Research Article

Optimization of Base Location and Patrol Routes for Unmanned Aerial Vehicles in Border Intelligence, Surveillance, and Reconnaissance

Yao Liu,[1] Zhong Liu,[1] Jianmai Shi ⬝,[1] Guohua Wu,[2] and Chao Chen[1]

[1]*Science and Technology on Information Systems Engineering Laboratory, College of System Engineering, National University of Defense Technology, Changsha 410073, China*
[2]*School of Traffic and Transportation Engineering, Central South University, Changsha 410075, China*

Correspondence should be addressed to Jianmai Shi; jianmaishi@gmail.com

The location-routing problem (LRP) of unmanned aerial vehicles (UAV) in border patrol for Intelligence, Surveillance, and Reconnaissance is investigated, where the locations of UAV base stations and the UAV flying routes for visiting the targets in border area are jointly optimized. The capacity of the base station and the endurance of the UAV are considered. A binary integer programming model is developed to formulate the problem, and two heuristic algorithms combined with local search strategies are designed for solving the problem. The experiment design for simulating the distribution of stations and targets in border is proposed for generating random test instances. Also, an example based on the practical border in Guangxi is presented to illustrate the problem and the solution approach. The performance of the two algorithms is analysed and compared through randomly generated instances.

1. Introduction

Border patrol is one of military Intelligence, Surveillance, and Reconnaissance (ISR) missions. In many ISR missions especially monitoring mission, Unmanned Aerial Vehicles (UAVs) have become a natural choice with the deployment of air surveillance technology [1, 2]. In border patrol, some areas located on or inside the borderline which have the high rates of illegal activities would be focused. Thus, it is feasible to deploy UAVs to detect the targets regularly and ferry the information such as images, videos, sensor data, etc. to base stations.

However, patrol mission faces severe challenges with rampant crimes such as smuggling. And an increasingly concern is raised with more diversified forms of crimes. To be able to cope with the complexity and diversity of threats effectively, much higher requirements need to be satisfied to ensure the peace and security near the borderline, such as high patrol efficiency and fast reaction speed. Furthermore, due to the harsh environment and complex geographical

conditions, it is dangerous for border guards to patrol manually. And the widespread use of the camera is also difficult with its high cost. Therefore, it is meaningful to apply the unmanned aerial vehicles (UAVs) to the border patrol.

Even though unmanned patrol is of great significance, there are still some challenges. Initially, the capacity constraints of base stations should be taken into account. To make full use of each established station, it is not practicable for the amount of its UAVs to be under a lower limit, while it is also unreasonable to exceed an upper limit with the limitations of military ranks. Moreover, since more than one base station should be established to ensure the mission complement, both the amount and the location of base stations need to be optimized. Meanwhile, the flight paths of UAVs are also supposed to be jointly planned. These factors are all involved to minimize the total cost, which increases the complexity of the problem.

After considering the above-mentioned factors, this paper provides a solution for border patrol, which is structured as follows: Section 2 presents the literature review, and

Section 3 illustrates the main assumptions and constructs the optimization model. Section 4 introduces two constructive heuristics and the local search improving strategies. The experimental design and computational results are proposed in Section 5. At last, Section 6 presents the conclusion and future works.

2. Literature Review

In regard to location-routing problem, there are many scholars doing the research after Jacobsen and Madsen [3] integrated the study of locations and routing in 1980. Tuzun et al. [4] promoted that LRP is an NP-hard problem. Min et al. [5] proposed a classification of location-routing problems based on facility capacities, vehicle capacities, and so on. Afterwards, LRP with capacity constraints on depots and vehicles was considered more often, called capacitated LRP (CLRP) [6]. Due to the complexity of the problem, a few exact approaches are available, such as branch-and-cut algorithm [7] and column generation approach [8], but can only be used to solve small instances. Therefore, heuristics are designed, which can present an appropriate solution in acceptable running time for the large-scale instances. Barreto et al. [9] proposed a constructive heuristic and employed clustering analysis. There are also many metaheuristics based on neighbourhood search [10]. Greedy randomized adaptive search procedure (GRASP) is designed [11] and even improved with evolutionary local search (ELS) [12]. Hybrid genetic algorithms (GA) [13], simulated annealing heuristic [14], and two-phase hybrid heuristics [15] are also proposed. Nevertheless, only the upper bound of the depot's capacity is considered in most of the current literatures. It is not economical nor appropriate if the depot's capacity occupied actually is too small. Thus, this paper takes the lower bound of the depot's capacity into account to ensure the efficient utilization of the depot's capacity. Besides, most current researches are conducted in the commercial background such as parcel delivery, and the potential depots are located within the region and the customers are usually around the depots. But in some situations, such as border patrol, the potential base stations are always located inside the border, while the targets are located in the border lines.

For the route planning of UAVs, many researches have been carried out. In 2004, Harder et al. [16] put forward the general architecture of UAVs' route problem and designed the components of the architecture. The constraints of ammunition load are first considered by Shetty et al. [17]. To scheme the attack path in the war, the main idea is distributing the attack targets to different UAVs and visiting targets in the sequence of importance-differentiation. Mufalli et al. [18] also take the limitations of drone payload into account and use the column-generated heuristics for both sensor selection and flight path. Besides, multiple vehicles and time windows are solved [19]. As for the military applications, unmanned combat aerial vehicles are applied to destruct predetermined targets with the constraints of munitions [17]. Moreover, Avellar, Gustavo S. C. et al. [20] present a solution for the problem of using a group of unmanned air vehicles (UAVs) equipped with image sensors to gather intelligence information, which the objective is to minimum time coverage of ground areas. Persistent Intelligence, Surveillance, and Reconnaissance (PISR) routing problem is also considered by minimizing the time of delivering the collected data to the control station [2]. Unfortunately, most of these researches are based on the traditional routing problem, with little touch and exploration on the optimization of base location and flight path.

From the current literature, there are a few studies on the location-routing problem for UAVs, with mainly two having investigated it. The first one is that İnci Sarıçiçek et al. [21] studied border patrols of UAVs in Turkey in 2014, which uses a two-stage solution method, leading to ineffectiveness of integrating the base location into path planning. The second one is in 2016. Yakıcı [22] studied the base location and path planning under the background of maritime target reconnaissance. However, the constraints of the amount of UAVs are not contained in his study.

