Collaborative Delivery Route Planning for Trucks and Drones Utilizing an Optimized Genetic Algorithm

Abstract: Collaborative delivery involving trucks and drones symbolizes one of the emerging trends in future logistics and distribution. This paper proposed a collaborative delivery model that integrated trucks and drones to address the path planning challenges for diverse scenarios. In detail, this model encompasses multifarious practical constraints, including vehicle load limits, drone flight range restrictions, and time window constraints. To explore optimized solution for this model, this paper developed an improved genetic algorithm, which integrated the simulated annealing algorithm with a large-scale neighborhood search approach to bolster algorithm performance. Two numerical case studies have validated the feasibility and efficiency of the proposed algorithm. The results show that the collaborative delivery model can substantially reinforce delivery efficiency, which brings about an average abatement in delivery cost by 13.61%. This research offers a brand new viewpoint and innovative methodology for optimizing urban logistics distribution systems.

Keywords: Vehicle Routing Planning, Collaborative Delivery, Drone; Optimized Genetic Algorithm, Large-scale Neighborhood Search.

1. Introduction

The rapid development of e-commerce, coupling with the escalating consumer demand for swift delivery, has posed considerable challenges to traditional delivery mode. Meanwhile, Logistics companies are also increasingly focusing on cost reduction and efficiency promotion. Advanced drone technologies have offered innovative standpoints to deal with this dilemma [1]. The collaborative delivery model, which integrates drones with conventional trucks, can effectively combine the advantages of both and is becoming a principal method of urban delivery in the future. It is worth noting that the essence of this collaborative delivery model lies in utilizing trucks as mobile hubs, carrying drones to execute delivery tasks. Trucks undertake long-distance transportation and serve major customer points, whereas drones can serve nearby, hard-to-reach customer spots or those with small delivery volumes [2]. The collaborative delivery model not only raises delivery efficiency but also reduces labor cost, alleviates traffic congestion, and mitigates environmental pollution to a certain extent.

In the past decades, extensive research has been conducted by scholars on collaborative delivery issues. Truck-Drone collaborative delivery mode takes the trucks as carriers, storage hubs, and energy sources for drones, reinforcing last-mile delivery capabilities, thus optimizing transportation efficiency [3]. In comparison with traditional truck deliveries, the collaborative delivery approach of trucks and drones can reduce cost by 50.25% and raise delivery efficiency [4-5]. How to efficient and rational planning for collaborative delivery routes is a key technology for this emerging delivery method. The challenge lies in efficiently planning the routes for both drones and trucks, and minimizing total distribution cost under multiple constraints, including path, payload, endurance, and depreciation cost for trucks and drones. At present, most related studies are concentrated upon a single constraint, neglecting the intricacies of real-world scenarios and resulting in a lot of deficiencies in addressing multiple constraints, providing efficient solutions, and conducting economic benefit analyses. In recent years, the issue of collaborative distribution, defined as a logistics strategy that integrates multiple transportation modes to optimize delivery efficiency, has garnered increasing attention from academia, leading to notable research advancements. Among these, public transportation-based logistics solutions have emerged as cost-effective and sustainable alternatives, particularly for urban and rural last-mile delivery.

Massion et al. [6] and Ghilas et al. [7] investigated a serial collaborative distribution model involving public buses and enterprise-owned vehicles, wherein goods are first transported by buses to designated transfer stations before being delivered to customers by company-owned vehicles. To address this problem, the researchers applied large neighborhood search (LNS) algorithms and branch-and-price approaches to optimize delivery routes. Their findings indicate that LNS efficiently handles large-scale routing problems, while branch-and-price provides high-quality solutions for complex, constrained delivery networks. The results confirm that collaborative distribution significantly reduces costs compared to direct truck-based delivery.

Further, Demir et al. [9] considered stochastic demand variations and employed a sample average approximation (SAA) method alongside Adaptive Large Neighborhood Search (ALNS) to optimize the bus-enterprise delivery model. The study demonstrated that, even under demand uncertainty, SAA effectively captures probabilistic demand fluctuations, enabling ALNS to generate more resilient and cost-efficient routing plans. Additionally, He Yunzhu et al. [10] proposed a hybrid distribution strategy, wherein certain shipments are transported via truck–bus collaboration, while others rely on direct truck delivery. Using an ant colony optimization (ACO) algorithm, their study found that this bio-inspired metaheuristic efficiently balances cost minimization and delivery timeliness, outperforming conventional vehicle routing methods in dynamic delivery environments.

Expanding on these approaches, Kerim et al. [11] developed a comprehensive multimodal logistics framework, integrating public transportation as the backbone and supplementing it with automated service points, crowdsourced delivery fleets, and proprietary new-energy vehicles. By employing a branch-and-price algorithm, the study demonstrated that this environmentally sustainable model significantly reduces express delivery costs while maintaining service quality.

Despite the cost benefits of collaborative distribution models that integrate public transportation and enterprise-owned vehicles, several logistical challenges persist in rural areas. First, the introduction of proprietary vehicles entails high initial investment and ongoing maintenance expenses, making it less viable for logistics firms operating in sparsely populated regions. Second, the vast geographical expanse and dispersed population density in rural areas, combined with low, irregular delivery demand, contribute to high operational costs and route inefficiencies. Zhou Yue [12] and Cui Yong [13] demonstrated that leveraging idle social transport capacity can effectively reduce transportation costs while enhancing delivery efficiency. Further studies [14–18] have highlighted the significant advantages of collaborative distribution, including reducing vehicle procurement and labor costs for logistics enterprises, improving profitability, and enhancing the adaptability of capacity allocation. Building upon these findings, Kou et al. [19] proposed a multimodal transportation model tailored to rural logistics. Under this framework, specialized goods, such as large household appliances, continue to be delivered via enterprise-owned vehicles, whereas general cargo is transported through a joint distribution network integrating public transportation. Experimental results indicate that this collaborative distribution model can effectively reduce logistics costs while expanding service coverage in rural areas. Moreover, while truck–drone collaborative models [20–22] have been proposed for rural logistics, they face significant operational constraints related to:

Limited battery life and payload capacity restricting delivery range.

Regulatory barriers governing drone flights in sparsely connected road networks.

Infrastructure challenges, as rural regions often lack dedicated drone launch and landing zones.

Given these challenges,

By taking varied constraints such as load capacity, driving range, and delivery time windows into consideration, this paper addresses the real-world scenario of truck-drone collaborative delivery. A collaborative delivery model was established under these multiple constraints. To solve this model and obtain the optimal route, an improved genetic algorithm, integrating genetic operators, probabilistic adaptation, and simulated annealing was put forward. The superiority of this approach in dealing with multiple constraints has also been demonstrated through large-scale numerical experiments.

