Unmanned aerial vehicle inspection routing and scheduling for engineering management

Lu Zhen [a], Tianyi Fan [a], Gilbert Laporte [b,c], Wen Yi [d*], Zhiyuan Yang [a]

[a] *School of Management, Shanghai University, Shanghai, China*
[b] *Department of Decision Sciences, HEC Montréal, Montréal, Canada*
[c] *School of Management, University of Bath, Bath, United Kingdom*
[d] *Department of Building and Real Estate, The Hong Kong Polytechnic University, Hong Kong, China*

Abstract：Technological advances in unmanned aerial vehicles (UAVs) have enabled the extensive application of UAVs in various industrial domains. For example, UAV-based inspection in engineering management is a more efficient means of searching for hidden dangers in high-risk construction environments than traditional inspections in the artifactual field. Against the above background, this paper investigates the optimization of the UAV inspection routing and scheduling problem. A mixed-integer linear programming model is devised to optimize decisions on the assignment of inspection tasks, the monitoring sequence schedule, and charge decisions. The comprehensive consideration of no-fly zones, monitoring-interval time windows and multiple monitoring rounds distinguish the devised problem from the typical vehicle routing problem and make the mathematical model intractable for a commercial solver in the case of large-scale instances. Thus, a tailored variable neighborhood search metaheuristic is designed to solve the model efficiently. Extensive numerical experiments are conducted to validate the efficiency of the proposed algorithm for large-scale and real-scale instances. In addition, sensitivity experiments and a case study based on an engineering project are conducted to derive insights that will enable an engineering manager to improve the efficiency of inspection works.

Keywords: engineering management, unmanned aerial vehicle, inspection routing and scheduling optimization, MILP model, VNS metaheuristic

1. Introduction

Due to advances in the remote sensing technology of unmanned aerial vehicles (UAVs), UAVs have recently been applied in a wide range of fields, such as package delivery (Sajid et al., 2022; Salama and Srinivas, 2022), military reconnaissance (Manyam et al., 2020; Sundar et al., 2017), post-disaster rescue (Zheng et al., 2019), traffic monitoring (Li et al., 2018), and medical aid (Khan et al., 2021). UAVs are typically small, highly mobile and low cost, which means they are well suited for use in inspection and monitoring missions. Moreover, using UAVs to conduct such missions has the potential not only to

* Corresponding author. Email: yiwen0906@gmail.com (W. Yi)

greatly reduce labor investment, but also to avoid some of the risks inherent to manual inspection processes. However, although there has been extensive research on the routing and scheduling optimization of UAVs performing inspection and monitoring tasks in various fields (Chowdhury et al., 2021; Liu et al.,2022; Shen et al., 2022; Munishkin et al., 2022), there has been little work on UAV-based inspection in engineering management, especially for mega-projects in engineering construction. This paper develops a novel approach to study the optimization of the inspection and monitoring mode of UAVs in an engineering project, aiming to improve the quality and efficiency of engineering management.

In engineering construction, managers commonly need to perform inspection tasks periodically within a certain construction area to judge whether there are hidden dangers—such as unlocked fences, cracks in concrete, or objects that could fall—that could result in major accidents. However, the application of the inspection approach adopted in the traditional artifactual field to engineering mega-projects can result in high labor costs, low inspection efficiency, and slow responses. In addition, it can be dangerous for people to carry out inspection tasks in areas featuring complex facilities or terrain. Fortunately, using UAVs for inspections effectively eliminates the above problems, as UAVs are a non-contact solution (Spencer et al., 2019). In addition, using UAVs in inspections enables not only the collection of real-time images of monitoring sites within a short time, but also the uploading of information to a cloud platform for historical archiving. Moreover, in order to meet periodic inspection demands, managers can use UAVs to visit monitoring sites repeatedly instead of having to manually perform inspections. Most importantly, UAV-based inspection prevents the managers of an inspection task being placed in danger.

Nevertheless, although UAV-based inspections are highly efficient in an engineering project, UAV battery limitations require managers to plan scientific inspection routes for UAVs by taking many factors into account. First, inspection missions cannot be completed by one UAV powered by one battery, owing to the wide scope of inspections required in a construction project. Therefore, scheduling of UAVs of multiple types and their charging demands must be considered. Second, a construction environment varies dynamically, so inspection tasks must be conducted periodically to search for hidden dangers. Therefore, a scientific inspection route for UAVs needs to ensure not only that UAVs return to charging stations before their power runs out, but also that all monitoring sites are visited by UAVs at intervals within a planning horizon (e.g., one day). Third, to prevent collisions between UAVs and engineering facilities, no-fly zones are needed to limit the route accessibility of UAVs traveling from one monitoring site to another. All of the above factors constrain the planning of inspection routing and scheduling of UAVs, and it is difficult to obtain an optimal solution through experience, especially for

a real-scale engineering project.

Against the above background, this paper explores the application of UAV-based inspection in engineering projects by considering the inspection routing and scheduling optimization of UAVs. A mathematical model and a tailored algorithm are designed for planning UAV inspection routes to search for hidden dangers in an engineering project. In contrast with classical vehicle routes for target surveillance or delivery of parcels, where each customer or target is visited only once, UAV inspection routes for an engineering project require each UAV visits monitoring sites in multiple rounds. Thus, consecutive inspection tasks for a given site need to be separated by a time window. In addition, a mathematical model is used to consider the limitations of electric batteries and no-fly zones in the planning of realistic and practical inspection routes that allow UAVs to monitor various sites. The need for comprehensive consideration of the above factors and constraints leads to complex decisions having to be made on monitoring task assignment and monitoring sequence scheduling for UAVs.

The scientific contribution of this paper is as follows. A mixed integer linear programming (MILP) model is formulated to determine inspection routing and scheduling of heterogeneous UAVs, with consideration of periodic inspection demands and charging and no-fly zone constraints. A tailored algorithm based on a variable neighborhood search (VNS) is designed to solve the devised model, which can only be solved for small-scale instances by commercial solvers. In addition, numerical experiments are conducted to confirm the effectiveness of the model and the efficiency of the algorithm. Sensitivity analyses are performed to provide engineering managers with insights into inspection tasks. Furthermore, the Shiziyang Bridge (SZYB) project, a world-class engineering project, is used as an example of the application of the devised mathematical model and designed algorithm to inspection works in a real-life engineering project.

The remainder of this paper is organized as follows. The related work is reviewed in the next section. Problem backgrounds for the inspection routing and scheduling optimization in mega-bridge project are elaborated in Section 3. The mathematical model and the tailored algorithm are described in Section 4 and Section 5, respectively. Experimental results are reported and discussed in Section 6. Section 7 presents the case study on the Shiziyang Bridge. Conclusions are then outlined in the last section.

2. Related works

In recent years, many studies have investigated the application of UAVs in various fields, from the viewpoint of technology frameworks (Perry et al., 2020; Hu et al., 2022; Bailon-Ruiz et al., 2022; Meng et al., 2022) and operational research (Xia et al., 2017; Zheng et al., 2019; Mbiadou Saleu et al., 2022). This section reviews three streams of research. The first stream is the application fields and tasks of

UAVs. The second stream is the optimization of the routing and scheduling of UAVs. The third stream is the optimization of inspection and monitoring tasks performed by UAVs.

The first research stream focuses on the application fields and tasks of UAVs. The tasks assigned to UAVs can be categorized as emergency management, product or package delivery, real-time monitoring and patrolling missions, and military combat missions. Chowdhury et al. (2021) studied UAV inspection routing optimization for post-disaster management. They developed a MILP model to minimize the post-disaster inspection cost for a disaster-affected area. Khan et al. (2021) addressed the vital role of UAVs in the efficient delivery of first aid and medical supplies and demonstrated the value of UAVs in a medical rescue response. Studies on package delivery have mainly focused on the scheduling of trucks and drones for cooperative delivery (Sacramento et al., 2019; Salama and Srinivas, 2022; Zhen et al., 2022). The last-mile delivery problem is well solved by adopting a delivery mode of cooperative trucks and drones, according to the scheduling solution provided by scholars. Shen et al. (2022) conducted a pioneering study on the application of UAVs to the monitoring of vessel air emissions. They jointly optimized routing and scheduling decisions for ship-deployed multiple UAVs in the monitoring of pollution from vessels. Rajan et al. (2022) used a two-stage stochastic programming approach to optimize the routing problem of UAVs in the context of a patrolling mission, revealing the value of UAVs in information collection. In addition, Li et al. (2018) investigated the UAV scheduling problem for the multi-period real-time monitoring of road traffic under demand uncertainty, where UAVs equipped with different imaging sensors are used for capturing multi-period targets images of road traffic. A MILP model is formulated to minimize the operating cost of UAVs. In the field of military combat missions, UAVs commonly perform target surveillance and reconnaissance missions, in which they depart from a depot and travel to targets to collect information, within their power limitations. Liu et al. (2022) designed a hybrid optimization framework, in which the reconnaissance mission planning problem is decomposed into a target selection subproblem and a path planning subproblem. They respectively devised an evolutionary algorithm and a deep reinforcement learning method to solve the target selection subproblem and the path planning subproblem, respectively. However, although UAVs have been widely applied in medical, logistics, urban management and military fields, there has been little discussion on applying UAVs to engineering fields, especially in terms of exploring the value of UAVs in engineering management. The present paper therefore explores the UAV routing and scheduling problem for an engineering project.

The second stream is optimization research on the UAV routing and scheduling problem. Most UAV routing and scheduling problems can be regarded as variants of the classical vehicle routing problem (VRP) or of the traveling salesman problem (TSP). However, decision making in various scenarios with

different targets varies greatly in the planning of the routing and scheduling of UAVs, compared with decision making in the VRP and the TSP. Xia et al. (2017) studied routing planning for a fleet of UAVs collecting information from a set of regions in enemy territory within a given mission time window. The optimization objective was to maximize the total expected reward collected by the fleet. Similarly, Xu et al. (2020) investigated multi-UAV cooperative routing planning within a complex confrontation environment. They developed a multi-constraint objective optimization model to guarantee that each UAV reaches the mission area rapidly, and hence reduce the possibility of a UAV being captured and destroyed. Other research (Qin et al., 2021; Hu et al., 2019) in the military field has considered joint planning of the routing and scheduling of UAVs and vehicles (i.e., the use of mobile charging vehicles or inspection-assisted vehicles). Zhen et al. (2019a) formulated an integer programming model to minimize the total time for completing monitoring tasks in a routing problem by adopting UAVs to monitor a set of areas with different accuracy requirements, thereby optimizing routing and scheduling for monitoring. Wang et al. (2022) investigated the UAV monitor routing problem in the context of unattended offshore oil platforms. They established a multi-objective mathematical model with the shortest flight path and minimum correction times for the intelligent UAVs. The field of logistic delivery involves the scheduling of multiple heterogenous UAVs (Coelho et al., 2017; Mbiadou Saleu et al., 2022), and the joint planning of the routing and scheduling of UAVs and trucks (Moshref-Javadi et al., 2020; Zhen et al., 2022). The main optimization objectives are the sum of the waiting times of customers, the total distance traveled by UAVs, the total delivery time of UAVs and the total cost of cooperative delivery. In the engineering management field, UAVs possess remarkable advantages for performing inspection and monitoring missions. Similar to the above-described applications of UAVs, the routing and scheduling of UAVs in an engineering project must be determined by considering the limitations of batteries and charging stations, the complexity of multiple heterogenous UAVs, and the sets of monitoring areas. In addition, inspection routes must be planned for the collection of images of each monitoring site in multiple rounds at certain intervals, and there must be consideration of terrain obstacles (i.e., construction facilities) that make routes inaccessible. However, no study has comprehensively considered the afore-mentioned factors and a realistic decision demand, so there is a dearth of guidance for making UAV inspection routing and scheduling decisions in engineering projects.

The final stream reviews the optimization works about the inspection and monitoring tasks. Huang et al. (2018) investigated the bridge inspection routing problem. Different from the traditional VRP, the bridge inspection problem also needs to consider the lodging accommodation selection for inspection team to rest overnight since the visited bridges are located in a large geographical area. Therefore, the inspection route not only needs to determine the sequence for visiting bridges, but also the

accommodation selection for lodging. Zhen et al. (2019b) studied an inspection routing problem under a background of coal mine safety personnel in underground mines, which is defined as a multi-depot VRP with time windows. Tasks and the workload balance, multiple inspector-vehicle docking sites and the time windows for inspection tasks are considered in a MILP model. Similar to inspection tasks, routing optimization for patrolling missions (Rajan et al., 2022) could be a novel VRP variant. The distinction lies in the decisions on the routes for patrolling missions needed to visit a set of targets in the initial routes and a set of additional targets in the adjusted routes when information collected by UAVs is incomplete. As for monitoring tasks, the related optimization research (Li et al., 2018; Zhen et al., 2019a; Shen et al., 2022) is commonly dedicated to minimizing the total monitoring time for completing the tasks. Data capacity for storing information, specific monitoring requirements for monitoring sites or areas and power limitation of UAVs are major factors that are considered in their mathematical models. However, there is little optimization research on the routing problem associated with inspecting each monitor site at intervals with a fixed time window. Besides, different from the environment where no construction would obstruct the flying of UAVs, there are many obstacles resulting from engineering construction. Thus, a no-fly zone is novel constraint factor that needs to be considered when determining the UAVs inspection routing in an engineering project.

