Dynamic Truck–UAV Collaboration and Integrated Route Planning for Resilient Urban Emergency Response

Yuying Long, Gangyan Xu, *Member, IEEE*, Jinqiu Zhao, Binglei Xie, and Meng Fang

***Abstract*—Efficient urban emergency response is vital for saving lives and property after disasters. However, urban emergency response is challenging because it involves many demands, with a very tight time for making decisions, and is frequently threatened by the disruption of road networks and infrastructures. Taking these challenges into consideration, this article proposes a dynamic truck-UAV (DTU) collaboration strategy to realize efficient and resilient urban emergency response. Specifically, a DTU collaboration strategy is first proposed based on the characteristics of urban emergency management. Then, an integrated truck-UAV collaborative scheduling model is developed and indicated to be an NP-hard problem. Then, a tabu search-based integrated (TSI) scheduling algorithm is developed, with several tailored local search operators and a recursion-based evaluation (RE) algorithm. Finally, comprehensive experiments are conducted to demonstrate the superiority of our proposed strategy over existing truck-UAV collaboration strategies and verify the performance of our proposed solution algorithm.**

***Managerial Relevance Statement*—This article proposes a dynamic truck-UAV (DTU) collaboration strategy and a corresponding route planning method for urban emergency response. The findings of this work demonstrate that emergency management agencies can utilize the DTU collaboration strategy for efficient emergency response in urban areas with high-density road networks. Meanwhile, the emergency response under this strategy is resilient to cope with different degrees of road network disruptions. In addition, the DTU collaboration strategy shows a distinct improvement over the parallel and truck-UAV flying sidekick strategies in large-scale emergency response scenarios, and it will perform better when demand density increases. Furthermore, the proposed integrated scheduling algorithm is demonstrated to be suitable and efficient for solving the routing problems associated with the DTU collaboration strategy.**

Manuscript received 31 July 2022; revised 20 January 2023, 7 June 2023, and 12 July 2023; accepted 20 July 2023. Date of publication 24 August 2023; date of current version 4 June 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 72174042, in part by the Natural Science Foundation of Guangdong Province under Grant 2023A1515011402, in part by RGC Research Impact Fund under Grant R7036-22, and in part by RGC Theme-based Research Scheme under Grant T32-707/22-N. Review of this manuscript was arranged by Department Editor A. O. Solis. *(Corresponding author: Gangyan Xu.)*

Yuying Long and Gangyan Xu are with the Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: yuying_long519@163.com; gagexgy@gmail.com).

Jinqiu Zhao and Binglei Xie are with the School of Architecture, Harbin Institute of Technology (Shenzhen), Shenzhen 150001, China (e-mail: jinqiuzhao@126.com; xiebinglei@126.com).

Meng Fang is with the Department of Computer Science, University of Liverpool, L69 3BX Liverpool, U.K. (e-mail: meng.fang@liverpool.ac.uk).

Digital Object Identifier 10.1109/TEM.2023.3299693

***Index Terms*—Disaster management, emergency responses, resilience, truck–UAV collaboration, unmanned aerial vehicle (UAV).**

Notation

Notation	Description
Sets	
K	Set of all trucks.
U	Set of all UAVs.
C	Set of all rescue demands.
Parameters	
$d_{i,j}$	Travel time of trucks from node i to j.
$\hat{d}_{i,j}$	Travel time of UAVs from node i to j.
T_i	Latest service time of the ith demand.
t^i_{serve}	Service time of the ith demand.
q_i	Number of UAVs at the ith demand.
z	Makespan of the whole process in scheduling.
x^k_i	Binary parameter indicating whether the ith demand is served by truck k.
x^u_i	Binary parameter indicating whether the ith demand is served by UAV u.
M	Arbitrarily large positive number.
L	Flight range of UAVs.
N_3	Number of UAVs that each truck carries.
α	Penalized large coefficient of service delay.
c	Number of all rescue demands.
Decision Variables	
$y^k_{i,j}$	Binary variable indicating whether truck k moves from node i to node j.
$y^u_{i,j}$	Binary variable indicating whether UAV u flies from node i to node j.
t_i	Time that the truck or UAV arrives at the ith demand.
l_i	Time that the truck or UAV departures from the ith demand.

I. Introduction

EFFECTIVE emergency response is crucial for saving lives and protecting property in urban areas, which are densely populated and consist of many vital infrastructure components [1], [2]. With the rapid urbanization and development of urban agglomerations, governments around the world have recognized the significance of urban emergency response and are making extensive efforts to improve its performance through various initiatives, policies, and technologies. The focus is on regulating the preparation of emergency resources and action plans after disasters. However, transportation networks and infrastructures are always vulnerable to destruction during disasters, such as road blockage by floods or collapsed buildings [3], which can hinder the implementation of general response activities using ground vehicles [4]. For example, in the 2021 Henan flood, many roads were damaged, preventing ground vehicles from delivering relief goods to victims. Similarly, the 2022 Guangdong flood destroyed many urban roads, severely decreasing the efficiency of emergency response and threatening the safety of rescue teams during transportation. Therefore, there is an urgent need to develop efficient and resilient urban emergency response systems that can effectively cope with disruptions in transportation networks and infrastructures.

The recent increase of unmanned aerial vehicles (UAVs), which have been successfully adopted in various fields, such as urban logistics [5], traffic surveillance [6], infrastructure monitoring [7], risk management [8], and healthcare systems [9], provides an opportunity for more flexible emergency response solutions. Specifically, due to their intrinsic flexibility and safety, UAVs have been piloted in disaster responses, including monitoring the disaster scenario of the 2008 Wenchuan earthquake, restoring communications in the 2021 Henan flood, and providing medicine delivery services during the 2022 COVID-19 outbreak in Shanghai. Nevertheless, due to their limited endurance mileage and loading capacity, UAVs may not be efficient and effective at working independently. Therefore, many efforts have been made to integrate ground vehicles with UAVs to take advantage of both [10], [11], and various routing and scheduling methods have been proposed for different collaboration modes [12], [13], [14].

However, previous studies have mainly focused on fixed binding modes between trucks and UAVs or small-scale scenarios, which may not be suitable for urban emergency response scenarios. First, due to the dense population and various vital infrastructures in urban areas, the emergency response system usually involves a large number of rescue demands after disasters, making the scale of problems relatively large and complex. Second, along with the development of disasters, emergency scenarios can hardly be predicted and are always highly dynamic, requiring efficient scheduling of trucks and UAVs after a disaster occurs. Third, the high density of infrastructures in urban areas makes it challenging to construct sufficiently large temporary depots for trucks and UAVs, further complicating the problem.

Taking into account the above characteristics and challenges of urban emergency response, this article develops a dynamic truck–UAV (DTU) collaboration strategy and a scheduling method to improve the response efficiency and resilience under transportation network disruptions. Specifically, this work considers the scenario of delivering homogeneous small-sized relief materials (e.g., relief kits, medicines, instant foods and water, etc.), which can be easily carried by UAVs [9], [15], [16]. Meanwhile, leveraging the technological advantages of UAVs regarding dynamic takeoff and landing on different trucks or even moving trucks [17], this work proposes a loosely coupled truck and UAV collaborative response strategy and develops an effective solution algorithm for it. The contributions of this work lie in the following four aspects.

1) A novel DTU collaboration strategy is proposed for urban emergency response in which UAVs can takeoff and land on different trucks.
2) A mixed integer linear programming (MILP) model is developed for the integrated scheduling of trucks and UAVs with the objective of minimizing both the response time and delay of service.
3) An integrated truck–UAV collaborative scheduling approach is developed based on the tabu search (TS), which can efficiently generate high-quality scheduling solutions for the DTU collaboration strategy.
4) Extensive experiments are carried out with different emergency scenarios and road network conditions. The results show that the DTU collaboration strategy is more efficient and resilient than the existing strategies, and the proposed scheduling algorithm outperforms many other methods.

The rest of this article is organized as follows. A detailed literature review is presented in Section II. The DTU collaboration strategy for urban emergency response is discussed in Section III. Subsequently, the model of integrated truck–UAV scheduling and the tabu search-based integrated (TSI) scheduling algorithm are proposed in Section IV. Experimental case studies that verify the effectiveness of the proposed method are conducted in Section V. Finally, Section VI concludes this article.

II. Literature Review

In this section, the relevant literature will be reviewed from two streams: urban emergency response and truck–UAV collaboration.