In this paper, both capacity constraints on base stations and endurance limitations on UAVs are taken into account in the location-routing problem. And a programming model and two heuristic algorithms are constructed to solve the problem.

3. Model Formulation

In the border patrol mission, there exist a set of potential base stations and a set of determined patrol targets distributed in the border area. Considering the capacity constraints of bases and the endurance limitations of UAVs, the objective of mission planning is to locate the base stations and plan the UAVs' routes in an effort to minimize the overall cost. For starters, this section presents the assumptions, defines the problem, and builds the mathematical model.

3.1. Problem Assumptions

  (i) Assume that all UAVs are in the same type, which means that they have same fight endurance and flying speed.

 (ii) The flying speed of UAVs is assumed to be constant.

(iii) The UAV must depart from and return to the same base station. The situation of swapping the UAVs between different stations is not considered.

(iv) Each UAV can visit multiple targets, while each target can only be visited by one UAV.

 (v) Patrol cost for each mile is only concerned with UAV's property. Uncertain or unexpected situations, such as weather changes, are not taken into consideration.

3.2. Problem Description.

Figure 1 presents an example of the optimization problem for the UAV base location and patrol routes. In patrol missions, a series of base stations with the corresponding equipment are essential. These stations are established inside the borderline: can be reconstructed with the existing sentry posts for reducing the construction cost or be built near the border to reduce the flight distance of UAVs.
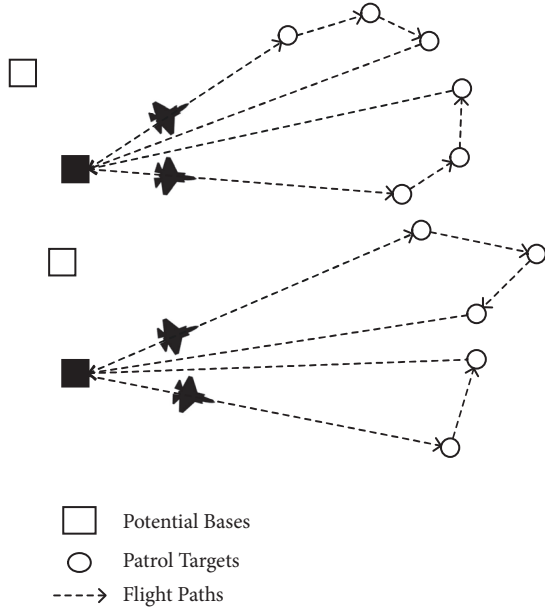
Figure 1: Sketch map for UAV border patrol.

Thus, the set of all potential locations can be constituted as $M = \{1, 2, \ldots, m\}$. Once a base station is determined, the equipped facilities, such as takeoff and recovery device, would generate a fixed establishing cost C_i $(i \in M)$. Furthermore, under the considerations of economic cost and military establishment, there exists a limitation on the capacity of base stations. For each base station, it is not reasonable for the amount of its UAVs to be under the lower limit a^L which the station would be underutilized, nor to exceed the upper limit a^U with the limitations of military ranks.

Another necessary element in this problem is patrol targets, which are some small areas located on or inside the borderline where have the high rates of illegal activities like smuggling. Since the sensor of the drone detects one area at one time, these areas can be simplified into nodes. Then we get a set $N = \{1, 2, \ldots, n\}$. All target points in this set must be detected once in this mission. And the distance d_{ij} between any two targets (or target and base) i, j $(i, j \in M \cup N)$ is known. Moreover, UAVs would be consumed some service time to spy on each target, denoted by s_i $(i \in N)$.

As for UAVs, $V = \{1, 2, \ldots, v\}$ is used to denote the set of all UAVs. Since UAVs require maintenance before and after usage, there is a fixed use cost F_k $(k \in V)$. It could be different for each UAV, but with all UAVs in the same type, it is assumed to be constant to simplify the calculation. Furthermore, knowing that UAVs would fly at a constant speed, the flying time d_{ij} from node i to node two nodes j $(i, j \in M \cup N)$ can be given. Besides, the total time containing flying time and service time cannot exceed the UAVs' maximum flight duration D.

The final objective function is formed via the sum of three primary costs: (1) minimize the base establishing cost; (2) accomplish the patrol mission with as few drones as possible; (3) minimize the flight cost of UAVs.

3.3. Mathematical Model. The parameters and variables used in the model are listed as follows.

Sets

$M := \{1, 2, \ldots, m\}$, set of all potential base stations

$N := \{1, 2, \ldots, n\}$, set of all patrol targets

$V := \{1, 2, \ldots, v\}$, set of all UAVs

Parameters

C_i: fixed establishing cost of base i $(i \in M)$

F_k: fixed use cost of UAV k $(k \in V)$

q: unit patrol cost of UAVs per kilometre

d_{ij}: the distance between two targets (or target and base) i, j $(i, j \in M \cup N)$

t_{ij}: the flying time between two targets (or target and base) i, j $(i, j \in M \cup N)$

s_i: detecting time for target i $(i \in N)$

a^U: upper limit of the capacity of bases

a^L: lower limit of the capacity of bases

D: the UAVs' maximum flight duration

Decision Variables

$x_{ijk} = 1$, if UAV k $(k \in V)$ fly from node i to $j (i, j \in M \cup N)$; 0 otherwise.

$y_i = 1$, if a base station is determined to be built on node i $(i \in M)$; 0 otherwise.

z_{ik}: auxiliary variable for subtour elimination constraints in the route of UAV k $(k \in V)$.