2. Mathematical Modeling of Truck-drone Collaborative Delivery

2.1 Problem description

The traditional delivery mode relies on logistics personnel driving trucks, which can serve multiple demand points on account of their large capacity but accompanied by high cost and inefficiencies [25]. On the other hand, drones offer swift point-to-point delivery services to customers through their compact size and agile flight capabilities, but its capacity and endurance restrict the scope of services and the number of delivery destinations [26]. On this basis, this paper combines each other's disadvanteges and proposes a hybrid delivery model to raise efficiency [27].

This paper is dedicated to resolving the delivery issues in mountainous areas, where the terrain constraints prevent trucks from accessing directly. The collaborative delivery mode comprises of a truck and a drone. To be more specific, the truck undertakes long-distance transportation and serves as the supply station of the drone. The drone provides services at specific demand points and returns to the truck for reloading or battery replacement. In this context, truck route planning must accommodate the takeoff and landing sequences of the drone mission paths, while striking a balance between cost and efficiency. The special terrain of mountainous areas makes the planning of drone flight paths complicated. Since the battery capacity is limited, the maximum flight distance of the drone poses a prominent constraint.

To establish the collaborative delivery model, the following assumptions are defined:

1. All of the road information is known and certain demand points are inaccessible for truck.

2. The velocities of the truck and the drone are different constants.

3. Each truck is equipped with a drone, forming a one-to-one correspondence and calling a collaborative vehicle. The truck departs from the distribution center and eventually returns to it, while the drone can only take off and land on the matching truck or service points.

4. Both trucks and drones have load capacity constraints.

5. For a specified point-to-point transportation, the distance traveled by a truck is different from that by a drone.

6. Each demand point can only be serviced once, either by a truck or a drone.

7. Subjected to the battery capacity, drones have mileage constraints.

8. The time required for drone battery replacement is a constant.

9. The service time for each demand point is known.

10. The electricity consumption during waiting and service time is ignored, that is, the electricity consumption is merely dependent on the flight distance.

2.2 Mathematical modeling

To reduce the collaborative delivery cost and raise the delivery efficiency under a variety of constraints, this paper utilizes a mixed-integer programming model to address the challenge. The collaborative delivery cost comprises of the fixed cost, battery replacement cost, delivery cost, and so on. The fixed cost consists of the expenses associated with the use of vehicles and drones, including depreciation of equipment, employee salaries, and daily maintenance. The fixed cost can be expressed as

$$f_1 = \sum_{c \in C} X_c (\alpha_1 + \alpha_2) \tag{2.1}$$

where f_1 is the total fixed cost; α_1, α_2 is the fixed cost for trucks and drones; $X_c \in \{0,1\}$ symbolizes whether the truck or the drone is employed; C refers to the collaborative vehicle collection (truck, drone)

As illustrated in Equation 2.2, the battery replacement cost for drones encompasses expenses associated with replacing the batteries, including the cost associated with maintaining battery degradation:

$$f_2 = \alpha_3 \sum_{c \in C} X_c \sum_{n \in N} U^b_{c,n} \tag{2.2}$$

where α_3 defines the cost of replacing the battery of a drone; $U^b_{c,n} \in \{0,1\}$ refers to whether the drone in the collaborative vehicle c lands at the demand point; N symbolizes the collection of demand points.

Driving cost, including vehicle driving cost and drone flight cost, such as fuel consumption, electricity consumption, and distance related factors, can be expressed as Equation 2.3

$$f_3 = \sum_{c \in C} X_c (\alpha_4 \sum_{n1 \in R} \sum_{n2 \in R} D^{car}_{n_1,n_2} Z^{c,car}_{n_1,n_2} + \alpha_4 \sum_{n1 \in N} \sum_{n2 \in N} D^{uav}_{n_1,n_2} Z^{c,uav}_{n_1,n_2}) \tag{2.3}$$

where α_3 symbolizes the cost per kilometer of the drone; $D^{car}_{n_1,\ n_2}$ defines the distance traveled by the truck node to node;

Where, α_3 symbolizes the cost per kilometer of the drone; $D^{car}_{n_1,\ n_2}$ defines the distance $n_1, n_2 \in R$ traveled by the truck node n_1 to node n_2; $D^{uav}_{n_1,n_2}$ denotes the distance $n_1, n_2 \in R$ the drone flies from node n_1 to node n_2; $Z^{c,car}_{n_1,n_2} \in \{0,1\}$ refers to whether the truck in the collaborative vehicle c goes from node n_1 to node n_2, $c \in C$, $n \in N$;

The remaining cost is mainly related to customer demand and can be calculated as follows:

$$f_4 = \sum_{n \in N} (\alpha_6 Max(0, T^S_n - S^t_n) + \alpha_7 Max(0, S^t_n - T^e_n)) \tag{2.4}$$

where T^S_{n} refers to the expected earliest service time $n \in N$ for the demand point n; S^t_n symbolizes the start time $n \in Ns$ when the demand point n is served; T^e_n defines the expected latest service time $n \in N$ for the demand point n; α_6 denotes the punishment for arriving early at the workplace; α_7 symbolizes the punishment for arriving late at the workplace;

Based on the above analysis, the total cost can be derived

$$\begin{aligned} &F = f_1 + f_2 + f_3 + f_4 \\ &= \sum_{c \in C} X_c (\alpha_1 + \alpha_2) + \alpha_3 \sum_{c \in C} X_c \sum_{n \in N} U^b_{c,n} + \sum_{c \in C} X_c (\alpha_4 \sum_{n_1 \in R} \sum_{n_2 \in R} D^{car}_{n_1,n_2} Z^{c,car}_{n_1,n_2} + \\ &\alpha_4 \sum_{n_1 \in N} \sum_{n_2 \in N} D^{uav}_{n_1,n_2} Z^{c,uav}_{n_1,n_2}) + \sum_{n \in N} (\alpha_6 Max(0, T^S_n - S^t_n) + \alpha_7 Max(0, S^t_n - T^e_n)) \end{aligned} \tag{2.5}$$

The above constraints are defined as follows for actual scenarios and algorithm design:

1. Each demand point can only be served once

$$\forall n \in N, \sum_{c \in C} Y^{car}_{c,n} + \sum_{c \in C} Y^{uav}_{c,n} = 1 \tag{2.6}$$

Where, $Y^{car}_{c,n} \in \{0,1\}$ denotes whether the demand point n is served through the truck in the collaborative vehicle c, $c \in C$, $n \in N$

2. Truck line connection constraints, flow balance

$$\forall c \in C, \forall n_1 \in N, \sum_{n_2 \in R} Z_{n_1,n_2}^{c,car} = \sum_{n_2 \in R} Z_{n_2,n_1}^{c,car} = Y_{c,n_1}^{car} \tag{2.7}$$

