Annals of Operations Research (2024) 339:693–739
https://doi.org/10.1007/s10479-023-05418-y

ORIGINAL RESEARCH

Collaborative truck multi-drone delivery system considering drone scheduling and en route operations

Teena Thomas[1,4] · Sharan Srinivas[2,3] · Chandrasekharan Rajendran[1]

Accepted: 23 May 2023 / Published online: 15 June 2023

Abstract
The integration of drones into the conventional truck delivery system has gained substantial attention in the business and academic communities. Most existing works restrict the launch and recovery of unmanned aerial vehicles (UAVs) to customer locations (or nodes) in the delivery network. Nevertheless, emerging technological advances can allow drones to autonomously launch/land from a moving vehicle. In addition, majority of the current literature assumes multiple UAVs to be deployed and/or recovered simultaneously, thereby ignoring the associated scheduling decisions, which are essential to ensure safe, collision-free operations. This research introduces the single truck multi-drone routing and scheduling problem with en route operations for last-mile parcel delivery. A mixed integer linear programming (MILP) model is developed to minimize the delivery completion time. In addition, a variant is introduced to minimize the total delivery cost. Since the problem under consideration is NP-hard, a relax-and-fix with re-couple-refine-and-optimize (RF-RRO) heuristic approach is proposed, where the associated decisions (truck routing and drone scheduling) are decomposed into two stages and solved sequentially. Besides, a deep learning-based clustering procedure is developed to establish an initial solution and accelerate the convergence speed of the RF-RRO heuristic. Notably, the proposed approach is extended to solve a multi-truck multi-drone variant using a deep learning-based cluster-first route-second heuristic. Our numerical results show that the proposed MILP model is able to solve problem instances with up to 20 customers optimally in a reasonable time. The proposed RF-RRO heuristic can achieve optimal (or near-optimal) solutions for small instances and is computationally effi-

✉ Sharan Srinivas
SrinivasSh@missouri.edu

Teena Thomas
annateenathomas@gmail.com

Chandrasekharan Rajendran
craj@iitm.ac.in

1 Department of Management Studies, Indian Institute of Technology Madras, Chennai 600036, India

2 Department of Industrial and Systems Engineering, University of Missouri, Columbia, MO 65211, USA

3 Department of Marketing, University of Missouri, Columbia, MO 65211, USA

4 Department of Civil Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal 576104, Karnataka, India

cient for large cases. Extensive experimental analysis shows 30% average savings in delivery completion time, and an average drone utilization of 62% if en route drone operations are considered. In addition, numerical results provide insights on the impact of heterogeneous drone fleet and customer density.

Keywords Drone delivery · Last-mile logistics · Decomposition · Heuristics · En route operations · Deep learning

1 Introduction

The last-mile delivery is the most expensive segment in the distribution process and accounts for 53% of the total shipping cost (Dolan, 2022). Given the rising traffic congestion and fuel costs, the conventional method of using truck alone may not be ideal for improving delivery efficiency and satisfying the growing customer expectations (i.e., fast and cheap shipping options) (Bosona, 2020; Shukla et al., 2010). The use of unmanned aerial vehicles (UAVs) or drones in last-mile delivery has the potential to bring revolutionary improvements in customer accessibility, urban greenhouse gas emissions, operating cost, and delivery speed (Goodchild & Toy, 2018; Perera et al., 2020). As a result, several companies have been piloting drone-based deliveries. For instance, Alphabet's autonomous drone unit, Wing, the first drone company to obtain FAA approval, is about to achieve the 100,000 delivery milestone (Melendez, 2021). Likewise, Amazon's Prime Air delivery program aims to complete customer orders within 30 min (Intelligence, 2022). Drone-based deliveries also help in serving customers in remote and inaccessible regions, thereby addressing accessibility issues in rural areas (Kim et al., 2017). About 44% of people live in rural areas, and only one-third of this population is located at least two kilometers from all-season roads (Kennedy, 2020). Delivering medicines and other necessary amenities to these stranded communities can be arduous for a ground-based transportation network. In such challenging situations, airborne logistics technologies can provide fast and cost-effective solutions. For example, UPS Flight Forward, the first drone company to receive full Part 135 certification from FAA, is delivering vaccines using drones (Wolf, 2021). Likewise, Zipline achieves the world's first long-range delivery of the COVID-19 vaccine in Ghana by collaborating with Pfizer and BioNTech (Zipline, 2021).

Although drones can travel faster than trucks and avoid ground traffic, they are restricted by their payload capacity and short range. Nevertheless, the typical parcel weight of e-commerce orders is within the drone's load-carrying capacity. For example, it is estimated that 86% Amazon's packages are below five pounds and can be easily transported by drones (Pierce, 2013). To circumvent the short flying range, the UAVs can be integrated with the traditional truck-only delivery system, where the truck serves as a moving depot and docking/battery-swapping station for the drones. Such a van-drone combination has been initiated by companies, such as Mercedes Benz, Matternet, and Siroop, to deliver products on-demand for the online marketplace (Rosenthal, 2017). Electric vehicle manufacturer, Workhorse, is planning to enhance their last-mile deliveries using their Horsefly drones mounted on trucks and anticipates 80% cutback in the total delivery costs (Adler, 2020). Amazon's new patent describes a delivery truck that can release drones for completing last-mile deliveries (Irving, 2021).

In addition to gaining significant attention from businesses, the integration of drones into the last-mile delivery system has also been widely researched by the academic community, especially the coordinated routing of the truck-drone tandem. Several recent works have demonstrated the potential to achieve substantial delivery time savings for a truck-drone

system as opposed to the conventional truck-only system (Agatz et al., 2018; Chung, 2018; Murray & Raj, 2020). Based on these existing works, the subproblems associated with truck multi-drone tandems can be broadly categorized as assignment, routing and scheduling. The *assignment problem* deals with the determination of the vehicle (truck or drone) that must serve a customer location. The *truck routing problem* pertains to determining the best order of visiting customers assigned to the truck, while *drone routing problem* deals with identifying the best sorties (i.e., launch location, customer to be visited, recovery location) to serve customers assigned to it. Finally, the *scheduling problem* involves solving for the best launch and retrieval time of each drone sorties between any two nodes in the network. Most existing research has predominantly focused on the assignment and routing problems (Es Yurek & Ozmutlu, 2018; Moshref-Javadi et al., 2020a; Poikonen & Golden, 2020; Salama & Srinivas, 2020), whereas a few notable works have considered all three subproblems together (Murray & Raj, 2020; Salama & Srinivas, 2022). Besides, the drone launch and receipt are predominantly restricted to customer nodes in a majority of the previous work (Ha et al., 2018; Murray & Raj, 2020; Wang & Sheu, 2019). Nevertheless, few research articles have incorporated en route drone operations but did not consider the associated drone scheduling problem (Li et al., 2022; Masone et al., 2022; Marinelli et al., 2018; Schermer et al., 2019).

In this research, we consider the assignment, routing and scheduling decisions associated with a delivery system involving a truck and heterogeneous drone fleet, while allowing en route drone launch/recovery operations. To illustrate the impact of the proposed research, we consider an example with 10 customers and compare the optimal delivery plan for two cases (a) without and (b) with en route drone operations. Figures 1a and 1b show the optimal plan without and with en route drone operations, respectively. The number of truck stops is established as four in both cases, but drone operations take place only with respect to two truck stops (nodes 7 and 9) if en route operations are prohibited, whereas it occurs with respect to all four stops when such a restriction is relaxed. In this paper, a *focal node* refers to a truck stop to which a set of customers served by drones are assigned. In other words, a truck stop is a focal node if a (i) drone launch and/or recovery operation takes place at this stop or (ii) drone launch takes place when the truck is en route to this stop and drone recovery happens once the truck has departed this stop (i.e., while en route to the next stop). However, if the truck stops at a customer location only for parcel delivery and does not involve any of the above-mentioned drone operations, then this truck-stop location is not considered to be a focal node. In this example, in Fig. 1a, the focal nodes are nodes 7 and 9, and in Fig. 1b, all four truck stops (i.e., 9, 1, 8, and 7) are focal nodes. The Gantt chart shown in Fig. 1 illustrates the scheduling of the truck and drones in the collaborative truck-drone system without and with en route drone operations. The parcel delivery takes place at a customer location during the residence time of the vehicle (truck or drone) at the customer location. The optimal delivery completion time with en route operations is 111 min, which is 29% faster than restricting drones to launch and recover at customer nodes. Besides, the drone utilization is observed to be 46% higher in the case of allowing en route operations. Given the potential impact, this research seeks to develop novel approaches to enable last-mile delivery planning for a hybrid truck-drone fleet by considering en route operations. Specifically, this paper contributes to the literature on truck-drone routing in the following ways.

- We introduce a new variant of the truck multi-drone delivery system called the Collaborative Truck-multi-Drone Routing and Scheduling Problem with en route Launch and Recovery operations (CTDRSP-ELR). In addition, variants of the CTDRSP-ELR model based on different drone launch/recovery patterns are introduced.

- To the best of our knowledge, this research is the first to integrate the three logistical decisions (assignment, routing and scheduling) associated with truck-drone delivery with en route operations. Moreover, we consider a heterogeneous drone fleet, where each drone is characterized by its payload-carrying capacity and endurance. We define drone endurance that is dependent on the path and weight of the package to be delivered.
- A new mixed integer linear programming (MILP) model is developed with the objective of minimizing the last-mile delivery completion time to solve the CTDRSP-ELR by jointly optimizing the three logistical decisions. The MILP model is formulated in a modular fashion to provide flexibility in adding or removing certain problem characteristics and allows all possible drone launch and recovery combinations (e.g., launching and recovering en route, launching from a customer node, and recovering en route). In addition, a new valid inequality is developed to accelerate the model convergence.
- Given the NP-hard nature of the CTDRSP-ELR, the proposed MILP model becomes computationally intractable for large problem instances. We propose a heuristic that decomposes the problem into two phases and uses a relax-and-fix with re-couple-refine-and-optimize approach to efficiently solve large instances. Moreover, a valid lower bound for the CTDRSP-ELR is established based on the first phase of the heuristic.
- A deep learning-based approach for obtaining an initial feasible solution to the CTDRSP-ELR is proposed, where an effective representation of the customer dataset is extracted using deep autoencoders and is given as input to k-means clustering technique. The proposed initialization strategy is used to warm start the two-phase decomposition heuristic, and is generalizable to establish an initial solution to other variants of truck multi-drone routing problem in the literature (Gomez-Lagos et al., 2021; Luo et al., 2021; Moshref-Javadi et al., 2020b; Vu et al., 2022).
- We introduce a new variant of collaborative truck-drone systems minimizing total delivery cost. We propose Min-Cost CTDRSP-ELR adapted from CTDRSP-ELR to reduce the operating and idle time costs of the truck and drone fleet enabled with en route launch and recovery operations (LRO). To the best of our knowledge, this is the first time a cost minimization approach is proposed for a single truck-heterogeneous set of drones where en route LRO is permitted.
- A variant to the CTDRSP-ELR is proposed by considering a multi-truck multi-drone delivery setting. We extend our solution approach to solve the Collaborative multi-Truck-multi-Drone Routing and Scheduling Problem with en route Launch and Recovery Operations (mCTDRSP-ELR) using a deep-learning enabled cluster-first route-second heuristic.
- Several new problem instances for CTDRSP-ELR are introduced and extensive numerical analysis is conducted. Besides, the performance of the proposed model is benchmarked against a truck-drone delivery model without en route drone operations, also referred to as Collaborative Truck-multi-Drone Routing and Scheduling Problem with Node Launch and Recovery operations (CTDRSP-NLR).

The remainder of the paper is structured as follows. Section 2 reviews related literature on truck-drone parcel delivery systems. The proposed MILP model for CTDRSP-ELR is presented in Sect. 3, and the two-phase decomposition heuristic with deep learning based initialization procedure is described in Sect. 4. The experimental setup and results of the numerical analysis are provided in Sect. 5. Finally, the concluding remarks and directions for future research are given in Sect. 6.

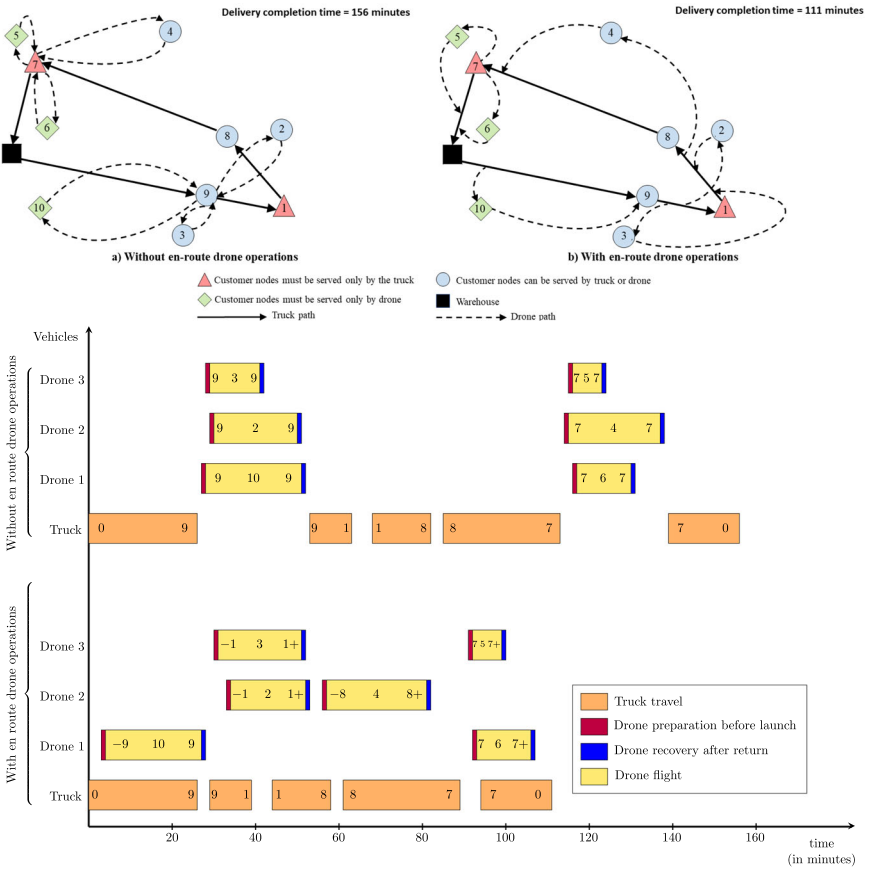

Fig. 1 Illustration of truck-drone delivery system for 10 customers without and with en route drone operations. The numbers inscribed in the Gantt chart indicate customer locations in truck/drone route. The minus sign indicates en route drone launch and the plus sign indicates en route drone recovery

2 Literature review

The literature on truck-drone tandem for parcel delivery has grown considerably in recent years. A comprehensive survey of research works on truck-drone routing is provided in Otto et al. (2018), Chung et al. (2020), Moshref-Javadi and Winkenbach (2021) and Benarbia and Kyamakya (2021). In this section, we focus our review on prior works that employ a single truck and one or more drones for package delivery since they closely align with the proposed research.

One of the earliest works on truck-drone tandem was proposed by Murray and Chu (2015), where a single-truck single-drone system for package delivery was considered. Specifically, they introduced the flying sidekick traveling salesman problem (FSTSP), where the drone launched from a truck stop to serve a customer is recovered at a different truck stop. They also proposed the parallel drone scheduling traveling salesman problem (PDSTSP), where the drone and truck operate independently (no synchronization) to serve customers in a delivery region. The authors developed MILP models and heuristic approaches to minimize the

delivery completion time for both variants, and demonstrated their superiority over the traditional traveling salesman problem. This seminal work led to several variants of collaborative truck-drone delivery systems (Carlsson & Song, 2017; Dell'Amico et al., 2022; Ferrandez et al., 2016; Ha et al., 2018; Ponza, 2016). Ponza (2016) proposed a strengthened formulation for the FSTSP and developed a simulated annealing algorithm to solve relatively large instances. Ham (2018) extended Parallel Drone Scheduling TSP considering a fleet of trucks and a set of drones. However, the study does not consider the truck and drone to operate in a collaborative manner and restricts the LRO of drones to the depot. Agatz et al. (2018) introduced the traveling salesman problem with drone (TSP-D) in which the drone was allowed to return to the same stop from where it was launched. The authors developed an integer programming model and several route-first, cluster-second heuristics to minimize the tour length. This work was subsequently extended by Bouman et al. (2018), where a dynamic programming-based approach was used to obtain the optimal solutions for relatively larger problem instances.

Several algorithmic approaches and variants were later proposed to solve the single-truck single-drone delivery system (de Freitas & Penna, 2020; Es Yurek & Ozmutlu, 2018; Marinelli et al., 2018; Ponza, 2016). Es Yurek and Ozmutlu (2018) proposed a decomposition-based iterative approach to solve the TSP-D problem and solved instances with up to 15 nodes in an average time of 15 min. de Freitas and Penna (2020) proposed a hybrid heuristic, where the initial solution was taken from the optimal TSP solution, and a general variable neighborhood search was employed to obtain the truck and drone routing. This was effective in solving larger problem instances. While the aforementioned works focused on minimizing the delivery completion time (or makespan), Ha et al. (2018) proposed heuristic methods to minimize the total operational costs for the TSP-D. They developed a Greedy Randomised Adaptive Search Procedure for solving problem instances with up to 100 customers, and demonstrated its capability to outperform existing heuristics.