$$\min \quad \sum_{i \in M} C_i y_i + \sum_{i \in M \cup N} \sum_{j \in M \cup N} \sum_{k \in V} q d_{ij} x_{ijk} \tag{1}$$
$$+ \sum_{k \in V} F_k \sum_{i \in M} \sum_{j \in N} x_{ijk}$$

$$\text{St.} \quad \sum_{k \in V} \sum_{i \in M \cup N} x_{ijk} = \sum_{k \in V} \sum_{i \in M \cup N} x_{jik}, \quad j \in M \cup N \tag{2}$$

$$\sum_{k \in V} \sum_{j \in M \cup N} x_{ijk} = 1, \quad i \in N \tag{3}$$

$$\sum_{i \in M \cup N} \sum_{j \in M \cup N} \left(t_{ij} + s_i \right) x_{ijk} \leq D, \quad k \in V \tag{4}$$

$$\sum_{i \in N} \sum_{j \in M} x_{ijk} \leq 1, \quad k \in V \tag{5}$$

$$\sum_{j \in N} \sum_{k \in V} x_{ijk} \leq a^U y_i, \quad i \in M \tag{6}$$

$$\sum_{j \in N} \sum_{k \in V} x_{ijk} \geq a^L y_i, \quad i \in M \tag{7}$$

$$\sum_{i \in N} x_{ijk} = \sum_{i \in N} x_{jik}, \quad j \in M, \ k \in V \tag{8}$$

```
(1)   Allocate targets: allocate every target to its nearest station
(2)   Sort base: sort base stations by the number of allocated targets
(3)   while (unvisited targets) do
(4)       Select the base with the most targets
(5)       Detect targets with Nearest Neighbour
(6)       if (unvisited targets in this base) then
(7)           Reallocate: allocate the remaining targets to the second nearest station
(8)       end if
(9)   end while
(10)  while (bases with the number of UAVs lower than limit) do
(11)      Select the base with the least targets and close the station
(12)      Reallocate: allocate targets of the station to the nearest neighbour station
(13)      Re-detect targets: add the new targets for the neighbour station
(14)  end while
```

ALGORITHM 1: Heuristic based on clustering and nearest point search.

$$z_{ik} - z_{jk} + |N| \, x_{ijk} \leq |N| - 1,$$
$$i, j \in N, \ k \in V \tag{9}$$

$$x_{ijk} \in \{0, 1\}, \quad i, j \in M \cup N, \ k \in V \tag{10}$$

$$y_i \in \{0, 1\}, \quad i \in M \tag{11}$$

$$z_{ik} \geq 0, \quad i \in N, \ k \in V \tag{12}$$

The objective function (1) minimizes the sum of base establishing cost, UAVs' use cost and its patrol cost. Constraint (2) expresses the limitation in flow conservation. Constraint (3) ensures that each target point must be visited and assigned to only one UAV. Constraint (4) restricts the time elapsed in each UAV's flight, containing both flying time and detecting time. Constraint (5) requires that each UAV can be scheduled at most once in one mission planning. Constraints (6) and (7) define the limitations on the capacity of base station, of which the number of equipped UAVs cannot exceed the upper and lower limit. Constraint (8) makes sure that each UAV must turn back to the base station which it departs from. Subtour elimination constrains are expressed in (9). Constraints (10) to (12) declare the variable domains.

4. Algorithms

In this section, two different hybrid heuristics are introduced to construct the feasible solution: nearest point searching algorithm or saving algorithm. Then neighbourhood search serves as an optimization tool to finalize the optimization solution.

4.1. Hybrid Heuristics

4.1.1. Heuristic Based on Clustering and Nearest Point Search (H1). Heuristic based on clustering and nearest point search (H1) utilizes the strategy of Nearest Neighbour. Nearest Neighbour is a well-known constructive search algorithm that is one of the earliest methods proposed for TSP problems [23]. It adopts the principle of selecting the next nearest unvisited node until all nodes have been covered. It runs

fast; however, the optimality of the tours it produces highly depends on the layout of the given nodes.

In H1, every target point is allocated to the nearest base station at first. Then for each station which has assigned targets, the path of UAV is arranged one by one until all assigned targets are detected or the number of UAVs reaches the upper limit of the station. If there are unvisited targets, then allocate the remaining points to the second nearest station. At last, there is a check function to find whether there are some stations that the number of UAVs does not reach lower limit. If so, close the station and reallocate the targets.

The corresponding pseudocode is shown in H1 and the detailed explanation of the heuristic is shown in Algorithm 1.

As shown in pseudocode, every target is allocated to its nearest station with the known positions of the potential base stations and target points (Line (1)). Then sort the selected base stations (Line (2)) and plan the path of UAVs in each station. While detecting the targets, prioritize the base with the most targets (Line (4)). As for the details (Line (5)), launch one UAV at a time, and choose the nearest unvisited target as the next access point. Judge whether the UAV can return back to the original station after visiting the next point. If so, the UAV travels to the next point and keeps detecting, while turning back to the start if not. Repeat the step until all assigned targets are detected or the number of UAVs reaches the upper limit of the station. At this time, check whether there is assigned targets unvisited after sending out all UAVs (Line (6)). If so, allocate the remaining targets to the second nearest station (Line (7)).

After all targets have been detected, find whether there are some stations not having enough numbers of UAVs (Line (10)). If so, give priority to the base with the least targets and close the station (Line (11)). Then allocate targets of the station to the Nearest Neighbour station (Line (12)). After that, redetect targets for the new neighbour station (Line (13)) like what have done in Line (5).

4.1.2. Heuristic Based on Clustering and CW Saving Search (H2). Clarke and Wright proposed the saving algorithm in 1964 [24]. This algorithm provides an easy way to solve the

```
(1)   Allocate targets: allocate every target to its nearest station
(2)   Sort base: sort base stations by the number of allocated targets
(3)   while (unvisited targets) do
(4)       Select the base with the most targets and calculate the saving matrix
(5)       Detect targets with CW
(6)       if (unvisited targets in this base) then
(7)           Reallocate: allocate the remaining targets to the second nearest station
(8)       end if
(9)   end while
(10)  while (bases with the number of UAVs lower than limit) do
(11)      Select the base with the least targets and close the station
(12)      Reallocate: allocate targets of the station to the nearest neighbour station
(13)      Re-detect targets: add the new targets for the neighbour station
(14)  end while
```

ALGORITHM 2: Heuristic based on clustering and CW saving search.

vehicle routing problem, but it considers neither the location problem nor the restriction on the numbers of vehicles.

Similar to H1, heuristic based on clustering and CW saving search (H2) first allocates each target point to its nearest base station and has a check function at last. Different from H1 using Nearest Neighbour, CW saving search algorithm is applied in H2 while planning paths to detect targets. The corresponding pseudocode is shown in H2 and the detailed explanation of the heuristic is shown in Algorithm 2.

Since the overall framework of H2 is similar to that of H1, the same process will not be repeated in this part. After selecting the base with the most targets, calculate the corresponding distance matrix and saving matrix (Line (4)). In terms of the details (Line (5)), launch one UAV at a time. Refer to the saving matrix and merge the target points as many as possible under the limitation of UAVs' flight endurance, which would generate a flight path for a UAV. Repeat the step until all assigned targets are detected or all UAVs have been sent out. Moreover, CW saving search also works in redetecting targets in Line (13).