3. Only collaborative vehicles are employed, the corresponding trucks and drones can travel along the path

$$\forall c \in C, \forall n_1, n_2 \in R, Z_{n_1,n_2}^{c,car} \le X_c, Z_{n_1,n_2}^{c,uav} \le X_c \tag{2.8}$$

4. Constraints on drone wiring links, flow balance:

$$\forall c \in C, \forall n_1 \in N, \sum_{n_2 \in N} Z_{n_1,n_2}^{c,uav} = Y_{c,n_1}^{uav} + U_{c,n}^{a} \tag{2.9}$$

Equation 2.9 $U_{c,n}^{a} \in \{0,1\}$ indicates whether the drone in the collaborative vehicle takes off at the demand point. $U_{c,n}^{b}$ indicates whether the drone in the collaborative vehicle landing at the demand point, $c \in C$, $n \in N$

$$\forall c \in C, \forall n_1 \in N, \sum_{n_2 \in N} Z_{n_2,n_1}^{c,uav} = Y_{c,n_1}^{uav} + U_{c,n}^{b} \tag{2.10}$$

5. Load constraints

$$\forall c \in C, \sum_{n_1 \in N} Q_{n_1} Y_{c_n n}^{car} + \sum_{n_1 \in N} Q_{n_1} Y_{c_n n}^{uav} \le \gamma^{car} \tag{2.11}$$

In Equation 2.11, Q_{n} symbolizes the quantity demanded at the demand point n , $n \in N$; γ^{car} denotes the maximum load of the truck.

This investigation is principally intended to materialize the minimum total cost (Equation 2.5) under the constraint of Equation 2.6-2.11. The objective function takes into account factors such as fixed cost, transportation cost, and penalty cost incurred by violations of time windows, which is intended to minimize the total delivery cost [3]. The mathematical model in this paper is a mixed-integer programming problem with high nonlinear, combinatorial explosion, and the coupling of multiple constraints. Traditional algorithms is inefficient when applied to large-scale programming problem. As a consequence, this paper introduces an improved genetic algorithm.

3. The Improved Genetic Algorithm

The operational principle of genetic algorithms involves simulating the animal reproduction process to seek the optimal solution to a problem [29]. Genetic algorithms draw inspiration from the principles of biological evolution, iteratively refine and adapt potential solutions through mechanisms such as selective breeding, genetic crossover, and spontaneous mutation, ultimately calculating the most feasible solution to a specified problem[30]. Although genetic algorithms exhibit prominent performance for complex optimization problems, they also have some drawbacks [31]. For example, the slow convergence rates, a tendency to get trapped in local optima, a heightened sensitivity to parameter settings, and so on. [32] [33].

The improved genetic algorithm presented in this paper can markedly elevate optimization efficiency and expedite convergence. The Improved Genetic Algorithm proposed in this study introduces several key innovations specifically tailored to the truck-drone collaborative delivery problem, addressing critical constraints and enhancing computational efficiency. While prior improved genetic algorithm studies have primarily focused on domains such as robotic navigation[34] and aerospace applications[35][36], the present work explicitly targets path planning within a truck-drone collaborative delivery framework. For multi-drone logistics, the Improved Genetic Simulated Annealing Algorithm (IGSAA) integrates simulated annealing with genetic algorithms to minimize logistics cost, materializing conspicuous cost reductions in contrast to conventional methods [37],The proposed improved genetic algorithm is specifically designed to effectively manage critical constraints such as time windows, payload capacity, and drone endurance, which are often overlooked or insufficiently addressed in prior research. Aside from that, the algorithmic enhancements also incorporate the Improved Adaptive Large Neighborhood Search (IALNS) algorithm, which is employed to maximize work order completion rates. Empirical results demonstrate that IALNS outperforms Mixed-Integer Linear Programming (MILP) and Tabu Search in large-scale instances, further reinforcing its effectiveness in complex optimization scenarios[38].

GA is an approach that searches for the global optimal solution by randomly selecting feasible solution from the entire population. When dealing with complicated optimization and combination problems, the reliance on random selection for searching can result in prematurely finding local optimal solutions and sub-optimal search performance. Moreover, the most optimal solution may be lost during the crossover process as a consequence of the differences among individuals within distinct populations, which makes it difficult to obtain the optimal solution [12]. The simulated annealing algorithm demonstrates excellent performance in finding optimal solutions within a narrow range. As a result, it will be more effective to address practical problems if the simulated annealing algorithm is incorporated into genetic algorithms.

The procedure of the bettered genetic algorithm employed in this paper is outlined as follows:

This paper adopts a mixed coding representation to generate solutions for the initial population through both random and heuristic methods. To be specific, random generation primarily ensures the diversity of solutions, whereas heuristic generation improves the quality of the initial solutions. During the $(n+1)$th genetic operations, this paper incorporates probability-adaptive crossover and mutation operations, which dynamically adjust the crossover and mutation probabilities throughout the evolutionary process. It should be pointed out that this approach allows more individuals to be retained while increasing the likelihood of mutation for poorer individuals to maintain a balance between exploration and exploitation [39]. Subsequently, a simulated annealing strategy is introduced to avoid local optima [40]. In each iteration, there is a certain probability of accepting inferior solutions [41], and the temperature is updated in accordance with a geometric annealing schedule [42]. Finally, Adaptive Large Neighborhood Search (ALNS) [43-46] method is utilized to explore the solution space by iteratively disrupting and repairing the current solution. The simulated annealing process of the improved genetic algorithm utilized in this paper can be illustrated in Figure 1.