In summary, this paper explores the UAV inspection routing problem in the field of engineering management and comprehensively considers multiple heterogenous UAVs, the periodic monitoring requirement and no-fly zone constraints in a mathematical model. To the best of our knowledge, although the technologies for object identification and image collection are sufficiently mature for UAV inspection and monitoring tasks in engineering construction (Spencer et al., 2019; Jeong et al., 2022), little effort has been devoted to UAV inspection routing and scheduling optimization in this field. Thus, this paper fills a research gap by providing a practical approach on solving the UAV inspection routing and scheduling problem for engineering projects.

3 Problem description

We consider a set of sites to be periodically inspected by UAVs in a construction area. A fleet of UAVs with various performances, such as different flying velocities, battery capacities and charging times, are required to visit each site at certain intervals to collect images to assist the engineering manager to screen for hidden dangers in a construction environment. The engineering manager therefore needs to assign inspection tasks to each UAV and plan corresponding routes through the monitoring sites. First, the inspection routes should ensure that each monitoring site is visited by a UAV at certain intervals (e.g., one day) within a planning horizon. Second, the inspection routes should guarantee that

the UAVs return to a nearby charging station for charging before powering off. Finally, the inspection routes should prevent UAVs from colliding with construction facilities. Thus, there are no-fly zones that render some routes inaccessible. In addition, to simplify the problem, this paper considers two-dimensional routes of UAVs, so that the flying altitude of a UAV is approximately constant.

Figure 1 depicts an example of UAV inspection routes in which two UAVs monitor four sites in a construction area. Two sites need to be inspected in at least three rounds and two sites need to be inspected in at least two rounds. The UAVs upload the images collected at each monitoring site to a cloud platform in real time. The managers responsible for the inspection missions then record the inspection results in logs (shown in the bottom part of the figure) to trace the construction safety of an area. If a hidden danger is detected at a monitoring site, the managers immediately issue a warning to the supervisors on site and guide them to inspect the situation and take measures to eliminate the hidden danger. Thus, UAV-based inspection not only reduces the labor cost of wide patrolling but also efficiently and dynamically discovers the hidden dangers of an engineering project through the optimization of the inspection routes.

The objective of an engineering manager is to determine the inspection routes of heterogeneous UAVs to minimize the total working time for completing multiple-round inspection missions. The challenges in this decision problem are that (1) battery limitations require a UAV to return to a charging station for recharging in the process of inspection, which complicates decision making on inspection routes; (2) no-fly zones constrain the accessibility of flying routes, which complicates inspection task assignment and route planning; and (3) multiple-round monitoring demand with a time window requires the scheduling of an inspection sequence and time for each monitoring site to satisfy rolling interval constraints, which complicates decision making on routing and scheduling. An optimal solution should comprehensively address the above-mentioned challenges and schedule an appropriate sequence for visiting each monitoring site, in such a way that the time spent charging and waiting is as short as possible. In addition, no-fly zones sometime result in detours that increase the inspection time, so the optimal solution should ensure that UAVs fly from one site to the next (including the charging station) via the shortest route possible.

The following assumptions are made before formulating the mathematical model in Section 4.

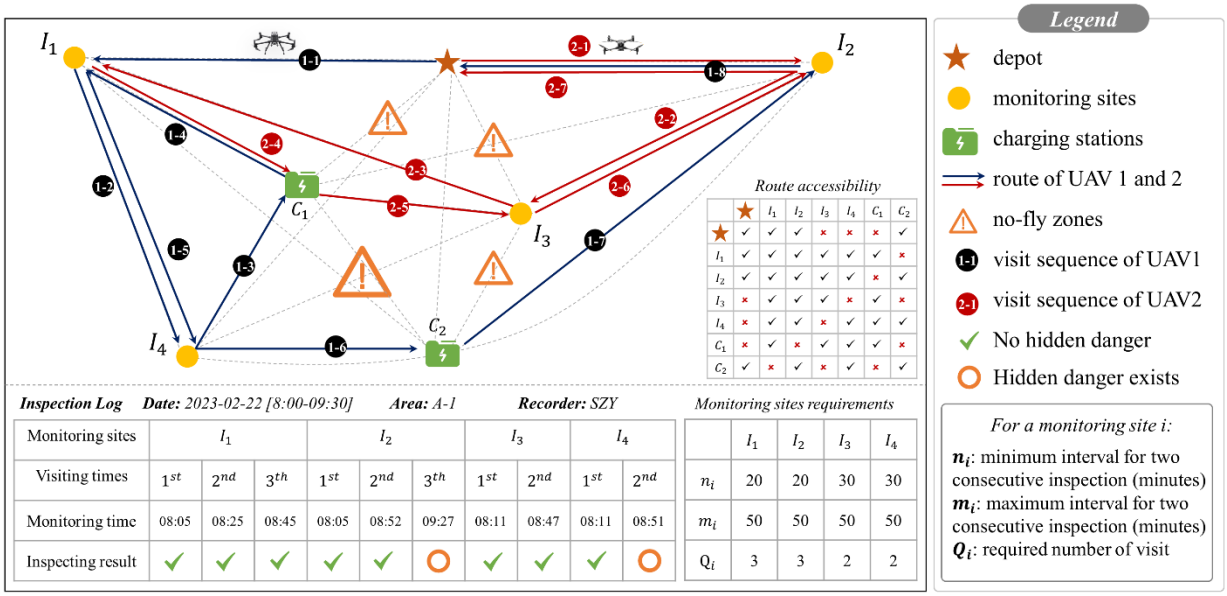
(1) Each UAV performs monitoring missions at an approximately constant altitude. Thus, the vertical movement of the UAVs is not considered.

(2) All of the UAVs start flying from an origin site and fly to a destination site only after completing all of their monitoring missions.

(3) The requirements for each monitoring site, such as the inspection-interval time window, the

required monitoring time and the number of inspection rounds, are known in advance.

(4) The flying distance of any pair of locations and the performances of heterogeneous UAVs are known, as they are deterministic parameters.



Monitoring sites	I_1			I_2			I_3		I_4	
Visiting times	1^{st}	2^{nd}	3^{th}	1^{st}	2^{nd}	3^{th}	1^{st}	2^{nd}	1^{st}	2^{nd}
Monitoring time	08:05	08:25	08:45	08:05	08:52	09:27	08:11	08:47	08:11	08:51
Inspecting result	✓	✓	✓	✓	✓	○	✓	✓	✓	○

	I_1	I_2	I_3	I_4
n_i	20	20	30	30
m_i	50	50	50	50
Q_i	3	3	2	2

Figure 1: An example of two UAVs performing inspection missions of four sites

4 Mathematical model

In this section, an MILP model is formulated for the integrated optimization problem. The objective is to minimize the total time—comprising the flying time, waiting time, inspection time, and charging time—for completing periodic inspection missions. As is usual practice for a transportation network formulation, the MILP model considers a direct connection for each origin-destination pair. The UAVs can fly from one node (where nodes are monitoring sites and charging stations) to another, provided that there is no no-fly zone between the two nodes. The transportation time within a node is set to zero.

4.1 Notations

The parameters and decision variables used in the model are first outlined and then the mathematical model is elaborated. For clarity, we use Roman and Greek letters to denote the parameters and decision variables, respectively.

Indices and sets

K	set of all UAVs, indexed by k, $k \in \{1, \dots, \lvert K \rvert\}$.
I	set of all monitoring sites, indexed by i, $i \in \{1, \dots, \lvert I \rvert\}$.
B	set of all charging stations, indexed by b, $b \in \{1, \dots, \lvert B \rvert\}$.
$e, \dot{e}$	origination and destination for all UAVs.
Q_i	set of all rounds that monitoring site i need to be inspected ($i \in I \cup \{e, \dot{e}\}$), indexed by l, $l \in \{1, \dots, \lvert Q_i \rvert\}, \lvert Q_i \rvert \leq Q$ which is the maximum monitoring rounds.

Parameters

T	duration of the planning period.
r_k	maximum battery capacity of UAV k.
h_k	power consumption per unit distance of UAV k.
v_k	average velocity of UAV k.
c_k	charge power per unit time of UAV k.
$f_{i,j}^1$	set to one if there are no-fly zones between monitoring site i and monitoring site j, otherwise zero ($i, j \in I \bigcup \{e, \dot{e}\}$).
$f_{i,b}^2$	set to one if there are no-fly zones between monitoring site i and charging station b, otherwise zero ($i \in I \bigcup \{e, \dot{e}\}, b \in B$).
$d_{i,j}^1$	distance from monitoring site i to monitoring site j, ($i, j \in I \bigcup \{e, \dot{e}\}$).
$d_{i,b}^2$	distance from monitoring site i to charging station b, ($i \in I \bigcup \{e, \dot{e}\}, b \in B$).
$t_{k,i}$	required time for completing inspection task of site i for UAV k.
$g_{k,i}$	required electricity to complete inspection task of site i for UAV k.
m_i	minimum interval time between two consecutive visits at monitoring site i.
n_i	maximum interval time between two consecutive visits at monitoring site i.
M	a sufficiently large positive number.

Decision variables

$\alpha_{k,i,l}$	binary, equal one if the l^{th} inspection task of monitoring site i is completed by UAV k, otherwise zero.
$\beta_{k,i,l,j,l'}$	binary, equal one if UAV k completes the l'^{th} inspection task of site j immediately after finishing the l^{th} inspection task of monitoring site i, otherwise zero.
$\varphi_{k,i,l,b}$	binary, equal one if UAV k flies to charging station b immediately after completing the l^{th} inspection task of monitoring site i, otherwise zero.
$\lambda_{k,i,l}$	time when UAV k starts the l^{th} inspection task of monitoring site i.
$\theta_{k,i,l}$	remaining electricity when UAV k starts the l^{th} inspection task of monitoring site i.
π_k	total operating time of UAV k, including the flying time, waiting time, inspection time, and charging time.

4.2 Model formulation

Based on the above introduced input parameter and decision variables, we formulate the following mixed integer programming model of this research.

$[M0]$ Minimize $\sum_{k\in K}\pi_k$ (1)

subject to:

$\sum_{k\in K}\alpha_{k,i,l}=1 \qquad \forall i\in I, l\in Q_i$ (2)

$\sum_{j\in I\cup\{\dot{e}\}}\sum_{l'\in Q_j}\beta_{k,i,l,j,l'}=\alpha_{k,i,l}=\sum_{j\in I\cup\{e\}}\sum_{l'\in Q_j}\beta_{k,j,l',i,l} \qquad \forall k\in K, i\in I, l\in Q_i$ (3)

$\sum_{b\in B}\varphi_{k,i,l,b}\le\sum_{j\in I\cup\{\dot{e}\}}\sum_{l'\in Q_j}\beta_{k,i,l,j,l'} \qquad \forall k\in K, i\in I, l\in Q_i$ (4)

$\sum_{j\in I\cup\{\dot{e}\}}\sum_{l'\in Q_j}\beta_{k,e,l,j,l'}=1 \qquad \forall k\in K,\ l\in Q_e$ (5)

$\sum_{i\in I\cup\{e\}}\sum_{l\in Q_i}\beta_{k,i,l,\dot{e},l'}=1 \qquad \forall k\in K, l'\in Q_{\dot{e}}$ (6)

$\beta_{k,i,l,j,l'}\le f_{i,j}^1+M\sum_{b\in B}\varphi_{k,i,l,b} \qquad \forall k\in K, i\in I\cup\{e\}, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (7)

$\beta_{k,i,l,j,l'}\le f_{i,b}^2+M(1-\varphi_{k,i,l,b}) \qquad \forall k\in K, i\in I\cup\{e\}, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (8)

$\beta_{k,i,l,j,l'}\le f_{j,b}^2+M(1-\varphi_{k,i,l,b}) \qquad \forall k\in K, i\in I\cup\{e\}, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (9)

$\lambda_{k,i,l}+t_{k,i}+\frac{d_{i,j}^1}{v_k}\le\lambda_{k,j,l'}+M(1-\beta_{k,i,l,j,l'}) \forall k\in K, i\in I\cup\{e\}, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (10)

$\lambda_{k,i,l}+t_{k,i}+\frac{d_{i,b}^2}{v_k}+\frac{(r_k-\theta_{k,i,l}+h_k d_{i,b}^2)}{c_k}+\frac{d_{b,j}^2}{v_k}\le\lambda_{k,j,l'}+M(2-\varphi_{k,i,l,b}-\beta_{k,i,l,j,l'})$

$\forall k\in K, i\in I\cup\{e\}, b\in B, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (11)

$\lambda_{k',i,l+1}\ge\lambda_{k,i,l}+t_{k,i}+m_i-M(2-\alpha_{k,i,l}-\alpha_{k',i,l+1}) \qquad \forall k,k'\in K, i\in I, l\in Q_i/\{|Q_i|\}$ (12)

$\lambda_{k',i,l+1}\le\lambda_{k,i,l}+t_{k,i}+n_i+M(2-\alpha_{k,i,l}-\alpha_{k',i,l+1}) \qquad \forall k,k'\in K, i\in I, l\in Q_i/\{|Q_i|\}$ (13)

$\theta_{k,j,l'}\le\theta_{k,i,l}-g_{k,i}-h_k d_{i,j}^1+M(\sum_{b\in B}\varphi_{k,i,l,b}+1-\beta_{k,i,l,j,l'})$

$\forall k\in K, i\in I\cup\{e\}, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (14)

$\theta_{k,j,l'}\ge\theta_{k,i,l}-g_{k,i}-h_k d_{i,j}^1-M(\sum_{b\in B}\varphi_{k,i,l,b}+1-\beta_{k,i,l,j,l'})$