A. Urban Emergency Response

With rapid urbanization and climate change, there is a growing emphasis on enhancing the performance of urban emergency response to ensure the safe development of urban areas. On the one hand, many operation models and optimization methods have been proposed for emergency response [18], [19], with various exact solution algorithms [20] and heuristic-based methods [21], [22] developed. On the other hand, various technologies have been introduced for efficient urban emergency response, including the Internet of Things [2], [23], wireless communications [24], [25], [26], and autonomous systems [22], [27]. In particular, the applications of UAVs in urban emergency response have attracted much attention, such as UAV applications in humanitarian logistics [28], [29], [30], disaster risk monitoring [31], [32], and searching and rescuing [33]. Specifically, Liu et al. [25] designed a multiantenna transceiver mechanism and multihop device-to-device communication to expand the communication range of UAVs in disasters. Zhao et al. [26] proposed a UAV-assisted framework for disaster response to facilitate communication in disaster areas [26]. To plan UAV routes for postdisaster assessment, Oruc and Kara [34] introduced a novel biobjective routing with a profit problem

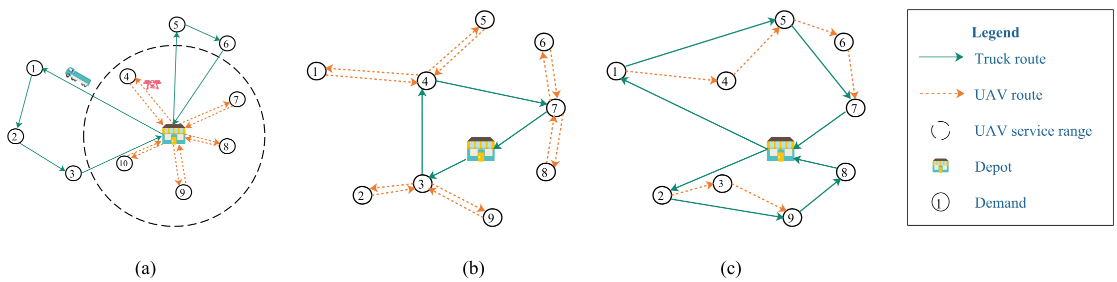


Fig. 1. Truck–UAV collaboration strategies. (a) PTU. (b) TSU. (c) TUFS.

considering arc and node profits and proposed a heuristic method to make the optimal solution stay close to the Pareto optimal. Nedjati et al. [35] proposed a new relief distribution system for emergency response after an earthquake and showed that the system has an efficient capability for urban areas with high population density. In addition, Huang et al. [36] proposed a novel UAV path planning framework for emergency data collection and transmission.

Previous studies have accumulated extensive knowledge on improving the performance of urban emergency response. However, strategies for resilient emergency response remain open for discussion. In addition, although many pilot studies have been performed using UAVs for urban emergency response, there is still a lack of research on efficient and flexible collaborative strategies with other transport modes after disasters.

B. Truck–UAV Collaboration

To overcome the shortcomings of UAVs in endurance mileage and loading capacity, there is a trend to integrate UAVs with trucks for complex tasks. Three typical truck–UAV collaboration strategies that are widely adopted in practice and discussed in the literature are illustrated in Fig. 1.

The first is the *parallel truck–UAV (PTU)* scheduling strategy, where trucks and UAVs work independently for a set of tasks. In the PTU strategy, UAVs are dispatched to the demands within their range and leave the others to trucks. Murray and Chu [37] modeled the PTU strategy as a parallel drone scheduling traveling salesman problem (PDSTSP) and built a mixed integer programming model for it. As an improvement of the basic PDSTSP, Ham [38] developed a constraint programming model that considered time windows, pickup and delivery processes, and multivisit issues, which has been applied to various emergency scenarios. The PTU strategy is relatively easy to implement and has been applied to various urban emergency response scenarios.

The second is the *truck-supported UAV (TSU)* scheduling strategy, where demands are served by UAVs, while trucks serve as mobile stations for UAVs. In the TSU strategy, a truck departs from the depot with UAVs on board. When the truck arrives at a site, UAVs depart from it to serve demands and return to the truck after completing tasks. The truck waits at the site until all UAVs fly back and then head to the next site. Wu et al. [12] depicted the TSU strategy as a carried vehicle supporting drone delivery problem and developed a cluster-based reinforcement learning framework for this problem. For the selection of truck parking sites, Salama and Srinivas [39] proposed two selection policies and a promoted heuristic algorithm based on unsupervised machine learning. In addition, Mercedes-Benz introduced Vision Van equipped with two UAVs, which can support the TSU strategy for parcel delivery [40].

The third is the *truck–UAV flying sidekick (TUFS)* scheduling strategy, where both trucks and UAVs respond cooperatively to demands. In the TUFS strategy, trucks respond to demands while serving as mobile stations for UAVs. In addition, UAVs can return to the truck at a different point, which enables trucks and UAVs to serve demands in parallel. Murray and Chu [37] defined the TUFS strategy as the flying sidekick traveling salesman problem and developed an efficient heuristic algorithm to solve it. Yurek and Ozmutlu [41] defined the TUFS strategy as a traveling salesman problem with drones (TSP-D) and developed a two-stage iterative solution algorithm based on heuristics. Furthermore, an algorithm combined with a TSP solver and a general variable neighborhood search (VNS) was employed to solve the scheduling problem based on TSP-D [42].

Previous studies have significantly enriched the methods and applications of truck–UAV collaboration. However, considering the characteristics of urban emergency response discussed in Section I, they may not be well-suited for emergency response scenarios. Meanwhile, dynamic bindings between trucks and UAVs are rarely discussed, leaving a great gap in further improving collaboration efficiency and resilience.

III. DTU Collaboration Scheduling Strategy

Among the three truck–UAV collaborative strategies discussed in Section II-B, the TUFS strategy is the most flexible. Nevertheless, with the advancement of UAV technologies and considering the characteristics of urban emergency response, the TUFS strategy can be further improved. First, the high density and high connectivity of urban road networks allow a more even distribution of trucks than in rural areas, which provides an ideal distribution of UAV stations if every truck serves as a mobile UAV station. Meanwhile, the mobile UAV stations (trucks) would in turn further counteract the shortcomings of UAVs in endurance mileage and loading capacity, as UAVs could hop among these trucks for more complex tasks. Second, the high density of rescue demands leads to more trucks within the affected areas, which provides more possibilities for UAVs to land on different trucks. Third, the development of UAV technologies and the standardization of interfaces between trucks and UAVs makes it possible for UAVs to land, recharge (or swap

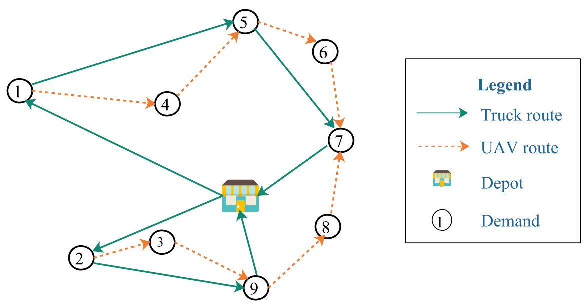


Fig. 2. DTU collaboration strategy.

battery), and reload on different trucks, which further improves flexibility.

With the above considerations, the *DTU* collaboration strategy is proposed based on the TUFS strategy for urban emergency response, as illustrated in Fig. 2. Different from the TUFS strategy, this strategy adopts dynamic binding between UAVs and trucks in which UAVs can land on any truck with available spaces nearby after completing one emergency task. Specifically, in the context of an emergency, a fleet of trucks departs from the depot, each with relief goods and several UAVs on board. These trucks follow the preassigned routes to serve the demands and act as mobile stations for their carrying UAVs. Once the truck completes the service of demand at one demand point, it moves to the next demand point instead of waiting for UAVs to takeoff or land. At each demand point, UAVs can depart from the truck to serve the assigned demand, dynamically determine which truck to land on, and then head to the chosen truck within its endurance for recharging and/or reloading.

Compared with the PTU, TSU, and TUFS strategies, the DTU collaboration strategy has a much higher resource utilization rate, response efficiency, and flexibility to cope with disruptions in the context of urban emergency response. Similar to the PTU and TUFS strategies, trucks and UAVs in the DTU collaboration strategy can serve demands simultaneously, making good use of the capability of trucks and UAVs. Meanwhile, because UAVs can takeoff and land on different trucks, it reduces the truck waiting time in the PTU and TUFS strategies, which further improves the utilization rates of both trucks and UAVs. Meanwhile, because the TUFS strategy can be treated as a special case of the DTU collaboration strategy with additional constraints on the fixed binding of trucks and UAVs, its solution space D_{TUFS} is a subset of the DTU collaboration strategy's solution space D_{DTU}, that is, $D_{\text{TUFS}} \subseteq D_{\text{DTU}}$. Therefore, the DTU collaboration strategy may have more efficient solutions and more flexibility to cope with disruptions.

As an example, we consider a simple case of urban emergency response with one depot and six demands, as shown in Fig. 3(a). The number on each edge indicates its distance. Assuming that there are two trucks with speed $v_t = 1$ and two UAVs with speed $v_u = 2$, the optimal scheduling plans for the trucks and UAVs in different collaboration strategies are illustrated in Fig. 3(b)–(d). The demand delays for the PTU, TUFS, and DTU collaboration strategies are 25.9, 26.7, and 25.7, respectively, which indicates that the DTU collaboration strategy has the highest efficiency.