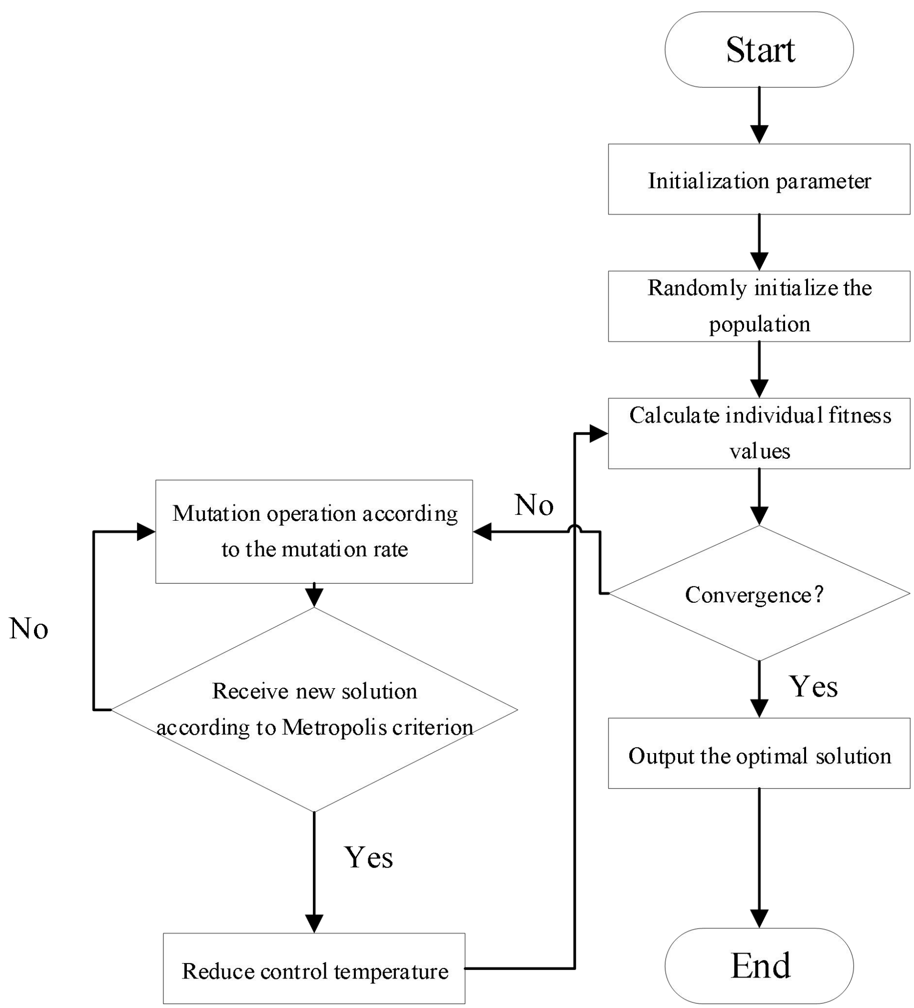

Figure 1 Flow chart of the improved genetic algorithm

The pseudo-code of the algorithm can be described as follows:

```
Set algorithm parameters, population size N, maximum iteration times T, minimum crossover probability pc1, maximum crossover probability pc2, minimum mutation probability pm1, maximum mutation probability pm2, initial temperature temp0, cooling rate cratio, initial scores for removal and destruction operators, and score adjustment mechanism, alns internal update times G
Randomly generate initial population, calculate objective function, update optimal solution, and set initial temperature temp = temp0
For t =1: T
    Update temperature temp = temp*cratio
    Select offspring population through binary tournament
    For  i = 1: N/2
        Extract two corresponding individuals from the offspring population
        % 1 self-adaptive
        On the basis of individual fitness values, adaptively calculate the crossover probability pc and mutation
    probability pm
        Randomly generate 0 to 1 decimal r1
        If      r1<pc
            Crossover through sub path crossover operator
        End if
        Randomly generate 0 to 1 decimal r2
        If      r2<pm
            Perform mutation through the 2-opt mutation operator
        End if
        Calculate the target value of the new solution and update the optimal solution
    End if
    Randomly generate 0 to 1 decimal r3
    If   r3<0.5
        Select the optimal solution as the initial solution x
    Else
        Randomly select a offspring as the initial solution x
    End if
    Introduction of % 2 simulated annealing
    For  i = 1: N
        Generate a new solution k by changing x through the insert operator
        Calculate the target value of the new solution k
        If      k is better than x
            x=k, Update the optimal solution
        Else
            In line with Metropolis' acceptance criteria, accept k
        End if
    End for
    If   r3<0.5
        Select the optimal solution as the initial solution s
    Else
        Randomly select a offspring as the initial solution s
    End if
    Introduction of % 3 alns
    For  i = 1: G
        Select the removal operator by roulette in accordance with the score (random removal, correlation removal)
        Select repair operator by roulette in line with score value (greedy repair, maximum regret repair)
        Generate a new solution k by changing the current solution s grounded in the selected removal and repair
    operators
        Calculate the new solution target value and record the score update through the current solution s, new
    solution k, optimal solution target value
        If the target value of the new solution k is better than the current solution s, update the current solution s
    End for
End for
Output the obtained optimal solution
```

4. Case Analysis

In this section, a sequence of numerical experiments were conducted to validate the effectiveness of mathematical modeling and the improved genetic algorithm. In detail, these experiments were modified from the Solomon standard benchmark set to accommodate the unique features of collaborative delivery problems. Tests were carried out on both small-scale problems (with 7 demand points) and large-scale problems (with 50 demand points), with comparisons made between the performances of multiple algorithms. This paper primarily distinguishes between large-scale and small-scale scenarios grounded in the number of demand points:

[1] Small-scale

The number of demand points typically falls within the range of 5 to 20, typically featured by comparatively low computational complexity and straightforward path optimization, making it suitable for basic algorithm validation and preliminary performance assessments.

[2] Large-scale

As the number of demand points increases, the computational complexity increases markedly. Path planning subsequently necessitates the management of an extensive array of potential combinations and constraints, thereby serving as a testament to evaluate the algorithm's scalability and expertise in handling large-scale optimization tasks.

4.1 Parameter setting

In a mountainous distribution scenario, a distribution center situated at the base of a valley is tasked with providing goods delivery services to multiple demand points scattered across dissimilar altitudes and geographical locations. Due to the challenging terrain of the mountainous roads, certain demand points are inaccessible by trucks, thereby necessitating the use of drones for delivery. The delivery tasks are accomplished through the collaborative efforts of trucks and drones, which is intended to minimize the overall cost—including fixed cost, travel expenses, and penalties for violating time windows—by optimizing the delivery routes.

Its main parameters are displayed below:

Truck speed: 50 km/h

Drone speed: 60 km/h

Maximum flight distance of drone: 40 km

Maximum payload of drone: 40 kg

Maximum payload of truck: 150 kg

Fixed cost of truck: $ 300 per meters

Fixed cost of drone: $ 50 per unit distance

Truck cost/unit distance: $ 1.5 per unit distance

Drone cost/unit distance: $0.5 per unit distance

Time penalty cost (Early arrival): $0.1 per minute

Time penalty cost (Late arrival): $ 0.5 per minute

Service time: 5 minutes

Battery swapping cost: $ 10 per time

Battery swapping duration: 1 minute

The distance traveled by the truck is multiple of the distance traveled by the drone: 1.3

Proportion of impassable nodes: 5%

Distribution center: Located at the bottom of a valley at an altitude of 200 meters, with coordinates of (0,0).

Demand point:

Point A: At an altitude of 300 meters, coordinates (3,4), the demand is 20 kg, and trucks can access to the demand point.

Point B: At an altitude of 450 meters, coordinates (7,8), the demand is 15 kg, and trucks can access to the demand point.