$\forall k\in K, i\in I\cup\{e\}, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (15)

$\theta_{k,j,l'}\ge r_k-h_k d_{j,b}^2-M\big(2-\varphi_{k,i,l,b}-\beta_{k,i,l,j,l'}\big)$

$\forall k\in K, i\in I\cup\{e\}, b\in B, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (16)

$\theta_{k,j,l'}\le r_k-h_k d_{j,b}^2+M\big(2-\varphi_{k,i,l,b}-\beta_{k,i,l,j,l'}\big)$

$\forall k\in K, i\in I\cup\{e\}, b\in B, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (17)

$\theta_{k,i,l}\ge g_{k,i}+h_k\min\{\min_{b\in B}d_{i,b}^2, d_{i,\dot{e}}^1\}-M(1-\alpha_{k,i,l})$

$\forall k\in K, i\in I, l\in Q_i$ (18)

$\pi_k=\lambda_{k,\dot{e},0}-\lambda_{k,e,0} \qquad \forall k\in K$ (19)

$\alpha_{k,i,l}\in\{0,1\} \qquad \forall k\in K, i\in I\cup\{e,\dot{e}\}, l\in Q_i$ (20)

$\beta_{k,i,l,j,l'}\in\{0,1\} \qquad \forall k\in K, i\in I\cup\{e\}, j\in I\cup\{\dot{e}\}, l\in Q_i, l'\in Q_j$ (21)

$\varphi_{k,i,l,b}\in\{0,1\} \qquad \forall k\in K, i\in I\cup\{e\}, b\in B, l\in Q_i,$ (22)

$$0 \le \lambda_{k,i,l} \le T \qquad \forall k \in K, i \in I \cup \{e, \dot{e}\}, l \in Q_i \tag{23}$$

$$0 \le \theta_{k,i,l} \le r_k \qquad \forall k \in K, i \in I \cup B \cup \{e, \dot{e}\}, l \in Q_i \tag{24}$$

$$0 \le \pi_k \le T \qquad \forall k \in K. \tag{25}$$

Objective (1) minimizes the total time for all of the UAVs to complete periodic inspection tasks, which comprises the flying time, the waiting time, the inspection time and the charging time. Constraints (2) guarantee that each inspection task at each monitoring site is assigned to one and only one UAV. Constraints (3) are the flow conservation constraints. Constraints (4) mean that if a UAV needs to fly to a charging station for recharging, this operation is scheduled after the completion of an inspection task. Constraints (5) and (6) ensure that each UAV departs from the origin site and arrives at the destination site. Constraints (7) guarantee that UAV k does not fly directly from monitoring site i to monitoring site j if there is a no-fly zone on the route from i to j. Constraints (8) and (9) mean that UAV k does not fly from monitoring site i to charging station b and then fly to monitor site j if there is a no-fly zone on the route from i to b or the route from b to j. Constraints (10) and (11) are fly-time conservation constraints for the UAVs. Constraints (12) and (13) guarantee that the interval between two consecutive inspection tasks at one monitoring site is no shorter than the minimum interval time and no longer that the maximum interval time. Constraints (14) and (15) enforce power conservation when a UAV flies directly from one monitoring site to another monitoring site. Constraints (16) and (17) state that power is conserved when a UAV flies from a charging station to a monitoring site. Constraints (18) ensure that each UAV has sufficient power to travel to the destination site or the charging station closest to its current monitoring site. Constraints (19) compute the total time taken by a UAV to complete all of its inspection tasks. Constraints (20)–(25) define the decision variables.

5 Solution methodology

As is well known, the classical VRP is an NP-hard problem. The present study comprehensively considers the charging decisions, no-fly zone constraints and periodic-interval time windows in a VRP framework. The initial model M0 is therefore difficult to solve directly using CPLEX or other commercial solvers within an acceptable time, even for small-scale instances. Accordingly, this section develops a VNS-based metaheuristic for solving the model efficiently, based on the formulation and embedding of some simplified models.

5.1. Algorithm framework

Our problem can be divided into three subproblems. The first subproblem concerns the assignment of monitoring tasks (e.g., $\alpha_{k,i,l}$). The second subproblem concerns the scheduling of each UAV to carry

out its task (e.g., $\beta_{k,i,l,j,l'}$). The third subproblem concerns the determination of which charging station each UAV should use for recharging after the UAV completes a monitoring task (e.g., $\varphi_{k,i,l,b}$). Since these three subproblems are intertwined, a non-optimized task-assignment plan could result in the monitoring interval constraints (12) and (13) not being satisfied, and a greedy scheduling plan could lead to the powering off of a UAV before it reaches a charging station or its next monitoring site. Therefore, a feasible initial solution must be generated to enable the model to be solved. In addition, effective neighborhood search strategies are needed to obtain improved solutions via iteration of the VNS algorithm. Finally, the charging decisions should be made from a global perspective, so that the performance of an inspection plan can be correctly evaluated.

To address the above-described challenges and solve the devised model, the VNS-based approach was implemented via the following steps.

Step 1: Initialize a feasible solution for conducting a neighborhood search.

Step 1.1: Obtain a feasible solution by iteratively solving the model $M1\_n$.

Step 1.2: Set the best inspection plan (IP^*) and the best objective value denoted $F(IP^*)$, and set the local best inspection plan ($\grave{IP}$) and local best objective value $F(\grave{IP})$.

Step 2: Repeat the following steps until one of the termination conditions is satisfied.

Step 2.1: Explore the neighborhood to find a new solution ($\widetilde{IP}$) based on $\grave{IP}$ by adopting variable neighborhood descent (VND).

Step 2.2: Evaluate the new solution $\widetilde{IP}$ by solving the model $M2$ and obtain a new objective value $F(\widetilde{IP})$.

Step 2.3: If $F(\widetilde{IP}) - F(IP^*) < 0$, set $IP^* = \grave{IP} = \widetilde{IP}$ and $F(IP^*) = F(\widetilde{IP})$.

Step 2.4: Generate a new feasible solution ($\dot{IP}$) adopting the shaking procedure and set $\widetilde{IP} = \dot{IP}$.

Step 3: Output the best inspection plan IP^* and its best objective $F(IP^*)$.

The stopping conditions are that (a) the iteration count exceeds a threshold and (b) $F(IP^*)$ has not been improved by a given threshold.

5.2. Initial solution generation

In this paper, the UAV inspection plan comprehensively considers the constraints of the time interval, power limitations and no-fly zones. It is difficult to generate a feasible solution using heuristic rules without global considerations because the local heuristic strategy struggles to balance the conflict between charging constraints and monitoring time interval constrains. In addition, compared with the classical electric vehicle routing problem (EVRP) and electric vehicle routing problem with time windows (EVRPTW), it is difficult to use the original model $M0$ to obtain a feasible solution within a

short time using the CPLEX solver, given the expansion of the scale of the problem (as demonstrated in Section 6.2). We solve the problem in a rolling optimization manner to generate a feasible solution within a short time. Our core approach is to schedule the inspection routes for monitoring tasks one by one, according to the monitoring rounds (as shown in Figure 2). The procedure of this approach is illustrated in Figure 2.

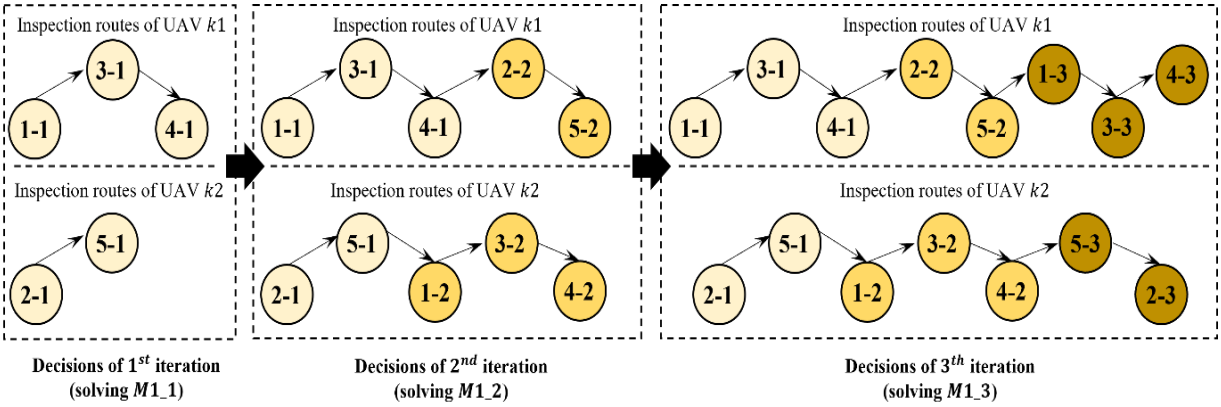


Figure 2: An example of the procedure for generating the initial inspection plan

Each iteration solves a submodel $M1\_n$ $(n = 1, \ldots, \max_{i \in I}\{|Q_i|\})$ listed in Appendix 1. The form of $M1\_n$ is similar to that of the original model $M0$. However, there is only one round of monitoring tasks that need to be conducted by a certain number of UAVs. The first iteration (solving model $M1\_1$) can be regarded as solving a classical EVRP and the subsequent iteration as solving a classical EVRPTW (solving model $M1\_2$ to $M1\_n$) with a determined task assignment and routing sequence. A commercial solver cannot obtain the optimal solution within a short time but can generate a feasible solution within a reasonable time for conducting a neighborhood search.

5.3. Neighborhood structure

In the VNS process, diverse neighborhood structures need to be considered to adequately explore the search space. We thus designed four strategies to find a new solution, with consideration of the problem characteristics of multiple monitoring rounds and no-fly constraints. We denote the j^{th} inspection task for monitoring site i as $i - j$, to clarify the neighborhood structure in the following section. We also consider the constraints of no-fly zones that limit route accessibility. Therefore, we first examine the route accessibility of a new solution. If the inspection route violates constraints (7)–(9), we search the corresponding neighborhood structure repeatedly until we obtain another new solution that does not violate constraints (7)–(9). A global tabu list is used to store the unsuccessful neighborhood searches and newly generated solutions to avoid the need for repeated searches.

5.3.1 Strategy 1: Swapping monitoring tasks

Strategy 1 involves changing the inspection plan by swapping two monitoring tasks. Such a strategy can change a task assignment or monitoring sequence to obtain an improved inspection plan. Figure 3 shows two options for applying Strategy 1 to search for a new solution. Under the first option, strategy 1 changes the solution structure of $\alpha_{k,i,l}$ and $\beta_{k,i,l,j,l'}$. However, under the second option, strategy 1 only changes $\beta_{k,i,l,j,l'}$. Specifically, taking option 1 as an example, $\alpha_{2,5,1}$, $\alpha_{1,3,2}$, $\beta_{2,5,1,4,2}$ and $\beta_{1,1,2,3,2}$ have a value of 1 in the initial solution and a value of 0 after the neighborhood search adopting strategy 1. Subsequently, $\alpha_{1,5,1}$, $\alpha_{2,3,2}$, $\beta_{2,3,2,4,2}$ and $\beta_{1,1,2,5,1}$ equal 1.

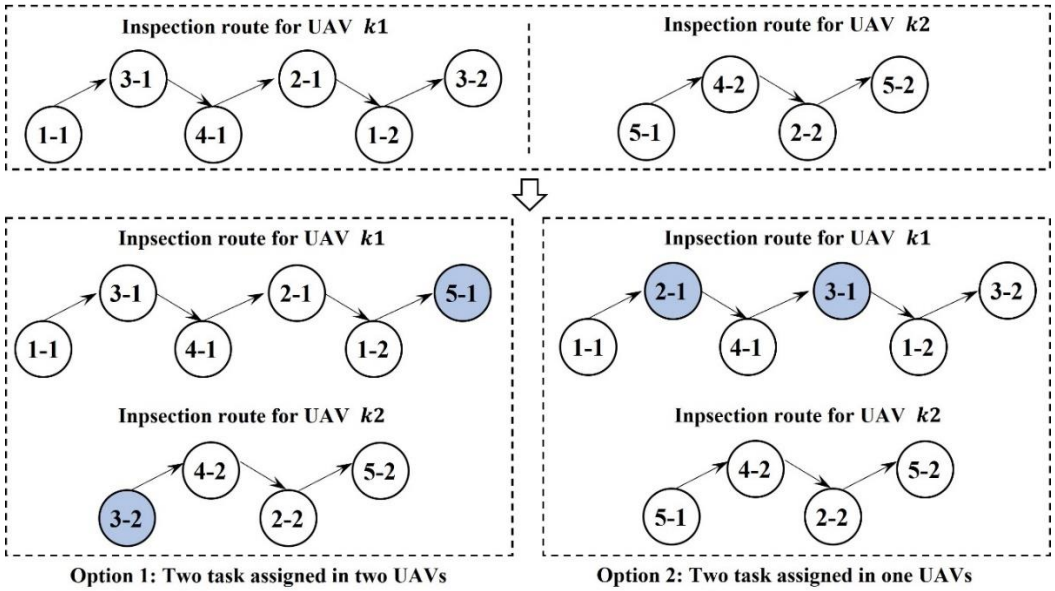


Figure 3: Examples for swapping monitoring tasks neighborhood structure

5.3.2 Strategy 2: Inserting monitoring tasks

In contrast to strategy 1, which keeps a constant total number of tasks assigned to a UAV, strategy 2 changes the number of tasks assigned to a UAV because a better-performing UAV is sometimes able to complete more monitoring tasks than a poorer-performing UAV within the same period. In addition, strategy 2 locally changes the sequence of a UAV to make the inspection route more scientific than it was before. The above two options are illustrated on the right and left of Figure 4, respectively,. When performing strategy 2, we randomly select a monitoring task and insert it in a random location of the original sequence. The specific changes in the decision variables $\alpha_{k,i,l}$ and $\beta_{k,i,l,j,l'}$ during a neighborhood search are similar to those described in Section 5.3.1.