In addition, some disruptions occur in road transportation networks, e.g., the road between the depot and node 5 and the road between node 5 and node 6 are broken (see Fig. 4), which makes node 5 isolated and unable to be accessed by trucks. Thus, the efficiency of the PTU and TUFS strategies will be decreased as their previous optimal solutions become infeasible. However, the optimal solution of the DTU collaboration strategy still works, revealing that it is more resilient to disruptions. More comprehensive demonstrations and verification will be further discussed in Section V.

IV. Integrated Truck–UAV Scheduling

In this section, the integrated truck–UAV scheduling problem under the DTU collaboration strategy will be modeled first, and then an efficient TS-based scheduling algorithm will be developed.

A. Problem Modeling

Consider an urban emergency response scenario with a set of rescue demands C, a set of trucks K, a set of UAVs U, and an integrated depot for both trucks and UAVs. The emergency response scenario can be represented as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N}$ refers to the nodes that consist of the rescue demands (node 1 to node $|C|$) and depot (node 0). The set of arcs connecting each node is defined as $\mathcal{A} := \{(i,j)|i,j \in \mathcal{N}, i \neq j\}$. The distances traveled by the trucks and UAVs in an arc $(i,j) \in \mathcal{A}$ are represented as $d_{i,j}$ and $\hat{d}_{i,j}$, respectively. Each truck $k \in K$ departs from the depot with n_k UAVs onboard. UAVs can depart from any node $i \in \mathcal{N}$ from a truck and return to another truck at node $j \in \mathcal{N}$. In addition, without the loss of generality, this work only considers one type of truck and one type of UAV. Meanwhile, trucks can serve several rescue demands in one trip, while UAVs can only serve one rescue demand at a time.

Considering that the top priority is to obtain an efficient response to all rescue demands, the objective of the integrated truck and UAV scheduling consists of two parts. The first is to minimize the delay of service for each rescue demand, which can be denoted as $(t_i - T_i)^+$. Here, T_i refers to the required arrival time of trucks or UAVs for demand i, while t_i refers to the actual arrival time. In practice, T_i can be easily calculated based on the service quality guarantee of the response time for different types of rescue demands. In addition, adopting the delay of service rather than the average response time as the objective ensures the fairness of emergency response, which is an important consideration in public services. The second objective is the total time needed to complete all rescue demands, which is denoted as z in this work. This is an important complement to the objective, as it can drive the scheduling process to finish the entire response process as soon as possible. Otherwise, the scheduling will become inefficient in small cases where delay can hardly happen. In addition, α is introduced to adjust the weight of the service delay so that the model can adapt to different scenarios.

According to the above analysis, truck and UAV scheduling under the DTU collaboration strategy is modeled as an MILP. The integrated model is given as follows:

$$\min\ \alpha \sum_{i \in C} (t_i - T_i)^+ + z \tag{1}$$

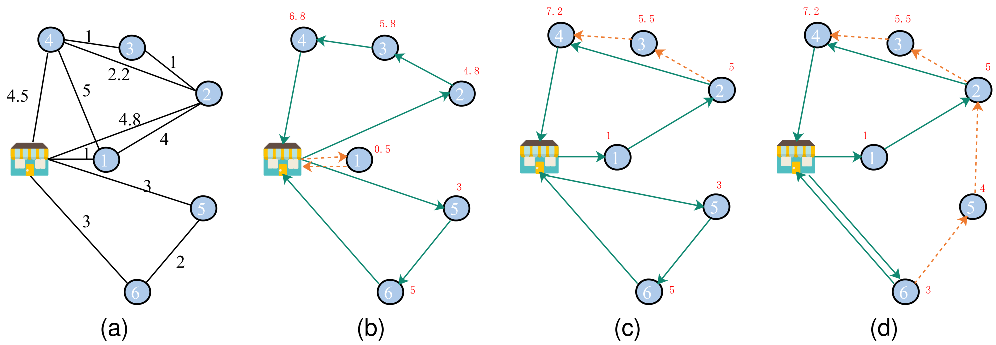


Fig. 3. Efficiency comparison of different collaboration strategies. (a) Simple case. (b) Solution for the PTU strategy. (c) Solution for the TUFS strategy. (d) Solution for the DTU strategy.

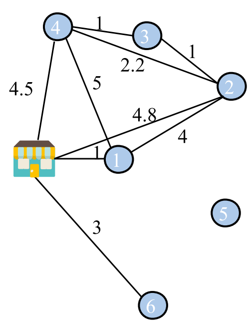


Fig. 4. Scenario with road network damage.

s.t.

$$\sum_{k\in K}\sum_{i\in C} y_{0,i}^{k} = |K|, \tag{2}$$

$$\sum_{i\in\mathcal{N}, i\neq j} y_{i,j}^{k} = \sum_{s\in\mathcal{N}, s\neq j} y_{j,s}^{k} \quad \forall j\in\mathcal{N}, \forall k\in K, \tag{3}$$

$$\sum_{u\in U}\sum_{i\in\mathcal{N}, j\neq i} y_{i,j}^{u} \leq 1 \quad \forall j\in C, \tag{4}$$

$$\sum_{u\in U}\sum_{j\in\mathcal{N}, i\neq j} y_{i,j}^{u} \leq 1 \quad \forall i\in C, \tag{5}$$

$$x_i^u + x_j^u - y_{i,j}^u = 0 \quad \forall i,j\in C, j\neq i, \tag{6}$$

$$z \geq t_i \quad \forall i\in C, \tag{7}$$

$$\sum_{k\in K} x_i^k + \sum_{u\in U} x_i^u = 1 \quad \forall i\in C, \tag{8}$$

$$\sum_{k\in K} y_{i,j}^k + \sum_{u\in U} y_{i,j}^u \leq 1 \quad \forall i,j\in C, j\neq i, \tag{9}$$

$$l_i + d_{i,j} - \left(1 - \sum_{k\in K} y_{i,j}^k\right) M \leq t_j \quad \forall i\in\mathcal{N},\ \forall j\in C, j\neq i, \tag{10}$$

$$t_i + \hat{d}_{i,j} - \left(2 - \sum_{u\in U} y_{i,j}^u - \sum_{u\in U} x_j^u\right) M \leq t_j \quad \forall i,j\in C, j\neq i, \tag{11}$$

$$t_i + t_{\text{serve}}^i \leq l_i \quad \forall i\in C, \tag{12}$$

$$l_j + \hat{d}_{j,i} - \left(2 - \sum_{u\in U} y_{j,i}^u - \sum_{u\in U} x_j^u\right) M \leq l_i \quad \forall i,j\in C, j\neq i, \tag{13}$$

$$\sum_{k\in K} x_i^k \cdot M \geq q_i \quad \forall i\in C, \tag{14}$$

$$q_i + \sum_{u\in U}\sum_{s\in C, s\neq i} \left(y_{s,i}^u - y_{i,s}^u\right) \leq N_3 \quad \forall i\in C, \tag{15}$$

$$q_i + \sum_{u\in U}\sum_{s\in C, s\neq i} \left(y_{s,i}^u - y_{i,s}^u\right) - \left(1 - \sum_{k\in K} y_{i,j}^k\right) M \leq q_j \quad \forall i\in C, \forall j\in C, j\neq i, \tag{16}$$

$$q_i + \sum_{u\in U}\sum_{s\in C, s\neq i} \left(y_{s,i}^u - y_{i,s}^u\right) + \left(1 - \sum_{k\in K} y_{i,j}^k\right) M \geq q_j \quad \forall i\in C, \forall j\in C, j\neq i, \tag{17}$$

$$\sum_{s\in C, s\neq i} \left(\hat{d}_{s,i}\cdot y_{s,i}^u\right) + \sum_{j\in C, j\neq i} \left(\hat{d}_{i,j}\cdot y_{i,j}^u\right) \leq L \quad \forall i\in C, \forall u\in U. \tag{18}$$

Constraints (2) and (3) ensure the balance of truck flows, while constraints (4) and (5) ensure the balance of UAV flows. Specifically, constraint (2) denotes that the number of trucks leaving the depot should not exceed the total number of trucks. Constraint (3) indicates that, for every node, the number of trucks departing from it should equal the number of trucks arriving at it. Constraints (4)–(6) guarantee that, for every rescue demand node, a UAV can only visit once. Constraint (7) determines the total time needed to finish serving all rescue demands. Constraints (8) and (9) ensure that each demand can be visited only once, either by a truck or UAV. In addition, constraints (10) and (11) give the relationship between travel time and arrival time, while constraint (12) ensures that each truck/UAV can only depart from a demand after service. Constraint (13) means that, at every node, the departure time of a UAV cannot be earlier than its arrival time. Constraints (14) and (15) indicate the restriction of the number of UAVs carried by each truck. Here, constraint (14) indicates that the number of UAVs on each truck at each point is greater than zero only when the truck serves the point. Constraint (15) ensures that the number of UAVs each truck carries does not

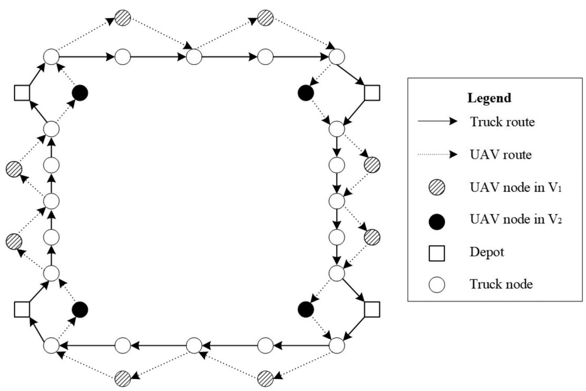


Fig. 5. Example of the reduction: $m = 4$ and $B = 6$.

exceed its capacity of carrying UAVs. Constraints (16) and (17) provide formulations for the calculation of the number of UAVs per truck. Finally, constraint (18) shows the restriction of the mileage endurance of each UAV.