Point C: At an altitude of 600 meters, coordinates (10, 12), the demand is 10 kg, and trucks cannot access to the demand point.

Point D: At an altitude of 550 meters, coordinates (12, 6), the demand is 25kg, and trucks cannot access to the demand point.

Point E: At an altitude of 400 meters, coordinates (15, 3), the demand is 18 kg, and trucks can access to the demand point.

Point F: At an altitude of 250 meters, coordinates (5, 2), the demand is 12 kg, and trucks can access to the demand point.

Topographic conditions

Road conditions: The mountainous roads are winding, and the actual distance of the truck path is 1.3 times the straight-line distance of the drone flight.

Obstacle ratio: Due to the mountainous terrain, about 5% of the demand points cannot be directly reached by trucks (such as points C and D).

Altitude difference: Drone flight needs to consider the impact of terrain height change on battery energy consumption.

4.2 Test results for small-scale problems

Table 2 lists the input data for small-scale problems, where x and y represent the horizontal and vertical coordinates of randomly generated points. Gurobi's exact solution method, the standard genetic algorithm (GA), and the improved genetic algorithm(GAIP) are adopted to deal with small-scale problems, respectively. Gurobi's exact solution method refers specifically to its capability to find precise solutions. Table 3 systematically compares the solution results for these three approaches, which principally concentrates on fixed cost, battery replacement cost, path cost, time window cost, and the total cost. In particular, both the standard GA and the bettered GA can attain accurate solutions for small-scale problems in these aspects. Given that the small-scale problem involves only 7 demand points, the path planning obtained through these three methods is identical. Consequently, the solution results presented are fully consistent with each other, as shown in Table 3. Here, X and Y represent the horizontal and vertical coordinates of points, severally, Moreover, they denote dimensionless distances.

Table 2 Demand point data for small-scale issues

Node ID	x	y	Demand	Earliest time (seconds)	Latest time (seconds)	Whether the truck is passable
0	35	35		0		1
1	41	49	10	0	974	1
2	35	17	7	0	972	1
3	55	45	13	678	967	1
4	55	20	19	0	801	0
5	15	30	26	0	969	1
6	25	30	3	0	978	1
7	20	50	5	0	968	1

Table 3 Comparison of solution results for small-scale problems

Algorithm	Fixed cost (RMB)	Battery swapping cost (RMB)	Path cost (RMB)	Time window cost (RMB)	Total cost (RMB)
Gurobi	350	10	225.88	58.25	644.14
GA	350	10	225.88	58.25	644.14
GAIP	350	10	225.88	58.25	644.14

4.3 Test results for large-scale problems

As shown in Table 4, the performance comparison between the GA and GAIP algorithms was conducted on a large-scale problem involving 50 client nodes. With regard to the total cost, the GAIP algorithm materialized a 15.5% reduction in comparison with the GA algorithm. The identical fixed cost suggests that both algorithms utilized the same number of vehicles. GAIP exhibited a higher battery replacement cost, indicating a greater reliance on drones. Notably, the path cost of GAIP was strikingly lower than that of GA, lessened by approximately 40.9%, which constitutes the predominant factor behind the overall cost reduction. The slightly higher time window cost tightly correlated with GAIP could be ascribable to the trade-offs made in pursuit of an optimal path.

Table 4 Comparison of GA and GAIP solutions for large-scale problems

Algorithm	Fixed cost (RMB)	Battery swapping cost (RMB)	Path cost (RMB)	Time window cost (RMB)	Total cost (RMB)
GA	1750	130	1628.25	121.39	3629.64
GAIP	1750	110	961.80	244.25	3066.33

Table 4 exhibits the detailed results of the GAIP algorithm on large-scale problems. The algorithm adopts 5 vehicles to serve all demand points. The time window cost of vehicle 4 is 0, indicating that it fully meets the time window requirements of all services. Vehicle 2 and Vehicle 5 have the highest battery replacement cost, which evidently suggests that these two vehicles may have employed more drones for delivery. Vehicle 5 has the highest time window cost, which possibly can be ascribed to the fact that it serves some demand points with tight time windows.

The cost ratio in the table is advantageous for understanding the importance and efficiency of each vehicle in the overall distribution process.

Table 5 Detailed results of GAIP algorithm

Vehicle	Fixed cost (RMB)	Battery swapping cost (RMB)	Path cost (RMB)	Time window cost (RMB)	Total cost (RMB)
1	350	10	240.50	3.45	603.96
2	350	30	218.77	58.08	656.85
3	350	20	178.18	39.82	588.00
4	350	20	132.45	0	502.45
5	350	30	191.98	143.06	715.05

To better distinguish the performance between the GAIP and GA algorithms, the results of path planning for both algorithms on a large-scale problem involving 50 demand points are shown in Figure 1, where the dashed lines represent the drone routes. The distribution center (node O) serves as both the starting and ending points of the truck routes, forming a closed-loop path. It should be noted that the path of the truck mainly presents a smooth and orderly structure, without any transitional intersections or repetitions. The drone mainly serves demand points that are far away from the truck owner's path or comparatively isolated in location, such as nodes 39, 47, and 49.