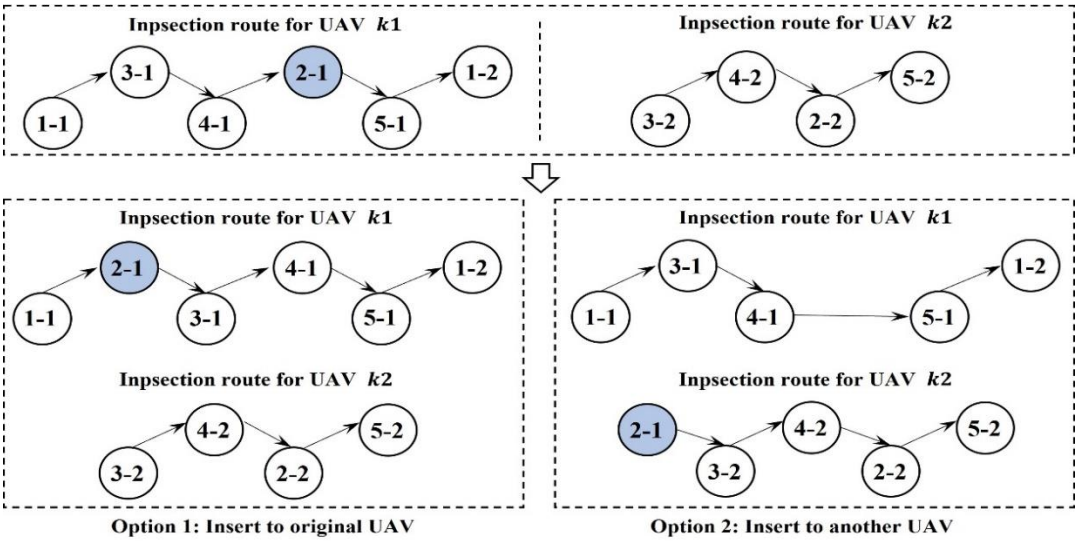


Figure 4: Example for inserting monitoring task neighborhood structure

5.3.3 Strategy 3: Inserting monitoring rounds

Strategy 3 involves changing the solution by inserting monitoring tasks that belong to one monitoring round, thereby changing the task assignment and monitoring sequence from the perspective of the monitoring round. We use a monitoring-interval time window $[n_i, m_i]$ between two consecutive visits to a monitoring site. An optimal inspection plan should ensure that each UAV performs inspection work continuously, i.e., a UAV waits as little as possible before performing its next round of monitoring at a site. Therefore, strategy 3 randomly selects a round of monitoring tasks and reassigns some of the tasks to another UAV. In the example presented in Figure 5, in the initial solution, the monitoring tasks 3-2, 4-2 and 2-2 are assigned to UAV 1 and conducted after task 1-2. After the neighborhood search via strategy 3, these tasks are inserted together into the start of the original sequence of UAV 2. The specific changes in the decision variables $\alpha_{k,i,l}$ and $\beta_{k,i,l,j,l'}$ during the neighborhood search are similar to those described in Section 5.3.1. Strategy 3 finds an improved solution by assigning tasks to a better-performing UAV and by optimizing the work continuity through separating the inspection tasks belonging to different rounds.

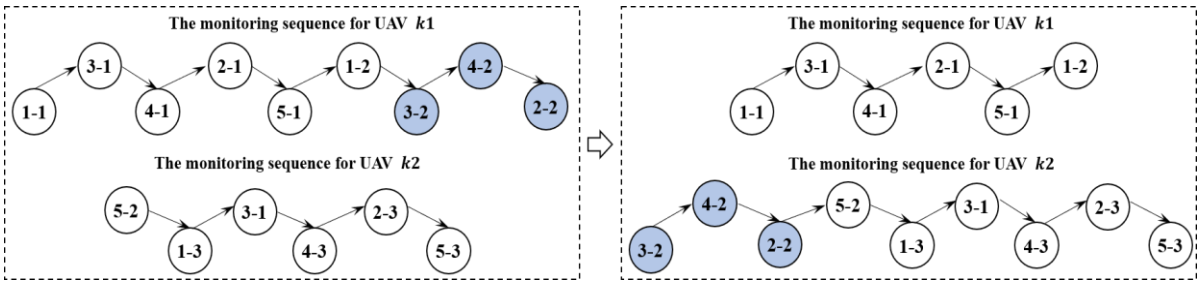


Figure 5: Example for inserting monitoring rounds neighborhood structure

5.3.4 Strategy 4: Swapping inspection routes

Strategy 4 involves changing the solution by swapping inspection routes of two UAVs. Such a strategy is motivated by the consideration of heterogenous UAVs. A better-performing UAV can complete the same monitoring tasks within a shorter time than a poorly performing UAV. Figure 4 shows an example of a neighborhood search adopting strategy 4. The use of strategy 4 affects the solution structure of $\alpha_{k,i,l}$. There are at most $|K|(|K|-1)/2$ neighborhood structures when $|K|$ UAVs perform the monitoring tasks. If all of the neighborhood structures have been searched for an input solution but no new solution has been generated, we use one of the other three strategies to search for a new solution.

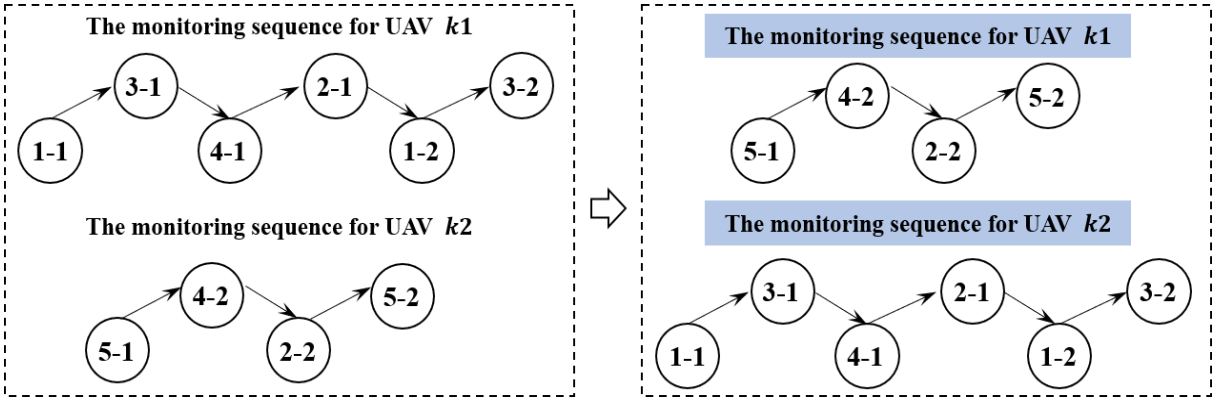


Figure 6: Example for swapping monitoring sequence neighborhood structure

5.4 VND framework

Using the neighborhood structures designed above, we devised a VND framework to search for a improved solution (where the pseudo-code is given in Appendix 3). The search procedure initially explores one neighborhood structure (i.e., it applies strategy 1 to generate a new solution) and then extends to the next neighborhood structure if no better solution is found. Once a better solution is found in the current neighborhood structure, the procedure returns to the first neighborhood and reiterates the above searching procedure until one of the stopping conditions is met.

In the VND process, a simplified submodel M2 is used to evaluate the performance of the new inspection plan, as presented in Appendix 2. The difference between models M2 and M0 is that the decision variables $\alpha_{k,i,l}$ and $\beta_{k,i,l,j,l'}$ are fixed in M2. In this way, most of the constraints, including the binary variables $\alpha_{k,i,l}$ and $\beta_{k,i,l,j,l'}$, are eliminated, and the model retains only the binary variable $\varphi_{k,i,l,b}$ and the continuous variables $\lambda_{k,i,l}, \theta_{k,i,l}, \pi_k$, and can be solved using a commercial solver within a short time, even for the large-scale instance. When the input inspection plan provides an infeasible solution, we set its objective value as M and continue to search for a new solution in the next neighborhood.

5.5 Shaking strategy: regenerating a feasible solution

At the beginning of each iteration, the shaking procedure needs to avoid becoming trapped in a local optimum. Thus, the shaking strategy for generating a new feasible solution is designed as the following heuristic, which is illustrated in Figure 7. In the first step, as far as possible, we assign the first round of inspection of each monitoring site to UAVs in a random but uniform manner to fix $\alpha_{k,i,1}$ for each monitoring site. We then solve model $M1\_1$ given in Appendix 1 with the fixed variables $\alpha_{k,i,1}$ to generate the inspection route for the first round of monitoring tasks. Finally, we generate the inspection routes for the subsequent rounds through the rolling optimization process described in Section 5.2.

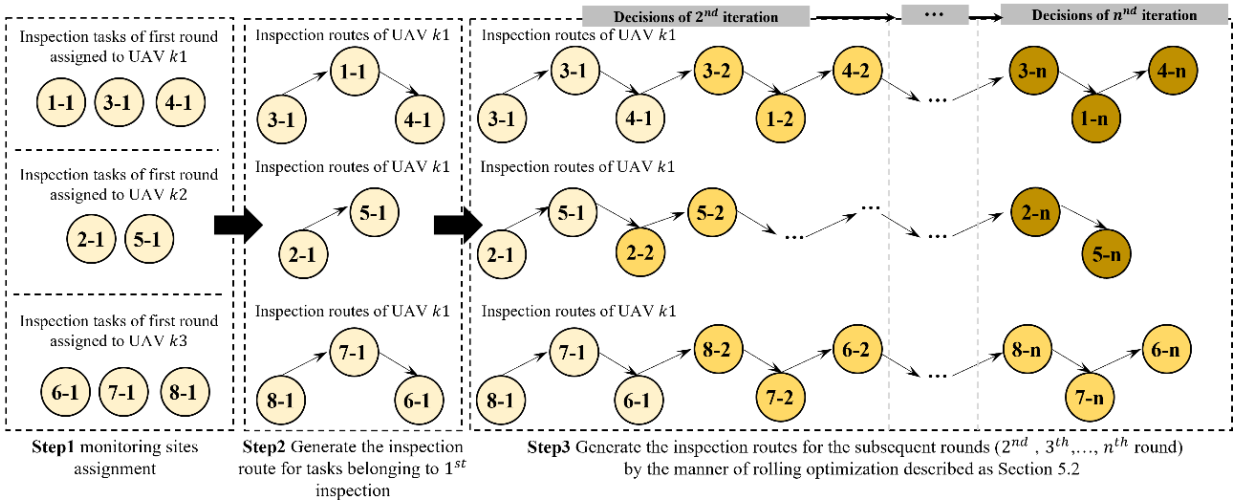


Figure 7: Example for shaking strategy

We take a problem with $|K|$ UAVs and $|I|$ monitoring sites as an example to illustrate the procedure for "assigning tasks in a random but uniform manner." We assign Q tasks of the first -round inspection (e.g., inspection tasks 1-1, 2-1 and 3-1) to UAVs one by one, in decreasing order of UAV velocity, until all of the monitoring tasks are assigned. Here, Q equals $\lceil |I|/|K| \rceil$. Thus, the task assignment variables $\alpha_{k,i,1}$ are fixed by adopting the above heuristic. However, the assignment plan for the following round of inspection tasks may be infeasible when iteratively solving the model $M1\_n(n > 2)$. Thus, we randomly change the number of tasks assigned to a UAV in the range of $[Q - \tau,\ Q + \tau]$, where τ is an integer parameter that is randomly generated in the range of $[1,\ \lceil Q/2 \rceil]$. We then repeat the above procedure until a new feasible solution is generated via such a shaking strategy.

6 Numerical experiments

This section reports on extensive numerical experiments conducted to assess the performance of the VNS algorithm, and sensitivity experiments are conducted to provide managerial insights. All of the experiments were conducted on a workstation equipped with two Intel Xeon Gold 5218R CPUs running at 2.10 GHz with 32 GB of memory on a Windows 10 operating system. The algorithm and models were programmed in C# (Visual studio 2022), and CPLEX 12.6.1 was used as the MILP solver. Considering that daily monitoring sites dynamically vary during construction (i.e., the sites can change daily), the design of a UAV inspection routing and scheduling plan should not be time consuming. The time limit was thus set to 3600 s to satisfy the demands of practical decision planning.

6.1. Generation of test parameters

Instances were generated to simulate a series of scenarios in which UAVs carry out inspection tasks in an engineering project. In practice, inspection tasks are normally performed during periods of construction work (e.g., from 8h to 12h or from 14h to 17h) and monitoring sites can be different in the morning and afternoon (or night). In addition, different construction areas have different types of hidden dangers that need to be monitored by UAVs. Thus, the present study took four hours as a planning horizon ($|T|$) for determining the inspection plan of UAVs. Monitoring sites and charging stations were distributed over a square area of 100 km^2 and the Euclidean distance was calculated as the distance between any two nodes for a UAV. Regarding the performance of UAVs used for inspection, Table 1 gives the parameter settings of UAVs, with reference to the M300 RTK UAV of Dajiang Company (Dji, 2023). The interval time windows of different monitoring sites were set as $[0.5 \times \frac{|T|}{\max\limits_{i \in I}\{|Q_i|\}}, \frac{|T|}{\max\limits_{i \in I}\{|Q_i|\}}]$ to enable UAVs to search for hidden construction dangers at monitoring sites periodically over one planning horizon. The number of no-fly zones affects the route accessibility for UAVs, and it was assumed that 20% of all arcs were inaccessible because of no-fly zones.