B. Complexity Analysis

As shown in Section IV-A, the truck–UAV collaboration scheduling problem is modeled as an MILP. To find a suitable solution method for this problem, its complexity is analyzed. Specifically, the NP-hardness of the DTU collaboration scheduling problem is analyzed through a pseudopolynomial-time reduction from the *three-partition problem* (refer to [43] for the specific definition).

Suppose the speed ratio of trucks and UAVs is a unit. A pseudopolynomial-time reduction from a three-partition problem to the DTU collaboration scheduling problem is given. Based on the definition of the three-partition problem, we construct an instance of the DTU collaboration scheduling problem (shown in Fig. 5). In this instance, there are m truck routes, each with B nodes (including $B-1$ truck nodes and a depot), $B-1$ edges with a unit length, and $3m$ UAV nodes. In addition, a dummy edge with a unit length is given to connect two disjoint truck routes by connecting the depot and the tail of the latter route. The UAV nodes are classified into two sets. The first set V_1 ($|V_1| = 2m$) is a collection of UAV serving nodes for UAV departures and landings on the same truck, while the second set V_2 ($|V_2| = m$) is a collection of UAV serving nodes for UAVs departing and landing on different trucks. For nodes i in V_1, $i = 1, 2, \ldots, 2m$, node v_i is connected with two nodes in the $\lceil \frac{i}{2} \rceil$th truck route via two edges of length $\frac{x_i}{2}$. For node j in V_2, node v_j is connected with both the head node of route j and route $j+1$ via two edges of length $\frac{x_j}{2}$. Based on the constructed instance, we have $\sum_{i=1}^{3m} x_i = mB$ and $x_i + x_{i+1} + x_{2m+i} = B$ for $i = 1, 3, \ldots, 2m-1$ when there is no truck waiting or UAV waiting, which corresponds to the three-partition problem. Furthermore, the time complexity of this construction is $O(mB)$, which is a pseudopolynomial time. Because the three-partition problem is NP-complete in a strong sense [44], the DTU collaboration scheduling problem is strongly NP-hard.

Algorithm 1: The TSI Scheduling Algorithm.

Input: Information on the depot, trucks, UAVs, and demands
Output: The DTU collaboration scheduling plan σ_{best}

```
Initialize neighborhood Ω, the set of neighborhood operators N, tabu list H, evaluation function F(σ)
σ ← Algorithm 1(N) ▷Initialize solution. Details are shown in Algorithm 2
σ_best = σ
for i = 1, 2, ..., n do
    Ω ← N(σ, H) ▷Generate the neighbors of solution σ. Details are shown in Section IV-C3.
    σ ← SelectBestSolution(Ω)
    Update the tabu list H
    if F(σ) < F(σ_best) then
        σ_best ← σ
    end if
end for
```

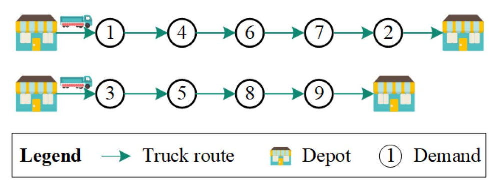


Fig. 6. Example of initial truck routes.

C. TS-Based Solution Algorithm

Due to the complexity of the integrated truck–UAV scheduling problem under the DTU collaboration strategy, the problem can hardly be solved by an exact algorithm considering the scale of problems in urban emergency response and the tight requirement on computing time. A TS-based solution algorithm is proposed because TS is widely adopted in various routing problems and has been demonstrated to be effective and efficient in many scenarios; and the operators of TS fit the properties of the problem well, which makes it easy to design new problem-specific operators [45], [46], [47].

1) Overall Framework: In general, the overall framework of the proposed TSI scheduling algorithm is shown in Algorithm 1. It starts with an initial solution (see Section IV-C2) that is considered the best solution in the original state. An iterative process is then followed to search for a better solution. At each iteration, several local search operators that are tailored for this problem (see Section IV-C3) are proposed to generate a neighbor solution. Subsequently, current solutions evolved from the local search are evaluated to explore better solutions, and the solution evaluation method is presented in Section IV-C4.

2) Initial Solution: A solution consists of a set of truck and UAV routes. In this article, the initial solution σ is obtained by the greedy algorithm, as illustrated in Algorithm 2. Specifically, a solution that serves all demands by trucks is randomly generated first. For example, in Fig. 6, all nine demands are served by two trucks, and two truck routes are randomly generated, namely,

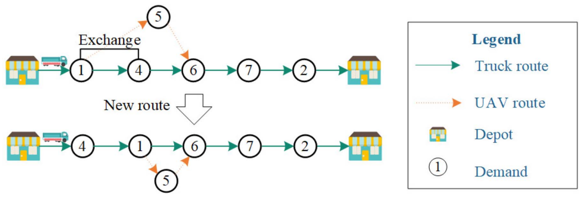


Fig. 7. Exchange of truck nodes within one truck route (EX-T1).

Algorithm 2: Greedy Algorithm.

Input: Information on the depot, trucks, UAVs, and demands

Output: The initial solution of the DTU collaboration scheduling σ

Randomly generate all truck routes serving all demands.
for $i = 1, 2, \ldots, n$ **do**
  if the ith matches the *point search* principle **then**
    if length$(i, j, s) \leq L$ **then**
      add UAV route (i, j, s)
    end if
  end if
end for

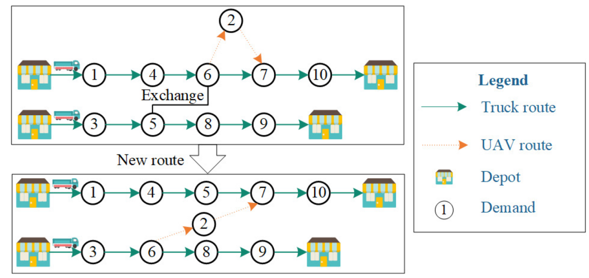
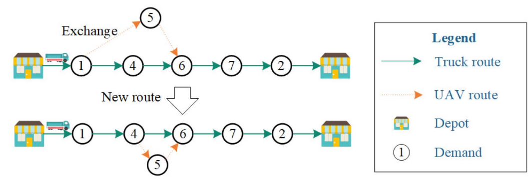
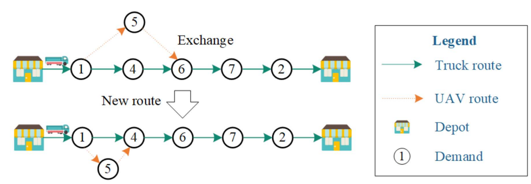


Fig. 8. Exchange of truck nodes between two truck routes (EX-T2).



(a)



(b)

Fig. 9. Exchange operators on UAV nodes. (a) Exchange of UAVs Origin node (EX-UO). (b) Exchange of UAVs Destination node (EX-UD).

[0, 1, 4, 6, 7, 2, 10] and [0, 3, 5, 8, 9, 10]. Then, UAV routes related to each truck are generated by *point search*. *Point search* is a principle for searching the demand node j whose adjacent nodes i and s are demand points served by trucks while no UAV is landing or takeoff at that node. For each demand node that conforms to the *point search* principle, an evaluation of whether the length of the UAV route (i, j, s) exceeds the endurance mileage of the UAV will be conducted. If not, the jth demand would be served by the UAV, which will takeoff at node i and land at node s. If yes, the UAV route will be treated as illegal, and the search should continue to the next node.

3) Local Search Operators: To improve the efficiency of the algorithm, three operators are carefully designed in this part.

a) Exchange of truck nodes: Because there is more than one truck route involved in the integrated truck–UAV scheduling problem, the exchange of truck nodes can occur within one truck route (EX-T1) or between two truck routes (EX-T2).

EX-T1 is an operator that exchanges two demand nodes, which are on the same truck route while not serving as the origin and destination of the same UAV route. Specifically, two nodes on a truck route are selected first. Then, the two nodes are swapped on the truck route, and a new truck route will be generated accordingly. Here, the related UAV routes (e.g., the UAVs origin or destination is the exchange node) are unchanged, as illustrated in Fig. 7.

