26) and (27)) requires the minimization of the maximum drone flight time within each cluster, thereby leading to a non-linear term. An exact linearization of this minmax term can be achieved by introducing a new variable (t_l and t_k for JOCR-R and JOCR-U models, respectively) to represent the maximum drone completion time per cluster, and adding the constraint (55) and (56) in JOCR-R and JOCR-U models, respectively. Thus, the objective functions (26) and (27) can be rewritten as in functions (57) and (58), respectively.

$$t_l \geq x_{ll} \times (2D_{ll}^E/V^D + S_l) \quad \forall l \in \mathcal{L} \quad (55)$$

$$t_k \geq 2d_{kk}^E/V^D + x_{lk} \times S_l \quad \forall l \in \mathcal{L}, k \in \mathcal{K} \quad (56)$$

$$\text{Minimize} \quad \mathcal{T}_{JOCR}^R = \sum_{l \in \mathcal{L}} t_l + \sum_{l' \in \mathcal{L}} \sum_{l \in \mathcal{L}} y_{l'l} \times D_{l'l}^R / V^T \quad (57)$$

$$\text{Minimize} \quad \mathcal{T}_{JOCR}^U = \sum_{k \in \mathcal{K}} t_c + \sum_{k \in \mathcal{K}} \sum_{k' \in \mathcal{K}} y_{kk'} \times d_{kk'}^R / V^T \quad (58)$$

Appendix C. Iterative k-means algorithm

The k-means clustering technique is an unsupervised machine learning algorithm, which can be used in logistics operations to partition a set of customer locations into k clusters. To establish a feasible cluster of customer locations, the following two conditions must be satisfied - (i) distance between cluster focal point and each assigned customer location should be within the flying range and (ii) number of locations per cluster must be less than the drones carried by truck. Thus, the number of clusters, k , increases iteratively until both conditions are satisfied, then the algorithm terminates. Following the notations in Section 4.2, the required input data for the algorithm are: depot location (A_{l_0}, B_{l_0}), set of customer locations that can be served by a drone/truck (\mathcal{L}^D) and their coordinates (A_l, B_l), drone flight range for each location ($R_l, \forall l \in \mathcal{L}^D$), and maximum number of drones can be carried by the truck (G). The pseudocode of the iterative k-means technique is provided in Algorithm 2.

Algorithm 2. Iterative k-means clustering

```

1: Initialize maximum allowable clusters to 1, ( $\hat{K} \leftarrow 1$  and  $\mathcal{K} = \{k_0, k_1\} \dagger$ ), and assign a random coordinate to the focal point  $k_1$ ,  $(a_{k_1}, b_{k_1})$ 
2: for each location  $l \in \mathcal{L}^D$  do
3:   Assign location  $l$  to the focal point  $k_1$  ( $x_{l,k_1} = 1$ )
4:   Compute the distance between the focal point  $k_1$  and delivery location  $l$  ( $d_{lk_1}^E$ )  $d_{lk_1}^E = \sqrt{(A_l - a_{k_1})^2 + (B_l - b_{k_1})^2}$ 
5: end for
6: while  $(x_{lk} \times d_{lk}^E > F_l, \forall l \in \mathcal{L}^D, k \in \mathcal{K})$  or  $(\sum_{l \in \mathcal{L}^D} x_{lk} > G, \forall k \in \mathcal{K})$ 
7:   Update  $\hat{K} \leftarrow \hat{K} + 1$ 
8:   Initialize iteration  $i = 1$ ;
9:   Randomly set  $\hat{K}$  focal points for iteration  $i$   $((a_{k_2,i}, b_{k_2,i}), (a_{k_3,i}, b_{k_3,i}), ..., (a_{k_{\hat{K}},i}, b_{k_{\hat{K}},i}))$ 
10:  while  $a_{k,i} \neq a_{k,i-1}, \forall k \in \mathcal{K}$  and  $b_{k,i} \neq b_{k,i-1}, \forall k \in \mathcal{K}$  do
11:    for each location  $l \in \mathcal{L}^D$  do
12:      for each cluster focal point  $k \in \mathcal{K}$  do
13:        Calculate distance between focal point  $k$  and delivery location  $l$  ( $d_{lk}^E$ )  $d_{lk}^E = \sqrt{(A_l - a_{k,i})^2 + (B_l - b_{k,i})^2}$ 
14:      end for
15:      Assign delivery location  $l$  to the nearest cluster focal point  $k$  ( $x_{lk}$ )  $x_{lk,i} = 1$  for focal point  $\operatorname{argmin}_k \{d_{lk}^E\}$  and 0 otherwise
16:    end for
17:     $i \leftarrow i + 1$ 
18:     $a_{k,i} = \left( \frac{\sum_{l \in \mathcal{L}^D} x_{l,k,i-1} \times a_{k,i-1}}{\sum_{l \in \mathcal{L}^D} x_{l,k}} \right)$ 
19:     $b_{k,i} = \left( \frac{\sum_{l \in \mathcal{L}^D} x_{l,k,i-1} \times b_{k,i-1}}{\sum_{l \in \mathcal{L}^D} x_{l,k}} \right)$ 
20:  end while
21: end while
22:  $a_k \leftarrow a_{k,i} \quad \forall k \in \mathcal{K}$ 
23:  $b_k \leftarrow b_{k,i} \quad \forall k \in \mathcal{K}$ 
24: return focal points of clusters  $(a_k, b_k)$  and assignments of delivery locations to clusters  $(x_{lk} \quad \forall l \in \mathcal{L}, k \in \mathcal{K})$ 