Table 1: Parameter setting for heterogenous UAVs

Parameter	Range setting	Unit	Parameter	Range setting	Unit
r_k	[5000,7000]	mA	c_k	[6000,6400]	mA/h
h_k	[180,220]	mA/km	$t_{k,i}$	[0.03,0.05]	h
v_k	[36,72]	km/h	$g_{k,i}$	[200,400]	mA/h

On the basis of the settings given in Table 1, we generated nine instance groups (ISGs) to assess the efficiency of our algorithm. Table 2 gives the key parameter settings of the different ISGs. ISG1, ISG2 and ISG3 were regarded as small-scale instances; ISG4, ISG5 and ISG6 were regarded as medium-scale instances; and ISG7, ISG8 and ISG9 were regarded as large-scale instances.

Table 2: Key parameters settings for different problem instance groups

Group ID	No.of UAVs ($\|K\|$)	No.of monitoring sites ($\|I\|$)	No.of charging stations ($\|B\|$)	No.of maximum monitoring round ($\|Q\|$)	No.of duration of planning horizon ($\|T\|$)
ISG1	2	4	3	2	4
ISG2	2	5	3	2	4
ISG3	2	6	3	3	4
ISG4	3	7	3	3	4
ISG5	3	8	3	3	4
ISG6	4	9	3	4	4
ISG7	5	12	4	4	4
ISG8	5	14	4	4	4
ISG9	6	16	5	4	4

6.2. Investigation of the solution quality of the algorithm

The first series of numerical experiments were conducted on small-scale instances to compare the results obtained using the CPLEX solver, the VNS-based metaheuristic and the initial generation strategy introduced in Section 5.2. Table 3 gives a comparison of the results for the objective value and the computing time, showing that CPLEX obtained an optimal solution only for the first two small-scale instance groups (i.e., ISG1 and ISG2) within 3600 s. In contrast, our algorithm obtained an same solution for ISG1 and ISG2 within 30 s. When CPLEX could not obtain the optimal solution within 3600 s, we compared its upper bound solution value with the solution value of our algorithm. The average difference between two values was −13.86%, and the computing time of the devised algorithm was less than 120 s, which indicates that it obtained a better solution than the CPLEX solver within a short time for the ISG3 problem scale. In addition, the average differences between the solutions of the devised algorithm and initial generation strategy for ISG1, ISG2 and ISG3 were 17.54%, 17.56% and 23.85% respectively. The neighborhood structures described in Section 5.3 were therefore effective in finding improved solutions within the VND search framework.

The second series of numerical experiments were conducted on the medium-scale instances. Table 4 shows that our algorithm obtained a solution far better than that obtained by CPLEX within 3600 s

(with the average difference being –30.74%). With increases in the number of monitoring sites and monitoring rounds, CPLEX could not obtain a feasible solution for ISG5 and ISG6 within 3600 s, whereas the devised algorithm solved the model within 300 s. In addition, the average difference between the solutions of devised algorithm and the initial generation strategy on the medium-scale instances was 19.21%, which confirms the good performance of the VNS-based metaheuristic in searching for better solutions.

The third series of numerical experiments were conducted on the large-scale instances. As the CPLEX solver could not obtain a feasible within a reasonable time on large-scale instances, Table 5 gives the differences between the solutions (i.e., Gap2) obtained via the VNS-based approach and via the initial generation strategy. The average value of Gap2 was 21.87%, which was larger than the value for the medium-scale instances (19.21%) and that for the small-scale instances (19.65%). This indicates that our algorithm was stable and effective in finding the best solution on different instances sizes. Most importantly, the computing times required for solving the nine ISGs were much shorter than 3600 s, indicating that our algorithm can effectively help an engineering manager plan a UAV inspection for application in a dynamically changing engineering schedule.

Table 3: Algorithm performance for small-scale instances

Instance		CPLEX		Strategy	VNS		Gap	
Group	ID	Obj (h)	Time (s)	Obj (h)	Obj (h)	Time (s)	Gap1	Gap2
ISG1	1	1.39	8	1.66	1.39	24	0.00%	20.03%
	2	1.51	7	1.83	1.51	13	0.00%	21.28%
	3	1.45	8	1.75	1.45	13	0.00%	20.57%
	4	1.78	13	1.94	1.78	15	0.00%	9.29%
	5	1.71	9	2.05	1.71	12	0.00%	19.87%
						Average	0.00%	17.54%
ISG2	1	1.59	143	1.82	1.59	12	0.00%	14.80%
	2	1.66	91	1.94	1.66	10	0.00%	17.25%
	3	1.53	294	1.90	1.53	25	0.00%	24.37%
	4	1.61	186	1.79	1.61	13	0.00%	11.51%
	5	1.56	271	1.87	1.56	19	0.00%	19.86%
						Average	0.00%	17.56%
ISG3	1	3.66	>3600	3.82	2.91	30	–20.56%	31.46%
	2	3.85	>3600	3.88	3.23	86	–16.04%	19.93%
	3	3.79	>3600	4.14	3.19	25	–15.72%	29.71%
	4	3.78	>3600	3.72	3.09	46	–18.28%	20.38%
	5	3.91	>3600	3.97	3.37	23	–13.64%	17.75%
						Average	–16.85%	23.85%
Average							–5.62%	19.65%

Note: (1) "Gap1" and "Gap2" represent the optimality gap between the objective values of solutions obtained by CPLEX, the initial generation strategy and the VNS-based solution approach, respectively; (2) ">3600" means that the computing time of obtaining the optimal solution for CPLEX is more than 3600 seconds; (3) "Obj" represents the objective value of proposed model solved by each solution methodology; (4) "ID" represents the different instances for the same instance size.

Table 4: Algorithm performance for medium-scale instances

Instance		CPLEX		Strategy	VNS		Gap	
Group	ID	Obj (h)	Time (s)	Obj (h)	Obj (h)	Time (s)	Gap1	Gap2
ISG4	1	5.53	>3600	4.70	4.23	76	–23.61%	11.16%
	2	4.39	>3600	3.99	3.18	72	–27.51%	25.45%
	3	6.11	>3600	4.55	3.64	74	–40.33%	24.93%
	4	6.27	>3600	4.76	4.32	89	–31.04%	10.10%
	5	4.55	>3600	3.91	3.13	79	–31.18%	24.64%
ISG5	1	——	>3600	7.06	6.16	134	——	14.56%
	2	——	>3600	5.23	4.65	187	——	12.35%
	3	——	>3600	5.39	4.80	177	——	12.23%
	4	——	>3600	7.13	6.28	74	——	13.50%
	5	——	>3600	6.35	4.99	174	——	27.22%
ISG6	1	——	>3600	8.33	6.50	288	——	28.27%
	2	——	>3600	5.18	4.40	197	——	17.75%
	3	——	>3600	7.85	6.72	166	——	16.90%
	4	——	>3600	8.56	6.86	155	——	24.87%
	5	——	>3600	5.59	4.50	218	——	24.20%
						Average	–30.74%	19.21%

Note: (1) A dash means the computation time of CPLEX for obtaining a feasible solution exceeds 3600 seconds; (2) "Gap1" and "Gap2" represent the optimality gap between the objective values of solutions obtained by CPLEX, the initial generation strategy and the VNS-based solution approach, respectively; (3) ">3600" means that the computing time of obtaining the optimal solution for CPLEX is more than 3600 seconds. (4) "Obj" represents the objective value of proposed model solved by each solution methodology; (5) "ID" represents the different instances for the same instance size.

Table 5: Algorithm performance for large-scale instances

Instance		Strategy	VNS		Gap
Group	ID	Obj (h)	Obj (h)	Time (s)	Gap2
ISG7	1	10.41	8.24	947	26.45%
	2	9.91	7.73	731	28.19%
	3	10.92	8.69	553	25.61%
	4	12.92	10.60	617	21.87%
	5	9.69	8.39	1029	15.46%
ISG8	1	12.94	10.72	1872	20.63%
	2	10.69	8.62	1495	24.05%
	3	11.16	9.40	878	18.74%
	4	10.47	8.36	1343	25.29%
	5	10.60	8.96	845	18.31%
ISG9	1	13.56	10.77	1684	25.91%
	2	14.32	11.64	1482	23.06%
	3	12.56	10.68	1194	9.44%
	4	13.25	11.95	1560	10.89%
	5	14.78	11.73	2548	25.97%
				Average	21.87%

Note: (1) "Gap2" represents the gap between the solutions obtained by optimality and the initial generation strategy; (2) "Obj" represents the objective value of proposed model solved by each solution methodology; (3) "ID" represents the different instances for the same instance size.

6.3 Derivation of managerial insights from sensitivity analyses

This subsection reports on sensitivity analyses performed to derive managerial insights into optimizing planning of UAV inspections. The sensitivity analyses were conducted from two perspectives. The first was the number of UAVs and the second was the speed of the UAVs.

6.3.1 Sensitivity analysis on the number of UAVs used for the inspections

The number of UAVs used for inspections affects the total working time of UAVs. However, too many UAVs could result in low efficiency (e.g., only a few inspection tasks being assigned to a UAV), whereas too few UAVs could require a long time to complete all of the inspection tasks (e.g., the UAVs need to recharge frequently during the process of inspection). It is thus necessary for an engineering manager to determine the suitable number of UAVs to use for conducting inspection tasks. We took the small-scale instances (ISG1, ISG2 and ISG3) as examples to investigate the relationship between the number of UAVs and the total working time. In addition, we excluded the effect of the UAV performance (e.g., the velocity or battery capacity) by assuming that the performance of the UAVs was homogenous when changing the number of UAVs. Figure 8 shows the effect of the number of UAVs used for the inspections on the total working time. It can be seen that two UAVs was suitable for ISG1 and ISG2 and five UAVs was suitable for ISG3.

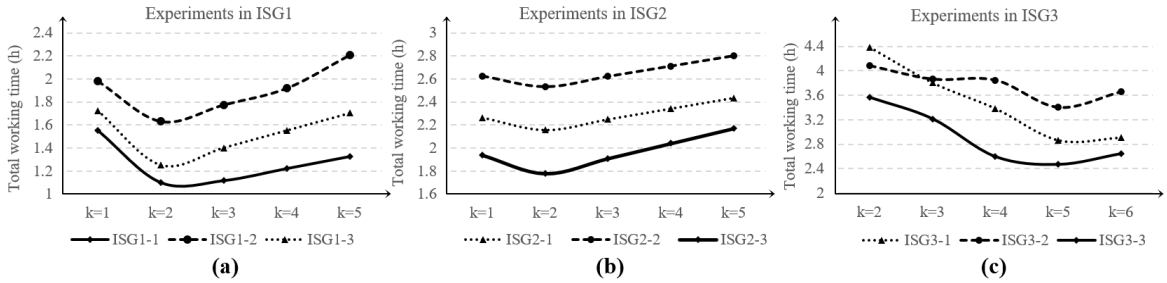


Figure 8: Sensitivity analysis on the number of UAVs for the inspections

The above results indicate that on the scale of ISG1 and ISG2, two UAVs were sufficient to complete all of the inspection tasks. Thus, investing in more UAVs to conduct the same number of inspection tasks would increase the total working time for no gain, as this would simply increase the amount of unnecessary flying time and prolong the waiting time. However, the maximum number of monitoring rounds and number of monitoring sites are both much larger for ISG3 than for ISG1 and ISG2, and the UAVs thus need to perform more inspection tasks at the same number of monitoring sites for ISG3 than for ISG1 and ISG2. Thus, increasing the number of UAVs for inspection tasks for ISG3 would reduce the total working time for completing all of the inspection tasks. These results reveal to an engineering manager that it is necessary to consider an appropriate ratio of the number of UAVs for inspection to the total number of inspection tasks, and that this ratio is affected by the performance of UAVs and the range of inspection areas.

6.3.2 Sensitivity analysis of the variation in UAV performance

The performance of UAVs used for the inspections greatly affects their total working time. This subsection discusses the effect of variations in different parameters of UAVs on their total working time, thereby showing each parameter's importance in terms of reducing the total working time. The battery power (r_k), the average velocity (v_k) and the power consumption per unit distance (h_k) were the three parameters considered in the sensitivity experiment. To explore the importance of variations in these three parameters, we set a benchmark for each and varied the parameters from their benchmark settings by –20% to 20%. In addition, it is intuitive that compared with a UAV traveling at a lower speed, a UAV traveling at a higher speed requires more power, which means that the power consumption per distance increases. We thus made the variation in h_k the same as that in v_k when changing the average velocity of a UAV. The above three parameters were varied on the scales of ISG1, ISG2 and ISG3, and the results are shown in Figure 9.