Another stream of research considered a single truck multi-drone setting and found it to be more efficient than TSP-D (Chang & Lee, 2018; Ferrandez et al., 2016; Jeong et al., 2019; Leon-Blanco et al., 2022; Moshref-Javadi et al., 2020b, a; Murray & Raj, 2020; Poikonen & Golden, 2020; Salama & Srinivas, 2020). Ferrandez et al. (2016) adapted the FSTSP to include multiple drones. They employed k-means algorithm to determine drone launch locations and genetic algorithm (GA) to fix the truck route. They found that multiple drones per truck are preferable, saving both energy and time. Chang and Lee (2018) also leveraged the k-means algorithm to identify customer clusters and shifted the cluster centroids towards the depot using a non-linear optimization model to shorten the truck route and widen the drone delivery area. Unlike prior works that adopt a sequential approach, Salama and Srinivas (2020) proposed a model to jointly optimize the clustering and routing decisions while considering two different objectives, namely delivery completion time and total operating cost. They found their integrated approach can yield substantial savings over the sequential approach. Similarly, Karak and Abdelghany (2019) implemented the hybrid Clarke and Wright heuristic to accelerate the optimal solution of the truck-drone collaborated delivery problem. While the aforementioned works involved multiple drones, they do not consider scheduling activities at each truck stop. A few studies have considered truck-drone synchronization constraints in their model but the scheduling of truck and drone operations referring to the launch and receipt time of drones in conjunction with truck operations is excluded (Boccia et al., 2021; Gonzalez-R et al., 2020). This was first addressed by Murray and Raj (2020), where the authors proposed a three-phase sequential heuristic to solve problem instances with up to 100 customers. Recently, Leon-Blanco et al. (2022) extended the work of Murray and Raj (2020) by allowing drones to visit multiple customers per launch. Wu et al. (2022) explored multiple

truck multiple drone problems and discussed partial scheduling of trucks and drones. The launch/receipt times are approximated to the truck arrival and departure times. However, the sequencing of drones with respect to a given node is not considered and hence the timetabling of drones is not determined.

Generally, in most research, drone operations occur when the truck is stationary at a stop. Although most of the research assumed the truck stops to be at a customer location, few notable works have considered them to be at non-customer locations (Carlsson & Song, 2017; Chang & Lee, 2018; Ferrandez et al., 2016; Salama & Srinivas, 2020). In contrast, few works have considered the drone operations to be along the truck path (referred to as en route operations) (Li et al., 2022; Marinelli et al., 2018; Masone et al., 2022; Schermer et al., 2019; Vásquez et al., 2021). Marinelli et al. (2018) extended the TSP-D to allow en route operations by adapting the GRASP procedure suggested by Ha et al. (2018). They conducted an extensive analysis and observed substantial savings in the makespan. Likewise, Vásquez et al. (2021) also considered en route drone operations and were the first to develop a Bender-type decomposition algorithm to solve the truck-drone delivery problem. Recently, Masone et al. (2022) considered single truck single UAV combination, while allowing en route drone operations and multiple customer visits per drone sortie. Li et al. (2022) proposed truck and drone routing problem with synchronization on arcs (TDRP-SA) that consider drone launch and recovery from moving vehicles. TDRP-SA is solved using a mixed integer non-linear programming model and an ALNS-based heuristic. They developed boundary analysis method to identify moving launch and recovery locations (LRL). Nevertheless, research that considers en route operations for a multi-drone setting is sparse, with Schermer et al. (2019) being a notable exception, where the authors consider up to two drones per truck. Besides, most studies that allow en route operations restrict them to be at predetermined points along the truck path. Finally, none of the prior works that consider en route operations have considered the scheduling activities involved during launch/recovery operations of multiple heterogeneous drones. Also, non interference of drone fleet at flexible launch/receipt locations along the truck route is not incorporated in existing literature works. In this research, we seek to tackle these aforementioned gaps and contribute to the literature on drone en route operations. Table 1 summarizes recent works on drone-assisted truck delivery systems which are most relevant to the problem discussed in the paper. The works are summarized based on the decisions considered (namely, assignment routing and scheduling), scalability to multi-vehicle problem, allowable locations for drone LRO, whether a lower bound is proposed, drone fleet employed, and objective functions considered.

3 Problem description and model formulation

The CTDRSP-ELR problem involves package delivery to a set of customer locations $i \in \mathcal{N}$ using a truck and heterogeneous fleet of $k \in \mathcal{K}$ UAVs. To account for operational restrictions and customer preferences, the locations are partitioned into three subsets: (i) customers who must be served only by the truck ($\mathcal{S}^T$), (ii) the set of customers who must be served only by drones ($\mathcal{S}^K$), and (iii) the set of locations that can be visited either by truck or drone ($\mathcal{S}^{TK}$). Likewise, the drone fleet is partitioned into three sets depending on the drone's payload carrying capacity (i.e., $\mathcal{K} = \{\mathcal{K}_{low}, \mathcal{K}_{medium}, \mathcal{K}_{high}\}$). It is assumed that customers belonging to S^{TK} have relatively heavier packages than customers belonging to S^K, and therefore low-payload drones, $k \in \mathcal{K}_{low}$ are considered incapable of serving these customers who can be

Table 1 Summary of relevant work on collaborative truck-drone system

References	Decisions[1]			Scalability	LRO locations[2]					LB	Drone settings[3]		Objective	
	A	R	S		C	N	P	L	U		M	H	Min-Time	Min-Cost
Murray and Chu (2015)	✓	✓	N/A	–	✓	–	–	–	–	–	–	N/A	✓	–
Ham (2018)	✓	✓	–	✓	✓	–	–	–	–	–	✓	–	✓	–
Es Yurek and Ozmutlu (2018)	✓	–	N/A	–	✓	–	–	–	–	–	–	N/A	✓	–
Marinelli et al. (2018)	✓	✓	N/A	–	–	–	–	✓	–	–	–	N/A	–	✓
Chang and Lee (2018)	–	✓	–	–		–	✓	–	–	–	✓	–	✓	–
Ha et al. (2018)	✓	✓	N/A	–	✓	–	–	–	–	–	–	N/A	–	✓
Bouman et al. (2018)	✓	✓	N/A	–	✓	–	–	–	–	–	–	N/A	–	✓
Agatz et al. (2018)	✓	✓	N/A	–	✓	–	–	–	–	✓	–	N/A	✓	–
Wang and Sheu (2019)	✓	✓	–	✓	–	–	✓	–	–	–	✓	–	–	✓
Schermer et al. (2019)	✓	✓	–	✓	✓	–	–	✓	–	–	✓	–	✓	
Salama and Srinivas (2020)	✓	✓	–	–	✓	✓	–	–	–	–	✓	✓	✓	✓
Gonzalez-R et al. (2020)	✓	✓	N/A	–	✓	–	–	–	–	–	–	N/A	✓	–
Murray and Raj (2020)	✓	✓	✓	–	✓	–	-	–	–	–	✓	✓	✓	–

Table 1 continued

References	Decisions[1]			Scalability	LRO locations[2]					LB	Drone settings[3]		Objective	
	A	R	S		C	N	P	L	U		M	H	Min-Time	Min-Cost
Poikonen and Golden (2020)	–	✓	–	–	–	–	✓	–	-	–	✓	–	✓	–
Boccia et al. (2021)	✓	✓	N/A	–	✓	–	–	–	–	–	–	N/A	✓	–
Luo et al. (2021)	✓	✓	✓	–	✓	–	–	–	–	–	✓	–	✓	–
Gomez-Lagos et al. (2021)	–	✓	–	–	–	–	✓	–	–	–	✓	–	✓	–
Wu et al. (2022)	✓	✓	✓	✓	✓	–	–	–	–	–	✓	–	✓	–
Salama and Srinivas (2022)	✓	✓	✓	–	✓		✓	–	–	–	✓	✓	✓	–
Masone et al. (2022)	✓	✓	N/A	–	–	–	✓	✓	–	✓	–	N/A	✓	–
This paper	✓	✓	✓	✓	✓	–	–	–	✓	✓	✓	✓	✓	✓

[1] *A* assignment, *R* routing, *S* scheduling

[2] *C* customer location, *N* undefined non-customer location, *P* predefined non-customer location, *L* restricted predefined points on truck route, *U* unrestricted points on truck route

[3] *M* multiple, *H* heterogeneous

served by either truck or drones. The truck capacity is assumed to be sufficient to carry all packages.

Each drone requires γ^L and γ^R time units for launch and recovery preparation, respectively. To allow safe separation between successive vehicle operations, a lag time of β is considered. The onward and return travel time of drone k with respect to node i when launched from and received at node j is given by τ^O_{ijk} and τ^R_{ijk}, respectively. Note that the travel times are determined based on the average drone speed. Prior works show that drone endurance is linearly related to the weight of the payload carried by the drone (Chowdhury et al., 2017; San et al., 2016). To account for this characteristic of drones, we consider onward and return drone endurance independently. We define $\hat{\tau}^O_{ijk}$ for drone k as the onward flight endurance to deliver the package to customer i while the truck is en route to node j. Likewise, the return drone endurance, $\hat{\tau}^R_{ijk}$ is considered for UAV k on its return after completing the delivery at i, while the truck is traveling from j to the next truck stop. It is to be noted that as the UAV returns to the truck, the weight of the drone reduces as the package is already delivered, and hence the return drone endurance is more than the onward endurance (Hong et al., 2018). Similar to prior works (Ham, 2018), we assume the drones to be fully charged before each sortie as battery swapping is feasible. On the other hand, the truck travel time between any two nodes j and j' is $\tau^T_{jj'}$. The drone is expected to take ρ^K_i time units to drop off the package at customer i, while the truck requires ρ^T_i time units.

The truck, along with the $|\mathcal{K}|$ drones on its roof, must start at the depot (denoted as starting node N+1), deliver all the packages, and return to the depot (ending node N+2). Each customer location must be visited exactly once, either by truck or UAV. The UAVs can be launched/recovered at truck stops or along truck routes. In this paper, we propose the following four drone launch/recovery cases: (i) dispatched and collected at a truck stop, (ii) en route launch but received at a truck stop, (iii) dispatched from a truck stop and recovered en route, (iv) en route launch as well as recovery. Figure 2 illustrates the four launch/recovery policies for a small instance. Customer nodes 1, 2, and 3 are visited by the truck, while the remaining customers are served by the drone fleet. The drone allocated to node 4 is launched and received while the truck is at node 2 (case i). In the case of customer node 6, the drone is launched en route and is retrieved when the truck is at node 2 (case ii). On the other hand, the drone flying to node 5 is launched from the truck stop at node 2 and is recovered while the truck is en route from node 2 to node 3 (case iii). Finally, the drone is launched to node 7 while the truck is en route to node 2 and is recovered when the truck is en route from node 2 to node 3 (case iv). In any case, multiple UAVs cannot be launched and/or recovered simultaneously, thereby requiring scheduling of these activities. The problem is to determine three logistical decisions, namely, assignment, routing (both truck route and drone sorties), and scheduling, such that the delivery completion time is minimized. Note that the delivery completion time is the truck return time at the warehouse with all drones recovered and having completed all deliveries. Moreover, the absolute precedence of vehicular operations is maintained throughout this model and guarantees non-interference of operations among delivery vehicles.

We use the following notations to formulate the objective and constraints pertaining to the CTDRSP-ELR.

Indices and Sets

$i, i', j, j' \in \mathcal{N}$	Set of customer locations (nodes) in the delivery network, where $\mathcal{N} = 1, 2, \ldots, N$

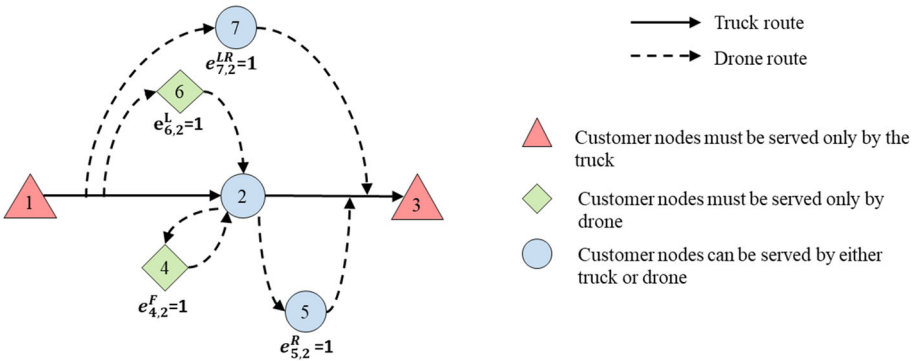

Fig. 2 Illustration of proposed launch and recovery operations (LRO)

$i, j, j' \in \mathcal{S}^T$	Set of customers that can be delivered only by the truck
$i, i' \in \mathcal{S}^K$	Set of customers that can be delivered only by a drone
$i, i', j, j' \in \mathcal{S}^{TK}$	Set of customers that can be delivered either by the truck or a drone
$k \in \mathcal{K}$	Heterogeneous fleet of low, medium and high payload drones, $\mathcal{K} = \{\mathcal{K}_{low}, \mathcal{K}_{medium}, \mathcal{K}_{high}\}$

Parameters

K^{MAX}	Maximum number of drones that can be loaded on the truck roof
C	Maximum number of truck stops, where $C \geq \frac{N}{K^{MAX}}$
τ^O_{ijk}	Onward travel time of drone $k \in \mathcal{K}$ to reach customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ from truck-stop $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$, when the truck is at customer location j
$\hat{\tau}^O_{ijk}$	Endurance time of drone $k \in \mathcal{K}$ to deliver package to customer location $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ when the truck is en route to truck-stop $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$
τ^R_{ijk}	Return travel time of drone $k \in \mathcal{K}$ from customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ to truck-stop $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$, when the truck is at customer location j
$\hat{\tau}^R_{ijk}$	Endurance time of drone $k \in \mathcal{K}$ to return from customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ after delivering its parcel, when the truck is en route from $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ to next truck stop
β	Lag time between two successive activities requiring truck operator assistance
γ^L	Launch preparation time of drone
γ^R	Time for receiving a drone and docking it on the truck
ρ^K_i	Drone service time at customer node $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ for unloading the payload
ρ^T_j	Truck unload time at customer node $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$
$\tau^T_{jj'}$	Truck travel time from customer $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ to customer $j' \in \mathcal{S}^T \cup \mathcal{S}^{TK}$
$\tau^T_{N+1,j}$	Truck travel time from the warehouse to customer $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$
$\tau^T_{j,N+2}$	Truck travel time from customer $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ to the warehouse
M	A large positive number

Decision variables

$x_{jj'}$ 1 if customer $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ and $j' \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ are truck stops, and when the truck proceeds to customer j' after visiting customer j, 0 otherwise

Δ_{ijk} 1 if drone $k \in \mathcal{K}$ serves customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ assigned to focal node $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$; 0 otherwise

δ_{ij} 1 if customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ is assigned to focal node $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$, 0 otherwise

δ_{jj} 1 when node $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ is visited by the truck, 0 otherwise

e^F_{ij} 1 if drone allocated to customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ is launched and received at truck-stop (focal node) $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ to which node i is assigned to; 0 otherwise

e^L_{ij} 1 if drone is launched to customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ when the truck is en route to truck-stop $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ and received back in the truck after truck arrival at node j but before its departure, 0 otherwise

e^R_{ij} 1 if drone is launched to customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ from truck-stop $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ and received while the truck is en route to next truck stop, 0 otherwise

e^{LR}_{ij} 1 if drone is launched to node $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ while the truck is en route to truck-stop $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ and received after the truck has departed from node j while en route to the next truck stop, 0 otherwise

$\Delta^O_{ii'}$ 1 when a drone is launched to node $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ before another drone is launched to node $i' \in \mathcal{S}^K \cup \mathcal{S}^{TK}$, 0 otherwise

$\Delta^R_{ii'}$ 1 when a drone after serving customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ returns to the truck before another drone from node $i' \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ returns to the truck, 0 otherwise

T^L_i Launch time of the drone to customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$

$\hat{T}^L_j$ Launch time of the last drone operated from focal node $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$

T^R_i Return time of the drone back to the truck after serving customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$

T^A_j Arrival time of the truck at customer location $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$

T^D_j Truck departure time from customer location $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$

t^O_i Onward drone travel time to customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$

t^R_i Return drone travel time from customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$

The objective of the MILP model is to minimize the delivery completion time, equivalent to the truck return time at the warehouse T^A_{N+2} as presented in the objective function (Eq. 1).

$$Minimize \quad Z = T^A_{N+2} \tag{1}$$

The model is presented in a modular approach to emphasize the following critical decisions involved: (i) assignment of customer locations to be served by the truck (i.e., determining truck stops), (ii) assignment of customers to be served by drones, (iii) routing and scheduling of truck through the selected stops, (iv) drone's onward and return travel times, and (v) scheduling drone activities considering en route operations, (vi) coordinating truck-drone operations.