4.2. Neighbourhood Search Improvement. Although the hybrid heuristics can construct a feasible solution quickly, the solution still has room for improvement. For example, some stations can be replaced by another station or some other stations and some UAV routes still can be optimized. Thus, neighbourhood search is introduced to optimize the solution obtained through the heuristics in Section 4.1 and reduce the overall cost. The framework of neighbourhood search is displayed in the pseudocode shown in Algorithm 3.

Given an initial solution s, the main process iterates over the parameter i until it reaches the preset value of maximum iterations i_{max} (Line (1)). In each interaction, a solution s' would be generated through the operation of closing base stations, which would be described in Section 4.2.1. Then, from Line (5) to Line (16), local search is performed. After initializing the neighbourhood list (Line (5)), the local search will not end until running all neighbourhoods without improvement (Line (7)). Neighbourhood list would be explored exhaustively every time, which returns the best improvement s'

```
(1)   Require: s, i_max
(2)   i ⟵ 1
(3)   while i ≤ i_max do
(4)       s' ⟵ CloseBaseStation(s)
(5)       Initialize Neighborhood List (𝒩)
(6)       k ⟵ 1
(7)       while k ≤ 𝒩 do
(8)           Find the best neighbor s' ∈ 𝒩(s)
(9)           if s' < s then
(10)              s ⟵ s'
(11)              Reinitialize 𝒩
(12)              k ⟵ 1
(13)          else
(14)              k ⟵ k + 1
(15)          end if
(16)      end while
(17)      i ⟵ i + 1
(18)  end while
```

ALGORITHM 3: Neighbourhood search.

(Line (8)). Compared with the current solution s, if s' is better, then s' is the new solution and reinitialize the neighbourhood list, which would restart the counter k (Lines (9)–(12)). The neighbourhoods used in this part are detailed in Sections 4.2.2–4.2.4.

Therefore, combining two heuristics with the neighbourhood research respectively, we can produce two algorithms and generate two optimal solutions.

4.2.1. Operation of Closing the Base Station. In the feasible solutions, there exists a possibility that some base stations have not fully utilized the UAVs. Therefore, it might useful to reduce the establishing cost through shutting down some stations. As Figure 2 shows, just two UAVs are launched for both station A and station B, which have not dispatched all UAVs. At this time, we can try to close one of these two stations and find whether these targets can be detected by the UAVs from only one station. If so, then station B can
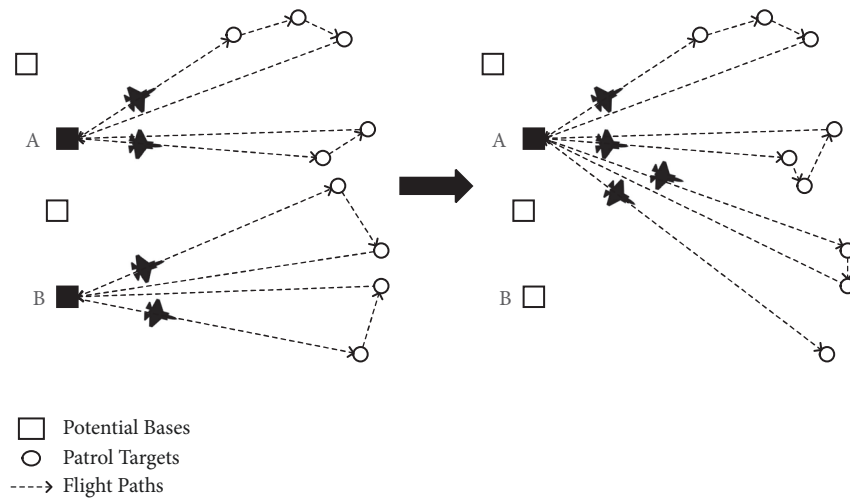
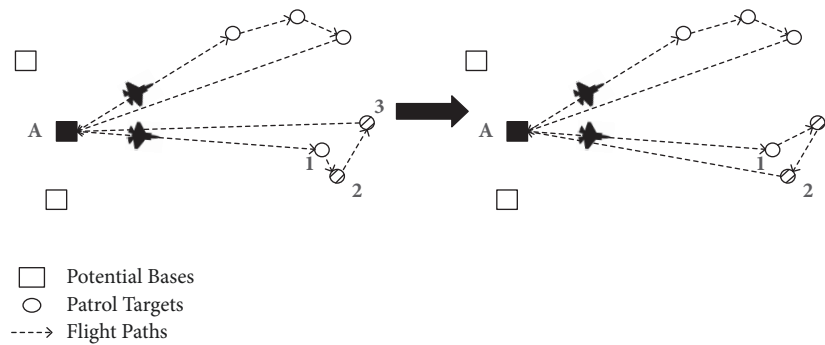FIGURE 2: The operation of closing the base station.
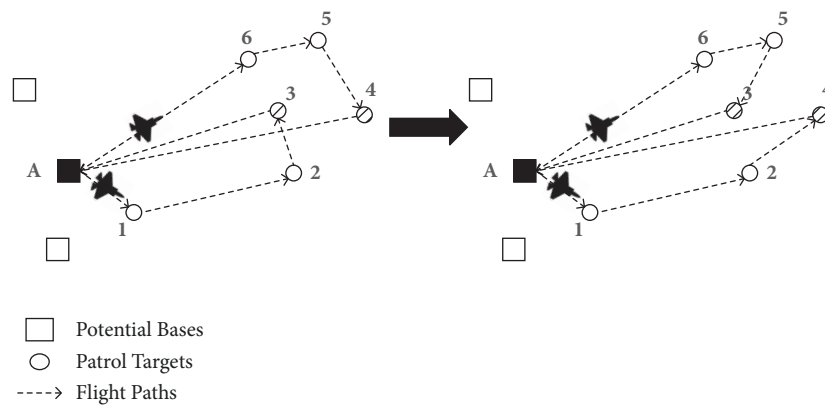

FIGURE 3: The operation of Opt2.

FIGURE 4: The operation of exchanging targets.

be closed. Furthermore, after closing the base station, the flight paths in station A would be rearranged. To save the calculation time, Nearest Neighbourhood algorithm is used, which is displayed in Figure 2.