EX-T2 is an operator that exchanges two demand nodes that are on different truck routes while not serving as the origin and destination of the same UAV route. Specifically, two nodes are selected from two different truck routes and swapped accordingly. Then, two new truck routes are formed. Similarly, these procedures will not affect the related UAV routes. Details are shown in Fig. 8.

b) Exchange of UAVs origin and destination nodes: Because each UAV route contains its origin node, demand node, and destination node, exchanges of origin and destination nodes of one UAV are introduced in two cases: Exchange of UAVs Origin node (EX-UO) and Exchange of UAVs Destination node (EX-UD).

EX-UO means the exchange of the origin node of a UAV route. In this operator, a UAV route (i, j, s) is chosen first. In addition, to ensure that the new route is feasible, the set of new feasible origins O is generated. Meanwhile, all demands in O satisfy that its occurrence time is earlier than the occurrence time of the jth demand. Then, node i is randomly swapped with node $i' \in O$ to form a new UAV route. Details are shown in Fig. 9(a).

EX-UD is an operator that exchanges the destination of the UAV route. Similar to *EX-UO*, a destination node s of a UAV route is selected first, and the set of feasible destination nodes is generated where the occurrence time of the demand is later than the occurrence time of the sth demand. Then, swapping operations are made between nodes s and s', where $s' \in D$. Details are shown in Fig. 9(b).

c) Mutation: There are two possibilities for this operator. One is that a truck-served demand j in the current solution can be converted to UAV served. In this case, there is a newly generated UAV route (i, j, s), which is a fragment of the truck route containing the jth demand. The other is that a UAV-served demand can be converted to truck served. In this case, the former

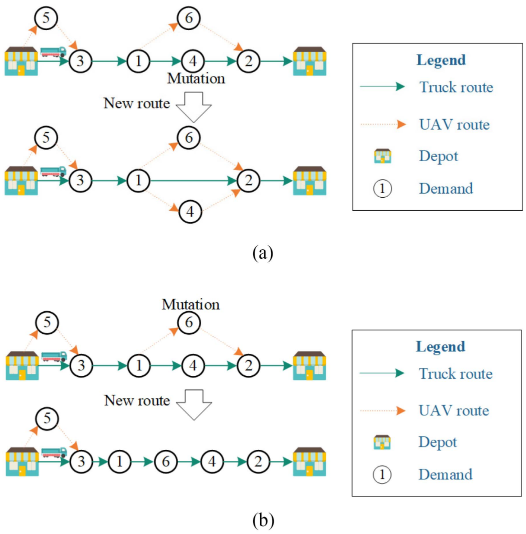


Fig. 10. Mutation. (a) Mutation of truck nodes. (b) Mutation of UAV nodes.

UAV route with the jth demand will be deleted, while the jth demand will be added to a truck route. Details are shown in Fig. 10(a) and (b), respectively.

4) Solution Evaluation: Due to the complexity of the problem, an recursion-based evaluation (RE) algorithm is designed to compute the arrival and departure times of trucks and UAVs. Integrating the ideas of dynamic programming and graph theory, the evaluation of each solution begins with the calculation of the arrival time of the depot after finishing all the demands and repeats the backtracking to calculate the arrival time of the previous node. If the previous arc is between two demands served by trucks, the arrival time of the calculated node is related to the departure time of its previous node. Otherwise, the arrival time is related to the arrival time of the previous node. The flow of the algorithm is shown in Algorithm 3.

Algorithm 3: The RE Algorithm.

Input: Compute node i, current solution σ, the set of predecessor nodes for compute node P

Output: The set of arrival times and departure times of the current solution

Assume X is the set of the candidates of l_i
if $i \in C_U$ **then**
  if t_{i-1} is unknown **then**
    RE($i-1, \sigma$)
  end if
  $t_i \leftarrow t_{i-1} + \hat{d}_{i,i-1}/\hat{v}$
  $l_i \leftarrow t_i + t^i_{\text{serve}}$
end if
if $i \in C_K$ **then**
  for $j \in P$ **do**
    if l_j is unknown **then**
      RE(j, σ)
    end if
    if $j \in C_U$ **then**
      $l'_i \leftarrow l_j + \hat{d}_{j,i}/\hat{v}$
      X.append(l'_i)
    end if
    if $j \in C_K$ **then**
      $t_i \leftarrow l_j + d_{j,i}/v$
      $l_i \leftarrow t_i + t^i_{\text{serve}}$
    end if
    $l_i \leftarrow \max(X)$
  end for
end if

V. Experimental Case Study

In this section, experimental case studies will be conducted to validate the effectiveness and resilience of the DTU collaboration strategy and the TSI scheduling algorithm.

A. Experimental Settings

To systematically verify the performance of the DTU collaboration strategy and the TSI scheduling algorithm, five types of datasets [48] are used: 15 demands with two trucks, 25 demands with three trucks, 50 demands with five trucks, 75 demands with eight trucks, and 100 demands with ten trucks. The demand increment in different datasets shows the increase in scales of the scenario settings, with not only a higher frequency of demands but also more demand points. In the experiment, each truck carries two UAVs. The speed of truck is set as $v_t = 8.5$ m/s, while the speed of UAV is $v_u = \beta v_t$. In consistent with the previous research [14], [49], [50], [51], [52], β is set as 1.0, 1.5, and 2.0, respectively, in the experiments. Besides, $\beta = 1.76$ is included based on the technical parameters of SF Express UAV and adopted in Section V-B1, B3, and C. In addition, the endurance mileage of the UAVs is set to 10 km. Each type of dataset contains ten instances; hence, the average performance under each type of dataset is compared. Some data examples are shown in Table I. Furthermore, all experiments stop when the number of iterations reaches 1000 or the results stay unchanged continuously for 200 iterations. The simulation environment is developed using Python, and the experiments are conducted on a workstation with a 2.10 GHz CPU and 16 GB RAM.

TABLE I
DATA EXAMPLES

Instance No.	Instance name	Start time	Service duration
0	CF141SW	540	0
1	CF144QJ	720	7
2	CF142XJ	660	7
3	CF146LW	540	6
4	CF144SJ	600	9
5	CF142LS	540	8
6	CF140SE	600	5
7	CF144XY	540	8
8	CF148NR	600	7
9	CF146SD	660	9

TABLE II
EFFICIENCY COMPARISONS OF DIFFERENT TRUCK–UAV STRATEGIES AT DIFFERENT INSTANCE SCALES

Dataset	Strategies	Avg_Obj (10^7)	Idling_time (min)	Total delay (min)	Improve_1	Improve_2
Cardiff_15	PTU	1.63	-	27.11	0.00%	-42.09%
	TUFS	1.14	26.59	19.08	29.62%	0.00%
	DTU	**1.13**	**27.34**	**18.78**	**30.73%**	**1.57%**
Cardiff_25	PTU	8.37	-	139.46	0.00%	4.89%
	TUFS	8.80	31.50	146.63	-5.14%	0.00%
	DTU	**7.69**	**31.52**	**128.18**	**8.09%**	**12.58%**
Cardiff_50	PTU	8.19	-	136.52	0.00%	-0.06%
	TUFS	8.19	72.48	136.44	0.06%	0.00%
	DTU	**6.11**	**69.96**	**101.74**	**25.48%**	**25.43%**
Cardiff_75	PTU	15.97	-	266.15	0.00%	6.73%
	TUFS	17.12	105.16	285.35	-7.21%	0.00%
	DTU	**10.45**	**104.31**	**174.23**	**34.54%**	**38.94%**
Cardiff_100	PTU	14.84	-	247.30	0.00%	0.00%
	TUFS	17.83	139.72	297.10	-20.14%	0.00%
	DTU	**8.41**	**135.94**	**140.11**	**43.35%**	**52.84%**

The bold parts show the response efficiency of the DTU collaboration strategy at different instance scales.

B. Performance of the DTU Collaboration Strategy

The performance of the DTU collaboration strategy is evaluated from two aspects: efficiency and resilience.

1) Efficiency Analysis at Different Instance Scales: To evaluate the efficiency of the DTU collaboration strategy, five scenarios are adopted with different numbers of demands, representing different scales of problems. In addition, ten experiments are conducted for each scenario, and the average objective, the idling time of trucks, and the total demand delay are compared. Table II presents the results of the above experiments. Improve_1 and Improve_2 indicate the improvement in the demand delay using the PTU and TUFS strategies as baselines, respectively.