3.1 Truck assignment constraints

$$\delta_{ij} \leq \delta_{jj} \quad \forall i \in \mathcal{S}^T \cup \mathcal{S}^K \cup \mathcal{S}^{TK},\ j \in \mathcal{S}^T \cup \mathcal{S}^{TK},\ i \neq j \tag{2}$$

$$\sum_{\substack{i \in \mathcal{S}^K \cup \mathcal{S}^{TK} \\ i \neq j}} \delta_{ij} \leq K^{MAX} \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{3}$$

$$\sum_{j \in \mathcal{S}^T \cup \mathcal{S}^{TK}} \delta_{jj} = C \tag{4}$$

$$\sum_{j \in \mathcal{S}^T \cup \mathcal{S}^{TK}} \delta_{ij} = 1 \quad \forall i \in \mathcal{S}^T \cup \mathcal{S}^K \cup \mathcal{S}^{TK} \tag{5}$$

$$\delta_{jj} = 1 \quad \forall j \in \mathcal{S}^T \tag{6}$$

$$\delta_{jj} = 0 \quad \forall j \in \mathcal{S}^K \tag{7}$$

Constraint (2) permits customer i to be allotted to customer j only if the latter is a truck stop. The number of customers assigned to a truck stop cannot exceed the truck's drone carrying capacity, K^{MAX} and is ensured by Constraint (3). The total number of truck stops is governed by Constraint (4). Each customer must be assigned to exactly one truck stop as regulated by Constraint (5). Constraint (6) mandates all customer locations in set $\mathcal{S}^T$ to be truck stops and Constraint (7) prevents customers in set $\mathcal{S}^K$ from being visited by the truck.

3.2 Drone assignment constraints

$$\sum_{k \in \mathcal{K}} \Delta_{ijk} = \delta_{ij} \quad \forall i \in \mathcal{S}^K, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{8}$$

$$\sum_{\substack{i \in \mathcal{S}^K \cup \mathcal{S}^{TK} \\ i \neq j}} \Delta_{ijk} \leq \delta_{jj} \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, k \in \mathcal{K} \tag{9}$$

$$\sum_{k \in \mathcal{K}} \Delta_{ijk} \leq \delta_{ij} \quad \forall i \in \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{10}$$

$$\sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} \sum_{k \in \mathcal{K}} \Delta_{ijk} + \delta_{ii} = 1 \quad \forall i \in \mathcal{S}^{TK} \tag{11}$$

$$\Delta_{ijk} = 0 \quad \forall i \in \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j, k \in \mathcal{K}_{low} \tag{12}$$

$$\sum_{\substack{i \in \mathcal{S}^K \cup \mathcal{S}^{TK} \\ i \neq j}} \sum_{k \in \mathcal{K}} \Delta_{ijk} \leq K^{MAX} \times \delta_{jj} \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{13}$$

A customer $i \in \mathcal{S}^K$ assigned to focal node $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ must be assigned exactly one drone as stipulated in Constraint (8). Drone k is assigned to customer i only if node j to which i is assigned to is a truck-stop as restricted in Constraint (9). Also, Constraint (10) allows a drone to serve customer $i \in \mathcal{S}^{TK}$ from truck-stop $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$, only if j is the focal node of customer i. Furthermore, Constraint (11) guarantees that customer $i \in \mathcal{S}^{TK}$ is served either by a drone or the truck. Constraint (12) restraints low-payload drones from serving the set of customers belonging to $\mathcal{S}^{TK}$. Constraint (13) ensures that the total number of drones dispatched from a truck stop does not exceed the maximum number of drones available.

3.3 Truck routing and scheduling constraints

$$\sum_{j \in \mathcal{S}^T \cup \mathcal{S}^{TK}} x_{N+1,j} = 1 \tag{14}$$

$$\sum_{j \in \mathcal{S}^T \cup \mathcal{S}^{TK}} x_{j,N+2} = 1 \tag{15}$$

$$\delta_{jj} - \delta_{j'j'} \leq 1 - x_{jj'} \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, j' \in \mathcal{S}^T \cup \mathcal{S}^{TK}, j \neq j' \tag{16}$$

$$\sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ j \neq j'}} x_{jj'} + x_{N+1,j'} = \delta_{j'j'} \quad \forall j' \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{17}$$

$$\sum_{\substack{j' \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ j' \neq j}} x_{jj'} + x_{j,N+2} = \delta_{jj} \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{18}$$

$$T_j^D \geq T_j^A + \rho_j^T + \beta - M\left(1 - \delta_{jj}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{19}$$

$$T_j^A \leq M\left(\delta_{jj}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{20}$$

$$T_j^D \leq M\left(\delta_{jj}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{21}$$

$$T_j^A \geq \tau_{N+1,j}^T - M\left(1 - x_{N+1,j}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{22}$$

$$T_{j'}^A \geq T_j^D + \tau_{jj'}^T - M\left(1 - x_{jj'}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, \forall j' \in \mathcal{S}^T \cup \mathcal{S}^{TK}, j \neq j' \tag{23}$$

$$T_{N+2}^A \geq T_j^D + \tau_{j,N+2}^T - M\left(1 - x_{j,N+2}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{24}$$

Constraint (14) ensures that the truck departs exactly once from the warehouse. Similarly, Constraint (15) confirms that the truck returns to the warehouse only once. Constraint (16) is introduced as a novel inequality for solving the truck tour. It guarantees that the truck precedence between two customer locations j and j' will be considered only if both nodes are truck stops. The precedence relationship between two nodes does not hold when atleast one of them is not a truck stop. However, when both nodes correspond to truck stops, $x_{jj'}$ can take values 0 or 1 based on the precedence of nodes, j and j'. Constraints (17) and (18) ascertain the inflow and outflow of the truck with respect to truck stops. Constraints (16), (17) and (18) ensure that when nodes j and j' are not truck stops, $x_{jj'}$ takes value 0. Constraints (16)–(18) in conjunction with Constraints (22)–(24) eliminate all sub tours. Constraints (19)–(21) determine the truck arrival and departure time at a truck stop. Truck arrival and departure times are not defined for those nodes which are not visited by the truck and ensured by Constraints (20) and (21), respectively. The truck departure time from the warehouse is assumed to be 0 and Constraints (22)–(24) determine the timetable of the truck tour.

3.4 Drone travel time constraints

Constraints (25)–(29) correspond to the onward drone travel time and Constraints (30)–(34) address the drone return travel time with respect to the delivery location.

$$t_i^O \geq \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} \sum_{k \in \mathcal{K}} \tau_{ijk}^O \times \Delta_{ijk} \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK} \tag{25}$$

$$t_i^O \geq \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, \\ i \neq j}} \sum_{k \in \mathcal{K}} (\tau_{ijk}^O \times \Delta_{ijk}) - M(1 - e_{ij}^F - e_{ij}^R) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{26}$$

$$t_i^O \leq \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} \sum_{k \in \mathcal{K}} (\tau_{ijk}^O \times \Delta_{ijk}) + M(1 - e_{ij}^F - e_{ij}^R) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{27}$$

$$t_i^O \leq \sum_{k \in \mathcal{K}} \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} (\hat{\tau}_{ijk}^O \times \Delta_{ijk}) + M(1 - e_{ij}^{LR} - e_{ij}^L) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{28}$$

$$t_j^O \leq M\left(1 - \delta_{jj}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{29}$$

$$t_i^R \geq \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} \sum_{k \in \mathcal{K}} (\tau_{ijk}^R \times \Delta_{ijk}) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK} \tag{30}$$

$$t_i^R \geq \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} \sum_{k \in \mathcal{K}} (\tau_{ijk}^R \times \Delta_{ijk}) - M(1 - e_{ij}^F - e_{ij}^L) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{31}$$

$$t_i^R \leq \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} \sum_{k \in \mathcal{K}} (\tau_{ijk}^R \times \Delta_{ijk}) + M(1 - e_{ij}^F - e_{ij}^L) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{32}$$

$$t_i^R \leq \sum_{k \in \mathcal{K}} \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} (\hat{\tau}_{ijk}^R \times \Delta_{ijk}) + M(1 - e_{ij}^R - e_{ij}^{LR}) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{33}$$

$$t_j^R \leq M\left(1 - \delta_{jj}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{34}$$

$$t_i^O + t_i^R \leq \sum_{k \in \mathcal{K}} \sum_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} (\hat{\tau}_{ijk}^O + \hat{\tau}_{ijk}^R) \times \Delta_{ijk} \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{35}$$

Constraint (25) stipulates the onward drone travel time from the focal node, j to the drone customer, i to be greater than or equal to τ_{ijk}^O. Constraints (26) and (27) restrict that for drone launch operations fixed at the focal node (e_{ij}^F and e_{ij}^R), the onward drone travel time is equal to τ_{ijk}^O. The onward drone travel time cannot exceed the drone endurance time between the focal node and the customer to be served by the drone ($\hat{\tau}_{ijk}^O$) as given by Constraint (28) for drone launches taking place while truck is en route to the focal node (e_{ij}^L and e_{ij}^{LR}). The onward drone travel time is not defined for customers visited by the truck and is guaranteed by Constraint (29). The return travel time of drone from customer, i to truck-stop, j must be greater than or equal to τ_{ijk}^R as enforced by Constraint (30). Constraints (31) and (32) impose that for drone recoveries taking place at focal node locations (e_{ij}^F and e_{ij}^L), the return travel time of drone is equal to τ_{ijk}^R. For drone receipts that take place en route (e_{ij}^R and e_{ij}^{LR}), Constraint (33) assures that the return drone travel time does not exceed the drone endurance time between the two locations ($\hat{\tau}_{ijk}^R$). Constraint (34) checks that return drone travel time is not defined for truck-stop locations. Finally, the total drone travel time (onward and return) with respect to customer, i and truck-stop, j must not exceed the drone endurance time (onward and return) between the two nodes, as stipulated by Constraint (35).

3.5 Constraints for scheduling drone launch and recovery

$$e^L_{ij} + e^F_{ij} + e^R_{ij} + e^{LR}_{ij} = \delta_{ij} \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{36}$$

$$T^L_{i'} \geq T^L_i + \gamma^L - M\left(2 - \delta_{ij} - \delta_{i'j}\right) - M\left(1 - \Delta^O_{ii'}\right) \tag{37}$$
$$\forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall i' \in \mathcal{S}^K \cup \mathcal{S}^{TK}, i' > i, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i' \neq j, i \neq j$$

$$T^L_i \geq T^L_{i'} + \gamma^L - M\left(2 - \delta_{ij} - \delta_{i'j}\right) - M \times \Delta^O_{ii'}$$
$$\forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall i' \in \mathcal{S}^K \cup \mathcal{S}^{TK}, i' > i, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j, i' \neq j \tag{38}$$

$$\hat{T}^L_j \geq T^L_i - M\left(1 - \delta_{ij}\right) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \ \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{39}$$

$$T^L_j \leq M\left(1 - \delta_{jj}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{40}$$

$$T^L_{i'} \geq T^R_i + \gamma^R - M(2 - \delta_{ij} - \delta_{i'j'}) - M(1 - x_{jj'})$$
$$\forall i, i' \in \mathcal{S}^K \cup \mathcal{S}^{TK}, i \neq i', \forall j, j' \in \mathcal{S}^T \cup \mathcal{S}^{TK}, j \neq j', i \neq j, i' \neq j' \tag{41}$$

$$T^R_i \geq \hat{T}^L_j - M\left(1 - \delta_{ij}\right) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{42}$$

$$T^R_i \geq T^L_i + t^O_i + t^R_i + \rho^K_i \times \sum\nolimits_{\substack{j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \\ i \neq j}} \delta_{ij} \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK} \tag{43}$$

$$T^R_{i'} \geq T^R_i + \gamma^R - M\left(2 - \delta_{ij} - \delta_{i'j}\right) - M\left(1 - \Delta^R_{ii'}\right)$$
$$\forall i, i' \in \mathcal{S}^K \cup \mathcal{S}^{TK}, i' > i, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i' \neq j, i \neq j \tag{44}$$

$$T^R_i \geq T^R_{i'} + \gamma^R - M\left(2 - \delta_{ij} - \delta_{i'j}\right) - M \times \Delta^R_{ii'}$$
$$\forall i, i' \in \mathcal{S}^K \cup \mathcal{S}^{TK}, i' > i, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i' \neq j, i \neq j \tag{45}$$

$$T^R_j \leq M\left(1 - \delta_{jj}\right) \quad \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK} \tag{46}$$

Constraint (36) manages the drone operation to customer i assigned to focal node j based on the launch/recovery policies. Constraints (37) and (38) ensure non-interference of drones during launch with respect to a focal node. Note that binary variable $\Delta^O_{ii'}$ is relevant and active only when nodes i and i' are assigned to the same focal node. Constraint (39) determines the last drone launch time with respect to a focal node. Constraint (40) guarantees that the drone launch time is not defined for customers visited by the truck. Constraint (41) confirms that drone launches with respect to a focal node commence only once all drone receipts with respect to the previous focal node are completed. Drones are received in the truck only after all drone launches with respect to that focal node are completed, and this is ensured by Constraint (42). Constraint (43) determines the drone receipt time which includes drone travel time and service time at customer location. Constraints (44) and (45) ensure non-interference of drones with respect to a focal node while returning from their respective delivery locations. Note that $\Delta^R_{ii'}$ is relevant and active only when both nodes i and i' are assigned to the same focal node. Finally, Constraint (46) ensures that drone recovery time is not defined for customer locations visited by the truck.

3.6 Truck-drone sequencing and routing constraints

This section consists of all constraints that deal with truck-drone sequencing and routing.

$$T^D_j \geq T^R_i + \gamma^R + \beta - M(1 - e^F_{ij} - e^L_{ij}) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \tag{47}$$

$$T_i^L \geq T_{j'}^D + \beta + \gamma^L - M(1 - e_{ij}^{LR} - e_{ij}^L) - M(1 - x_{j'j})$$
$$\forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j, j' \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j, j \neq j' \quad (48)$$

$$T_i^L \geq T_j^A + \beta + \gamma^L - M(1 - e_{ij}^F - e_{ij}^R) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \quad (49)$$

$$T_i^L + \beta \leq T_j^A + M\left(1 - e_{ij}^{LR} - e_{ij}^L\right) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, \ i \neq j \quad (50)$$

$$T_i^L \geq \beta + \gamma^L - M(1 - e_{ij}^{LR} - e_{ij}^L) - M(1 - x_{N+1,j})$$
$$\forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \quad (51)$$

$$T_j^D \geq T_i^L + \beta - M(1 - e_{ij}^R) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \quad (52)$$

$$T_i^R + \gamma^R + \beta \leq T_j^D + \tau_{jj'}^T + M(1 - e_{ij}^{LR} - e_{ij}^R) + M(1 - x_{jj'})$$
$$\forall j', j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, j \neq j', i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, i \neq j \quad (53)$$

$$T_i^R + \gamma^R + \beta \leq T_{j'}^A + M(1 - e_{ij}^{LR} - e_{ij}^R) + M(1 - x_{jj'})$$
$$\forall j', j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, j \neq j', i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, i \neq j \quad (54)$$

$$T_i^R \geq T_j^A + \beta - M(1 - e_{ij}^L) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \quad (55)$$

$$T_i^R \geq T_j^D + \beta - M(1 - e_{ij}^{LR} - e_{ij}^R) \quad \forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \quad (56)$$

$$T_i^R + \gamma^R + \beta \leq T_j^D + \tau_{j,N+2}^T + M(1 - e_{ij}^{LR} - e_{ij}^R) + M(1 - x_{j,N+2})$$
$$\forall i \in \mathcal{S}^K \cup \mathcal{S}^{TK}, \forall j \in \mathcal{S}^T \cup \mathcal{S}^{TK}, i \neq j \quad (57)$$

Constraint (47) stipulates that the truck departs from a node only once all drone recoveries fixed at the focal node are completed (e_{ij}^F and e_{ij}^L). Constraint (48) ensures that en route drone launches with respect to a focal node can begin only after the truck has departed from the previous truck stop. For drone launches fixed at the focal node (e_{ij}^F and e_{ij}^R), the launch operation begins after the truck has arrived at the focal node as restricted in Constraint (49). Constraint (50) forces completion of en route drone launches before the truck arrives at the focal node. Constraint (51) allows drone launches while the truck is en route to the first truck stop, only after the truck has departed from the warehouse. The drone is ready for launch right after the truck leaves the depot allowing for a very small duration of launch preparation time. Similarly, for en route drone recovery case (e_{ij}^R), their respective drone launch must take place before the truck departs the focal node, as enforced by Constraint (52). Constraints (53) and (54) prevent the truck from entering the next truck stop before completing all en route drone receipts with respect to the current truck stop. For en route drone launches (e_{ij}^L), Constraint (55) restricts their recovery only once the truck has arrived at the focal node. Constraint (56) guarantees that en route drone recoveries are engaged only after the truck has departed from the corresponding focal node. Constraint (57) assures that all en route drone receipts are completed before the truck arrives at the warehouse after completing the tour.

One of the unique characteristics of the proposed optimization model for CTDRSP-ELR is its modularity. The mathematical formulation is decomposed into various constituents that address different subproblems. The simple partition and recombination of these subproblems can be used to solve a variety of tasks. One of the distinct merits of this model is that the CTDRSP-ELR model is robust and self-sufficient to solve multiple variants of this problem which differ in the nature of drone launch and recovery operations (LRO). The same model can be implemented in different scenarios with minor changes, which thus helps in reducing the complexity of subsequent models. For instance, government regulations or business policies

may prohibit a drone carrying a package from being operated from a moving vehicle. In that case, the drone must always be launched from a truck stop, but can be recovered en route. This can be easily handled in the proposed model by setting the binary variables allowing en route launches (e^L_{ij}, and e^{LR}_{ij}) as zero. Likewise, the model can be adapted to handle other scenarios (e.g., en route operation for launch only, where binary variables e^R_{ij} and e^{LR}_{ij} are set to zero).