4.2.2. Neighbourhood 1: Opt2. This neighbourhood relocates two targets on one flight path in a tentative solution. This operation is mainly meant to reduce the cross-over routes or

overlapping routes. In Figure 3, the initial route is $1 \longrightarrow 2 \longrightarrow 3$. Then points 2 and 3 are selected and their positions are swapped to see whether there is a better solution.

4.2.3. Neighbourhood 2: Exchange Targets. This neighbourhood is set to swap a target with another one which locates on another path in a tentative solution. The two paths can belong to the same station or two adjacent stations. Figure 4
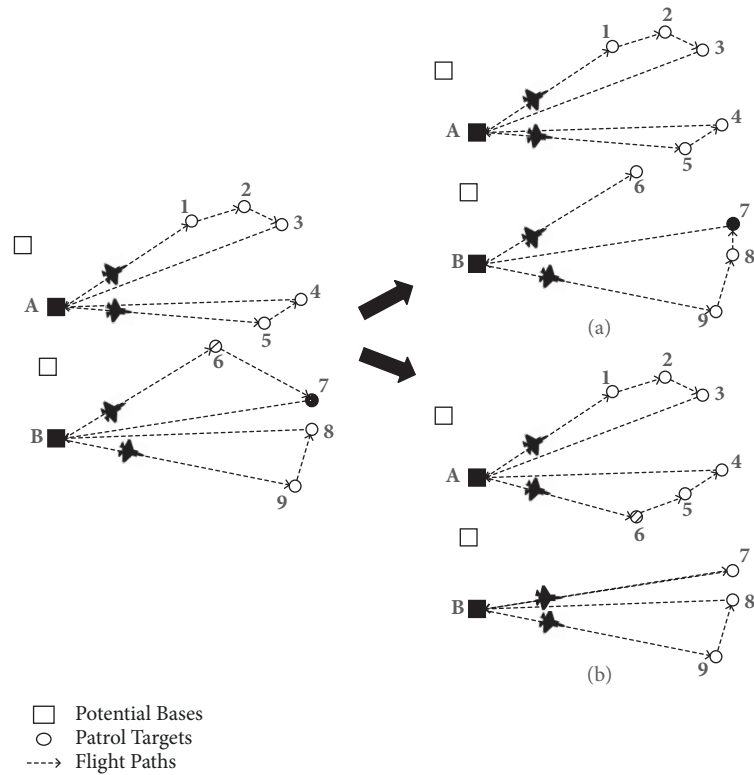
FIGURE 5: The operation of insertion.

Potential Bases
Patrol Targets
Flight Paths

presents the situation that two targets are in different paths of one station. After exchanging the points 3 and 4, the flight distance of both two flight paths would be decreased, which could help lower the flight cost.

4.2.4. Neighbourhood 3: Insertion. This neighbourhood removes a target and reinserts it in other position in a tentative solution, which may change the owner base station of the target. Figure 5(a) illustrates a relatively straightforward move in one station. Additionally, the path represented by Figure 5(b) relocates the target 6 from station B to station A.

5. Experiment Design and Results

In this section, experiments are designed based on actual characteristics of border patrol and two algorithms are tested with the constructed cases.

5.1. Experiment Design. When designing the experiment, the particularity of border patrol should be taken into account. Since the detection goal is part of borderline, the mission area should be set as an irregular strip area. Thus, as displayed in Figure 6, it is assumed that the detection area is a rectangle with aspect ratio of 2 : 1.

In the mission planning, the target points should be located on the borderline while the base stations are inside the borderline. It means that there would be a boundary between potential bases and target points, which is different from the cases in delivery system. Therefore, a random curve is first
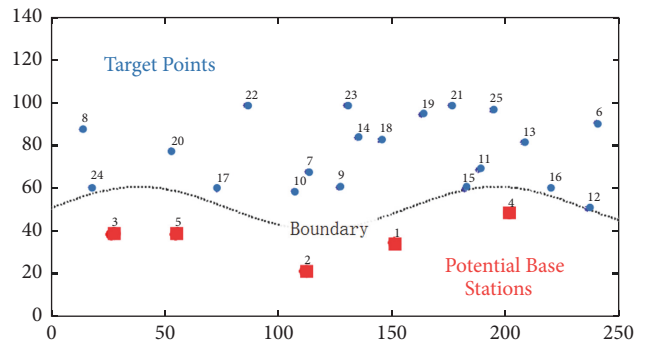

FIGURE 6: Experiment case generated under limited conditions.

generated as a boundary (note: this boundary differs from the borderline; the border is in the area of target points). Then base stations and target points are separately generated on both sides of the boundary, which is the dotted line in Figure 6. Five potential base stations are below the boundary line while twenty target points are above it.

As the target points are abstracted from a small area on the borderline, they should not be too close to each other; otherwise two points could be merged into one node. It is same for the potential stations. It would be impractical to build two stations in a close range. For the purpose of this characteristic, all nodes are generated one by one. Take the base stations as an example, every time a new station is produced, the distance between the station and every existing station would be judged. If the distance is too short, the

TABLE 1: Detailed parameters of two typical UVAs.

Parameters	UAV 1	UAV 2
Fuselage	0.7 meters in length	1.75 meters in length
	2 meters in wingspan	1.75 meters in length
Body Weight	8.5 kg	15 kg
Payload	1.5 kg	4 kg
Flight Speed	80-120 km / h	maximum: 158 km / h
		patrol: 29.5 km / h
Flight Altitude	working: 100-500 meters	working: 300 meters
Mission Radius	70 km	30 km
Flight Endurance	2.5 hours	≥ 4 hours

TABLE 2: Detailed comparison between feasible solution and improved solution of H1-NS.

	Feasible solution by H1			Improved solution after NS	
Base	UAV	Route	Base	UAV	Route
1	1	1⟶9⟶7⟶10⟶1	1	1	1⟶9⟶7⟶10⟶1
	2	1⟶18⟶14⟶23⟶1		2	1⟶14⟶23⟶18⟶19⟶1
4	1	4⟶15⟶11⟶13⟶4	4	1	4⟶15⟶11⟶21⟶4
	2	4⟶16⟶12⟶6⟶4		2	4⟶16⟶12⟶6⟶4
	3	4⟶25⟶21⟶4		3	4⟶13⟶25⟶4
	4	4⟶19⟶4			
5	1	5⟶24⟶8⟶5	5	1	5⟶24⟶8⟶5
	2	5⟶17⟶20⟶5		2	5⟶17⟶22⟶20⟶5
	3	5⟶22⟶5			
	Cost: 372992			Cost: 274386	

position would be abandoned and regenerated until there are enough base stations.