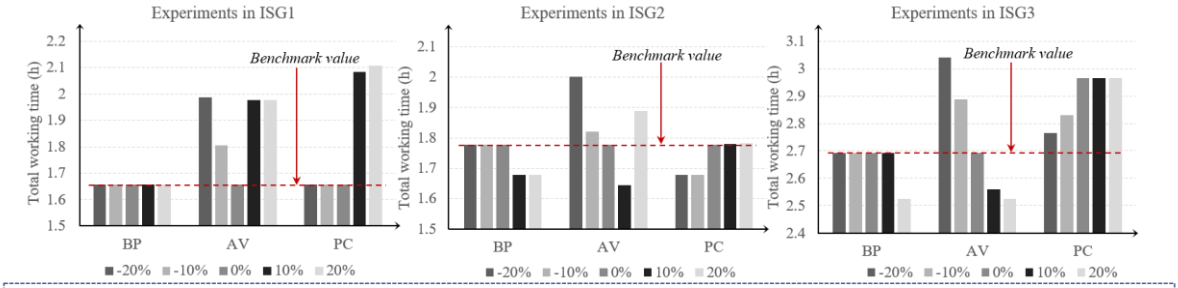


Figure 9: Sensitivity analysis on the variation of UAVs' performance

These experiments provided interesting insights. First, increasing only the battery power above a threshold decreased the total working time, owing to a decrease in the charging time. Second, increasing the average velocity of the UAVs did not always reduce the total working time when there were a low number of monitoring tasks and a low maximum number of monitoring rounds (Figures 9a and 9b). This is because the power consumption per unit distance typically increases with the average velocity for the current technology standard, resulting in an increase in the time spent charging or waiting. However, increasing the velocity of UAVs was an effective way to shorten the total working time when there were many inspection tasks (at least up to the scale of ISG3). Finally, an increase in power consumption per unit distance resulted in an increase in the working time required to completing the inspection tasks. This is because an increases in power consumption per unit distance results in a decrease in the distance that a UAV can fly within one charging cycle. The above analysis should be helpful to a project manager purchasing UAVs for inspection tasks in an engineering project, as it shows that to effectively improve the efficiency of inspection works, UAV velocity should be the primary consideration, followed by UAV endurance.

7 Case study

In this section, the SZYB engineering project is taken as an example of the use of the devised model and algorithm to decide on an inspection plan. As the SZYB is still under construction, it provides a simulated real-world problem scale for our investigation. Numerical experiments were conducted using real-world parameter settings, and several analyses were performed.

7.1. Case background

As mentioned in the Section 1, the UAV-based inspection approach has great advantages for performing inspection tasks in such complex and high-risk construction environment to continuously screen out the potential hidden-danger (illustrated by the Figure 10). The proposed mathematical model and algorithm provide a scientific tool for a project manager to determine the inspection plan of UAVs (inspection tasks assignment, monitoring sequence and charging decisions). Thus, in order to show the practical application of the solution approach for determining the UAVs' inspection plan, we take SZYB for case study and simulate the decision-making based the estimated data from the real-world cases.

Figure 10: The application of the UAV-based inspection approach in the engineering management

The SZYB crosses the Pearl River and has a length of 2180 m. It is located in Guangzhou, Guangdong province, and serves as an engineering control project of the Shiziyang Channel dedicated to strengthening the connectivity of the surrounding cities and regions. Figure 11 shows the approximate location and a concept graph of the SZYB. Such an ultra-large span, high load and ultra-wide deck suspension bridge is technically difficult to construct to a world-class standard. In addition, there are additional difficulties associated with construction of the SZYB, since engineering characteristics such as the ultra-large scope of construction, high technical difficulties and long construction period mean that there are many construction areas at high risk throughout the construction cycle. It is therefore essential for project managers to carry out continuous screening operations to identify hidden dangers and hence reduce the incidence of major accidents.

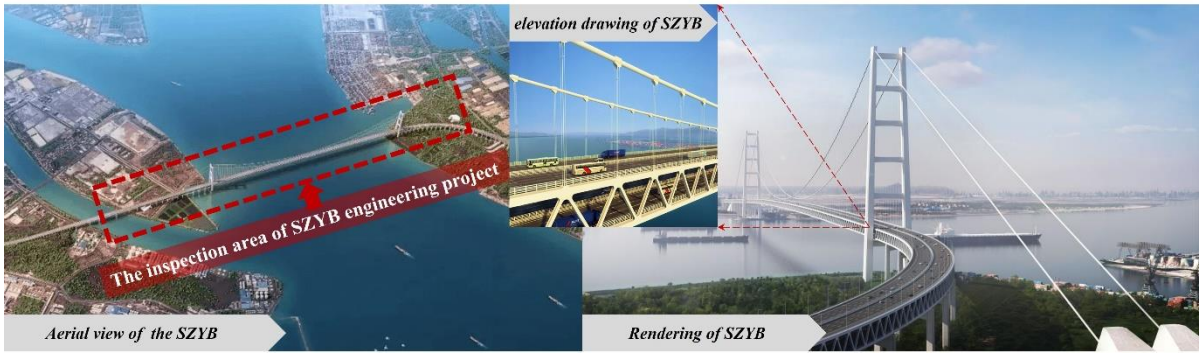


Figure 11: The concept graph of Shiziyang Bridge

7.2. Case results

This subsection describes the parameter settings based on estimation data for the real-scale case of the SZYB. The performance of the devised approach for solving the real-world instance is as follows.

7.2.1. Performance in solving the real-scale instance

The parameter settings for the real-world instance were first clarified before we conducted numerical experiments to validate the efficiency of the devised algorithm in solving our model.

According to planning files kept by the Department of Transport of Guangdong Province (DTGP, 2021), the construction areas of the SZYB could be approximately simulated as a 12 km long and 2 km wide rectangle (as illustrated in Figure 12). We estimated a likely problem scale that was much larger than the scale of ISG9 (listed in Table 6) for determining the UAV inspection plan in the case of the SZYB, and we refer to this new instance group as ISG10. The other parameters were set the same as in Section 6.1. The following computing results were obtained for ISG10 and from the problem simulation shown in Figure 12.

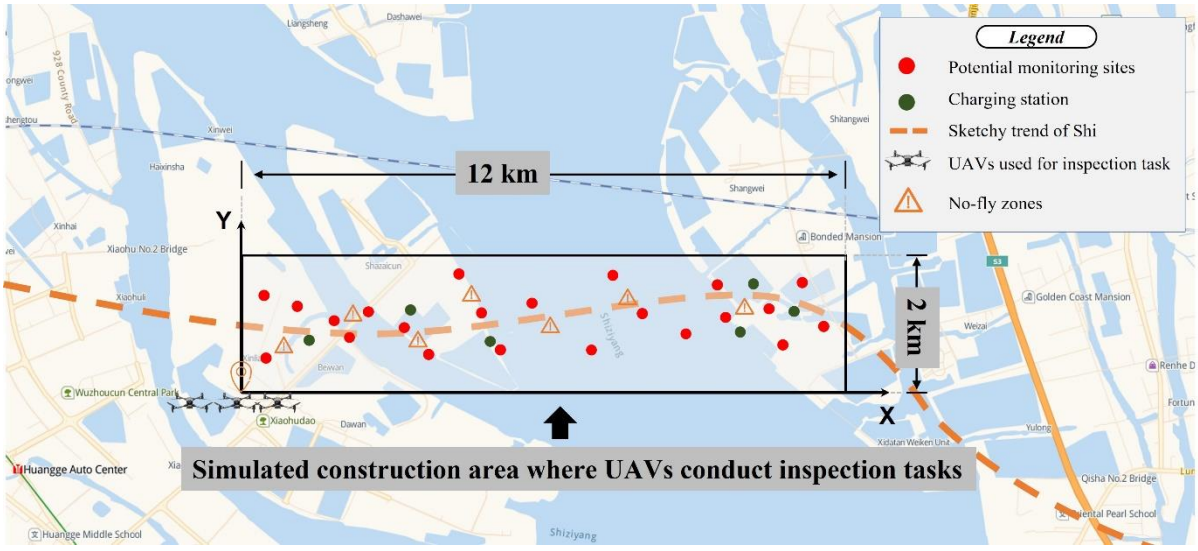


Figure 12: An example of the distribution of monitoring tasks based the real case of SZYB

Table 6: Parameter setting in the real-scale instance

Group ID	No.of UAVs ($\|K\|$)	No.of monitoring sites ($\|I\|$)	No.of maximum monitoring round ($\|Q\|$)	No.of charging stations ($\|B\|$)	No.of duration of planning period ($\|T\|$)
ISG10	8	22	4	6	4

Table 7 shows that the devised algorithm obtained the best solution for most of the real-world instance within 3600 s. There was an average difference of 23.64% between the results obtained using the devised algorithm and those obtained using the strategy described in Section 5.2. The results confirmed the efficiency of the devised algorithm in solving the real-world instance.

Table 7: Results of solving the proposed model on the real-scale instance

Instance		Strategy	VNS		GAP
Group	ID	Obj (h)	Obj (h)	Time (s)	$Gap2$
ISG10	1	15.87	12.69	2403	25.08%
	2	14.12	11.67	1900	20.95%
	3	14.29	11.70	3447	22.09%
	4	14.53	11.42	2271	27.15%
	5	15.66	12.74	2697	22.92%
				Average	23.64%

If the computing time for solving the devised model were required to be much shorter than 3600 s, it would be helpful to cluster the monitoring sites into a series of smaller areas (i.e., separate to four rectangular areas that are 4 km long and 2 km wide). Then, the devised model and algorithm could be adopted to obtain the inspection plan of the corresponding areas separately; i.e., we could solve four subproblems (involving two UAVs, five or six monitoring sites and two charging stations) simultaneously to obtain a series of inspection plans for the different smaller areas. Although this would incur some loss of optimality, owing to the separate solving of a series of medium-scale problems, it would be a practical approach for reducing the computing time of the real-world instance having many monitoring sites, because the computing time for any medium-scale instance is no more than five min. An effective method of clustering monitoring sites to different sub-areas is left as future work.

7.2.2. Practical application of our approach for making an inspection plan

The above numerical experiments provided the computing results of the objective value to validate the effectiveness and efficiency of the devised algorithm in solving a real-world instance, but they did not present the obtained UAV inspection plan. This subsection demonstrates the practical application our approach for making real-world inspection plans.

To simulate the real-world instance as realistically as possible, we generated an instance with six

UAVs, 20 monitoring sites and eight charging stations, based on the realistic situation where (1) the monitoring sites were averagely placed within the construction area of the SZYB, (2) the charging stations were distributed in corresponding construction areas to satisfy the charging demand, and (3) each UAV begins its inspection tasks on one side of the SZYB and ends these tasks on the other side. On the basis of these requirements, a real-world instance is illustrated in Figure 13. Here, we needed to generate a practical inspection plan for a certain number of UAVs.

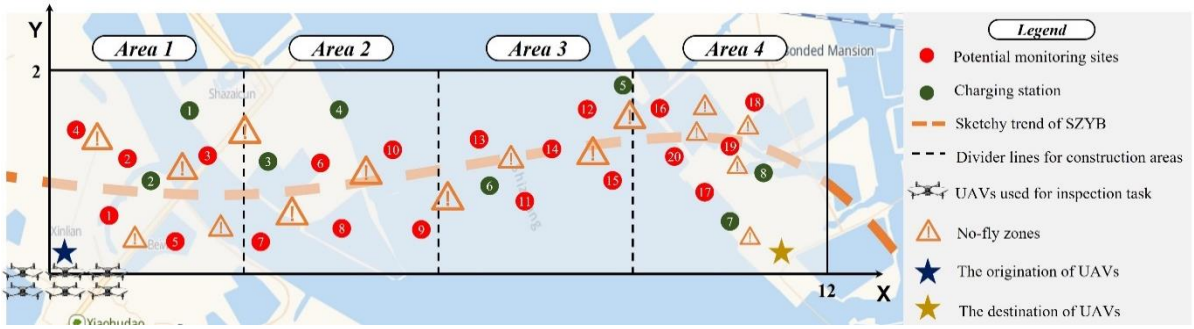


Figure 13: A real-world instance for making the practical inspection plan of UAVs

The inspection plan obtained using our approach is presented in Table 8, and the Gantt chart of the inspection plan is presented in Figure 14. Owing to space limitations, information on the charging decisions of UAVs is omitted from the chart. The following insights were gained from the results for the real-world case. First, it was found that all of the UAVs, except UAV 3 and UAV 4, continuously performed inspection tasks for a long time. UAV 3 and UAV 4 could be regarded as *emergency workers* dedicated to performing inspection tasks that may not be completed by other *busy* UAVs. Second, the time intervals between two consecutive inspection tasks of a UAV were usually shorter than 10 min but sometimes much longer than 10 min (e.g., 18 min between tasks 13-1 and 2-2). Except in a case of a long distance between two inspection tasks, the time was mainly spent on charging. All time intervals were shorter than 30 min, meaning that UAVs tended to charge frequently instead of returning to a charging station once nearly depleted of power. The latter charging strategy would result in UAVs spending a long time waiting to fully charge, which would violate the time interval requirement.