To evaluate the impact of allowing en route operations, we benchmark the CTDRSP-ELR solution with a variant of the proposed problem that prohibits all en route operations. In other words, the LRO must happen only at truck stops. We refer to this benchmark problem as the Collaborative Truck-multi-Drone Routing and Scheduling Problem with Node Launch and Recovery operations (CTDRSP-NLR) throughout this paper.

4 Proposed heuristic solution approach: relax-and-fix with re-couple-refine-and-optimize (RF-RRO)

The truck-drone routing and scheduling problem is NP-hard and hence the proposed MILP can obtain provable optimal solutions only for small test instances in a reasonable time (Dell'Amico et al., 2020; Murray & Chu, 2015). In particular, the proposed CTDRSP-ELR is a complex combinatorial optimization problem with a large solution space due to the simultaneous consideration of three key decisions (assignment, routing and scheduling) along with en route LRO of drones. Therefore, a decomposition-based heuristic is proposed to efficiently handle large instances of CTDRSP-ELR. Prior research works have proposed various decomposition methods to divide the overall problem into smaller subproblems. For instance, a two-phase decomposition heuristic to solve the TSP with drone resupply was presented by Pina-Pardo et al. (2021), where the truck routing was solved in the first phase, and drone resupply decisions were determined in the second phase. Likewise, a decomposition-based relax-and-fix heuristic was proposed by Brahimi and Aouam (2015) to solve the production routing problem. Dellaert et al. (2021) employs a decomposition method to solve the two-echelon capacitated vehicle routing problem by decoupling and re-coupling its components. Similarly, we introduce a two-phase decomposition heuristic to solve CTDRSP-ELR. The problem under study is computationally intensive due to the presence of scheduling and routing decisions associated with truck-drone operations. Specifically, the presence of scheduling activities coupled with en route LRO leads to increased complexity (e.g., constraints (37), (38), (41), (44), (45)). Therefore, our solution approach seeks to decompose the CTDRSP-ELR into smaller subproblems by relaxing/separating the complicated constraints. Toledo et al. (2015) implemented a relax-and-fix and fix-and-optimize heuristic to solve multi-level lot sizing problems where binary variables undergo the relax-and-fix process. The application of decomposition heuristic and relax-fix and optimize methods have been experimented in other domains with appropriate adaptations and are specific to such problems. In this paper, we propose a new optimization-based two-phase heuristic for collaborative truck-drone delivery problems. The proposed solution approach consists of (i) relax-and-fix phase, where selected complicating constraints are relaxed and binary decision variables are fixed, and (ii) recouple-refine-and-optimize phase, where the first phase solutions are refined by recoupling the relaxed constraints and optimizing the binary variables.

In the first phase, a relax-and-fix (RF) heuristic is developed, where the drone scheduling activities and en route launch or/and recovery operations are relaxed in the optimization model. The first phase of the heuristic determines truck stops from the set of customer

locations and solves the routing and scheduling of the truck. It also ensures feasible solutions to the following drone-related decisions: (i) assignment of customers to be served by drones along with the associated focal node for those customers and (ii) assignment of specific drone $k \in \mathcal{K}$ to these customers. In the second phase, a re-couple-refine-and-optimize (RRO) heuristic is proposed, where the truck-drone routing and scheduling decisions are optimized by re-coupling the relaxed set of first-phase decisions while exploring potential improvements (refinements) to the decisions established in the first phase. To achieve good solutions to NP-hard problems in reasonable time, prior works have used a feasible solution to initialize the heuristic algorithms (Ali et al., 2021; Bai et al., 2018; Hu et al., 2018; Kosanoglu et al., 2022; Pichka et al., 2018; Salama & Srinivas, 2020). Given the complexity of CTDRSP and associated optimization subproblems, this research proposes a deep clustering approach to generate a feasible initial solution for the RF-RRO heuristic. Specifically, the proposed deep clustering procedure uses a combination of autoencoders, k-means clustering and post-processing techniques to establish an initial solution. Figure 4 provides an overview of the proposed solution approach for the CTDRSP-ELR.

4.1 Deep Learning-based clustering algorithm for generating a feasible solution

Existing literature has shown that deep clustering-based solution techniques with autoencoders are effective for extracting information and learning latent representation of high-dimensional data sets (Dairi et al., 2018; Zeng et al., 2019, 2021). Motivated by these findings, we develop a deep learning-based clustering approach to establish a feasible initial solution as shown in Fig. 4. The initialization procedure involves four key steps—data preparation, feature generation using autoencoder network, k-means clustering to establish meaningful grouping of customer locations, post-processing of established clusters to guarantee feasibility.

4.1.1 Data preparation

The input information for deep clustering-based initialization procedure consists of the distance matrix between customer locations and mode of delivery of their parcels. These sets of features are normalized using the z-score normalization method, making the scales of all features within the same interval with mean 0 and standard deviation 1. The normalized data is used to train the autoencoder network for feature learning of the input data and to reduce the dimensional complexity.

4.1.2 Deep autoencoder network

Autoencoders are feed-forward neural networks capable of extracting information from dense input data. Specifically, it is effective in generating a compressed representation of the input data based on unsupervised learning algorithms and is able to reconstruct the input with minimum information loss. A deep autoencoder has an input layer, two or more hidden layers, and an output layer. The number of neurons in the input and the output layers is equal, while hidden layers have a varying number of neurons. An autoencoder has two parts: encoder and decoder. The encoder focuses on the extraction and encoding of input information and dimensionality reduction. It is the network between the input and the encoded layer, also called the bottleneck. The pre-processed data is given as input to a stacked autoencoder. A typical stacked autoencoder network is displayed in Fig. 3.

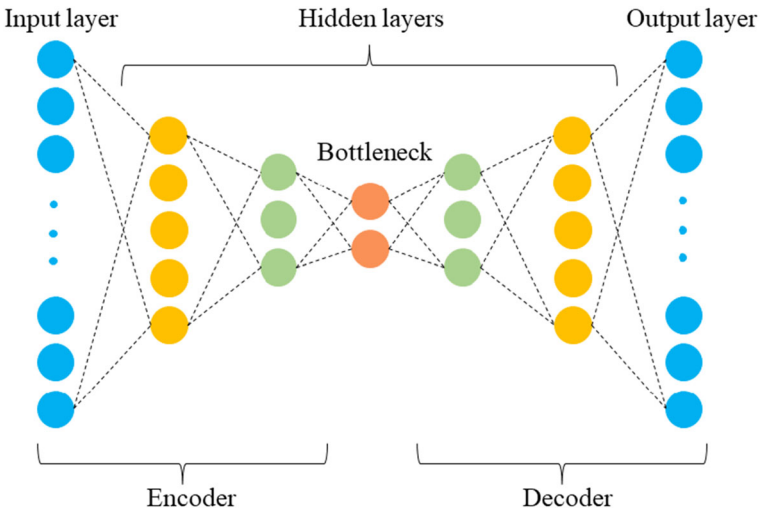

Fig. 3 Stacked autoencoder network structure

Let $X_1, X_2, \ldots, X_N$ be the pre-processed data corresponding to the set of N customers where $X_i \in \mathbb{R}^n$ is fed to the input layer of the autoencoder. The input layer is connected to the first hidden layer with n_1 neurons and activation of its i^{th} neuron is defined by Eq. (58), where h_i^1 is the value of i^{th} unit in the first hidden layer, $\sigma(.)$ is the activation function, $w_i^{input} \in \mathbb{R}^{n \times n_1}$ and $b_i^{input} \in \mathbb{R}^{n_1}$ are the weight matrix and the bias vector respectively.

$$h_i^1 = \sigma(w_i^{input} X_i + b_i^{input}) \tag{58}$$

Stacked autoencoders have multiple hidden layers where the output of a hidden layer is connected to the input of the next hidden layer. The value of the j^{th} neuron in the l^{th} hidden layer, h_j^l is given by Eq. (59), where $\sigma(.)$ is the activation function, $w_j^{(l)}$ is the matrix with weights from layer $l-1$ to neuron j of hidden layer l, where $w_j^{(l)} \in R^{n_{(l-1)} \times n_{(l)}}$. For $(l-1)^{th}$ hidden layer, $h^{(l-1)}$ is the vector containing values of its neuron units. Also, the bias vector of layer l is $b_j^{(l)}$, where $b_j^{(l)} \in R^{n_{(l)}}$ and the number of neurons in $(l-1)^{th}$ and l^{th} layer is $n_{(l-1)}$ and $n_{(l)}$, respectively. The output of layer $l-1$ becomes the input of the next layer, l.

$$h_j^l = \sigma(w_j^{(l)} h^{(l-1)} + b_j^{(l)}) \tag{59}$$

At bottleneck, the encoder output is the compressed representation of the input data with number of neurons, $n_{bottleneck}$, where $n_{bottleneck} < n$. We use the rectified linear unit (ReLU) activation function, a piecewise linear function that overcomes the problem of vanishing gradient in deep neural networks. The number of hidden layers and neurons are fixed using trial and error method which vary with respect to the size of customers considered in the problem instance.

In order to improve the convergence speed, Adam Optimizer, a gradient descent method is used to train the hidden layers. This moment-based optimization algorithm employs an adaptive learning rate after considering the exponential weighted average of the gradients

evaluated on the current batch. If g_t is the gradient at time t, then the first and second moment vectors, m_t and v_t, are given by Eqs. (60) and (61), respectively. Also, ν_1 and ν_2 are hyperparameters representing decay rates of first and second moment vectors, respectively. The bias corrected values of first and second moments, $\hat{m}_t$ and $\hat{v}_t$ are introduced in Eqs. (62) and (63), respectively. Equation (64) defines the weight vector, $w \in \mathbb{R}^n$, where Δw_t represents the weight change and $t \in \mathbb{N}$ is the time stamp for the concerned training step. Further, the change in weight is given by Eq. (65), where η is the learning rate, and ε is a small positive number to avoid division by zero.

$$m_t = \nu_1 m_{t-1} + (1 - \nu_1) g_t \tag{60}$$

$$v_t = \nu_2 v_{t-1} + (1 - \nu_2) g_t^2 \tag{61}$$

$$\hat{m}_t = \frac{m_t}{1 - \nu_1^t} \tag{62}$$

$$\hat{v}_t = \frac{v_t}{1 - \nu_2^t} \tag{63}$$

$$w_t = w_{t-1} + \Delta w_t \tag{64}$$

$$\Delta w_t = -\eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \varepsilon} \tag{65}$$

Decoder is the autoencoder network between the encoder and the output layer. It reconstructs the input information by learning the compressed data in the bottleneck, minimizing the loss function. The mean squared error loss function is defined for this autoencoder architecture and is described in Eq. (66), where N is the input size, $\hat{X}_i$ is the reconstructed output of i^{th} input observation, X_i, where $\hat{X}_i, X_i \in \mathbb{R}^n$. The features extracted from hidden layers are decoded or reconstructed back to the actual input space $\mathbb{R}^n$ by the output layer.

$$L = \frac{1}{N} \sum_{i=1}^{N} (\hat{X}_i - X_i)^2 \tag{66}$$

The reconstruction of the input value X_i is given in Eq. (67), where $f(.)$ is the activation function for decoder, $h_i^{(L)}$ is the value of i^{th} unit in hidden layer, L and $w_i^{(L)}$ and $b_i^{(L)}$ are weight matrix and bias vector of the decoder, $w_i^{(L)} \in \mathbb{R}^{n_L \times n}$ and $b_i^{(L)} \in \mathbb{R}^n$.

$$\hat{X}_i = f(w_i^{(L)} h_i^{(L)} + b_i^{(L)}) \tag{67}$$

Algorithm 1 explains the feature extraction process of the customer data set using the autoencoder model. The features encoded by the stacked autoencoder are given as input to the k-means clustering algorithm.

Algorithm 1 Dimension reduction and feature extraction from data based on autoencoders

Inputs:
  Standardized customer data, $X \in R^n$
  Number of hidden layers, L and number of neurons in each hidden layer $n_1, n_2, ..., n_L$
Autoencoder algorithm:
  Initialize parameters of encoder and decoder: weight matrix, w and bias vector, b
  Define rectified linear unit activation function for the encoder, σ
  Define number of epochs and learning rate
  for each epoch:
    Unsupervised training of hidden layers using Adam optimizer
    Determine encoder parameters $\theta = \{w, b\}$ by minimizing the MSE loss function
    For the optimal parameter setting, θ record the extracted features at bottleneck
  end for
Output:
  Compressed low dimension data at bottleneck, $Y \in R^d$, where $d < n$

4.1.3 *k*-means clustering and post-processing

The encoder network in the autoencoder model extracts the latent features of the original input information. The output of the encoding layer, a low-dimensional representation of input data, becomes the input to the next stage of the initialization procedure, k-means clustering. The number of clusters is predefined and this parameter is set to maximum number of truck stops, C, in the MILP model to ensure feasibility. Algorithm 2 describes the clustering algorithm partitioning customers into suitable groups based on the autoencoder learning. Clusters are formed such that the distance between the customers and the cluster center is minimum.

Algorithm 2 k-means clustering

Inputs:
  Features extracted by encoder, $Y \in R^d$ where $d < n$
  Number of clusters, k
k-means algorithm:
  Choose k customers arbitrarily from Y as cluster centroids
  Repeat:
    Assign each customer to the nearest cluster center
    Recompute cluster centers
  Until:
    Convergence criteria is met
Output:
  k set of clusters

To guarantee feasible grouping of customers, the clusters obtained from the k-means algorithm undergo post-processing to ensure an initial feasible solution for the RF-RRO heuristic. During post-processing, customers are reassigned to other clusters until the following conditions are satisfied:

- cluster size is limited to $K^{MAX}+1$
- no cluster has more than one customer belonging to $\mathcal{S}^T$
- no cluster exists whose members belong to only $\mathcal{S}^K$
- number of customers belonging to $\mathcal{S}^{TK}$ in a set must not exceed the available medium- and high-payload drones, $k \in \mathcal{K}_{medium} \cup \mathcal{K}_{high}$
- drone flight duration to customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ from $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$, j being the nearest customer to the cluster centroid, is within the drone endurance.

After post-processing, for each cluster,

- the closest customer, $j \in \mathcal{S}^T \cup \mathcal{S}^{TK}$ to the cluster centroid is assigned $\delta_{jj}=1$
- all remaining customers, $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$ are assigned $\delta_{ij}=1$, j being the focal node.

4.2 Phase 1: Relax-and-fix heuristic (RF)

In this section, we propose the relax-and-fix (RF) approach to solve the MILP subproblem, which optimizes the truck tour by relaxing the drone scheduling decisions. The set of relaxed decisions are constraints related to drone travel time, scheduling drone launch and recovery, and truck-drone sequencing and routing (Sect. 3.4–3.6). No en route drone operations are considered in the first phase of the heuristic as the RF phase focuses on solving the truck routing and scheduling decisions. The output of the initialization procedure consists of the feasible customer clusters. The cluster solutions obtained after the deep clustering technique are given as an initial solution to accelerate the first phase of the decomposition heuristic. Additionally, the input parameters and sets of customers from Sect. 3 are also given as input as shown in Fig. 4. The problem identifies the truck stops among the set of customers and solves the truck tour among the truck-stop locations. Specifically, this subproblem seeks to fix the binary variables $x_{jj'}$ (truck tour) and δ_{jj} (truck stops) with the goal of minimizing delivery completion time. The set of constraints that govern the first phase are focal node assignment (Sect. 3.1), drone assignment (Sect. 3.2), and routing/scheduling of truck (Sect. 3.3), as illustrated in Fig. 4. Besides, the subproblem considered in Phase 1 (i.e., decomposition of CTDRSP-ELR) ensures that the solutions obtained, especially focal node assignments (δ_{ij}) and drone assignments (Δ_{ijk}), are feasible for the original MILP model, since the constraints related to these decisions are considered in this phase. The truck travel schedule obtained as the output of Phase 1 does not consider the collaborative truck-drone delivery timetable. Although the truck tour is optimized in this phase, the truck timetable and delivery completion time obtained do not account for drone sequencing and timetable.