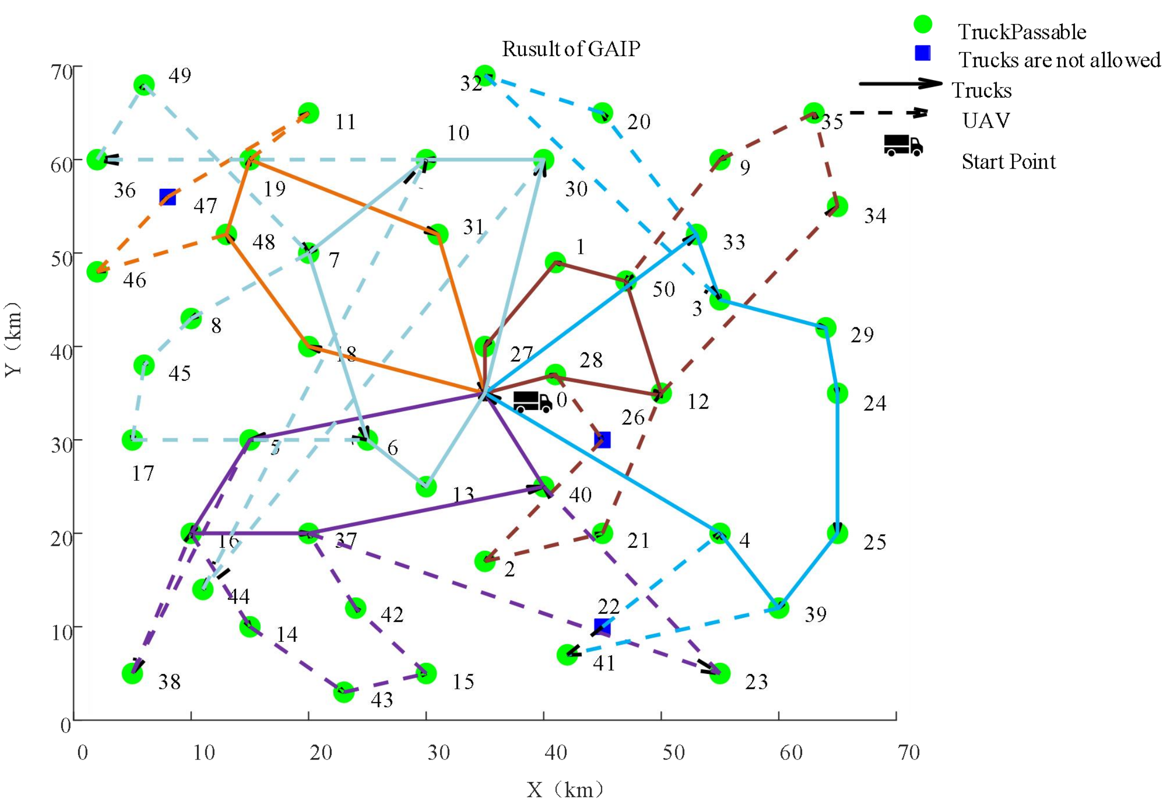

Figure 1 GAIP algorithm path planning results

Figure 2 discribes the path visualization results when adopting GA algorithm to cope with large-scale problems. Compared to the GAIP algorithm, the GA-generated truck routes exhibit a higher degree of redundancy and intersection, resulting in a less structured and less efficient overall path. Additionally, the GA algorithm fails to optimize the sequence of customer visits, leading to unnecessarily prolonged delivery routes and increased total travel distance. The comparison between Figures 1 and 2 clearly highlights the superior performance of the GAIP algorithm in route optimization.

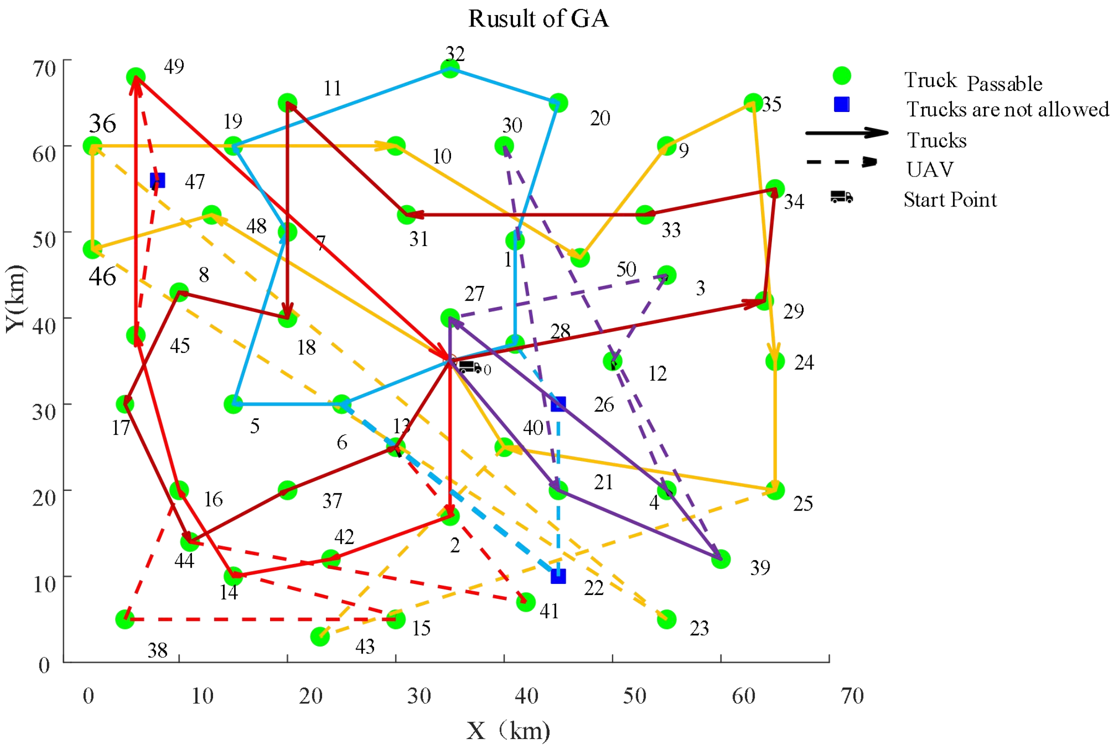

Figure 2 GA algorithm path planning results

Figure 3 describes the convergence curves of the GAIP and GA algorithms. As demonstrated in the above figure 3, the initial solution of the GAIP algorithm is superior to that of the GA algorithm, which primarily due to the superior initialization strategy of GAIP in contrast to GA. On this basis, the convergence rate of the GAIP algorithm is superior to that of the GA algorithm. Furthermore, the objective function experiences a speedy decline within the first 10 generations, suggesting that the GAIP algorithm noticeably surpasses the GA algorithm on problem-solving. In the 15th generation, the objective function tends to be stable, gradually lessening until it converges by the 25th generation. After 50 iterations, the final solution of GAIP is 3300, which is 15% more exceptional than that of the GA algorithm. In comparison, this convergence curve clearly highlights the advantages of the GAIP algorithm in both convergence speed and solution quality, validating the effectiveness of amelioration measures such as adaptive strategies, local search, and ALNS.