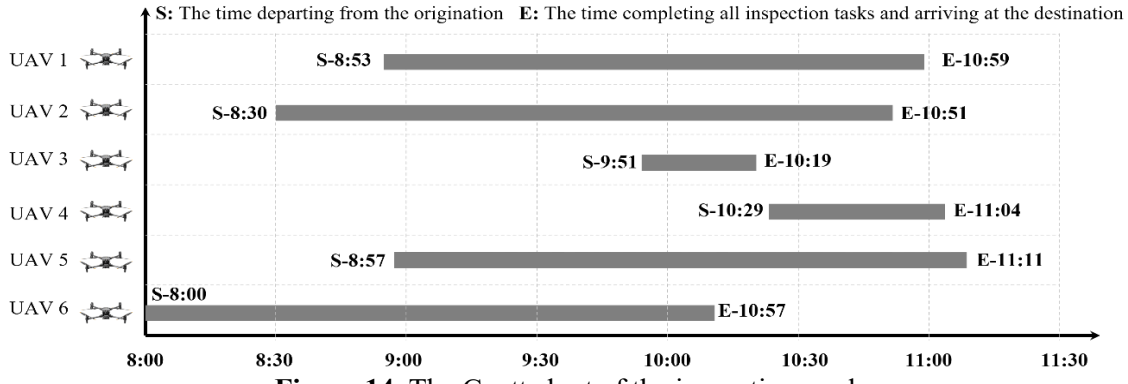


Figure 14: The Gantt chart of the inspection work

Table 8: The timetable of inspection work for the real-world instance illustrated at Figure13

UAV	Inspection tasks of 1^{st} round		Inspection tasks of 2^{nd} round		Inspection tasks of 3^{rd} round		Inspection tasks of 4^{th} round	
	I_index	I_time	I_index	I_time	I_index	I_time	I_index	I_time
1	0	8:53	2	9:27	2	10:05	2	10:38
	2	8:54	3	9:31	3	10:09	3	10:42
	3	8:58	9	9:37	16	10:16	9	10:49
	9	9:04	13	9:41	13	10:21	13	10:53
	13	9:09						
	21	10:59					***Total Time: 2h6min***	
2	0	8:30	10	9:09	17	9:58	17	10:31
	17	8:36	17	9:20	14	10:04	14	10:36
	6	8:41	14	9:25	18	10:09	18	10:42
	14	8:47	18	9:31	8	10:26	12	10:46
	18	8:52	12	9:36				
	12	8:56						
	7	9:03						
	8	9:06						
	21	10:51					***Total Time: 2h21min***	
3	0	9:51	8	9:54	12	10:08	10	10:13
	21	10:19					***Total Time: 28min***	
4	0	10:26			5	10:27	1	10:48
	21	11:04			7	10:30	6	10:54
							Total Time: 38min	
5	0	8:57	1	9:23	11	10:06	11	10:46
	11	9:01	11	9:34	9	10:10	16	10:51
	16	9:09	6	9:38	1	10:16	5	11:00
	5	9:17	16	9:43	6	10:21	7	11:03
	21	11:11	5	9:52				
			7	9:58				
			15	9:09			***Total Time: 2h14min***	
6	0	8:00	20	9:16	15	9:09	15	9:45
	15	8:05	19	9:19	20	9:16	20	9:49
	20	8:09	4	9:28	19	9:19	19	9:52
	19	8:12	10	9:41	4	9:28	4	10:00
	1	8:21			10	9:41		
	4	8:24						
	10	8:31						
	21	10:10					***Total Time: 2h10min***	

Note: (1) "I_index" and "I_time" denote the index of different monitoring sites and corresponding monitoring time, respectively; (2) The time of monitoring sites "0" and "21" indicate the departure time of the corresponding UAV from the origination and the arriving time at the destination, respectively; (3) The planning horizon is set as 8h to 12h in the above experiment.

8 Conclusion

This paper explored the UAV inspection routing and scheduling problem in the context of engineering project construction, and an MILP model was formulated to optimize the total working time of UAVs completing a set of monitoring tasks. The devised model comprehensively considers the limitations of no-fly zones, interval windows and the power of a UAV when developing UAV inspection routing and scheduling plans. A VNS-based metaheuristic was developed to solve the devised model efficiently. Numerical experiments were performed to validate the effectiveness of the devised model and algorithm. Finally, a case study of the SZYB project was conducted to derive practical insights for engineering managers that would enable them to optimize the efficiency of inspection works. The major contributions of the study are summarized as follows.

(1) From a modeling perspective, we comprehensively considered specific constraints in determining a UAV inspection plan using a devised MILP model; i.e., no-fly zone limitations, rolling monitoring interval time window constraints and power limitations. These specific constraints distinguish the devised model from typical models dedicated to the EVRP or EVRPTW. To the best of our knowledge, few papers have incorporated the above practical constraints in the context of engineering management into a mathematical model or focused on the routing and scheduling of UAVs against the above background.

(2) From an algorithmic perspective, we developed a VNS-based metaheuristic to efficiently solve the devised model, as even a commercial solver struggles to find a feasible solution for this model on the large-scale instances. Tailored neighborhood structure and initialization methods based on the specific features of the problem were developed. Extensive numerical experiments validated the efficiency of the devised algorithm for solving large-scale instances within 30 min.

(3) From a practical perspective, we gained several managerial insights from sensitivity experiments. For example, the suitable number of UAVs for performing different numbers of inspection tasks should be evaluated. Investing extensively in purchasing UAVs for inspections may not necessarily reduce the total working time; instead, it may increase unnecessary costs. A comparative experiment showed that an engineering manager purchasing UAVs within a limited budget should prioritize the velocity of UAVs, followed by their power consumption per unit distance, and finally their battery power. Finally, it is best for UAVs—especially those conducting many inspection tasks—to charge frequently, as this reduces the time they have to wait to become fully charged.

This paper makes contributions to the inspection routing and scheduling optimization of UAVs in the context of engineering management, but there remain topics deserving of investigation. As mentioned,

the optimal number of UAVs for different scales of inspection work should be further explored. In addition, the uncertainty in inspection time for a monitoring site and in power consumption should be considered. Finally, a dynamic or stochastic programming method should be introduced so that the model and decision making are practical.

Acknowledgement

This research was supported by the National Natural Science Foundation of China (Grant Nos. 72201229, 72025103, 72071173).

Appendices

Appendix 1: A submodel for iteratively generating a feasible inspection plan

Before addressing the model, a new set are defined: $P_n^F = \{1, \dots, n-1\}, P_n^B = \{n\}, n \in \{1, \dots, \max_{i \in I}\{|Q_i|\}\}$. For the first iteration, $P_1^F = \emptyset$. The model for the n^{th} iteration is formulated as follows. It is noted that the decisions on the inspection routes for the first round of monitoring sites would not consider the time interval constraints (40)–(41), thus the $M1\_1$ could be regarded as a classical EVRP and is easily to generate a feasible solution for CPLEX within a short time. Begin with the second iteration, constraints (40)–(41) would be considered to guarantee that the time interval of consecutive visit of monitoring site i is limited in the range of $[m_i, n_i]$. It could be regarded as an EVRPTW for which it may be difficult to generate a feasible solution with the CPLEX solver within a short time under large-scale instances. Thus, we further fix the $\alpha_{k,i,l}$ and $\beta_{k,i,l,j,l'}$, $\forall k \in K, i, j \in I, l, l' \in P_n^B$ with the corresponding value as set P_n^F to speed up the solving. The procure is illustrated as the Figure 15.

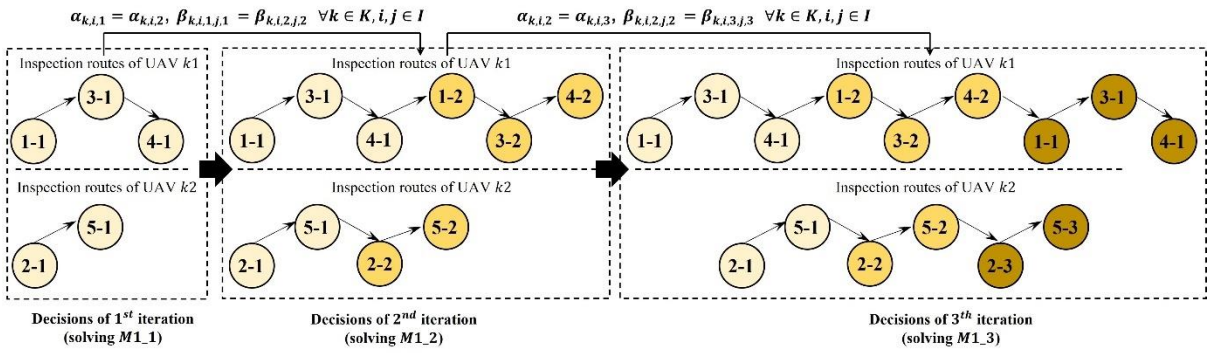


Figure 15: Strategy for speeding up the generation of a feasible solution in the large-scale instances

$Input\ data$: all the input data for the original model

$\alpha_{k,i,l}, \beta_{k,i,l,j,l'}, \lambda_{k,i,l}, \theta_{k,j,l'}$ $\forall k \in K, i, j \in I, l, l' \in P_n^F$

$[M1\_n]$ Minimize $\sum_{k\in K} \pi_k$ (26)

subject to:

$\sum_{k\in K} \alpha_{k,i,l} = 1$ $\forall i \in I, l \in P_n^B$ (27)

$\sum_{j\in I\cup\{\dot{e}\}} \sum_{l'\in Q_j} \beta_{k,i,l,j,l'} = \alpha_{k,i,l} = \sum_{j\in I\cup\{e\}} \sum_{l'\in Q_j} \beta_{k,j,l',i,l}$ $\forall k \in K, i \in I, l \in P_n^B$ (28)

$\sum_{b\in B} \varphi_{k,i,l,b} \le \sum_{j\in I\cup\{\dot{e}\}} \sum_{l'\in Q_j} \beta_{k,i,l,j,l'}$ $\forall k \in K, i \in I, l \in P_n^F$ (29)

$\beta_{k,i,l,j,l'} \le f_{i,j}^1 + M \sum_{b\in B} \varphi_{k,i,l,b}$ $\forall k \in K, i \in I\cup\{e\}, j \in I\cup\{\dot{e}\}, l, l' \in P_n^B$ (30)

$\beta_{k,i,l,j,l'} \le f_{i,b}^2 + M(1 - \varphi_{k,i,l,b})$ $\forall k \in K, i \in I\cup\{e\}, j \in I\cup\{\dot{e}\}, l, l' \in P_n^B$ (31)

$\beta_{k,i,l,j,l'} \le f_{j,b}^2 + M(1 - \varphi_{k,i,l,b})$ $\forall k \in K, i \in I\cup\{e\}, j \in I\cup\{\dot{e}\}, l, l' \in P_n^B$ (32)

$\lambda_{k,i,l} + t_{k,i} + \frac{d_{i,j}^1}{v_k} \le \lambda_{k,j,l'} + M(1 - \beta_{k,i,l,j,l'})$ $\forall k \in K, i \in I\cup\{e\}, j \in I\cup\{\dot{e}\}, l, l' \in P_n^F \cup P_n^B$ (33)

$\lambda_{k,i,l} + t_{k,i} + \frac{d_{i,b}^2}{v_k} + \frac{(r_k - \theta_{k,i,l} + h_k d_{i,b}^2)}{c_k} + \frac{d_{b,j}^2}{v_k} \le \lambda_{k,j,l'} + M(2 - \varphi_{k,i,l,b} - \beta_{k,i,l,j,l'})$

$\forall k \in K, i \in I\cup\{e\}, b \in B, j \in I\cup\{\dot{e}\}, l, l' \in P_n^F \cup P_n^B$ (34)

$\lambda_{k',i,l+1} \ge \lambda_{k,i,l} + t_{k,i} + m_i - M(2 - \alpha_{k,i,l} - \alpha_{k',i,l+1})$ $\forall k, k' \in K, i \in I, l \in P_n^B$ (35)

$\lambda_{k',i,l+1} \le \lambda_{k,i,l} + t_{k,i} + n_i + M(2 - \alpha_{k,i,l} - \alpha_{k',i,l+1})$ $\forall k, k' \in K, i \in I, l \in P_n^B$ (36)

$\theta_{k,j,l'} \le \theta_{k,i,l} - g_{k,i} - h_k d_{i,j}^1 + M(\sum_{b\in B} \varphi_{k,i,l,b} + 1 - \beta_{k,i,l,j,l'})$

$\forall k \in K, i \in I\cup\{e\}, j \in I\cup\{\dot{e}\}, l, l' \in P_n^F \cup P_n^B$ (37)

$\theta_{k,j,l'} \ge \theta_{k,i,l} - g_{k,i} - h_k d_{i,j}^1 - M(\sum_{b\in B} \varphi_{k,i,l,b} + 1 - \beta_{k,i,l,j,l'})$

$\forall k \in K, i \in I\cup\{e\}, j \in I\cup\{\dot{e}\}, l, l' \in P_n^F \cup P_n^B$ (38)

$\theta_{k,j,l'} \ge r_k - h_k d_{j,b}^2 - M\big(2 - \varphi_{k,i,l,b} - \beta_{k,i,l,j,l'}\big)$

$\forall k \in K, i \in I\cup\{e\}, b \in B, j \in I\cup\{\dot{e}\}, l, l' \in P_n^F \cup P_n^B$ (39)

$\theta_{k,j,l'} \le r_k - h_k d_{j,b}^2 + M\big(2 - \varphi_{k,i,l,b} - \beta_{k,i,l,j,l'}\big)$

$\forall k \in K, i \in I\cup\{e\}, b \in B, j \in I\cup\{\dot{e}\}, l, l' \in P_n^F \cup P_n^B$ (40)

$\theta_{k,i,l} \ge g_{k,i} + h_k \min\{\min_{b\in B} d_{i,b}^2, d_{i,\dot{e}}^1\} - M(1 - \alpha_{k,i,l})$

$\forall k \in K, i \in I, l \in P_n^B$ (41)

$\pi_k = \lambda_{k,\dot{e},0} - \lambda_{k,e,0}$ $\forall k \in K$ (42)