4.3 Phase 2: Re-couple-refine-and-optimize heuristic (RRO)

In Phase 2, the drone travel time and scheduling (Sect. 3.4–3.5) as well as the truck-drone coordination/scheduling (Sect. 3.6) are addressed and are re-coupled into the heuristic as described in Fig. 4. Although customer assignment to focal nodes (δ_{ij}) and drone allocation (Δ_{ijk}) obtained in Phase 1 are feasible, they may not necessarily yield the best delivery completion time as these decisions are initially established by repressing the drone scheduling constraints, and therefore does not holistically factor the drone operations. In addition to this, Phase 1 does not consider en route drone operations. Superior solutions are guaranteed by finding efficient focal node assignments and drone assignments after integrating the related constraints into the model in Phase 2. Therefore, the truck travel sequence ($x_{jj'}$) and truck stops (δ_{jj}) obtained in Phase 1 are fixed in Phase 2. However, the following decisions

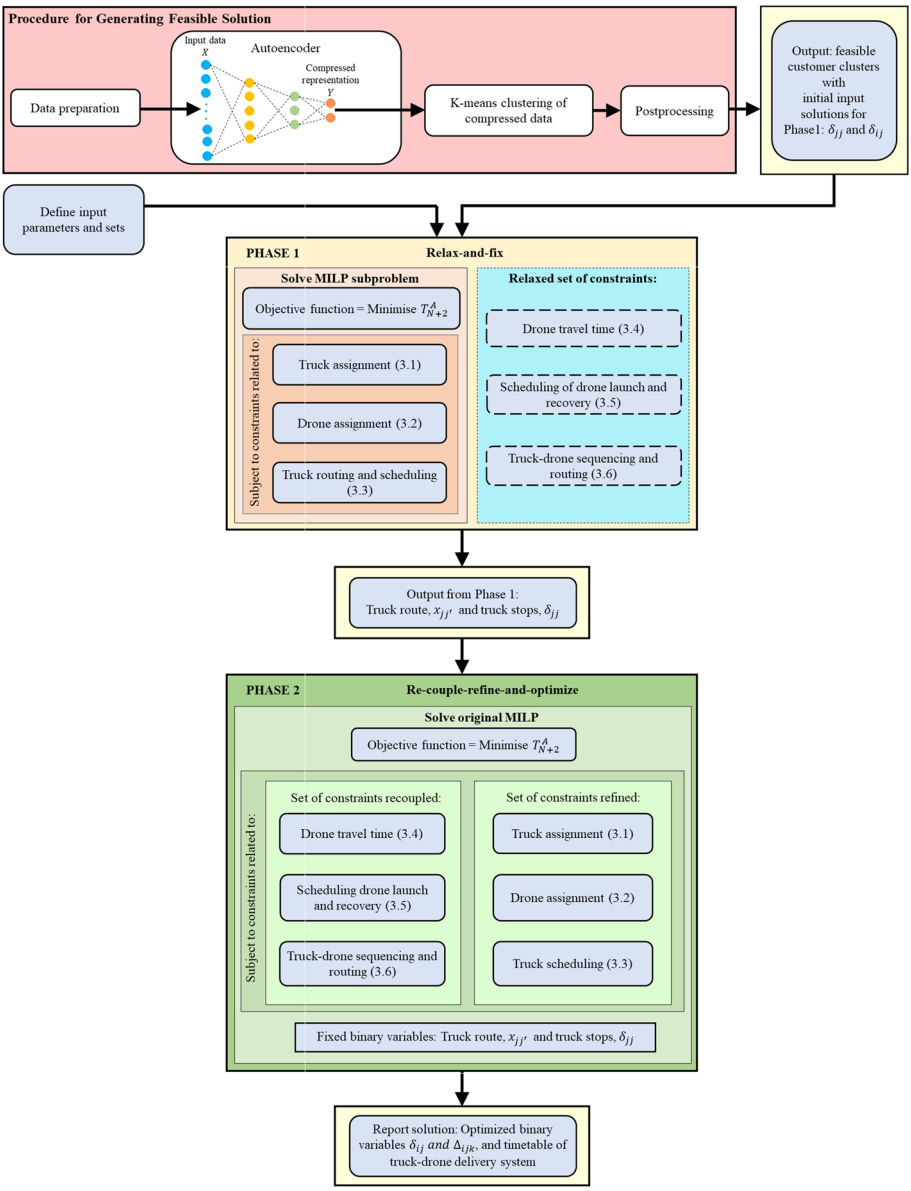

Fig. 4 Flowchart illustrating RF-RRO heuristic

addressed in the first phase are refined in Phase 2 with the goal of achieving faster delivery completion (i) focal node assignment (δ_{ij}), (ii) drone assignment (Δ_{ijk}), and (iii) truck schedule. In other words, the second phase of the heuristic seeks to improve the RF decisions by re-coupling drone travel and scheduling decisions in the presence of en route operations, and optimizing the associated decision variables to minimize the delivery completion time. Thus, the binary variables Δ_{ijk} and δ_{ij} are optimized in Phase 2 while taking into consideration

the spatial distribution of customers, proximity to truck visited nodes, and allowing en route drone operations.

Figure 5 illustrates the RF-RRO heuristic for a problem instance of customer size 10. The input data consist of output from the deep learning based initialization procedure constituting customer clusters, the set of customer locations and the warehouse location, and parameters listed under Sect. 3. RF heuristic solves the truck routing and scheduling with temporary decisions on focal node assignment (δ_{ij}) and drone assignment (Δ_{ijk}). The truck tour ($x_{jj'}$) and the set of truck stops (δ_{jj}) obtained after solving Phase 1 are given as input to the second phase of the heuristic. RRO solves the collaborative delivery of truck-drone system by optimizing the binary variables, Δ_{ijk} and δ_{ij}. Thus, the decisions on focal node assignment and drone assignment obtained in the RF phase are refined and optimized in the RRO phase along with the truck timetable after incorporating drone scheduling constraints. The objective function solution of the RF phase is 61 min, and the RRO phase is 67 min. However, if the refining of focal node assignments and truck schedule is not considered (i.e., RO instead of RRO), then the delivery completion time is increased to 87 min. Thus, the refinement of Phase 1 solution in Phase 2 leads to better solutions.

Proposition The optimal objective value to the relax-and-fix step (or Phase 1) of the solution approach (Z^*_{RF}) is a valid lower bound to the optimal objective value of the CTDRSP (Z^*).

Proof Consider an optimal solution of RF heuristic, Z^*_{RF}, which ensures the best truck route $x^*_{jj'}$ for a set of truck stops (δ^*_{jj}=1), while minimizing the delivery completion time of truck-stop locations. Thus, Z^*_{RF} solves the subproblem of truck routing and scheduling with the prospective assignment of customers to be served by drones from a specific focal node (δ_{ij} and Δ_{ijk}). Since drone sorties and scheduling are relaxed in this phase, the trucks will leave the stop immediately after serving the customer. In other words, there is no need for the truck to wait at a stop to launch or recover the drone. Therefore, the optimal delivery completion time for Phase 1, Z^*_{RF}, has no waiting time for trucks.

If Z^* is the solution of the CTDRSP, then the plan must include drone sorties, and each customer must be visited by a truck or drone. Thus, if the drone LRO is perfectly synchronized with the truck, then there is zero wait time for the truck, in which case, $Z^* = Z^*_{RF}$. In all other cases, the truck must wait for drones, which then increases the delivery completion time (i.e., $Z^* > Z^*_{RF}$). Thus, the optimal objective value for Phase 1 serves as a lower bound to the CTDRSP, $Z^* \geq Z^*_{RF}$. □

4.4 Variant 1: Min-cost CTDRSP-ELR

CTDRSP-ELR focuses on ensuring faster deliveries to customers while minimizing costs incurred during the delivery process remains a major concern for companies. In this section, we present the Min-Cost CTDRSP-ELR emphasizing on minimization of total costs involved in the delivery process, which includes the vehicle operating costs and idle time costs. A few researchers have considered minimization of cost in collaborative truck drone systems (Bouman et al., 2018; Ha et al., 2018). However, a cost minimization problem of truck-drone duo consisting of heterogeneous drones with flexible LRO along the truck route is proposed for the first time. The min time CTDRSP-ELR proposed in Sect. 3 can be adapted to formulate a MILP model to minimize the total costs of the truck-drone delivery system. It takes into account the total run time of the truck, including the waiting time (for package delivery and drone recovery) and the total flight duration of each drone as well as their idle times during hovering.

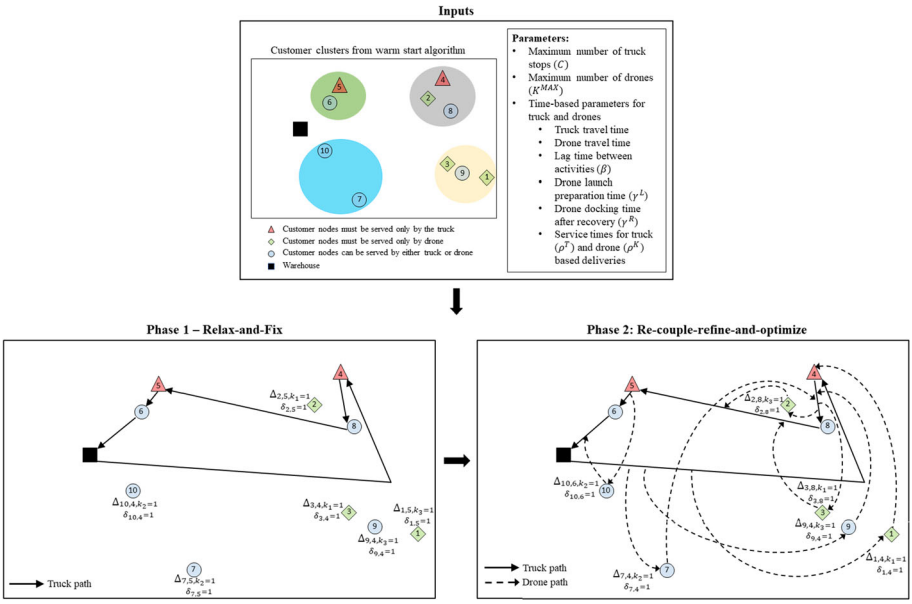

Fig. 5 Illustration of RF-RRO heuristic for a problem instance with 10 customers

We use the following additional notations to formulate the Min-Cost CTDRSP-ELR.

Sets

$l \in \mathcal{L}$ Set of depot and customer locations, $\mathcal{L} = N + 1, 1, 2, \ldots, N$
$l' \in \mathcal{L}'$ Set of customer locations and depot $\mathcal{L}' = 1, 2, \ldots, N, N + 2$

Parameters

α^T Operating cost of the truck per unit distance
V^T average speed of the truck
λ^T Idle time cost of the truck per unit time
α_k Operating cost of drone $k \in \mathcal{K}$ per unit distance
V_k Average speed of drone $k \in \mathcal{K}$
λ_k Idle time cost of drone $k \in \mathcal{K}$ per unit distance

Decision variables

O_{ki} Operating cost of drone $k \in \mathcal{K}$ serving customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$
I_{ki} Idle time cost of drone $k \in \mathcal{K}$ serving customer $i \in \mathcal{S}^K \cup \mathcal{S}^{TK}$

The objective function of the MILP model for Min-Cost CTDRSP-ELR minimizing the overall operating costs and idle time costs of the truck and heterogeneous drone fleet is given

by Eq. (68).

$$Minimize \quad Z = \sum_{l} \sum_{l'} (\tau_{ll'}^{T} \times x_{ll'}) \times V^{T} \times \alpha^{T} + (T_{N+2}^{A} - \sum_{l} \sum_{l'} (\tau_{ll'}^{T} \times x_{ll'})) \times \lambda^{T} + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{S}^{K} \cup \mathcal{S}^{TK}} O_{ki} + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{S}^{K} \cup \mathcal{S}^{TK}} I_{ki} \tag{68}$$

$S.t$ to following constraints, in addition to constraints discussed in Sect. 3.

$$O_{ki} \geq (t_i^{O} + t_i^{R}) \times V_k \times \alpha_k - M \times (1 - \sum_{j \in \mathcal{S}^{T} \cup \mathcal{S}^{TK}} \Delta_{ijk}) \quad \forall i \in \mathcal{S}^{K} \cup \mathcal{S}^{TK}, k \in \mathcal{K} \tag{69}$$

$$I_{ki} \geq ((T_i^{R} - T_i^{L}) - (t_i^{O} + t_i^{R})) \times V_k \times \lambda_k - M \times (1 - \sum_{j \in \mathcal{S}^{T} \cup \mathcal{S}^{TK}} \Delta_{ijk}) \quad \forall i \in \mathcal{S}^{K} \cup \mathcal{S}^{TK}, k \in \mathcal{K} \tag{70}$$

The operating cost of drones while delivering parcels to customer nodes is captured by Constraint (69). Similarly, the idle time cost of drones with respect to a customer node is determined in Constraint (70).

4.5 Variant 2: Collaborative multi-Truck-multi-Drone Routing and Scheduling Problem with en route Launch and Recovery operations (mCTDRSP-ELR)

CTDRSP-ELR considers a single-truck, multi-drone delivery problem with flexible drone operations along the truck route. In this section, we extend it to a multi-truck multi-drone setting referred to as Collaborative multi-Truck, multi-Drone Routing, and Scheduling Problem with en route Launch and Recovery operations (mCTDRSP-ELR). The solution technique involves an initial clustering phase of customers followed by routing and scheduling of vehicles (trucks and drones). A Deep-Learning-enabled Cluster-First Route-Second (DL-CFRS) heuristic is developed to solve the multi-truck variant of CTDRSP-ELR. Subsequently, the Relax-and-Fix with Re-couple-Refine-and-Optimize (RF-RRO) heuristic is adapted to the clusters formed.

Algorithm 3 presents the DL-CFRS heuristic proposed to solve the VRP variant of CTDRSP-ELR. The customer data, which includes the distance between the customer locations and customer categories along with the desired number of customer clusters, are entered as input data. The Deep Learning Technique presented in Algorithm 1, followed by k-means clustering detailed in Algorithm 2, is applied to determine the customer clusters. Each cluster is regarded as a CTDRSP-ELR comprising a truck and its fleet of drones. Consequently, the routing and scheduling of the truck-drone duo for each cluster are solved with the aid of Relax-and-Fix with Re-couple-Refine-and-Optimize (RF-RRO) heuristic illustrated in Fig. 4. The delivery completion time of each cluster is recorded, and the arrival time of the last truck to return to the warehouse is given as the output of DL-CFRS heuristic.

Algorithm 3 DL-CFRS heuristic to solve mCTDRSP-ELR

Inputs:
  Customer data, number of clusters
Establish customer clusters by sequentially executing Algorithm 1 and 2
for each customer cluster:
    Initialize CTDRSP model parameters
    Phase 1: RF heuristic
    Phase 2: RRO heuristic
    Report T^A_{N+2}
end for
Output:
  $max(T^A_{N+2})$

5 Numerical analysis

This section presents the numerical analysis performed to evaluate the performance of MILP and the two-phase heuristic for CTDRSP-ELR. The optimization models are developed on GAMS 33.1 and solved using IBM ILOG CPLEX version 12.10, and the deep clustering initialization procedure as well as the RF-RRO heuristic are coded in Python 3.7.6. The experiments are executed on high-performance computing systems with an Intel Xeon machine having two 2.4 GHz processors and 64 GB RAM memory. To assess the benefit of allowing en route operations, the solution obtained for CTDRSP-ELR is compared against CTDRSP-NLR (i.e., launch/recovery at truck stops only) for all test instances. Similar to previous work, the MILP models are run for up to 3600 s of CPU time, and the proposed heuristic (deep clustering + RF-RRO) is executed for a maximum CPU time of 1800 s.

5.1 Experiment setup

In this section, the different test instances to implement and analyze performances of MILP and RF-RRO heuristic are introduced. We generate problem instances of sizes 10, 20, 50, and 100 customers. Each level of customer size is tested for three different delivery areas to understand the impact of different demand densities. We consider the truck to carry 3 drones for instances with 10 customers, and 6 drones in all the remaining cases. Thus, a problem (or network) setting is characterized by the number of customers, delivery area, and drone fleet size. Also, we consider 10 instances for each network setting. Thus, a total of 12 network settings (A-L) and 120 instances are generated, as summarized in Table 2. It is assumed that 50% of customers can be served by both the truck and drones (i.e., set $\mathcal{S}^{TK} \subset \mathcal{N}$). The remaining share of customers is allocated among truck- and drone-only customers (i.e., sets $\mathcal{S}^{T} \subset \mathcal{N}$ and $\mathcal{S}^{K} \subset \mathcal{N}$). Further, to analyze the influence of the number of drones on delivery completion time, an additional set of instances are generated with 10 customers.

Consistent with prior work, the truck speed is assumed to be 40 km/h (Godwin et al., 2016; Ha et al., 2018; Salama & Srinivas, 2020; Wang & Sheu, 2019). The distance between customer nodes in the 2D delivery space is calculated using rectilinear and Euclidean metrics for truck and drone travel, respectively. Such an approach serves as a good approximation for the road network for truck and drone travel in low-altitude airspace (Holeczek, 2021;

Table 2 Summary of instances to assess the performance of MILP and heuristic

Network	Customer size	Number of drones	Area (km^2)	Density
A1 to A10	10	3	20×20	0.025
B1 to B10	10	3	30×30	0.011
C1 to C10	10	3	40×40	0.00625
D1 to D10	20	6	20×20	0.05
E1 to E10	20	6	30×30	0.022
F1 to F10	20	6	40×40	0.0125
G1 to G10	50	6	20×20	0.125
H1 to H10	50	6	30×30	0.055
I1 to I10	50	6	40×40	0.03125
J1 to J10	100	6	30×30	0.111
K1 to K10	100	6	40×40	0.0625
L1 to L10	100	6	50×50	0.04

Kitjacharoenchai et al., 2019; Murray & Chu, 2015; Salama & Srinivas, 2020). The lag time between two activities in the problem (β), which requires driver intervention, is considered to be 1 min. We consider a heterogeneous set of drones for experiments, motivated by the capability of commercial drones available in the market (such as DJI Matrice 300 RTK (DJI Matrice (2022)), Evo II Autel (Robotics (2022)), and eBeeX (eBee (2022))). Consistent with commercial drone specifications, the speeds of low-, medium- and high-payload drones are set to 80, 70, and 60 km/h, respectively. The onward flight endurance of each drone considered in the paper is with respect to the endurance limits specified by the manufacturer. Two drones of each drone type are considered for problem instances of customer size 20 and above. For problem instances of customer size 10, the number of drones considered for the study is three (i.e., one of each drone type). Both the launch preparation and recovery time of drones are set to 1 min. The service time at customer nodes that belong to $\mathcal{S}^T$, $\mathcal{S}^K$, and $\mathcal{S}^{TK}$ are assumed to be 4 min, 1 min, and 2 min, respectively.