From Table II, the average objective Avg_Obj and the demand delay of the DTU collaboration strategy are lower than those of the PTU and TUFS strategies for all the scales of problems, with demand numbers from 15 to 100. For example, the improvement of the DTU collaboration strategy compared with the PTU collaboration strategy ranges from 8.09% to 43.35%, and the improvement compared with the TUFS collaboration strategy ranges from 1.57% to 52.84%. Meanwhile, the advantage in efficiency becomes more obvious with the increase in demand number, with over 40% improvement in the total delay for dataset Cardiff_100. This means that more demands provide a higher probability of using UAVs, thus improving the efficiency of the DTU collaboration strategy. In terms of the idling time of trucks, the TUFS strategy and DTU collaboration strategy are compared (as there is no idling time of trucks in the PTU strategy). The results show that there is no obvious difference in the idling time of trucks between the two strategies, while the system efficiency of the DTU collaboration strategy improves. The reason is that the DTU collaboration strategy provides a higher probability for the use of UAVs than the TUFS strategy. Besides, the DTU collaboration strategy can further reduce the idling time of trucks when the instance scale is large. In general, the experiments indicate that the DTU collaboration strategy performs better than both the PTU and TUFS strategies in efficiency. Moreover, the DTU collaboration strategy shows more obvious superiority in large-scale instances.

2) Efficiency Analysis With Different Speed Ratios: To verify the efficiency of the proposed strategy at different speed ratios, β is set to four values (1.0, 1.5, 1.76, and 2.0) with dataset Cardiff_50. All scenario settings are the same as Section V-B1. Table III presents the experimental results, which show that the DTU collaboration strategy can reduce the total delay by at least 19.20% compared with the PTU strategy and 6.33% compared with the TUFS strategy at different speed ratios. This means that the DTU collaboration strategy outperforms the PTU and TUFS strategies at different and commonly adopted speed ratios.

3) Resilience Analysis: Because road networks are frequently destroyed during emergencies, the accessibility of trucks will be affected, which would impede the delivery of emergency response services. Therefore, the ability to cope with road network disruptions and ensure the quality of emergency response services will be very important, which is commonly known as the resilience of emergency systems [21]. In this part, scenarios with 100 rescue demands are adopted to evaluate the resilience of the DTU collaboration strategy. Three damage scales of the road network are considered, namely, 10% cutting off, 20% cutting off, and 40% cutting off, which means that 10%, 20%, and 40% of roads cannot be accessed by trucks due to road damage; thus, some demands have to be served by UAVs. Based on these three scenarios, a series of experiments are carried out to compare the resilience of different truck–UAV collaboration strategies. The results are shown in Table IV. In this table, the average objectives and the demand delay of three strategies are compared in different scenarios (a fully equipped road network, 10% cutting off roads, 20% cutting off roads, and 40% cutting off roads). In addition, the gaps among the DTU, PTU, and TUFS collaboration strategies in different scenarios are given.

From Table IV, the DTU collaboration strategy reduces the average objectives by at least 6.05×10^7 and demand delay by at least 100.84 min, which shows that it performs better in reducing

TABLE III
EFFICIENCY COMPARISONS OF DIFFERENT TRUCK–UAV STRATEGIES AT DIFFERENT SPEED RATIOS

Speed ratio (β)	Strategies	Avg_Obj (10^7)	Total delay (min)	Improve_1	Improve_2
1.0	PTU	9.27	154.50	0.00%	-9.14%
	TUFS	8.49	141.56	8.38%	0.00%
	DTU	**6.94**	**115.71**	**25.11%**	**18.26%**
1.5	PTU	8.89	148.18	0.00%	-21.41%
	TUFS	7.32	122.05	17.63%	0.00%
	DTU	**6.86**	**114.33**	**22.84%**	**6.33%**
1.76	PTU	8.19	136.52	0.00%	-0.06%
	TUFS	8.19	136.44	0.06%	0.00%
	DTU	**6.11**	**101.74**	**25.48%**	**25.43%**
2.0	PTU	8.14	135.62	0.00%	-7.81%
	TUFS	7.55	125.80	7.24%	0.00%
	DTU	**6.59**	**109.82**	**19.20%**	**12.70%**

The bold parts show the response efficiency of the DTU collaboration strategy at different speed ratios.

TABLE IV
RESILIENCE COMPARISONS OF DIFFERENT STRATEGIES IN DIFFERENT SCENARIOS

Strategies	PTU		TUFS		DTU	
Scenarios	Avg_obj (10^7)	Total delay (min)	Avg_obj (10^7)	Total delay (min)	Avg_obj (10^7)	Total delay (min)
Fully-equipped	14.84	247.30	17.83	297.10	8.41	140.11
Gap	**6.43**	**107.19**	**9.42**	**157.00**	**0**	**0**
10% disruption	16.70	278.38	18.49	308.16	9.41	156.84
Gap	**7.29**	**121.54**	**9.08**	**151.32**	**0**	**0**
20% disruption	18.55	309.22	20.36	339.25	12.50	208.38
Gap	**6.05**	**100.84**	**7.85**	**130.87**	**0**	**0**
40% disruption	28.96	482.61	40.67	677.83	15.73	262.10
Gap	**13.23**	**220.51**	**24.94**	**415.73**	**0**	**0**

The bold parts show the gaps of average objective and total delay in the PTU strategy and TUFS, respectively, compared with the DTU collaboration strategy.

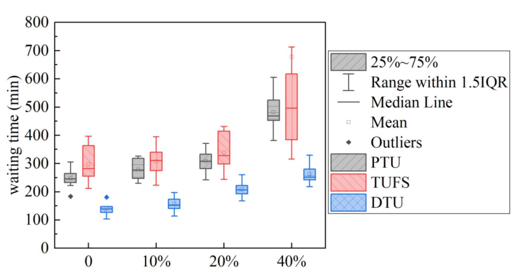


Fig. 11. Resilience analysis.

both the average objectives and the demand delay than the PTU and TUFS strategies.

Meanwhile, from the box-plot figure of the result (see Fig. 11), the interquartile range of the DTU collaboration strategy is much smaller than those of the TUFS and PTU strategies, which means that the DTU collaboration strategy can perform steadily in different instances with different topologies. Therefore, the DTU collaboration strategy has better resilience and can cope well with road network disruptions. There are two reasons why the DTU collaboration strategy performs better. One is that the dynamic collaboration between trucks and UAVs provides a more feasible scheduling solution for truck–UAV collaboration scheduling problems. The other is that perfect road conditions are not a necessary condition for the DTU collaboration strategy. Dynamic collaboration takes full advantage of the high-density and disrupted road network, making it possible for trucks and UAVs to collaborate efficiently even after road disruptions.

C. *Performance of the TSI Scheduling Algorithm*

Considering that local search algorithms are the most prevalent algorithms in the relevant literature (e.g., TS [53], simulated annealing (SA) [52], and VNS [54], [55]), the performances of TS, SA, and VNS are taken for comparison to evaluate the performance of the TSI scheduling algorithm. The results are indicated in Table V. Here, TS is regarded as a baseline with the comparisons of the average objectives and demand delays. The datasets show the names of the used instance data and the scales of the demands. In addition, the elements Avg_Obj (average objective), total delay (demand delay), and C_time (computation time) are all obtained from the average of ten instances at the same scale.

As Table V presents, TS and SA do not perform as well as desired with large average objectives and demand delays, although the computation times of TS and SA are extremely small. In comparison, VNS and TSI scheduling algorithms perform well in different scales of instances, with up to 69.22% improvement compared with the TS algorithm. In addition, the TSI scheduling algorithm outperforms VNS in most instances within a much smaller computation time.

TABLE V
EFFECTIVENESS OF THE TSI SCHEDULING ALGORITHM

Datasets	Algorithms	Avg_Obj (10^7)	Total delay (min)	Delay_Improve	C_time (min)
Cardiff_15	TS	2.07	34.43	0.00%	1.05
	SA	1.66	27.69	19.58%	0.81
	VNS	1.20	20.02	41.84%	5.20
	TSI	**1.13**	**18.78**	**45.46%**	**2.78**
Cardiff_25	TS	10.58	176.36	0.00%	1.05
	SA	10.84	180.64	-2.42%	1.55
	VNS	6.80	113.36	35.72%	8.52
	TSI	**7.69**	**128.18**	**27.32%**	**6.27**
Cardiff_50	TS	13.13	218.85	0.00%	1.35
	SA	16.55	275.82	-26.03%	2.84
	VNS	7.00	116.69	46.68%	32.39
	TSI	**6.11**	**101.74**	**53.51%**	**21.60**
Cardiff_75	TS	23.88	398.02	0.00%	2.58
	SA	35.69	594.83	-49.45%	4.81
	VNS	12.34	205.72	48.31%	107.17
	TSI	**10.45**	**174.23**	**56.23%**	**43.00**
Cardiff_100	TS	27.31	455.21	0.00%	5.42
	SA	49.79	829.82	-82.29%	8.15
	VNS	11.82	197.05	56.71%	261.22
	TSI	**8.41**	**140.11**	**69.22%**	**78.80**

The bold parts show the performance of the TSI scheduling algorithm at different instance scales.