```

[†]Note: Cluster focal point k_o is set as the depot by default with coordinates $(a_{k_o}, b_{k_o}) = (A_{l_0}, B_{l_0})$

References

- Agatz, N., Bouman, P., Schmidt, M., 2018. Optimization approaches for the traveling salesman problem with drone. *Transport. Sci.* 52 (4), 965–981. <https://doi.org/10.1287/trsc.2017.0791>.
- Amazon, 2016. Amazon Prime Air. URL: <https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011>.
- Bäckman, A., Hollenberg, J., Svensson, L., Ringh, M., Nordberg, P., Djärv, T., Forsberg, S., Hernborg, O., Claesson, A., 2018. Drones for provision of flotation support in simulated drowning. *Air Medical J.* 37 (3), 170–173. <https://doi.org/10.1016/j.amj.2018.01.007>.
- Bouman, P., Agatz, N., Schmidt, M., 2018. Dynamic programming approaches for the traveling salesman problem with drone. *Networks* 72 (4), 528–542. <https://doi.org/10.1002/net.21864>.
- Calvete, H.I., Galé, C., Irazo, J.A., 2013. An efficient evolutionary algorithm for the ring star problem. *Eur. J. Oper. Res.* 231 (1), 22–33. <https://doi.org/10.1016/j.ejor.2013.05.013>.
- Campbell, J.F., Sweeney II, D.C., Zhang, J., 2017. Strategic Design for Delivery with Trucks and Drones, Tech. Rep.
- 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. <https://doi.org/10.1016/j.eswa.2018.06.027>.