5.2 Results for small problem instances

This section presents the performance of MILP model and RF-RRO heuristic for small test instances (i.e., network configuration A-F consisting of 10 and 20 customer nodes as shown in Table 2). For problem instances with 10 customers (network configurations A-C), the MILP model for CTDRSP-ELR is solved to optimality in an average time of 10 s, while the optimization model for the benchmark problem with fixed drone operations (i.e., CTDRSP-NLR) requires an average time of 16 s. On the other hand, the proposed RF-RRO heuristic algorithm is able to obtain a solution for both CTDRSP-ELR and CTDRSP-NLR in less than two seconds.

Table 3 presents the optimal delivery completion time obtained using MILP model and RF-RRO heuristic for both CTDRSP-ELR and CTDRSP-NLR for problem instances with 10 customers. The percentage difference between the MILP and heuristic solution is referred

to as "GAP". The percentage reduction in delivery completion time achieved by the heuristic solution for CTDRSP-ELR as opposed to CTDRSP-NLR is referred to as "Time Savings". It can be observed that allowing en route operations leads to an average time savings of about 30.13% (min:24.68%, max:39.02%). Moreover, the proposed RF-RRO is able to achieve optimal solutions for all 10-customer instances, except one (network A6), in the case of CTDRSP-ELR. The RF-RRO adapted for CTDRSP-NLR also yields superior performance and has an average gap of about 0%-3% from the optimal solution. Table 4 compares the performance of MILP and RF-RRO heuristic for instances with 20 nodes. Given the runtime limit of 3600 s, the MILP model is able to optimally solve all instances in the case of CTDRSP-ELR, in an average time of 530 s. On the contrary, the MILP model for CTDRSP-NLR did not converge to the optimal solution within the runtime limit for 8 out 30 instances (D5, D10, E6, E7, E8, F2, F6, and F9). However, the RF-RRO heuristic is able to obtain a solution within 20 s for both CTDRSP-ELR and CTDRSP-NLR. Similar to the findings involving 10 customers, the new variant introduced in this paper (CTDRSP-ELR) achieves substantial reduction in delivery completion time over CTDRSP-NLR for network settings with 20 customers. Likewise, the RF-RRO heuristic performs well for both cases and even achieves a better solution than the MILP model (since it is terminated at 3600 s and not solved to optimality) in the case of CTDRSP-NLR. In summary, for 60 instances (networks A-F) evaluated, the heuristic approach achieves optimal (or near-optimal) solutions for both CTDRSP-ELR and CTDRSP-NLR, while taking substantially shorter runtime than the optimization model. The advantage of employing a math-based heuristic for complex problems is the enhanced possibility of near-optimal solutions or quality solutions in a limited time. For small instances of size 10 and 20 described in the results section, the ability to obtain solutions similar to the MILP model in limited CPU time is the strength of the proposed approach, thereby elucidating the efficiency of the RF-RRO heuristic. More than 85% of small test instances are found to be solved optimally. The following subsections discuss the key characteristics observed in optimal solutions for CTDRSP-ELR.

5.2.1 Influence of focal nodes on delivery completion time

We observed that the optimal solution for CTDRSP-ELR leveraged most truck stops to act as a focal node for drone launch/recovery (either en route or at nodes). However, in general, for the same instances, the number of focal nodes is fewer in the case of fixed drone operations (CTDRSP-NLR). This may be because the number of sites that a drone can reach from a truck stop is fewer due to its limited flight range, thereby reducing the number of stops that can serve as focal nodes. However, with en route operations, the drones are likely to be launched/recovered at points that minimize the total distance (or time) traveled, thereby facilitating drone sorties that were previously not feasible. Figure 6 illustrates the utilization of focal nodes in both models, which is the share of truck visited nodes functioning as focal nodes to drone visited nodes. For both CTDRSP-NLR and CTDRSP-ELR, the utilization of focal nodes increases as customer size increases. This can be attributed to the increased demand for focal nodes among truck stops with increased delivery requests. The percentage improvement of CTDRSP-ELR over CTDRSP-NLR is 11.4%, 51%, and 20% for networks A, B, and C, respectively. It is observed that CTDRSP-ELR achieves substantial improvement over CTDRSP-NLR in utilizing most of the truck stops as focal nodes for drone LRO. Likewise, the percentage improvement in focal node utilization with the inclusion of en route operations is about 1%, 40%, and 18% for networks D, E and F, respectively.

Table 3 Delivery completion time for N = 10

Instance	CTDRSP-NLR			CTDRSP-ELR			Time savings (%)
	MILP	RF-RRO	Gap (%)	MILP	RF-RRO	Gap (%)	
A1	104	104	0.00	72	72	0.00	30.77
A2	109	109	0.00	82	82	0.00	24.77
A3	105	105	0.00	75	75	0.00	28.57
A4	113	113	0.00	78	78	0.00	30.97
A5	115	118	2.61	84	84	0.00	26.96
A6	101	101	0.00	66	67	1.52	34.65
A7	97	97	0.00	70	70	0.00	27.84
A8	113	115	1.77	82	82	0.00	27.43
A9	106	113	6.60	74	74	0.00	30.19
A10	98	98	0.00	67	67	0.00	31.63
Average	106.1	107.3	1.10	75	75.1	0.13	29.38
B1	164	164	0.00	100	100	0.00	39.02
B2	155	155	0.00	108	108	0.00	30.32
B3	159	159	0.00	111	111	0.00	30.19
B4	166	166	0.00	119	119	0.00	28.31
B5	156	156	0.00	111	111	0.00	28.85
B6	154	154	0.00	116	116	0.00	24.68
B7	147	147	0.00	107	107	0.00	27.21
B8	145	145	0.00	105	105	0.00	27.59
B9	157	157	0.00	108	108	0.00	31.21
B10	153	153	0.00	108	108	0.00	29.41
Average	155.6	155.6	0.00	109.3	109.3	0.00	29.68
C1	183	183	0.00	129	129	0.00	29.51
C2	216	227	5.09	147	147	0.00	31.94
C3	183	183	0.00	134	134	0.00	26.78
C4	203	214	5.42	142	142	0.00	30.05
C5	199	204	2.51	129	129	0.00	35.18
C6	201	217	7.96	143	143	0.00	28.86
C7	208	208	0.00	136	136	0.00	34.62
C8	197	216	9.64	146	146	0.00	25.89
C9	175	175	0.00	110	110	0.00	37.14
C10	192	192	0.00	128	128	0.00	33.33
Average	195.7	201.9	3.06	134.4	134.4	0.00	31.33

Table 4 Delivery completion time for N = 20

Instance	CTDRSP-NLR			CTDRSP-ELR			Time savings (%)
	MILP	RF-RRO	Gap (%)	MILP	RF-RRO	Gap (%)	
D1	140	141	0.71	108	108	0.00	22.86
D2	137	137	0.00	104	104	0.00	24.09
D3	143	145	1.40	111	111	0.00	22.38
D4	140	140	0.00	107	107	0.00	23.57
D5	143*	140	− 2.10	106	106	0.00	25.87
D6	145	147	1.38	112	112	0.00	22.76
D7	146	151	3.42	112	112	0.00	23.29
D8	150	150	0.00	114	114	0.00	24.00
D9	144	150	4.17	111	111	0.00	22.92
D10	135*	134	− 0.74	100	100	0.00	25.93
Average	142.3	143.5	0.82	108.5	108.5	0.00	23.75
E1	195	199	2.05	137	137	0.00	29.74
E2	205	206	0.49	137	137	0.00	33.17
E3	167	167	0.00	113	113	0.00	32.34
E4	191	203	6.28	132	132	0.00	30.89
E5	195	201	3.08	137	137	0.00	29.74
E6	204*	203	− 0.49	137	137	0.00	32.84
E7	202*	200	− 0.99	137	137	0.00	32.18
E8	189*	184	− 2.65	129	129	0.00	31.75
E9	192	192	0.00	131	131	0.00	31.77
E10	187	191	2.14	131	131	0.00	29.95
Average	192.7	194.6	0.99	132.1	132.1	0.00	31.45
F1	264	264	0.00	192	192	0.00	27.27
F2	260*	255	− 1.92	186	186	0.00	28.46
F3	247	247	0.00	179	179	0.00	27.53
F4	259	259	0.00	182	182	0.00	29.73
F5	250	253	1.20	189	189	0.00	24.40
F6	280*	276	− 1.43	197	197	0.00	29.64
F7	259	263	1.54	185	185	0.00	28.57
F8	263	263	0.00	191	191	0.00	27.38
F9	270*	263	− 2.59	191	191	0.00	29.26
F10	256	259	1.17	183	183	0.00	28.52
Average	260.8	260.2	− 0.20	187.5	187.5	0.00	28.11

*Optimal solution not obtained within preset CPU time

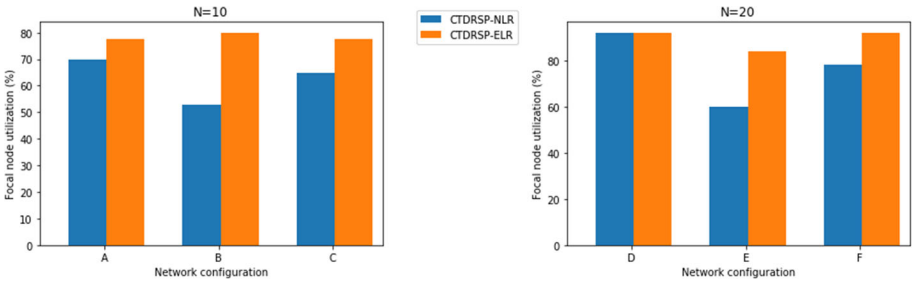

Fig. 6 Focal node utilization in small problem instances

5.2.2 Impact of different en route policies on truck/drone waiting time

Another notable characteristic that leads to a shorter delivery completion time is the synchronization of drone LRO with the truck movement in the case of CTDRSP-ELR. This reduces (or eliminates) the truck waiting time for drone arrival at a stop and vice versa. The proposed CTDRSP-ELR variant allows different en route operations—launch-only (e^L), recovery-only (e^R), en route LRO (e^{LR}). Besides, the LRO can also be at truck stops (e^F). The proposed solution approach (MILP and RF-RRO heuristic) determines the most appropriate drone LRO operation to coordinate with the truck route. Figures 7 and 8 summarize the distribution of en route operations for small instances and illustrate the reduction in the waiting time of the truck achieved by CTDRSP-ELR over CTDRSP-NLR for different problem networks, respectively. It can be observed that for instances with 10 and 20 customers, only en route operations are engaged (e^L, e^R and e^{LR}). Furthermore, en route operations for LRO of drones (e^{LR}) are more preferred over cases where either en route drone launch (e^L) or recovery (e^R) only takes place. It is observed that drone LRO fixed at node (e^F) is not preferred at all for both 10 and 20 customers. Therefore, these instances where en route drone operations subdue fixed drone operations at focal nodes reduce the wait time at a focal node for truck-drone reunification. The reduction in waiting time increases as the size of the delivery area increases. This may be because the customer density decreases (or distance between customer locations increases) with increasing delivery area, which could lead to longer flying time for drones when the LRO is restricted to the nodes (i.e., CTDRSP-NLR). However, the decrease in customer density is likely to have a lesser impact in the case of CTDRSP-ELR since the drone LRO can be scheduled to take place at strategically chosen points along the truck route such that the distance traveled is minimized. It is observed that the reduction achieved improves by an average of 22% as the problem size increases from 10 to 20 customers. This can be attributed to the increase in the number of drone operations with problem size, thus increasing the savings from each truck-drone launch/recovery synchronization.

The following subsections present other characteristics of optimal solutions (such as drone utilization, impact of fleet size, and drone types).

5.2.3 Impact of drone utilization and fleet size

Drone utilization is the ratio of average drone flying time to the total delivery time. The average drone utilization is over 60% for CTDRSP-ELR, and is about three times higher than CTDRSP-NLR for small problem instances. In other words, drone contribution in reducing the parcel delivery time is substantial in the case of the CTDRSP-ELR model, and it delineates

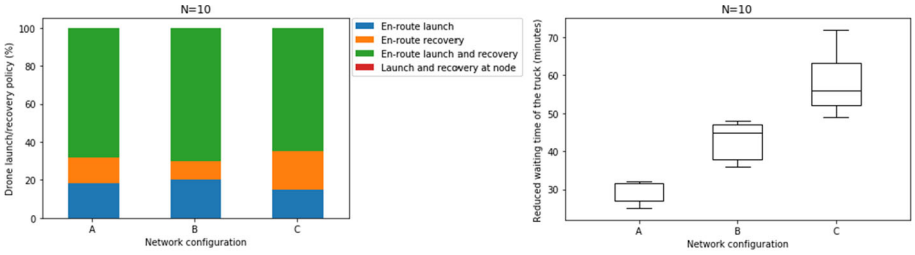

Fig. 7 Impact of different en route policies on truck/drone waiting time for $N = 10$ (i.e., networks $A - C$)

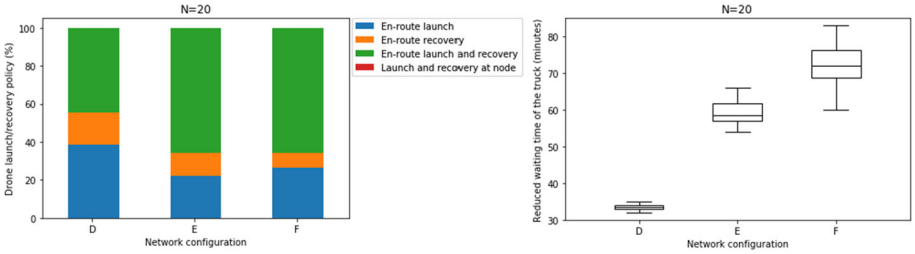

Fig. 8 Impact of different en route policies on truck/drone waiting time for $N = 20$ (i.e., networks $D - F$)

the impact of en route drone operations for efficient last-mile deliveries. Therefore, allowing en route operations enables drones, which are faster than the truck, to take the primary role in the last-mile delivery. On the other hand, several prior works that employ truck-drone tandems still rely on truck, while utilizing drones for aiding (rather than driving) the delivery operations. Thus, the CTDRSP-ELR model offers a novel approach and contributes to the existing literature on truck-drone delivery systems.

To understand the impact of the fleet size on the delivery completion time, we consider the impact of increasing the number of drones from three to six for problem instances with 10 customers. As expected, increasing the fleet size reduces the delivery completion time, as shown in Fig. 9. Specifically, it reduces by 4.9% and 7.9% for CTDRSP-NLR and CTDRSP-ELR, respectively, when the number of UAVs is increased from three to six. In other words, although the fleet size is doubled, the improvement in delivery completion time is not at a similar scale. Thus, it is clear that increasing the fleet size beyond a certain threshold may not be beneficial, especially considering the added operating costs and larger fleet management.

5.2.4 Insights on different drone types

In this research, we considered three different types of drones (low, medium and high) based on their payload capacity. The choice of drone types differs based on flexibility in the drone launch/return policy and spatial distribution of customers in a delivery area. Figure 10 presents the usage of different drone types for both CTDRSP-NLR and CTDRSP-ELR when the number of customers is 10 and 20.

For problem instances with 10 customers, all drone types are almost equally favored except for two cases, where low-payload drones are favored (network A of CTDRSP-NLR and network C of CTDRSP-ELR). When the number of nodes is increased to 20, the drone preference is not uniform, which indicates that as the density of customers increases, the diversity among them also increases, leading to a varied demand for drones. It is observed

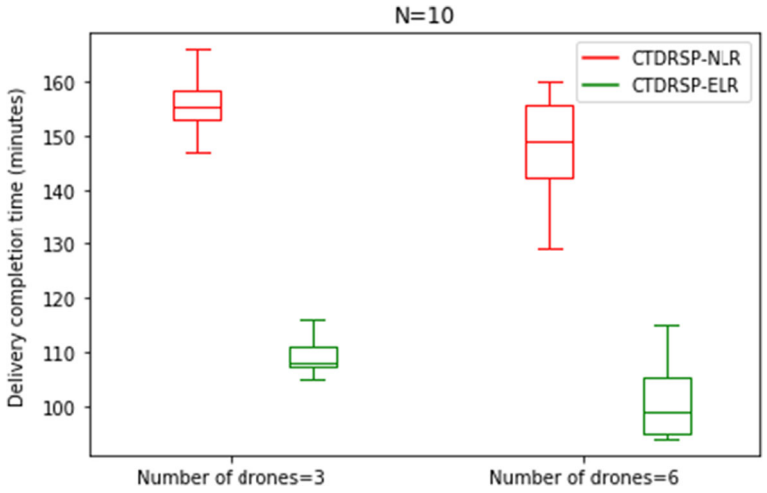

Fig. 9 Impact of number of drones on the performance of CTDRSP-NLR and CTDRSP-ELR models

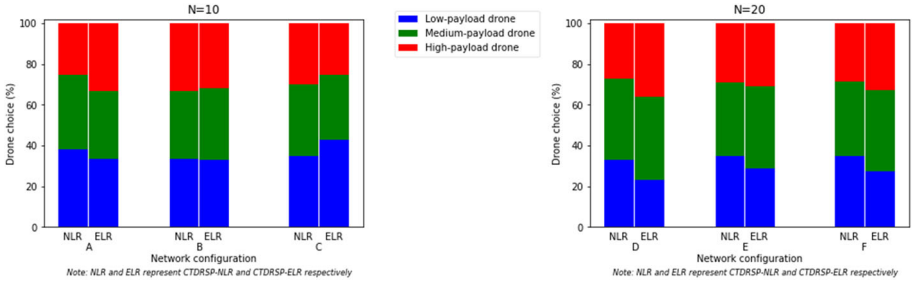

Fig. 10 Choice of drone type for small problem instances

that high-payload drones are least preferred for problem instances with 20 customers for CTDRSP-NLR (28%), and medium-payload drones are employed mostly for CTDRSP-ELR (40%) when the number of customers is 20. In the case of CTDRSP-ELR for both 10 and 20 customers, medium-payload drones have the highest preference, while usage of other drone types is almost comparable. The demand for medium-payload drones could be higher due to their advantage in terms of speed and payload features. They provide a shorter travel time than high-payload drones, and these drones are capable of serving all customers owing to their superior payload characteristics compared to low-payload drones. The variability in drone choice improves as the delivery area increases due to the more dispersed distribution of customers and their respective delivery demands, which is proportional to the payload capacity of drones.