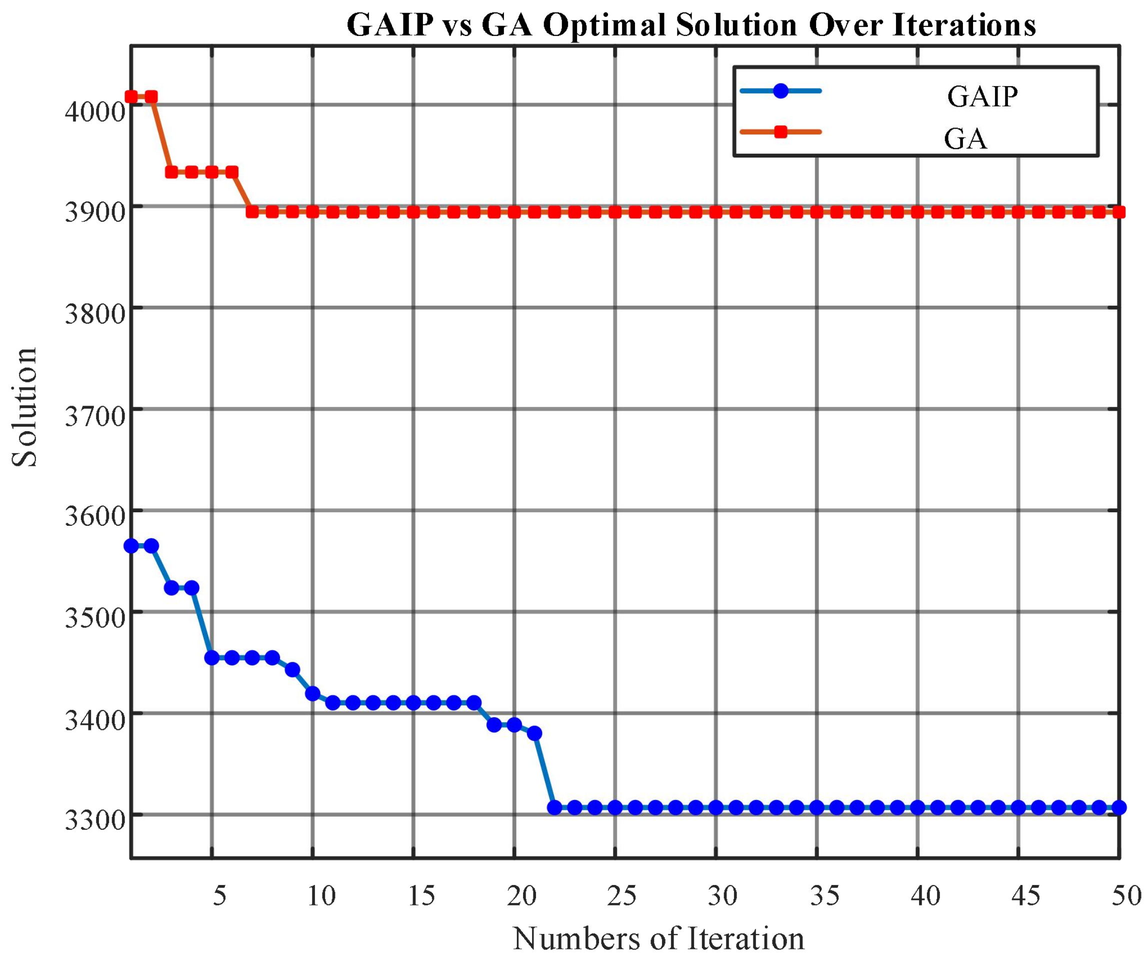

Figure 3 Comparison of convergence curves between GAIP and GA algorithms

5. Conclusions

This paper discusses the collaborative delivery issue between trucks and drones under the constraints such as load capacity, flight mileage, and time windows which align with practical application scenarios. A mathematical model is established to address the path planning problem. In addition, an improved genetic algorithm is developed to solve the model. The improved genetic algorithm incorporates features such as adaptive probabilities for genetic operators, simulated annealing, and ALNS strategy.

The case studies demonstrate that the collaborative delivery mode can decrease average cost by 13.81% in contrast to the traditional truck delivery mode, showcasing promising application prospects. The main cost savings originate from the reduction in both the total travel distance and the number of vehicles. The GAIP algorithm demonstrates impressive performance on both small-scale and large-scale problems.

The collaborative delivery mode shows its outstanding performance in dealing with large-scale problems, with GAIP's path cost lower than that of GA by approximately 40.9%, and the total cost lowered by 15.52%. Drones mainly serve demand points that are distant from the trucks' routes or comparatively isolated. The GAIP algorithm makes a good balance between path cost, time windows, and battery swapping cost, offering high-quality solutions.

Notwithstanding the above merits, the following demerits are conspicuous, which can be addressed and refined in future research. With respect to the solution algorithm, given the increased constraints in practical scenarios, the diversity of situations, and the intensity of planning tasks, it's imperative to further elevate the algorithm's performance in regard to convergence speed. On that account, it's advisable and preferable to take into consideration a hybrid algorithm grounded in machine learning, incorporating both cost and service level optimization objectives, to facilitate multi-objective optimization. future studies should focus on the following directions:

(1):Dynamic pricing models for collaborative delivery, ensuring fair and reasonable compensation for crowdsourced drivers while minimizing logistics costs.

(2):AI-driven demand forecasting and real-time fleet optimization, leveraging artificial intelligence to enable adaptive scheduling based on fluctuations in delivery demand.

References

[1] Montemanni, R., Dell'Amico, M., & Corsini, A. (2024). Parallel drone scheduling vehicle routing problems with collective drones. Computers & Operations Research.

[2] Kong, F., Jiang, B., Wang, J. Q., Wang, H., & Song, H. (2024). Collaborative delivery optimization with multiple drones via constrained hybrid pointer network. IEEE Internet of Things Journal.

[3] Bi, Z., Guo, X., Wang, J., Qin, S., & Liu, G. (2024). Truck-drone delivery optimization based on multi-agent reinforcement learning. Drones.

[4] Shu, Z., Li, S., & Li, J. (2023). Truck-and-drone collaborative delivery method based on adaptive simulated annealing algorithm.

[5] Gao, J., Zhen, L., & Wang, S. (2023). Multi-trucks-and-drones cooperative pickup and delivery problem. Transportation Research Part C: *Emerging Technologies. https://doi.org/10.1016/j.trc.2023.104407*

[6] Liu Tingcheng , Yu Qing.(2023). " Transport and Postal Integration+ " Assists Rural Revitalizatio[N]

[7] Masson R, Trentini A, LehUéDé F, et al. Optimization of a city logistics transportation system with mixed passengers and goods [J]. EURO Journal on Transportation and Logistics, 2017, 6(1): 81-109.

[8] Ghilas V, Demir E, Van Woensel T. An adaptive large neighborhood search heuristic for the Pickup and Delivery Problem with Time Windows and Scheduled Lines [J]. Computers & operations research, 2016, 72: 12-30.

[9] Demir, Emrah, Woensel V, et al. A scenario-based planning for the pickup and delivery problem with time windows scheduled lines and stochastic demands [J]. Transportation scheduled lines and stochastic demands [J]. Transportation.

[10] Yunzhu, Yang Zhongzhen. Optimization of express distribution by cooperatively using private trucks and buses[J]. Journal of Traffic and Transportation Engineering, 2017, 17(06): 97-103.

[11] KıZıL K U, YıLDıZ B. Public transport-based crowd-shipping with backup transfers [J]. Transportation Science, 2023, 57(1): 174-96.

[12] Zhou Yue. The Research on Vehicle Routing Problem Combinedwith Crowdsourcing under Dynamie Demand [D]. Wu Han: Huazhong University of Science and Technology, 2019..