The parameters of UAV mainly refer to the public parameters from two typical UAVs which have been applied to border patrol. The detailed parameter is presented in Table 1. After some proper randomization, the experiment parameters are generated, such as the flight endurance, patrol speed, and so on.

5.2. Experiment Result. With the experiment data generated, two algorithms are tested and compared with each other. For the sake of illustration, H1-NS represents the combination of the heuristic based on clustering and nearest point search (H1) and the neighbourhood search while H2-NS contains the heuristic based on clustering and CW saving search (H2) and the neighbourhood search.

5.2.1. Results of H1-NS on Test Case. Take a small-scale case (5 base stations and 20 target points) generated randomly. Figure 7(a) shows the feasible solution given by H1 while the improved solution is provided after neighbourhood search as Figure 7(b) depicts.

In Figure 7(a), it is can be seen that the performance of Nearest Neighbour is flawed. Obviously, when the UAV departures from Station 5 and then visits Target 17 and Target 20, the rest of energy cannot support it to return to the base if continuing detecting Target 22, which causes one more UAV to be sent to specially visit Target 22.

However, after neighbourhood search, the solution is improved a lot. Some fight paths are merged and adjusted, which reduces the number of UAVs used as well as the flight distance of UAV. For example, the routes 5⟶17⟶20⟶5 and 5⟶22⟶5 are merged into 5⟶17⟶22⟶20⟶5. And the detecting order of Target 14 and 23 is interchanged, decreasing the UAV's redundant flight. Although the number of stations has not changed, the number of drones has decreased after optimization. Only 7 UAVs are used in the final solution, which is two less than that of the initial solution. As displayed in Table 2, the overall costs have been decreased from 372992 to 274386 down by 26 percent, proving the feasibility of H1 and effectiveness of neighbourhood search.

5.2.2. Results of H2-NS on Test Case. The same case is also applied to test H2-NS. The feasible solution given by H2 is presented by Figure 8(a) and the improved after NS in Figure 8(b).
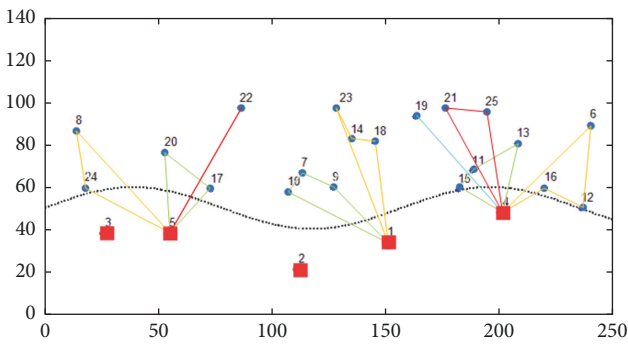
The conclusion can be drawn that the CW saving algorithm has utilized the flight endurance as far as possible, which is fairly obvious in the feasible solution. Compared with the cost of feasible solution in H1, H2 reduces the cost from 372992 to 336496.

After the adjustment of the neighbourhood search, the structure of the solution has changed a lot. Although the number of the UAVs has not decreased, the base station 1 is
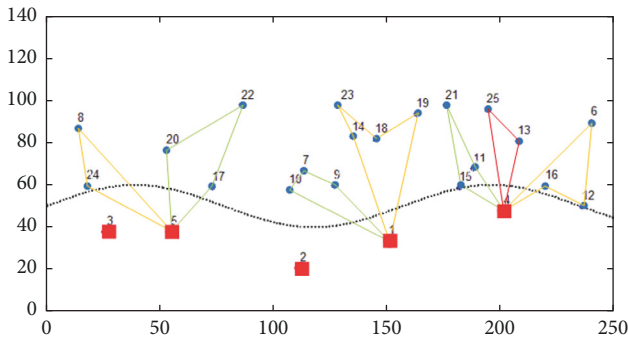
TABLE 3: Detailed comparison between feasible solution and improved solution of H2-NS.

	Feasible solution by H1			Improved solution after NS	
Base	UAV	Route	Base	UAV	Route
1	1	1⟶9⟶10⟶1	1	1	1⟶9⟶7⟶10⟶1
	2	1⟶18⟶23⟶14⟶7⟶1		2	1⟶14⟶23⟶18⟶19⟶1
4	1	4⟶13⟶25⟶21⟶19⟶4	4	1	4⟶11⟶25⟶21⟶19⟶4
	2	4⟶15⟶11⟶4		2	4⟶15⟶18⟶23⟶14⟶4
	3	4⟶16⟶6⟶12⟶4		3	4⟶16⟶13⟶6⟶12⟶4
				4	4⟶9⟶4
5	1	5⟶24⟶8⟶5	5	1	5⟶24⟶8⟶5
	2	5⟶17⟶20⟶22⟶5		2	5⟶17⟶20⟶22⟶5
				3	5⟶7⟶10⟶5
	Cost: 336496			Cost: 189061	

FIGURE 7: Illustration of the results of H1-NS for the small-scale example.

(a) Feasible solution given by H1

(b) Improved solution after NS

FIGURE 8: Illustration of the results of H2-NS for the small-scale example.

(a) Feasible solution given by H1

(b) Improved solution after NS

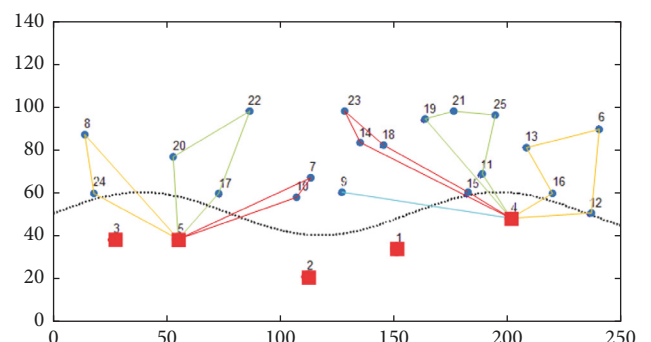closed and replaced by the other two stations, which reduce the base establishing cost.

The detailed comparison is shown in Table 3. The overall costs have been reduced from 336496 to 189061, dropping by 44%. Therefore, H2-NS seems to be more accurate than H1-NS and can solve the problem better.