5.3 Results for large problem instances

In this section, the computational experiments executed on instances with 50 and 100 customers are discussed. Due to the computational complexity associated with larger instances, MILP models may not be able to produce a feasible solution in a reasonable time. Therefore, we employ the RF-RRO heuristic for all large instances. Tables 5 and 6 present the

results for instances with 50 and 100 customers, respectively. Similar to the findings established for smaller instances, allowing en route operations leads to faster delivery completion. CTDRSP-ELR model leads to an average time savings of 30–40% over CTDRSP-NLR model for instances with 50 nodes, and this gain reduces to 22–24% as customer nodes increase to 100. All heuristic solutions reported for problem instances described in Tables 5 and 6 are obtained in less than 1800 s.

5.3.1 Impact of drone fleet composition

In all numerical experiments, an equal number of low, medium, and high-payload drones are maintained. In this section, the effect of varying drone composition on delivery completion time is studied for instances with 50 customers distributed in an area of size 30×30 km^2. In this experiment, the total number of drones considered is six, where at least one of each drone type is considered. The maximum number of drones for a drone type is limited to three. Table 7 summarizes the influence of different combinations of drones on delivery completion time.

It is observed that when a drone fleet composition with one low-payload drone, three medium-payload drones, and two high-payload drones is utilized, CTDRSP-ELR model reports the best delivery completion time. The most inferior drone composition for CTDRSP-ELR model consists of three low-payload drones, two medium-payload drones, and one high-payload drone. These observations cannot be generalized as the objective function value is dependent on Constraint (12) which prohibits low-payload drones from shipping parcels to a specific set of customers. Nevertheless, based on customer size, customer demands, and spatial distribution of nodes in the delivery area, a definite set of heterogeneous drones can be recommended based on insights from computational experiments to reduce the delivery completion time effectively.

5.3.2 Impact of spatial distribution of customers in delivery area

The delivery completion time is likely to be affected by the dispersion of nodes within the delivery region. Dry et al. (2012) studied the effect of spatial distribution of nodes on human performance in solving the TSP. Similarly, to assess the impact of customer distribution in a delivery area on delivery completion time, we consider three types of spatial distribution, namely random, clustered, and evenly distributed for a representative set of instances with 50 customers distributed over a region of 30×30 km^2 (as shown in Fig. 11). The random distribution assumes complete spatial randomness of the customer locations and is the most commonly considered case in the literature (Jeong et al., 2019; Wang & Sheu, 2019; Wang et al., 2019). In the case of clustered spatial distribution, there exist multiple clusters of customer locations distributed across the delivery region, where the intra-distance between customer locations are smaller within each cluster. On the other hand, spatial distribution is considered highly regular if customer locations are uniformly distributed across the delivery region (i.e., minimal randomness). Table 8 summarizes the results for the three types of spatial distribution. It is observed that the proposed heuristic model for CTDRSP-ELR is able to achieve substantial time savings irrespective of distribution pattern in the delivery area. With regard to CTDRSP-NLR, the delivery completion time is lowest for problem instances with the clustered distribution of customer locations and is maximum in regular distribution. Likewise, for CTDRSP-ELR, the delivery completion time is minimum for clustered locations, but unlike CTDRSP-NLR, the objective function values do not increase

Table 5 Delivery completion time for N = 50

Instance	CTDRSP-NLR / RF-RRO	CTDRSP-ELR / RF-RRO	Time savings (%)
G1	299	170	43.14
G2	297	193	35.02
G3	278	176	36.69
G4	264	176	33.33
G5	266	188	29.32
G6	273	163	40.29
G7	313	209	33.23
G8	278	185	33.45
G9	285	325	14.04
G10	279	188	32.62
Average	283.2	197.3	30.31
H1	412	253	38.59
H2	383	241	37.08
H3	391	254	35.04
H4	401	225	43.89
H5	407	219	46.19
H6	386	242	37.31
H7	407	236	42.01
H8	394	242	38.58
H9	450	252	44.00
H10	432	251	41.90
Average	406.3	241.5	40.46
I1	397	265	33.25
I2	389	249	35.99
I3	443	289	34.76
I4	457	269	41.14
I5	447	274	38.70
I6	405	259	36.05
I7	399	234	41.35
I8	482	261	45.85
I9	448	249	44.42
I10	452	247	45.35
Average	431.9	259.6	39.69

Table 6 Delivery completion time for N = 100

Instance	CTDRSP-NLR / RF-RRO	CTDRSP-ELR / RF-RRO	Time savings (%)
J1	933	752	19.40
J2	929	658	29.17
J3	970	663	31.65
J4	921	693	24.76
J5	967	663	31.44
J6	914	726	20.57
J7	919	678	26.22
J8	927	643	30.64
J9	932	674	27.68
J10	960	667	30.52
Average	937.2	681.7	27.20
K1	899	703	21.80
K2	919	678	26.22
K3	744	627	15.73
K4	910	765	15.93
K5	932	682	26.82
K6	965	663	31.30
K7	1013	738	27.15
K8	942	708	24.84
K9	1051	825	21.50
K10	971	798	17.82
Average	934.6	718.7	22.91
L1	1040	773	25.67
L2	919	753	18.06
L3	1015	804	20.79
L4	1107	812	26.65
L5	989	719	27.30
L6	913	774	15.22
L7	829	727	12.30
L8	1081	628	41.91
L9	1147	777	32.26
L10	1060	798	24.72
Average	1010	756.5	24.49

Table 7 Impact of drone fleet composition on delivery completion time for N = 50

Number of drones used			CTDRSP-NLR	CTDRSP-ELR	Time savings (%)
Low-payload	Medium-payload	High-payload	RF-RRO	RF-RRO	
1	2	3	394	236	40.10
1	3	2	388.4	230	40.78
2	1	3	378	234.4	37.99
2	3	1	389.2	236.8	39.16
3	1	2	414	233.4	43.62
3	2	1	400.2	240.2	39.98
2	2	2	393.6	236.2	39.99

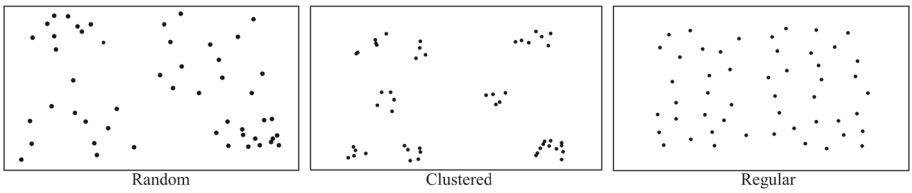

Fig. 11 Illustration of different spatial distributions of customer locations

Table 8 Impact of spatial distribution of customers on delivery completion time for N = 50

Spatial distribution	CTDRSP-NLR / RF-RRO	CTDRSP-ELR / RF-RRO	Time savings (%)
Random	396.4	233.4	40.95
Clustered	374.2	234.8	36.95
Regular	402.6	239.8	40.37

substantially for randomly and regularly distributed customers. Since a clustered distribution allows more opportunities for drone deliveries at each truck stop (due to most customers being within the drone's flying range within a cluster), it is naturally an effective spatial distribution for truck-drone tandem. On the other hand, a uniformly distributed region restricts such opportunities, especially when the drone LRO are restricted to the nodes. Since en route LRO facilitates drone-assisted deliveries to locations that are impossible with restricted drone launch/receipt operations, its impact on the delivery completion time for different spatial distribution is not greatly impacted for the instances evaluated.

5.4 Results for Min-cost CTDRSP-ELR

In this section, we demonstrate the performance of Min-Cost CTDRSP-ELR proposed in Sect. 4.4. For numerical experiments, the truck travel cost is assumed to be \$2.25/km, and drone travel costs are assumed to be \$0.08/km, \$0.09/km, and \$0.1/km, respectively, for low-, medium- and high-payload drones. The truck idle time cost is considered to be \$10/hour to account for parking and driver charges (Ha et al., 2018). The drone idle time cost is assumed to be the same as their operating cost as the idle time includes hovering (Tiniç et al., 2023). The numerical experiments for Min-Cost CTDRSP-ELR is tested on problem instances of size 10 and 50 dispersed in a delivery area of size 30×30 km^2. The drone fleet considered is the same as in earlier instances to minimize the delivery completion time. The results are displayed in Tables 9 and 10 for problem instances of size 10 and 50, respectively. The "Cost savings" mentioned is the percentage reduction in total delivery costs achieved by Min-Cost CTDRSP-ELR over Min-Cost CTDRSP-NLR. Small problem instances are solved using the MILP model proposed in Sect. 4.4 and larger instances are solved by adapting RF-RRO heuristic with the inclusion of cost components in Phase 2.

For problem size 10, the average time for solving the cost minimization model is 30 s. For problem instances of size 50, the CPU time limit is 1800 s. Referring to Tables 9 and 10, it is observed that the Min-Cost CTDRSP-ELR achieves better solutions than Min-Cost CTDRSP-NLR except for case B4, where node based drone LRO and en route LRO provide

Table 9 Total delivery cost for N = 10

Instance	Min-Cost CTDRSP-NLR	Min-Cost CTDRSP-ELR	Cost savings (%)
B1	124.52	114.08	8.39
B2	127.21	119.22	6.28
B3	137.36	127.90	6.89
B4	136.06	136.06	0.00
B5	133.31	124.46	6.64
B6	141.37	133.74	5.40
B7	131.68	124.66	5.33
B8	131.34	123.91	5.65
B9	136.95	127.10	7.19
B10	131.50	121.46	7.64
Average	133.13	125.26	5.94

Table 10 Total delivery cost for N = 50

Instance	Min-Cost CTDRSP-NLR	Min-Cost CTDRSP-ELR	Cost savings (%)
H1	202.10	163.55	19.08
H2	193.49	153.75	20.54
H3	237.61	190.11	19.99
H4	198.32	158.91	19.87
H5	213.46	155.64	27.09
H6	242.42	177.87	26.63
H7	215.25	168.38	21.77
H8	216.80	187.63	13.46
H9	236.85	203.21	14.20
H10	258.67	196.62	23.99
Average	221.50	175.57	20.66

the same solution. This implies that node-based LRO works best for the particular problem instance due to the location of customer nodes in the delivery area even though en route LRO is enabled. The average cost savings is 6% for instances with 10 customers and 21% for instances with 50 customers when the LRO of drones is relaxed to take place along the truck route in addition to customer nodes visited by the truck. The cost savings improve with an increase in customer locations leading to a rise in en route drone deliveries for their reduced operating cost. The experimental results show that the model tries to reduce drone travel costs, including hovering costs, by ensuring en route LRO. The drone travel costs are further reduced by utilizing medium-payload drones for most drone delivery trips. It is observed that at least 50% drone sorties are served by medium-payload drones. Low-payload and heavy-

Table 11 Delivery completion time for N = 50

Instance	mCTDRSP-NLR / RF-RRO	mCTDRSP-ELR / RF-RRO	Time savings (%)
H1	163	104	36.20
H2	151	97	35.76
H3	145	104	28.28
H4	150	117	22.00
H5	190	113	40.53
H6	182	106	41.76
H7	186	137	26.34
H8	160	112	30.00
H9	169	116	31.36
H10	178	116	34.83
Average	167.4	112.2	32.97

payload drones are less preferred by the Min-Cost model due to their low payload capacity and cost, respectively.

5.5 Results for mCTDRSP

In this section, the performance of DL-CFRS heuristic proposed in Sect. 4.5 to solve mCTDRSP is demonstrated. The customers dispersed in the delivery area are divided into different clusters with the help of k-means clustering technique of encoded customer data which is obtained from an autoencoder network. Each cluster is served by a truck and its heterogeneous drone fleet. Numerical experiments are executed on instances with 50 and 100 customers for a delivery area of size 30×30 sq.km. Also, trucks are considered to be homogeneous in their operational characteristics in every instance considered for experimental analysis. Tables 11 and 12 summarize the results of computational experiments for instances with 50 and 100 customers, respectively. All solutions are reported in less than 1800 s. The tables exhibit how mCTDRSP-ELR outperforms mCTDRSP-NLR in all cases. For problem instances of size 50 and 100, the total delivery time saved is about 33% with the inclusion of en route LRO.

6 Conclusion

This paper presents a new variant of truck multi-drone tandem called the Collaborative Truck-multi-Drone Routing and Scheduling Problem with en route Launch and Recovery operations (CTDRSP-ELR). Specifically, we take into account both the routing and scheduling decisions pertaining to the truck-multi-drone delivery system, while allowing drone launch and recovery operations (LRO) to be along the truck route. A mixed integer linear programming (MILP) model is formulated to optimize the decisions associated with CTDRSP-ELR. To handle large instances, a two-phase decomposition heuristic that involves a relax-and-fix with re-

Table 12 Delivery completion time for N = 100

Instance	mCTDRSP-NLR / RF-RRO	mCTDRSP-ELR / RF-RRO	Time savings (%)
J1	173	122	29.48
J2	191	134	29.84
J3	217	132	39.17
J4	179	126	29.61
J5	196	126	35.71
J6	193	132	31.61
J7	182	126	30.77
J8	190	125	34.21
J9	213	132	38.03
J10	193	126	34.72
Average	192.7	128.1	33.31

couple-refine-and-optimize (RF-RRO) approach is also proposed. A deep autoencoder-based clustering is developed to obtain a feasible solution to the CTDRSP, which is then used as an initial solution to warm start the RF-RRO heuristic. Extensive numerical experiments have been conducted by benchmarking the performance of the proposed CTDRSP-ELR against a truck-multi-drone tandem that restricts en route drone operations (denoted as CTDRSP-NLR). The MILP model is able to obtain an optimal solution for most small problem instances. Besides, the proposed RF-RRO heuristic also yields optimal (or near-optimal) solutions for these small instances. We also propose a lower bound to the CTDRSP-ELR model based on the proposed heuristic. We have also introduced a cost minimization variant of the proposed model, Min-Cost CTDRSP-ELR to minimize the total delivery cost which includes vehicle operating costs and idle time costs. The proposed CTDRSP-ELR is further extended to mCTDRSP-ELR which takes into account multiple trucks and their fleet of heterogeneous drones. This vehicle routing variant is solved by applying a deep learning-based clustering technique, followed by routing of the truck in each cluster using RF-RRO heuristic which also ensures feasible scheduling of truck-drone operations. This extension of CTDRSP enables the model to be applied in scenarios where vehicle (trucks and drones) utilization is in great demand. Numerical results suggest an average reduction of about 30% in delivery completion time with en route drone operations. Moreover, drone utilization and total waiting time are substantially better for CTDRSP-ELR. Our study also showed that the drone fleet preference varies depending on drone operation policies and customer size. Furthermore, the impact of different drone combinations and spatial distribution of customers in a delivery area on the objective function are also analyzed, where CTDRSP-ELR is shown to be efficient in substantially reducing the delivery completion time.

We believe that this study opens a plethora of opportunities for future research. The CTDRSP-ELR model can be improved to accommodate multiple parcel deliveries by drones in a single dispatch. Also, the proposed model can be reinforced by adding time window constraints to address the real-life challenges of last-mile deliveries. There is also scope to enhance the model by integrating simultaneous delivery and pick-up. The inclusion of these

additional features will strengthen the model and enable a more flexible truck-drone delivery system.

Acknowledgements The authors are extremely grateful for the valuable feedback from the editor and reviewers for helping us to improve the content and presentation of the paper.

Declarations

Conflict of interest None of the authors have conflicts of interest or any competing interest..