- [10.1016/j.eswa.2018.03.032.](https://doi.org/10.1016/j.eswa.2018.03.032)
- Derigs, U., Pullmann, M., Vogel, U., 2013. Truck and trailer routing - Problems, heuristics and computational experience. *Comput. Oper. Res.* 40 (2), 536–546. <https://doi.org/10.1016/j.cor.2012.08.007>.
- DHL, 2014. DHL parcelcopter launches initial operations for research purposes. URL: http://www.dhl.com/en/press/releases/releases_2014/group/dhl_parcelcopter_launches_initial_operations_for_research_purposes.html.
- Fehr, Peers, 2019. Drone Delivery — Fehr & Peers. URL: <https://www.fehrandpeers.com/drone-delivery/>.
- Gross, D., 2013. Amazon's drone delivery: How would it work?. URL: <https://www.cnn.com/2013/12/02/tech/innovation/amazon-drones-questions/>.
- Guglielmo, C., 2013. Turns Out Amazon, Touting Drone Delivery, Does Sell Lots of Products That Weigh Less Than 5 Pounds. URL: <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>.
- Ha, Q.M., Deville, Y., Pham, Q.D., Ha, M.H., 2018. On the min-cost Traveling Salesman Problem with Drone. *Transport. Res. Part C: Emerg. Technol.* 86 (May 2016), 597–621. <https://doi.org/10.1016/j.trc.2017.11.015>. arXiv:1509.08764.
- Hale, T.S., Hipkin, I., Huqb, F., 2012. An improved facility layout construction method. *Int. J. Prod. Res.* 50 (15), 37–41.
- Hong, I., Kuby, M., Murray, A.T., 2018. A range-restricted recharging station coverage model for drone delivery service planning. *Transport. Res. Part C: Emerg. Technol.* 90 (October 2017), 198–212. <https://doi.org/10.1016/j.trc.2018.02.017>.
- 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. <https://doi.org/10.1016/j.ijpe.2019.01.010>.
- Joerss, M., Neuhaus, F., Schroder, J., 2016. How customer demands are reshaping last-mile delivery. URL: <https://www.mckinsey.com/industries/travel-transport-and-logistics/our-insights/how-customer-demands-are-reshaping-last-mile-delivery>.
- Karak, A., Abdelghany, K., 2019. The hybrid vehicle-drone routing problem for pick-up and delivery services. *Transport. Res. Part C: Emerg. Technol.* 102 (March), 427–449. <https://doi.org/10.1016/j.trc.2019.03.021>.
- Katariya, M., Chung, D.C.K., Minife, T., Gupta, H., Zahidi, A.A.A., Liew, O.W., Ng, T.W., 2018. Drone inflight mixing of biochemical samples. *Anal. Biochem.* 545 (December 2017), 1–3. <https://doi.org/10.1016/j.ab.2018.01.004>.
- Kirschstein, T., 2020. Comparison of energy demands of drone-based and ground-based parcel delivery services. *Transp. Res. Part D* 78<https://doi.org/10.1016/j.trd.2019.102209>.
- Lin, S.W., Yu, V.F., Chou, S.Y., 2009. Solving the truck and trailer routing problem based on a simulated annealing heuristic. *Comput. Oper. Res.* 36 (5), 1683–1692. <https://doi.org/10.1016/j.cor.2008.04.005>.
- Mathew, N., Smith, S.L., Member, S., Waslander, S.L., 2015. Planning Paths for Package Delivery in Heterogeneous Multirobot Teams. *IEEE Trans. Autom. Sci. Eng.* 12 (4), 1298–1308.
- Mercedes-Benz, 2018. The Mercedes-Benz Vision Van. URL: <https://www.mercedes-benz.com/en/mercedes-benz/vehicles/transporter/vision-van/>.
- Mikrokopter, 2018. The MK8-3500 Standard. URL: <http://www.mikrokopter.de/en/products/nmk8stden/nmk8techdaten>.
- Mourelo Fernandez, S., 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. Manage.* 9 (2), 374. <https://doi.org/10.3926/jiem.1929>. URL: <http://www.jiem.org/index.php/jiem/article/view/1929>.
- Murray, C.C., Chu, A.G., 2015. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transport. Res. Part C: Emerg. Technol.* 54, 86–109. <https://doi.org/10.1016/j.trc.2015.03.005>.
- Murray, C.C., Raj, R., 2020. The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. *Transport. Res. Part C: Emerg. Technol.* 110 (February 2019), 368–398. <https://doi.org/10.1016/j.trc.2019.11.003>.