5.2.3. Comparison. See Table 4.

5.3. Example Based on the Sino-Vietnamese Border. In this section, a practical border is used as the example case and solved by the preceding algorithms.

As is shown in Figure 9, there are about 8 cities and 103 towns bordering in Guangxi Zhuang Autonomous Region. With the development of economy and the implementation of opening-up policy, especially under "Belt and Road" initiative, Guangxi develops an increasingly flourishing border trade. However, smuggling also increases at the same time. Since the border area is mainly delimited by rivers or mountains and has few natural barriers, this part of the border is prone to smuggling which is difficult to monitor. According to the official report, 6726 smuggling cases were seized in Guangxi in 2006. Thus, it is meaningful to apply UAVs on the border patrol, which can improve the patrol

Table 4: Algorithm comparison in cases of three scales.

Case Scale	H1-NS			H2-NS		
	Cost of feasible solution	Cost of improved solution	Time /10^-2 (s)	Cost of feasible solution	Cost of improved solution	Time /10^-2 (s)
	348882	300435	2.325	336890	238473	3.928
	386308	336416	2.392	362091	263009	3.815
	372992	274386	2.463	336496	189061	3.786
	322293	273432	2.313	287293	239058	3.682
5 stations & 20 targets	372327	276420	2.342	325443	179350	3.871
	372439	274322	2.335	336807	238975	5.487
	372369	273543	1.892	337965	240234	3.635
	361322	261822	2.247	348843	250545	3.670
	372538	322555	2.327	336976	237811	3.787
	384262	285167	2.196	336890	237784	3.811
	842265	692451	3.037	793872	544058	6.950
	731291	631663	3.081	681403	531504	6.895
	792129	592620	4.232	754694	505265	6.627
	730762	631054	3.064	670777	471234	7.285
20 stations & 50 targets	730340	630746	3.149	681039	481442	7.002
	856295	656052	3.173	792201	492593	7.867
	807587	608259	3.128	733063	483691	7.419
	804006	654817	3.503	719919	520651	7.432
	828288	628671	3.385	732245	482620	7.494
	804770	655033	2.620	719798	570285	7.242
	1340095	1028130	1.040	1196218	791338	4.590
	1272888	923026	0.993	1219406	819554	4.819
	1314565	952735	1.061	1143538	843628	4.490
	1396086	946150	1.073	1308837	858909	4.931
50 stations & 100 targets	1197487	897623	1.064	1075171	725519	5.414
	1430365	1091920	1.012	1368315	818606	4.525
	1331166	1031168	1.049	1219333	819335	4.508
	1354155	1004220	1.000	1208915	809037	4.550
	1418603	1080944	1.029	1319784	869953	4.446
	1343357	943478	1.023	1256920	807064	4.897

TABLE 5: Comparison between two results of practical cases.

H1-NS			H2-NS		
Cost of feasible solution	Cost of improved solution	Time /10^-2 (s)	Cost of feasible solution	Cost of improved solution	Time /10^-2 (s)
817492	619800	3.091	837997	480309	6.840

TABLE 6: Detailed bases and flight paths of the final solution.

Base	UAV	Route
3	1	3⟶21⟶22⟶3
	2	3⟶23⟶24⟶3
	3	3⟶25⟶27⟶26⟶3
6	1	6⟶28⟶29⟶30⟶6
	2	6⟶31⟶32⟶33⟶34⟶35⟶6
	3	6⟶36⟶37⟶6
8	1	8⟶38⟶39⟶40⟶41⟶8
	2	8⟶42⟶43⟶44⟶45⟶8
11	1	11⟶46⟶47⟶48⟶49⟶50⟶11
	2	11⟶51⟶52⟶53⟶11
15	1	15⟶54⟶55⟶56⟶57⟶15
	2	15⟶58⟶59⟶60⟶15
18	1	18⟶61⟶62⟶18
	2	18⟶63⟶64⟶65⟶67⟶66⟶18
	3	18⟶68⟶70⟶69⟶18

FIGURE 9: The map of Guangxi Zhuang Autonomous Region (the marked red line is the Guangxi section of the Sino-Vietnamese border).

FIGURE 10: Guangxi section of the Sino-Vietnamese border with the ranging tool.

efficiency and reaction speed. Therefore, take the Guangxi section of the border as an example.

As displayed in Figure 10, with the ranging tool of Google Map, the linear distance of this part is about 320 km. Then Paint software is used to discretize this line. Fifty target points are marked on the border and twenty base stations are chosen randomly inside the line. Furthermore, the relative positions of these 70 nodes are obtained via the pixel measurement.

It is assumed that all targets and potential bases are located in the area of 320 kilometres multiplied by 160 kilometres. Mapping these nodes into this area, the distribution map looks like Figure 11. The blue dots are target points and the red blocks are potential base stations.

Two algorithms are applied to solve the practical case. The results are shown in Table 5. Since the result of H2-NS is obviously superior to another one, take the result of H2-NS as the final solution, in which the cost is 480309 and the calculation takes 0.06840 seconds.

The potential base stations are numbered 1 through 20 when the target points are represented by 21 to 70. Then the detailed bases and flight paths can be listed in Table 6 and the corresponding route map is shown in Figure 12. The final result selects 6 bases and launches 15 UAVs.
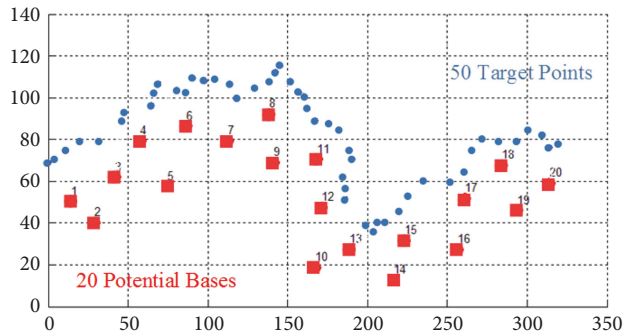
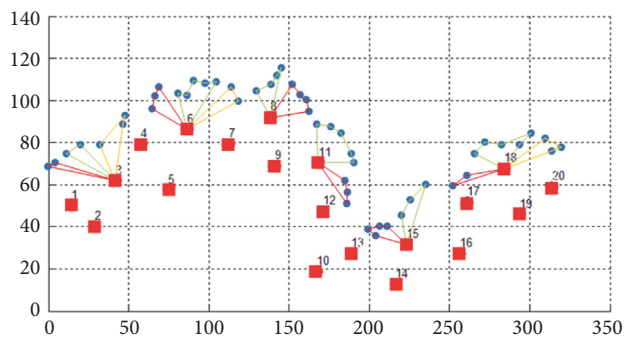FIGURE 11: 50 target points and 20 base stations after discretization.