References

Adler, A. (2020). Workhorse perfecting HorseFly truck-based drone delivery—FreightWaves. https://www.freightwaves.com/news/workhorse-perfecting-truck-based-autonomous-drone-delivery

Agatz, N., Bouman, P., & Schmidt, M. (2018). Optimization approaches for the traveling salesman problem with drone. *Transportation Science, 52*(4), 965–981. https://doi.org/10.1287/TRSC.2017.0791

Ali, A., Gajpal, Y., & Elmekkawy, T. Y. (2021). Distributed permutation flowshop scheduling problem with total completion time objective. *OPSEARCH, 58*(2), 425–447. https://doi.org/10.1007/s12597-020-00484-3

Bai, R., Woodward, J. R., Subramanian, N., & Cartlidge, J. (2018). Optimisation of transportation service network using κ-node large neighbourhood search. *Computers and Operations Research, 89*, 193–205. https://doi.org/10.1016/j.cor.2017.06.008

Benarbia, T., & Kyamakya, K. (2021). A literature review of drone-based package delivery logistics systems and their implementation feasibility. *Sustainability, 14*(1), 360. https://doi.org/10.3390/SU14010360

Boccia, M., Masone, A., Sforza, A., & Sterle, C. (2021). A column-and-row generation approach for the flying sidekick travelling salesman problem. *Transportation Research Part C: Emerging Technologies, 124*, 102913. https://doi.org/10.1016/J.TRC.2020.102913

Bosona, T. (2020). Urban freight last mile logistics-challenges and opportunities to improve sustainability: A literature review. *Sustainability, 12*(21), 8769. https://doi.org/10.3390/su12218769

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

Brahimi, N., & Aouam, T. (2015). Multi-item production routing problem with backordering: A MILP approach. *International Journal of Production Research, 54*(4), 1076–1093. https://doi.org/10.1080/00207543.2015.1047971

Carlsson, J. G., & Song, S. (2017). Management science coordinated logistics with a truck and a drone. *Management Science, 64*(9), 4052–4069. https://doi.org/10.1287/mnsc.2017.2824

Chang, Y. S., & Lee, H. J. (2018). Optimal delivery routing with wider drone-delivery areas along a shorter truck-route. *Expert Systems with Applications, 104*, 307–317. https://doi.org/10.1016/J.ESWA.2018.03.032

Chowdhury, S., Emelogu, A., Marufuzzaman, M., Nurre, S. G., & Bian, L. (2017). Drones for disaster response and relief operations: A continuous approximation model. *International Journal of Production Economics, 188*, 167–184. https://doi.org/10.1016/J.IJPE.2017.03.024

Chung, J. (2018). Heuristic method for collaborative parcel delivery with drone. *Journal of Distribution Science, 16*(2), 19–24. https://doi.org/10.15722/JDS.16.2.201802.19

Chung, S. H., Sah, B., & Lee, J. (2020). Optimization for drone and drone-truck combined operations: A review of the state of the art and future directions. *Computers and Operations Research, 123*, 105004. https://doi.org/10.1016/j.cor.2020.105004

Dairi, A., Harrou, F., Sun, Y., & Senouci, M. (2018). Obstacle detection for intelligent transportation systems using deep stacked autoencoder and k-nearest neighbor scheme. *IEEE Sensors Journal, 18*(12), 5122–5132. https://doi.org/10.1109/JSEN.2018.2831082

de Freitas, J. C., & Penna, P. H. V. (2020). A variable neighborhood search for flying sidekick traveling salesman problem. *International Transactions in Operational Research, 27*(1), 267–290. https://doi.org/10.1111/itor.12671

Dellaert, N., Van Woensel, T., Crainic, T. G., & Dashty Saridarq, F. (2021). A multi-commodity two-Echelon capacitated vehicle routing problem with time windows: Model formulations and solution approach. *Computers & Operations Research, 127*, 105154. https://doi.org/10.1016/J.COR.2020.105154

Dell'Amico, M., Montemanni, R., & Novellani, S. (2020). Matheuristic algorithms for the parallel drone scheduling traveling salesman problem. *Annals of Operations Research*. https://doi.org/10.1007/s10479-020-03562-3

Dell'Amico, M., Montemanni, R., & Novellani, S. (2022). Exact models for the flying sidekick traveling salesman problem. *International Transactions in Operational Research, 29*(3), 1360–1393. https://doi.org/10.1111/ITOR.13030

DJI Matrice 300 RTK (2022). https://www.dji.com/de/matrice-300/specs

Dolan, S. (2022). Last mile delivery logistics explained: Problems & solutions. https://www.businessinsider.com/last-mile-delivery-shipping-explained?IR=T

Dry, M., Preiss, K., & Wagemans, J. (2012). Clustering, randomness, and regularity: Spatial distributions and human performance on the traveling salesperson problem and minimum spanning tree problem. *The Journal of Problem Solving*. https://doi.org/10.7771/1932-6246.1117

eBee X (2022). https://www.sensefly.com/drone/ebee-x-fixed-wing-drone/

Es Yurek, E., & Ozmutlu, H. C. (2018). A decomposition-based iterative optimization algorithm for traveling salesman problem with drone. *Transportation Research Part C: Emerging Technologies, 91*, 249–262. https://doi.org/10.1016/j.trc.2018.04.009

Ferrandez, S. M., Harbison, T., Weber, T., Sturges, R., & Rich, R. (2016). Optimization of a truck-drone in tandem delivery network using k-means and genetic algorithm. *Journal of Industrial Engineering and Management, 9*(2), 374–388. https://doi.org/10.3926/jiem.1929

Godwin, T., Sajeev, K., & George, A. C. (2016). Finding time-robust fuel-efficient paths for a call-taxi in a stochastic city road network. *Journal of Advanced Transportation, 50*(6), 1156–1180. https://doi.org/10.1002/atr.1395

Gomez-Lagos, J., Candia-Vejar, A., & Encina, F. (2021). A new truck-drone routing problem for parcel delivery services aided by parking lots. *IEEE Access, 9*, 11091–11108. https://doi.org/10.1109/ACCESS.2021.3050658

Gonzalez-R, P. L., Canca, D., Andrade-Pineda, J. L., Calle, M., & Leon-Blanco, J. M. (2020). Truck-drone team logistics: A heuristic approach to multi-drop route planning. *Transportation Research Part C: Emerging Technologies, 114*, 657–680. https://doi.org/10.1016/J.TRC.2020.02.030

Goodchild, A., & Toy, J. (2018). Delivery by drone: An evaluation of unmanned aerial vehicle technology in reducing CO_2 emissions in the delivery service industry. *Transportation Research Part D: Transport and Environment, 61*, 58–67. https://doi.org/10.1016/J.TRD.2017.02.017

Ha, Q. M., Deville, Y., Pham, Q. D., & Hà, M. H. (2018). On the min-cost traveling salesman problem with drone. *Transportation Research Part C: Emerging Technologies, 86*, 597–621. https://doi.org/10.1016/j.trc.2017.11.015

Ham, A. M. (2018). Integrated scheduling of m-truck, m-drone, and m-depot constrained by time-window, drop-pickup, and m-visit using constraint programming. *Transportation Research Part C: Emerging Technologies, 91*, 1–14. https://doi.org/10.1016/J.TRC.2018.03.025

Holeczek, N. (2021). Analysis of different risk models for the hazardous materials vehicle routing problem in urban areas. *Cleaner Environmental Systems, 2*, 100022. https://doi.org/10.1016/j.cesys.2021.100022

Hong, I., Kuby, M., & Murray, A. T. (2018). A range-restricted recharging station coverage model for drone delivery service planning. *Transportation Research Part C: Emerging Technologies, 90*, 198–212. https://doi.org/10.1016/J.TRC.2018.02.017

Hu, C., Lu, J., Liu, X., & Zhang, G. (2018). Robust vehicle routing problem with hard time windows under demand and travel time uncertainty. *Computers and Operations Research, 94*, 139–153. https://doi.org/10.1016/j.cor.2018.02.006

Intelligence, I. (2022). Amazon, UPS, Domino's & the future of drone delivery services. https://www.businessinsider.com/drone-delivery-services?IR=T

Irving, M. (2021). Amazon patents trucks that unleash and direct fleets of delivery drones. https://newatlas.com/drones/amazon-patent-fleets-delivery-drone/

Jeong, H. Y., Song, B. D., & Lee, S. (2019). Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones. *International Journal of Production Economics*. https://doi.org/10.1016/j.ijpe.2019.01.010

Karak, A., & Abdelghany, K. (2019). The hybrid vehicle-drone routing problem for pick-up and delivery services. *Transportation Research Part C: Emerging Technologies, 102*, 427–449. https://doi.org/10.1016/j.trc.2019.03.021

Kennedy, C. (2020). How drones are used for life-saving healthcare | World Economic Forum. https://www.weforum.org/agenda/2020/04/medicines-from-the-sky-how-a-drone-may-save-your-life/

Kim, S. J., Lim, G. J., Cho, J., & Côté, M. J. (2017). Drone-aided healthcare services for patients with chronic diseases in rural areas. *Journal of Intelligent & Robotic Systems, 88*(1), 163–180. https://doi.org/10.1007/S10846-017-0548-Z

Kitjacharoenchai, P., Ventresca, M., Moshref-Javadi, M., Lee, S., Tanchoco, J. M., & Brunese, P. A. (2019). Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. *Computers and Industrial Engineering, 129*, 14–30. https://doi.org/10.1016/j.cie.2019.01.020

Kosanoglu, F., Atmis, M., & Turan, H. H. (2022). A deep reinforcement learning assisted simulated annealing algorithm for a maintenance planning problem. *Annals of Operations Research*. https://doi.org/10.1007/s10479-022-04612-8

Leon-Blanco, J. M., Gonzalez-R, P., Andrade-Pineda, J. L., Canca, D., & Calle, M. (2022). A multi-agent approach to the truck multi-drone routing problem. *Expert Systems with Applications, 195*, 116604. https://doi.org/10.1016/J.ESWA.2022.116604

Li, H., Chen, J., Wang, F., & Zhao, Y. (2022). Truck and drone routing problem with synchronization on arcs. *Naval Research Logistics (NRL)*. https://doi.org/10.1002/NAV.22053

Luo, Q., Wu, G., Ji, B., Wang, L., & Suganthan, P. N. (2021). Hybrid multi-objective optimization approach with pareto local search for collaborative truck-drone routing problems considering flexible time windows. *IEEE Transactions on Intelligent Transportation Systems*. https://doi.org/10.1109/TITS.2021.3119080

Marinelli, M., Caggiani, L., Ottomanelli, M., & Dell'Orco, M. (2018). En route truck-drone parcel delivery for optimal vehicle routing strategies. *IET Intelligent Transport Systems, 12*(4), 253–261. https://doi.org/10.1049/IET-ITS.2017.0227

Masone, A., Poikonen, S., & Golden, B. L. (2022). The multivisit drone routing problem with edge launches: An iterative approach with discrete and continuous improvements. *Networks*. https://doi.org/10.1002/NET.22087

Melendez, S. (2021). Alphabet Wing drones near 100,000 delivery milestone. https://www.fastcompany.com/90669760/alphabet-wing-drones-chicken-delivery

Moshref-Javadi, M., Hemmati, A., & Winkenbach, M. (2020). A truck and drones model for last-mile delivery: A mathematical model and heuristic approach. *Applied Mathematical Modelling, 80*, 290–318. https://doi.org/10.1016/j.apm.2019.11.020

Moshref-Javadi, M., Lee, S., & Winkenbach, M. (2020). Design and evaluation of a multi-trip delivery model with truck and drones. *Transportation Research Part E: Logistics and Transportation Review, 136*, 101887. https://doi.org/10.1016/j.tre.2020.101887

Moshref-Javadi, M., & Winkenbach, M. (2021). Applications and Research avenues for drone-based models in logistics: A classification and review. *Expert Systems with Applications, 177*, 114854. https://doi.org/10.1016/J.ESWA.2021.114854

Murray, C. C., & Chu, A. G. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies, 54*, 86–109. 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. *Transportation Research Part C: Emerging Technologies, 110*, 368–398. https://doi.org/10.1016/J.TRC.2019.11.003

Otto, A., Agatz, N., Campbell, J., Golden, B., & Pesch, E. (2018). Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. *Networks, 72*(4), 411–458. https://doi.org/10.1002/NET.21818

Perera, S., Dawande, M., Janakiraman, G., & Mookerjee, V. (2020). Retail deliveries by drones: How will logistics networks change? *Production and Operations Management, 29*(9), 2019–2034. https://doi.org/10.1111/poms.13217

Pichka, K., Bajgiran, A. H., Petering, M. E., Jang, J., & Yue, X. (2018). The two echelon open location routing problem: Mathematical model and hybrid heuristic. *Computers and Industrial Engineering, 121*, 97–112. https://doi.org/10.1016/j.cie.2018.05.010

Pierce, D. (2013). Delivery drones are coming: Jeff Bezos promises half-hour shipping with Amazon Prime Air—The Verge. https://www.theverge.com/2013/12/1/5164340/delivery-drones-are-coming-jeff-bezos-previews-half-hour-shipping

Pina-Pardo, J. C., Silva, D. F., & Smith, A. E. (2021). The traveling salesman problem with release dates and drone resupply. *Computers & Operations Research, 129*, 105170. https://doi.org/10.1016/J.COR.2020.105170

Poikonen, S., & Golden, B. (2020). Multi-visit drone routing problem. *Computers & Operations Research, 113*, 104802. https://doi.org/10.1016/J.COR.2019.104802

Ponza, A. (2016). Optimization of drone-assisted parcel delivery. Masters degree thesis, UNIVERSIT'A DEGLI STUDI DI PADOVA.

Robotics, A. (2022). EVO II Autel. https://auteldrones.com/pages/evo-ii-specification

Rosenthal, T. C. (2017). Vans & Drones in Zurich: Mercedes-Benz Vans, Matternet and siroop start pilot project for on-demand delivery of e-commerce goods—Daimler Global Media Site. https://media.daimler.com/

marsMediaSite/en/instance/ko/Vans--Drones-in-Zurich-Mercedes-Benz-Vans-Matternet-and-siroop-start-pilot-project-for-on-demand-delivery-of-e-commerce-goods.xhtml?oid=29659139
Salama, M., & Srinivas, S. (2020). Joint optimization of customer location clustering and drone-based routing for last-mile deliveries. *Transportation Research Part C: Emerging Technologies, 114*, 620–642. https://doi.org/10.1016/j.trc.2020.01.019
Salama, M. R., & Srinivas, S. (2022). Collaborative truck multi-drone routing and scheduling problem: Package delivery with flexible launch and recovery sites. *Transportation Research Part E: Logistics and Transportation Review, 164*, 102788. https://doi.org/10.1016/J.TRE.2022.102788
San, K. T., Lee, E. Y., & Chang, Y. S. (2016). The delivery assignment solution for swarms of UAVs dealing with multi-dimensional chromosome representation of genetic algorithm. In *2016 IEEE 7th annual ubiquitous computing, electronics and mobile communication conference, UEMCON 2016*. https://doi.org/10.1109/UEMCON.2016.7777839.
Schermer, D., Moeini, M., & Wendt, O. (2019). A hybrid VNS/Tabu search algorithm for solving the vehicle routing problem with drones and en route operations. *Computers & Operations Research, 109*, 134–158. https://doi.org/10.1016/J.COR.2019.04.021
Shukla, A. C., Deshmukh, S. G., & Kanda, A. (2010). Flexibility and sustainability of supply chains: Are they together? *Global Journal of Flexible Systems Management, 11*(1–2), 25–38. https://doi.org/10.1007/bf03396576
Tiniç, G. O., Karasan, O. E., Kara, B. Y., Campbell, J. F., & Ozel, A. (2023). Exact solution approaches for the minimum total cost traveling salesman problem with multiple drones. *Transportation Research Part B: Methodological, 168*, 81–123. https://doi.org/10.1016/J.TRB.2022.12.007
Toledo, C. F. M., da Silva Arantes, M., Hossomi, M. Y. B., França, P. M., & Akartunalı, K. (2015). A relax-and-fix with fix-and-optimize heuristic applied to multi-level lot-sizing problems. *Journal of Heuristics, 21*(5), 687–717. https://doi.org/10.1007/s10732-015-9295-0
Vásquez, S. A., Angulo, G., & Klapp, M. A. (2021). An exact solution method for the TSP with Drone based on decomposition. *Computers & Operations Research, 127*, 105127. https://doi.org/10.1016/J.COR.2020.105127
Vu, L., Vu, D. M., Hà, M. H., & Nguyen, V. (2022). The two-echelon routing problem with truck and drones. *International Transactions in Operational Research, 29*(5), 2968–2994. https://doi.org/10.1111/itor.13052
Wang, D., Hu, P., Du, J., Zhou, P., Deng, T., & Hu, M. (2019). Routing and scheduling for hybrid truck-drone collaborative parcel delivery with independent and truck-carried drones. *IEEE Internet of Things Journal, 6*(6), 10483–10495. https://doi.org/10.1109/JIOT.2019.2939397
Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. *Transportation Research Part B: Methodological, 122*, 350–364. https://doi.org/10.1016/J.TRB.2019.03.005
Wolf, C. D. (2021). UPS starts drone delivery of COVID vaccines | Transport Topics. https://www.ttnews.com/articles/ups-starts-drone-delivery-covid-vaccines
Wu, G., Mao, N., Luo, Q., Xu, B., Shi, J., & Suganthan, P. N. (2022). Collaborative truck-drone routing for contactless parcel delivery during the epidemic. *IEEE Transactions on Intelligent Transportation Systems*. https://doi.org/10.1109/TITS.2022.3181282
Zeng, J., Wang, J., Guo, L., Fan, G., Zhang, K., & Gui, G. (2019). Cell scene division and visualization based on autoencoder and K-means algorithm. *IEEE Access, 7*, 165217–165225. https://doi.org/10.1109/ACCESS.2019.2953184
Zeng, W., Xu, Z., Cai, Z., Chu, X., & Lu, X. (2021). Aircraft trajectory clustering in terminal airspace based on deep autoencoder and gaussian mixture model. *Aerospace, 8*(9), 266. https://doi.org/10.3390/aerospace8090266
Zipline, Pfizer and BioNTech Collaboration Paves the Way for Automated, On-Demand Delivery of First mRNA COVID-19 Vaccines in Ghana | BioNTech (2021). https://investors.biontech.de/news-releases/news-release-details/zipline-pfizer-and-biontech-collaboration-paves-way-automated