[13] Cui Yong. Research on vehicle routing problem with pickupand delivery in crowdsourcing mode [D]. An Hui: Hefei University of Technology, 2020.

[14] Archetti C, Savelsbergh M, Speranza G. The Vehicle Routing Problem with Occasional Drivers [J]. European Journal of Operational Research, 2016, 254(2): 472-480.:

[15] Pugliese L D P, Ferone D, Festa P, et al. Combining variable neighborhood search and machine learning to solve the vehicle routing problem with crowd-shipping [J]. Optimization Letters, 2022,17(9) 1981-2003.

[16] Macrina G, Di Puglia Pugliese L, Guerriero F, et al. The vehicle routing problem with occasional drivers and time windows[C]// Proc of the Optimization and Decision Science: Methodologies and Applications. Cham: Springer, 2017:577-587.:

[17] Vincent F Y, Jodiawan P, Redi A P. Crowd-shipping problem with time windows, transshipment nodes, and delivery options [J]. Transportation Research Part E: Logistics and Transportation Review, 2022, 157:102545.

[18] Macrina G, Pugliese L D P, Guerriero F. A Variable Neighborhood Search for the Vehicle Routing Problem with Occasional Drivers and Time Windows[C]//Proc of the ICORES. Cham: Springer, 2020:270-277..

[19] Kou X, Zhang Y, Long D, et al. An investigation of multimodal transport for last mile delivery in rural areas [J]. Sustainability, 2022, 14(3): 1291.

[20] Ma Huawei, Ma Kai, Guo Jun. Research on Vehicle Routing Problem with Drones Considering Multi-Delivery[J]. Computer Engineering, 2022, 48(08): 299-305..

[21] Liang Shuang, Chen Yanru, Sun Zhibin. An Adaptive Large Neighborhood Search for the Drone-TruckCollection Point Collaborative Delivery Problem [J]. Industrial Engineering and Management, 2024, 29(01): 119-132.

[22] Yang Shuangpeng, Guo Xiuping, Gao Jiaojiao. Research on Contactless“Truck+Drone”Joint Distribution Problem [J]. Industrial Engineering and Management, 2022, 27(01): 184-194..

[23] Li Y, Dong X, Ding Q, et al. Improved A-STAR Algorithm for Power Line Inspection UAV Path Planning[J].Energies, 2024, 17(21): 5364-5364.

[24] Na Y, Li Y, Chen D, et al.Optimal Energy Consumption Path Planning for Unmanned Aerial Vehicles Based on Improved Particle Swarm Optimization[J].Sustainability, 2023, 15(16):

[25] Yisheng J, Xinchao Z, Junling H. A Novel UAV Path Planning Algorithm Based on Double-Dynamic Biogeography-Based Learning Particle Swarm Optimization[J]. Mobile Information Systems, 2022, 2022

[26] Pan Q, Zheng S, Liu X. Deep-coupling neural network and genetic algorithm based on Sobol-PR for reactor lightweight optimization[J]. Applied Soft Computing, 2024, 167(PC): 112458-112458.

[27] Yao Z, Pan L, Mu B, et al. Multi-objective optimization on bionic fractal structure for heat exchanging of single fluids by genetic algorithm[J]. International Journal of Heat and Fluid Flow, 2024, 11010652-10652.

[28] Ebrahimibagha D, Ray M, Datta S.Informatics-based design of polyaniline-carbon nanotube thermoelectric nanocomposite using ANN and GA[J]. Functional Composites and Structures, 2024, 6(4): 045008-045008.

[29] Sun X, Shen W, Fan J, et al. An improved non-dominated sorting genetic algorithm II for distributed heterogeneous hybrid flow-shop scheduling with blocking constraints[J]. Journal of Manufacturing Systems, 2024, 770-1008.

[30] Jiao L Y, Huang L, Xu B, et al.Modeling and balancing of parallel U-shaped assembly line based on Optimized Genetic Algorithms[J]. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 2024, 238(13): 11-2000.

[31] Wang, Z., Huang, Y., Fan, C., Lai, X., & Lu, P. (2024). Optimized Genetic Algorithm based on greedy and simulated annealing ideas for vascular robot ordering strategy. arXiv. https://doi.org/10.48550/arxiv.2403.19484

[32] (2022). Wide-area detection technology based on hybrid neighborhood search and Optimized Genetic Algorithm. https://doi.org/10.1109/tocs56154.2022.10015971

[33] Xia, C., Cheng, F., Li, G., & Wang, A. (2022). Wide-area detection technology based on hybrid neighborhood search and Optimized Genetic Algorithm. https://doi.org/10.1109/TOCS56154.2022.10015971

[34] Абдумўминов, A. A. (2023). Multi-UAV logistics planning problem based on improved genetic simulated annealing algorithm. https://doi.org/10.1007/978-981-19-6613-2_357

[35] Mao, Z., Xu, Y., Fang, K., Wang, C., & Huang, D. (2024). An adaptive large neighborhood search algorithm for parallel assembly lines scheduling problem with complex fixture constraints. Computers & Industrial Engineering. https://doi.org/10.1016/j.cie.2024.109900

[36] Jassadapakorn, C., & Chongstitvatana, P. (2011). Self-adaptation mechanism to control the diversity of the population in genetic algorithm. International Journal of Computer Science and Information Technology. https://doi.org/10.5121/IJCSIT.2011.3409

[37] Chatzizyrli E , Afentaki A , Hinkelmann M , et al.Transient opto-thermal simulation analysis and experimental validation of LERP systems[J].Journal of Optics, 2024, 26(12): 125402-125402.

[38] Yaibok C , Suwanno P , Pornbunyanon T , et al.Improving urban intersection safety insights from simulation analysis[J].IATSS Research, 2024, 48(4): 523-536.

[39] Yang M , Lian K , Luo S , et al.Simulation analysis and optimization of pilot type electro-hydraulic proportional directional valve based on multi field coupling [J]. Flow Measurement and Instrumentation, 2024, 100102726-102726.

[40] Olejniczak Lobsien, V. (2023). A new adaptation mechanism of the ALNS algorithm using reinforcement learning. https://doi.org/10.1007/978-3-031-29313-9_1

[41] Wouda, N. A., & Lan, L. (2023). ALNS: A Python implementation of the adaptive large neighborhood search metaheuristic. The Journal of Open Source Software. https://doi.org/10.21105/joss.05028

[42] Ji B , Song Y , Yu S S , et al.A modified adaptive large neighborhood search algorithm for solving the multi-port continuous berth allocation problem with vessel speed optimization[J].Computers & Industrial Engineering, 2024, 198110699-110699.

[43] Qiu A , Yang Z .Variable-depth large neighborhood search algorithm for cable routing in distributed photovoltaic systems[J].Automation in Construction, 2024, 168(PB): 105839-105839.