FIGURE 12: The route map of the final solution.

6. Conclusions

In this paper, considering the background of border patrol for ISR, the capacity constraints of bases are involved in the multidepot location-routing problem. Hence, a modified mathematical model has been presented. The primary objective of this research is to develop an efficient approach for solving this kind of problem. Thus, two hybrid heuristics with neighbourhood search are promoted. One is based on clustering and nearest point search, and the other one is based on clustering and CW saving search.

In addition, experiments have been carried out to compare the performance of two algorithms. It can be addicted that neighbourhood search plays an optimizing role to the solution. And the heuristic based on CW saving search provides a better solution while consuming more time than the other one. Furthermore, an example based on a practical borderline is proposed. Both algorithms are applied to solve the border patrol on the practical borderline and provide the final solution.

Finally, two improvements in solving this kind of problem can be envisaged. First, the current neighbourhood search can be scaled up to find a global optimal solution. Second, more variants of this kind of problem can be exploited. For example, dynamic detecting could be added.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] L. Miao, L. Zhen, S. Wang, W. Lv, and X. Qu, "Unmanned Aerial Vehicle Scheduling Problem for Traffic Monitoring," *Computers & Industrial Engineering*, vol. 122, pp. 15–23, 2018.

[2] S. G. Manyam, S. Rasmussen, D. W. Casbeer, K. Kalyanam, and S. Manickam, "Multi-UAV routing for persistent intelligence surveillance & reconnaissance missions," in *Proceedings of the 2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, pp. 573–580, June 2017.

[3] S. K. Jacobsen and O. B. G. Madsen, "A comparative study of heuristics for a two-level routing-location problem," *European Journal of Operational Research*, vol. 5, no. 6, pp. 378–387, 1980.

[4] D. Tuzun and L. I. Burke, "Two-phase tabu search approach to the location routing problem," *European Journal of Operational Research*, vol. 116, no. 1, pp. 87–99, 1999.

[5] H. Min, V. Jayaraman, and R. Srivastava, "Combined location-routing problems: a synthesis and future research directions," *European Journal of Operational Research*, vol. 108, no. 1, pp. 1–15, 1998.

[6] G. Nagy and S. Salhi, "Location-routing: issues, models and methods," *European Journal of Operational Research*, vol. 177, no. 2, pp. 649–672, 2007.

[7] J.-M. Belenguer, E. Benavent, C. Prins, C. Prodhon, and R. W. Calvoc, "A Branch-and-Cut method for the Capacitated Location-Routing Problem," *Computers & Operations Research*, vol. 38, no. 6, pp. 931–941, 2011.

[8] Z. Akca, R. T. Berger, and T. K. Ralphs, "A branch-and-price algorithm for combined location and routing problems under capacity restrictions," *Operations Research/ Computer Science Interfaces Series*, vol. 47, pp. 309–330, 2009.

[9] S. Barreto, C. Ferreira, J. Paixão, and B. S. Santos, "Using clustering analysis in a capacitated location-routing problem," *European Journal of Operational Research*, vol. 179, no. 3, pp. 968–977, 2007.

[10] C. Prodhon and C. Prins, "A survey of recent research on location-routing problems," *European Journal of Operational Research*, vol. 238, no. 1, pp. 1–17, 2014.

[11] C. Prins, C. Prodhon, and R. Wolfer-Calvo, "Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking," *4OR: A Quarterly Journal of Operations Research*, vol. 4, no. 3, pp. 221–238, 2006.

[12] C. Duhamel, P. Lacomme, C. Prins, and C. Prodhon, "A GRASP×ELS approach for the capacitated location-routing problem," *Computers & Operations Research*, vol. 37, no. 11, pp. 1912–1923, 2010.

[13] R. B. Lopes, C. Ferreira, and B. S. Santos, "A simple and effective evolutionary algorithm for the capacitated location-routing problem," *Computers & Operations Research*, vol. 70, pp. 155–162, 2016.

[14] V. F. Yu, S.-W. Lin, W. Lee, and C.-J. Ting, "A simulated annealing heuristic for the capacitated location routing problem," *Computers & Industrial Engineering*, vol. 58, no. 2, pp. 288–299, 2010.

[15] J. W. Escobar, R. Linfati, and P. Toth, "A two-phase hybrid heuristic algorithm for the capacitated location-routing problem," *Computers & Operations Research*, vol. 40, no. 1, pp. 70–79, 2013.

[16] R. W. Harder, R. R. Hill, and J. T. Moore, "A Java™ universal vehicle router for routing unmanned aerial vehicles," *International Transactions in Operational Research*, vol. 11, no. 3, pp. 259–275, 2004.

[17] V. K. Shetty, M. Sudit, and R. Nagi, "Priority-based assignment and routing of a fleet of unmanned combat aerial vehicles," *Computers & Operations Research*, vol. 35, no. 6, pp. 1813–1828, 2008.

[18] F. Mufalli, R. Batta, and R. Nagi, "Simultaneous sensor selection and routing of unmanned aerial vehicles for complex mission plans," *Computers & Operations Research*, vol. 39, no. 11, pp. 2787–2799, 2012.

[19] 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," *Transportation Research Part C: Emerging Technologies*, vol. 91, pp. 1–14, 2018.

[20] G. S. C. Avellar, G. A. S. Pereira, L. C. A. Pimenta, and P. Iscold, "Multi-UAV routing for area coverage and remote sensing with minimum time," *Sensors*, vol. 15, no. 11, pp. 27783–27803, 2015.

[21] İ. Sarıçiçek and Y. Akkuş, "Unmanned aerial vehicle hub-location and routing for monitoring geographic borders," *Applied Mathematical Modelling: Simulation and Computation for Engineering and Environmental Systems*, vol. 39, no. 14, pp. 3939–3953, 2015.

[22] E. Yakıcı, "Solving location and routing problem for UAVs," *Computers & Industrial Engineering*, vol. 102, pp. 294–301, 2016.

[23] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[24] G. Clarke and J. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *INFORMS*, 1964.