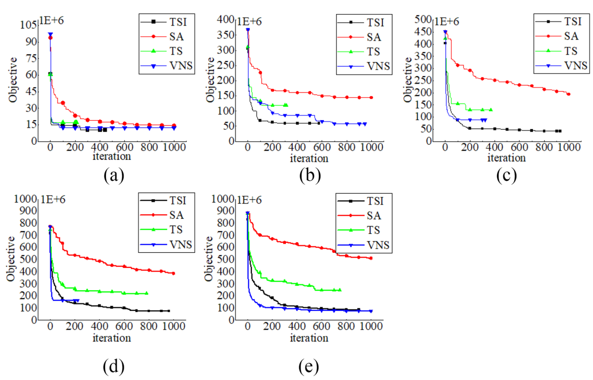


Fig. 12. Comparisons of the convergence between different algorithms.

To compare the searching efficiency and convergence performance of the above four algorithms, the total objectives for randomly chosen instances are compared with different problem scales. The convergence speed of these algorithms at different problem scales is shown in Fig. 12. In these figures, the search processes of all algorithms show the trend of rapid objective decline at the beginning and gradual convergence. However, SA cannot obtain an acceptable solution at large scales, while TS is very inclined to fall into local optima. Although VNS performs as well as the TSI scheduling algorithm in some instances, it is also liable to become trapped in local optima. In general, the TSI scheduling algorithm can obtain better solution among these four algorithms, and it avoids trapping into local optima with poor solutions.

D. Discussions

According to the results given above, several implications can be concluded as follows.

First, the proposed DTU collaboration strategy has been verified to be effective and efficient in the scenario of urban emergency response, resulting in improved response efficiency. In addition, it shows better resilience in coping with disruptions on road networks compared with the existing truck–UAV collaboration strategies.

Second, the TSI scheduling algorithm is effective and efficient in solving both small-scale and large-scale cases in urban emergency response. Meanwhile, compared with the existing popular heuristic methods, the algorithm has a relatively faster convergence speed, which can find better solutions even in a very short time, making it suitable for emergency response scenarios.

Third, the proposed DTU collaboration strategy and the TSI scheduling algorithm can generate solutions with less delay time, which means that most of the demands can be served within the required response time. The approach ensures the fairness of emergency services after disasters, which is very important for such public services.

However, although the DTU collaboration strategy can improve the response efficiency, the computing time for near-optimal solutions increases dramatically when facing large-scale problems. It is a common problem faced by most heuristic methods, and more efficient algorithms should be designed in the future.

VI. CONCLUSION

To realize efficient and resilient emergency response in urban areas, in this article, we proposed a DTU collaboration strategy and the corresponding scheduling algorithm. The DTU strategy leveraged both trucks and UAVs by allowing UAVs to dynamically takeoff, recharge/reload, and land on different trucks. The scheduling of trucks and UAVs was then modeled as an MILP, with the objective of minimizing the total service delay and response completion time. An efficient solution algorithm based on TS was proposed for the integrated truck–UAV scheduling problem, which had been verified to be effective in solving both small- and large-scale problems. Finally, comprehensive experiments were conducted that verify the efficiency and resilience of the DTU collaboration strategy, as well as the performance of the proposed TSI scheduling algorithm.

Future research will be addressed in the following perspectives. First, the efficiency of the solution algorithm for truck–UAV scheduling can be improved. Second, UAVs' takeoff and landing on moving trucks can be considered to further enhance the efficiency of truck–UAV collaborations. Third, the strategy can be adopted in many other scenarios, such as last-mile delivery and inspection of infrastructure.

References

[1] M. Zhong, C. Shi, T. Fu, L. He, and J. Shi, "Study in performance analysis of China urban emergency response system based on Petri net," *Saf. Sci.*, vol. 48, no. 6, pp. 755–762, 2010.

[2] G. Xu, G. Q. Huang, and J. Fang, "Cloud asset for urban flood control," *Adv. Eng. Inform.*, vol. 29, no. 3, pp. 355–365, 2015.

[3] R. Faturechi and E. Miller-Hooks, "Travel time resilience of roadway networks under disaster," *Transp. Res. B, Methodol.*, vol. 70, pp. 47–64, 2014.

[4] D. Yu et al., "Disruption of emergency response to vulnerable populations during floods," *Nature Sustain.*, vol. 3, no. 9, pp. 728–736, 2020.

[5] H. Huang, C. Hu, J. Zhu, M. Wu, and R. Malekian, "Stochastic task scheduling in UAV-based intelligent on-demand meal delivery system," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13040–13054, Aug. 2022.

[6] H. Huang, A. V. Savkin, and C. Huang, "Decentralized autonomous navigation of a UAV network for road traffic monitoring," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 4, pp. 2558–2564, Aug. 2021.

[7] C. Zhao, Y. Wang, X. Zhang, S. Chen, C. Wu, and K. L. Teo, "UAV dispatch planning for a wireless rechargeable sensor network for bridge monitoring," *IEEE Trans. Sustain. Comput.*, vol. 8, no. 2, pp. 293–309, Apr./Jun. 2023.

[8] G. Aiello, F. Hopps, D. Santisi, and M. Venticinque, "The employment of unmanned aerial vehicles for analyzing and mitigating disaster risks in industrial sites," *IEEE Trans. Eng. Manage.*, vol. 67, no. 3, pp. 519–530, Aug. 2020.

[9] Z. Lv, D. Chen, H. Feng, H. Zhu, and H. Lv, "Digital twins in unmanned aerial vehicles for rapid medical resource delivery in epidemics," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 25106–25114, Dec. 2022.

[10] Y. Pan, Q. Chen, N. Zhang, Z. Li, T. Zhu, and Q. Han, "Extending delivery range and decelerating battery aging of logistics UAVs using public buses," *IEEE Trans. Mobile Comput.*, Apr. 2022.

[11] H. Y. Jeong, B. D. Song, and S. Lee, "Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones," *Int. J. Prod. Econ.*, vol. 214, pp. 220–233, 2019.

[12] G. Wu, M. Fan, J. Shi, and Y. Feng, "Reinforcement learning based truck-and-drone coordinated delivery," *IEEE Trans. Artif. Intell.*, vol. 4, no. 4, pp. 754–763, Aug. 2023.

[13] H. Li, J. Chen, F. Wang, and M. Bai, "Ground-vehicle and unmanned-aerial-vehicle routing problems from two-echelon scheme perspective: A review," *Eur. J. Oper. Res.*, vol. 294, no. 3, pp. 1078–1095, 2021.

[14] Y. Liu, Z. Liu, J. Shi, G. Wu, and W. Pedrycz, "Two-echelon routing problem for parcel delivery by cooperated truck and drone," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 51, no. 12, pp. 7450–7465, Dec. 2021.

[15] M. Golabi, S. M. Shavarani, and G. Izbirak, "An edge-based stochastic facility location problem in UAV-supported humanitarian relief logistics: A case study of Tehran earthquake," *Natural Hazards*, vol. 87, no. 3, pp. 1545–1565, 2017.

[16] Q. Gu, T. Fan, F. Pan, and C. Zhang, "A vehicle-UAV operation scheme for instant delivery," *Comput. Ind. Eng.*, vol. 149, 2020, Art. no. 106809.

[17] S. V. Bagare, K. Mirza, M. Sharma, D. Shukla, and R. Pant, "Design of mobile docking mechanism for unmanned aerial vehicles capable of vertical take-off and landing," in *Proc. AIAA AVIATION*, 2022, Paper 4063.

[18] N. Altay and W. G. Green III, "OR/MS research in disaster operations management," *Eur. J. Oper. Res.*, vol. 175, no. 1, pp. 475–493, 2006.

[19] D. Tippong, S. Petrovic, and V. Akbari, "A review of applications of operational research in healthcare coordination in disaster management," *Eur. J. Oper. Res.*, vol. 301, pp. 1–17, 2022.

[20] M. Eftekhar, J.-S. J. Song, and S. Webster, "Prepositioning and local purchasing for emergency operations under budget, demand, and supply uncertainty," *Manuf. Service Oper. Manage.*, vol. 24, no. 1, pp. 315–332, 2022.

[21] G. Xu, J. Wang, G. Q. Huang, and C.-H. Chen, "Data-driven resilient fleet management for cloud asset-enabled urban flood control," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1827–1838, Jun. 2018.

[22] Y. Wan, Y. Zhong, A. Ma, and L. Zhang, "An accurate UAV 3-D path planning method for disaster emergency response based on an improved multiobjective swarm intelligence algorithm," *IEEE Trans. Cybern.*, vol. 53, no. 4, pp. 2658–2671, Apr. 2023.

[23] G. Xu, G. Q. Huang, J. Fang, and J. Chen, "Cloud-based smart asset management for urban flood control," *Enterp. Inf. Syst.*, vol. 11, no. 5, pp. 719–737, 2017.

[24] Z. Su, Y. Wang, Q. Xu, and N. Zhang, "LVBS: Lightweight vehicular blockchain for secure data sharing in disaster rescue," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 19–32, Jan./Feb. 2022.

[25] X. Liu et al., "Transceiver design and multihop D2D for UAV IoT coverage in disasters," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1803–1815, Apr. 2019.