- O'Muirgeasa, C., Kadipasaoğlu, S.N., Khumawala, B.M., 2013. An investigation into the effect of a more accurate measure of distance on the detailed facility layout problem. *J. Stat. Manage. Syst.* 4 (3), 327–340. <https://doi.org/10.1080/09720510.2001.10701046>.
- Oruc, B.E., Kara, B.Y., 2018. Post-disaster assessment routing problem. *Transport. Res. Part B: Methodol.* 116, 76–102. <https://doi.org/10.1016/j.trb.2018.08.002>.
- Otto, A., Agatz, N., Campbell, J., Golden, B., 2018. Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. *Networks* 72 (March), 411–458. <https://doi.org/10.1002/net.21818>.
- Poikonen, S., Wang, X., Golden, B., 2017. The vehicle routing problem with drones: Extended models and connections. *Networks* 70 (1), 34–43. <https://doi.org/10.1002/net.21746>.
- Rabta, B., Wankmüller, C., Reiner, G., 2018. A drone fleet model for last-mile distribution in disaster relief operations. *Int. J. Disaster Risk Reduct.* 28 (February), 107–112. <https://doi.org/10.1016/j.ijdrr.2018.02.020>.
- Raj, A., Sah, B., 2019. Analyzing critical success factors for implementation of drones in the logistics sector using grey-DEMATEL based approach. *Comput. Ind. Eng.* 138<https://doi.org/10.1016/j.cie.2019.106118>. 106118.
- Sacramento, D., Pisinger, D., Ropke, S., 2019. An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. *Transport. Res. Part C: Emerg. Technol.* 102 (March), 289–315. <https://doi.org/10.1016/j.trc.2019.02.018>.
- Salazar-Aguilar, M.A., Ríos-Mercado, R.Z., González-Velarde, J.L., 2011. A bi-objective programming model for designing compact and balanced territories in commercial districting. *Transport. Res. Part C: Emerg. Technol.* 19 (5), 885–895. <https://doi.org/10.1016/j.trc.2010.09.011>.
- Schermer, D., Moeini, M., Wendt, O., 2018. Algorithms for Solving the Vehicle Routing Problem with Drones, Lecture Notes in Artificial Intelligence, vol. 10751, Springer, January 2018, pp. 352–361. <https://doi.org/10.1007/978-3-319-75417-8>.
- Song, B.D., Park, K., Kim, J., 2018. Persistent UAV delivery logistics: MILP formulation and efficient heuristic. *Comput. Ind. Eng.* 120 (February 2017), 418–428. <https://doi.org/10.1016/j.cie.2018.05.013>.
- Stolaroff, J.K., Samaras, C., O'Neill, E.R., Lubbers, A., Mitchell, A.S., Ceperley, D., 2018. Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery. *Nature Commun.* 9 (1), 1–13. <https://doi.org/10.1038/s41467-017-02411-5>.
- Tuerk, M., 2019. Fixing Amazon's Drone Delivery Problem. URL: <https://www.forbes.com/sites/miriamtuerk/2019/05/16/fixing-amazons-drone-delivery-problem/#64f4da774d37>.
- Villegas, J.G., Prins, C., Prodhon, C., Medaglia, A.L., Velasco, N., 2013. A matheuristic for the truck and trailer routing problem. *Eur. J. Oper. Res.* 230 (2), 231–244. <https://doi.org/10.1016/j.ejor.2013.04.026>.
- Wang, Z., Sheu, J.B., 2019. Vehicle routing problem with drones. *Transport. Res. Part B: Methodol.* 122, 350–364. <https://doi.org/10.1016/j.trb.2019.03.005>.
- Wang, X., Poikonen, S., Golden, B., 2017. The vehicle routing problem with drones: several worst-case results. *Optim. Lett.* 11 (4), 679–697. <https://doi.org/10.1007/s11590-016-1035-3>.
- X-Company, 2016. Project Wing. URL: <https://x.company/projects/wing>.
- Xie, Y., Zhou, S., Xiao, Y., Kulturel-konak, S., Konak, A., 2018. A β -accurate linearization method of Euclidean distance for the facility layout problem with heterogeneous distance metrics. *Eur. J. Oper. Res.* 265 (1), 26–38. <https://doi.org/10.1016/j.ejor.2017.07.052>.
- Yurek, E.E., Ozmutlu, H.C., 2018. A decomposition-based iterative optimization algorithm for traveling salesman problem with drone. *Transport. Res. Part C: Emerg. Technol.* 91 (April), 249–262. <https://doi.org/10.1016/j.trc.2018.04.009>.