[26] N. Zhao et al., "UAV-assisted emergency networks in disasters," *IEEE Wireless Commun.*, vol. 26, no. 1, pp. 45–51, Feb. 2019.

[27] M. Aggravi, G. Sirignano, P. R. Giordano, and C. Pacchierotti, "Decentralized control of a heterogeneous human-robot team for exploration and patrolling," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 4, pp. 3109–3125, Oct. 2022.

[28] N. Pulsiri and R. Vatananan-Thesenvitz, "Drones in emergency medical services: A systematic literature review with bibliometric analysis," *Int. J. Innov. Technol. Manage.*, vol. 18, pp. 1–23, 2021.

[29] K. Yakushiji, H. Fujita, M. Murata, N. Hiroi, Y. Hamabe, and F. Yakushiji, "Short-range transportation using unmanned aerial vehicles (UAVs) during disasters in Japan," *Drones*, vol. 4, no. 4, 2020, Art. no. 68.

[30] I. Khoufi, A. Laouiti, and C. Adjih, "A survey of recent extended variants of the traveling salesman and vehicle routing problems for unmanned aerial vehicles," *Drones*, vol. 3, no. 3, 2019, Art. no. 66.

[31] C. Gomez and H. Purdie, "UAV-based photogrammetry and geocomputing for hazards and disaster risk monitoring—A review," *Geoenviron. Disasters*, vol. 3, no. 1, 2016, Art. no. 23.

[32] P. Tokekar, J. v. Hook, D. Mulla, and V. Isler, "Sensor planning for a symbiotic UAV and UGV system for precision agriculture," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1498–1511, Dec. 2016.

[33] Y.-J. Zheng, Y.-C. Du, W.-G. Sheng, and H.-F. Ling, "Collaborative human–UAV search and rescue for missing tourists in nature reserves," *Informs J. Appl. Anal.*, vol. 49, no. 5, pp. 371–383, 2019.

[34] B. E. Oruc and B. Y. Kara, "Post-disaster assessment routing problem," *Transp. Res. B, Methodol.*, vol. 116, pp. 76–102, 2018.

[35] A. Nedjati, B. Vizvari, and G. Izbirak, "Post-earthquake response by small UAV helicopters," *Natural Hazards*, vol. 80, no. 3, pp. 1669–1688, 2016.

[36] Z. Huang, C. Chen, and M. Pan, "Multiobjective UAV path planning for emergency information collection and transmission," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 6993–7009, Aug. 2020.

[37] C. C. Murray and A. G. Chu, "The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery," *Transp. Res. C, Emerg. Technol.*, vol. 54, pp. 86–109, 2015.

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

[39] M. Salama and S. Srinivas, "Joint optimization of customer location clustering and drone-based routing for last-mile deliveries," *Transp. Res. C, Emerg. Technol.*, vol. 114, pp. 620–642, 2020.

[40] H. Li, H. Wang, J. Chen, and M. Bai, "Two-echelon vehicle routing problem with time windows and mobile satellites," *Transp. Res. B, Methodol.*, vol. 138, pp. 179–201, 2020.

[41] E. E. Yurek and H. C. Ozmutlu, "A decomposition-based iterative optimization algorithm for traveling salesman problem with drone," *Transp. Res. C, Emerg. Technol.*, vol. 91, pp. 249–262, 2018.

[42] J. C. de Freitas and P. H. V. Penna, "A variable neighborhood search for flying sidekick traveling salesman problem," *Int. Trans. Oper. Res.*, vol. 27, no. 1, pp. 267–290, 2020.

[43] M. Dell'Amico and S. Martello, "Reduction of the three-partition problem," *J. Combinatorial Optim.*, vol. 3, no. 1, pp. 17–30, 1999.

[44] J. A. Hoogeveen, J. K. Lenstra, and B. Veltman, "Preemptive scheduling in a two-stage multiprocessor flow shop is NP-hard," *Eur. J. Oper. Res.*, vol. 89, no. 1, pp. 172–175, 1996.

[45] G. Barbarosoglu and D. Ozgur, "A tabu search algorithm for the vehicle routing problem," *Comput. Oper. Res.*, vol. 26, no. 3, pp. 255–270, 1999.

[46] S. Shao, G. Xu, M. Li, and G. Q. Huang, "Synchronizing e-commerce city logistics with sliding time windows," *Transp. Res. E, Logist. Transp. Rev.*, vol. 123, pp. 17–28, 2019.

[47] M. Gmira, M. Gendreau, A. Lodi, and J.-Y. Potvin, "Tabu search for the time-dependent vehicle routing problem with time windows on a road network," *Eur. J. Oper. Res.*, vol. 288, no. 1, pp. 129–140, 2021.

[48] C. Chen, E. Demir, and Y. Huang, "An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and delivery robots," *Eur. J. Oper. Res.*, vol. 294, no. 3, pp. 1164–1180, 2021.

[49] K. Kloster, M. Moeini, D. Vigo, and O. Wendt, "The multiple traveling salesman problem in presence of drone- and robot-supported packet stations," *Eur. J. Oper. Res.*, vol. 305, no. 2, pp. 630–643, 2023.

[50] N. Agatz, P. Bouman, and M. Schmidt, "Optimization approaches for the traveling salesman problem with drone," *Transp. Sci.*, vol. 52, no. 4, pp. 965–981, 2018.

[51] D. Schermer, M. Moeini, and O. Wendt, "A hybrid VNS/tabu search algorithm for solving the vehicle routing problem with drones and en route operations," *Comput. Oper. Res.*, vol. 109, pp. 134–158, 2019.

[52] G. Wu, N. Mao, Q. Luo, B. Xu, J. Shi, and P. N. Suganthan, "Collaborative truck-drone routing for contactless parcel delivery during the epidemic," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 25077–25091, Dec. 2022.

[53] Y. Wang, Z. Wang, X. Hu, G. Xue, and X. Guan, "Truck–drone hybrid routing problem with time-dependent road travel time," *Transp. Res. C, Emerg. Technol.*, vol. 144, 2022, Art. no. 103901.

[54] R. J. Kuo, S.-H. Lu, P.-Y. Lai, and S. T. W. Mara, "Vehicle routing problem with drones considering time windows," *Expert Syst. Appl.*, vol. 191, 2022, Art. no. 116264.

[55] M. R. Salama and S. Srinivas, "Collaborative truck multi-drone routing and scheduling problem: Package delivery with flexible launch and recovery sites," *Transp. Res. E, Logist. Transp. Rev.*, vol. 164, 2022, Art. no. 102788.

Yuying Long received the B.S. degree in traffic and transportation from Southwest Jiaotong University, Chengdu, China, and the M.E. degree in transportation engineering from the Harbin Institute of Technology (Shenzhen), Shenzhen, China, in 2020 and 2022, respectively. She is currently working toward the Ph.D. degree in aviation engineering from The Hong Kong Polytechnic University, Hong Kong.

Her research interests include UAV-assisted systems, emergency management, and emergency logistics.

Gangyan Xu (Member, IEEE) received the B.S. degree in automation and the M.E. degree in systems engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2009 and 2012, respectively, and the Ph.D. degree in systems engineering from The University of Hong Kong, Hong Kong, in 2016.

He was an Assistant Professor with the Harbin Institute of Technology, Shenzhen, China. He is a Research Fellow with Nanyang Technological University, Singapore; and a Research Assistant with the City University of Hong Kong, Hong Kong. He is currently an Assistant Professor with The Hong Kong Polytechnic University, Hong Kong. His research interests include data-driven optimization and control, intelligent transportation systems, resilience engineering, and emergency management.

Dr. Xu is an Editorial Board Member of Advanced Engineering Informatics.

Jinqiu Zhao received the B.S. degree in port channel and coastal engineering from the Ocean University of China, Qingdao, China, and the M.S. degree in port channel and coastal engineering from the Dalian University of Technology, Dalian, China, in 2013 and 2016, respectively. He is currently working toward the Ph.D. degree in transportation with the Harbin Institute of Technology (Shenzhen), Shenzhen, China.

His research interests include logistics transportation optimization, intelligent transportation systems, and emergency management.

Binglei Xie received the B.S. degree in applied mathematics and the Ph.D. degree in management science and engineering from Southwest Jiaotong University, Chengdu, China, in 1996 and 2003, respectively.

He is currently a Professor of transportation planning and management with the School of Architecture, Harbin Institute of Technology (Shenzhen), Shenzhen, China. His primary research interests are travel behavior analysis, emergency transportation management, and integrated land use/transport planning. His published articles appear in *Transportation Research Part E* and *Advanced Engineering Informatics,* among others.

Meng Fang received the Ph.D. degree in computer science from the University of Technology, Sydney, NSW, Australia, in 2015.

He was a Research Fellow with the School of Computing and Information Systems, University of Melbourne, Melbourne, VIC, Australia. He is currently an Assistant Professor with the Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands. His research focuses on reinforcement learning and natural language processing.