$\alpha_{k,i,l} \in \{0,1\}$ $\forall k \in K, i \in I\cup\{e,\dot{e}\}, l \in P_n^B$ (43)

$\beta_{k,i,l,j,l'} \in \{0,1\}$ $\forall k \in K, i \in I\cup\{e\}, j \in I\cup\{\dot{e}\}, l, l' \in P_n^F \cup P_n^B$ (44)

$\varphi_{k,i,l,b} \in \{0,1\}$ $\forall k \in K, i \in I\cup\{e\}, b \in B, l \in P_n^B$ (45)

$$0 \le \lambda_{k,i,l}, \pi_k \le T \qquad \forall k \in K, i \in I \cup \{e, \dot{e}\}, l \in P_n^B \tag{46}$$

$$0 \le \theta_{k,i,l} \le r_k \qquad \forall k \in K, i \in I \cup B \cup \{e, \dot{e}\}, l \in P_n^B \tag{47}$$

Appendix 2: **A submodel employed for evaluating the performance of a new inspection plan**

Input: $\bar{\alpha}_{k,i,l}$ and $\bar{\beta}_{k,i,l,j,l'}$ $\qquad \forall k \in K, i, j \in I, l, l' \in Q_i$

Output: $\varphi_{k,i,l,b}, \lambda_{k,i,l}, \theta_{k,i,l}$ and π_k

[$M2$] Minimize $\sum_{k \in K} \pi_k$ (48)

subject to:

$$\lambda_{k,i,l} + t_{k,i} + \frac{d_{i,j}^1}{v_k} \le \lambda_{k,j,l'} + M(1 - \bar{\beta}_{k,i,l,j,l'}) \forall k \in K, i \in I \cup \{e\}, j \in I \cup \{\dot{e}\}, l \in Q_i, l' \in Q_j \tag{49}$$

$$\lambda_{k,i,l} + t_{k,i} + \frac{d_{i,b}^2}{v_k} + \frac{(r_k - \theta_{k,i,l} + h_k d_{i,b}^2)}{c_k} + \frac{d_{b,j}^2}{v_k} \le \lambda_{k,j,l'} + M(2 - \varphi_{k,i,l,b} - \bar{\beta}_{k,i,l,j,l'})$$
$$\forall k \in K, i \in I \cup \{e\}, b \in B, j \in I \cup \{\dot{e}\}, l \in Q_i, l' \in Q_j \tag{50}$$

$$\lambda_{k',i,l+1} \ge \lambda_{k,i,l} + t_{k,i} + m_i - M(2 - \bar{\alpha}_{k,i,l} - \bar{\alpha}_{k',i,l+1}) \qquad \forall k, k' \in K, i \in I, l \in Q_i / \{|Q_i|\} \tag{51}$$

$$\lambda_{k',i,l+1} \le \lambda_{k,i,l} + t_{k,i} + n_i + M(2 - \bar{\alpha}_{k,i,l} - \bar{\alpha}_{k',i,l+1}) \qquad \forall k, k' \in K, i \in I, l \in Q_i / \{|Q_i|\} \tag{52}$$

$$\theta_{k,j,l'} \le \theta_{k,i,l} - g_{k,i} - h_k d_{i,j}^1 + M(\textstyle\sum_{b \in B} \varphi_{k,i,l,b} + 1 - \bar{\beta}_{k,i,l,j,l'})$$
$$\forall k \in K, i \in I \cup \{e\}, j \in I \cup \{\dot{e}\}, l \in Q_i, l' \in Q_j \tag{53}$$

$$\theta_{k,j,l'} \ge \theta_{k,i,l} - g_{k,i} - h_k d_{i,j}^1 - M(\textstyle\sum_{b \in B} \varphi_{k,i,l,b} + 1 - \bar{\beta}_{k,i,l,j,l'})$$
$$\forall k \in K, i \in I \cup \{e\}, j \in I \cup \{\dot{e}\}, l \in Q_i, l' \in Q_j \tag{54}$$

$$\theta_{k,j,l'} \ge r_k - h_k d_{j,b}^2 - M\big(2 - \varphi_{k,i,l,b} - \bar{\beta}_{k,i,l,j,l'}\big)$$
$$\forall k \in K, i \in I \cup \{e\}, b \in B, j \in I \cup \{\dot{e}\}, l \in Q_i, l' \in Q_j \tag{55}$$

$$\theta_{k,j,l'} \le r_k - h_k d_{j,b}^2 + M\big(2 - \varphi_{k,i,l,b} - \bar{\beta}_{k,i,l,j,l'}\big)$$
$$\forall k \in K, i \in I \cup \{e\}, b \in B, j \in I \cup \{\dot{e}\}, l \in Q_i, l' \in Q_j \tag{56}$$

$$\theta_{k,i,l} \ge g_{k,i} + h_k \min\{\min_{b \in B} d_{i,b}^2, d_{i,\dot{e}}^1\} - M(1 - \bar{\alpha}_{k,i,l})$$

$$\pi_k = \lambda_{k,\dot{e},0} - \lambda_{k,e,0} \qquad \forall k \in K \tag{57}$$

$$\varphi_{k,i,l,b} \in \{0,1\} \qquad \forall k \in K, i \in I \cup \{e\}, b \in B, l \in Q_i \tag{58}$$

$$0 \le \lambda_{k,i,l}, \pi_k \le T \qquad \forall k \in K, i \in I \cup \{e, \dot{e}\}, l \in Q_i \tag{59}$$

$$0 \le \theta_{k,i,l} \le r_k \qquad \forall k \in K, i \in I \cup B \cup \{e, \dot{e}\}, l \in Q_i. \tag{60}$$

Appendix 3: **Pseudo-code of the VND procedure**

The VND procedure	
Input:	local_best_solution $\grave{IP}$, best_solution $F(IP^*)$,
1:	**While** (true)
2:	**Switch** (ω)
3:	Case 1: Performing *insert monitoring task* to generate new inspection plan $\widetilde{IP}$ with
4:	the given $\grave{IP}$ and solving M2 to evaluate the objective value $F(\widetilde{IP})$
5:	**If** $(F(\widetilde{IP}) < F(IP^*))$
6:	$F(IP^*) \leftarrow F(\widetilde{IP}), IP^* \leftarrow \widetilde{IP}, \grave{IP} \leftarrow \widetilde{IP}$
7:	$\omega = 1$;
8:	break;
9:	Case 2: Performing *swap monitoring task* to generate new inspection plan $\widetilde{IP}$ with
10:	the given $\grave{IP}$ and solving M2 to evaluate the objective value $F(\widetilde{IP})$
11:	**If** $(F(\widetilde{IP}) < F(IP^*))$
12:	$F(IP^*) \leftarrow F(\widetilde{IP}), IP^* \leftarrow \widetilde{IP}$
13:	$\omega = 1$;
14:	break;
15:	Case 3: Performing *insert monitoring round* to generate new inspection plan $\widetilde{IP}$ with
16:	the given $\grave{IP}$ and solving M2 to evaluate the objective value $F(\widetilde{IP})$
17:	**If** $(F(\widetilde{IP}) < F(IP^*))$
18:	$F(IP^*) \leftarrow F(\widetilde{IP}), IP^* \leftarrow \widetilde{IP}$
19:	$\omega = 1$;
20:	break;
21:	Case 4: Performing *swap monitoring sequence* to generate new inspection plan $\widetilde{IP}$
22:	with the given $\grave{IP}$ and solving M2 to evaluate the objective value $F(\widetilde{IP})$
23:	**If** $(F(\widetilde{IP}) < F(IP^*))$
24:	$F(IP^*) \leftarrow F(\widetilde{IP}), IP^* \leftarrow \widetilde{IP}$
25:	$\omega = 1$;
26:	break;
27:	default;
28:	return;
29:	$\omega \leftarrow \omega + 1$
30:	**End while**

References

Bailon-Ruiz, R., Bit-Monnot, A., Lacroix, S. (2022) Real-time wildfire monitoring with a fleet of UAVs. *Robotics and Autonomous Systems* 152: 104071.

Chowdhury, S., Shahvari, O., Marufuzzaman, M., Li, X., Bian, L. (2021) Drone routing and optimization for post-disaster inspection. *Computers & Industrial Engineering* 159: 107495.

Coelho, B., Coelho, V., Coelho, I., Ochi, L., Haghnazar K., Lima, M., Zuidema, D., da Costa, A. (2017) A multi-objective green UAV routing problem. *Computers & Operations Research* 88: 306–315.

Dji. (2023) The technology parameter of M300 RTK. https://www.dji.com/cn/matrice-300/specs. Accessed 1 April 2023.

DTGP (2021) The Department of Transport of Guangdong Province (DTGP). http://td.gd.gov.cn/gkmlpt/content/3/3233/post_3233793.html#1479. Accessed 1 April 2023.

Hu, J., Niu, H., Carrasco, J., Lennox, B., Arvin, F. (2022) Fault-tolerant cooperative navigation of networked UAV swarms for forest fire monitoring. *Aerospace Science and Technology* 123: 107494.

Hu, M., Liu, W., Lu, J., Fu, R., Peng, K., Ma, X., Liu, J. (2019) On the joint design of routing and scheduling for vehicle-assisted multi-UAV inspection. *Future Generation Computer Systems* 94: 214–223.

Huang, S., Huang, Y., Blazquez, C., Paredes-Belmar, G. (2018) Application of the ant colony optimization in the resolution of the bridge inspection routing problem. *Applied Soft Computing* 65: 443–461.

Jeong, E., Seo, J., Wacker, J. (2022) UAV-aided bridge inspection protocol through machine learning with improved visibility images. *Expert Systems with Applications* 197: 116791.

Khan, S., Qadir, Z., Munawar, H., Nayak, S., Budati, A., Verma, K., Prakash, D. (2021) UAVs path planning architecture for effective medical emergency response in future networks. *Physical Communication* 47: 101337.

Li, M., Zhen, L., Wang, S., Lv, W., Qu, X. (2018) Unmanned aerial vehicle scheduling problem for traffic monitoring. *Computers & Industrial Engineering* 122: 15–23.

Liu, W., Zhang, T., Huang, S., Li, K. (2022) A hybrid optimization framework for UAV reconnaissance mission planning. *Computers & Industrial Engineering* 173: 108653.

Manyam, S., Sundar, K., Casbeer, D. (2020) Cooperative routing for an air-ground vehicle team-exact algorithm, transformation method, and heuristics. *IEEE Transactions on Automation Science and Engineering* 17(1): 537–547.

Mbiadou Saleu, R., Deroussi, L., Feillet, D., Grangeon, N., Quilliot, A. (2022) The parallel drone

scheduling problem with multiple drones and vehicles. *European Journal of Operational Research* 300(2): 571–589.

Meng, D., Xiao, Y., Guo, Z., Jolfaei, A., Qin, L., Lu, X., Xiang, Q. (2021) A data-driven intelligent planning model for UAVs routing networks in mobile Internet of Things. *Computer Communications* 179: 231–241.

Moshref-Javadi, M., Hemmati, A., Winkenbach, M. (2020) A truck and drones model for last-mile delivery: A mathematical model and heuristic approach. *Applied Mathematical Modelling* 80: 290–318.

Munishkin, A., Milutinović, D., Casbeer, D. (2020) Min–max time efficient inspection of ground vehicles by a UAV team. *Robotics and Autonomous Systems* 125: 103370.

Perry, B., Guo, Y., Atadero, R., Van de Lindt, J. (2020) Streamlined bridge inspection system utilizing unmanned aerial vehicles (UAVs) and machine learning. *Measurement* 164: 108048.

Qin, W., Shi, Z., Li, W., Li, K., Zhang, T., Wang, R. (2021) Multiobjective routing optimization of mobile charging vehicles for UAV power supply guarantees. *Computers & Industrial Engineering* 162: 107714.

Rajan, S., Sundar, K., Gautam, N. (2022) Routing problem for unmanned aerial vehicle patrolling missions: A progressive hedging algorithm. *Computers & Operations Research* 142: 105702.

Sacramento, D., Pisinger, D., Ropke, S. (2019) An adaptive large neighborhood search metaheuristic for the vehicle routing problem with drones. *Transportation Research Part C* 102: 289–315.

Sajid, M., Mittal, H., Pare, S., Prasad, M. (2022) Routing and scheduling optimization for UAV assisted delivery system: A hybrid approach. *Applied Soft Computing* 126: 109225.

Salama, M., Srinivas, S. (2022) Collaborative truck multi-drone routing and scheduling problem: Package delivery with flexible launch and recovery sites. *Transportation Research Part E* 164: 102788.

Shen, L., Hou, Y., Yang, Q., Lv, M., Dong, J., Yang, Z., Li, D. (2022) Synergistic path planning for ship-deployed multiple UAVs to monitor vessel pollution in ports. *Transportation Research Part D* 110: 103415.

Spencer, B., Hoskere, V., Narazaki, Y. (2019) Advances in computer vision-based civil infrastructure inspection and monitoring. *Engineering* 5(2): 199–222.

Sundar, K., Venkatachalam, S., Manyam, S. (2017) Path planning for multiple heterogeneous Unmanned Vehicles with uncertain service times. *International Conference on Unmanned Aircraft Systems* 480–487.

Wang, Y., Li, Y., Yin, F., Wang, W., Sun, H., Li, J., Zhang, K. (2022) An intelligent UAV path planning

optimization method for monitoring the risk of unattended offshore oil platforms. *Process Safety and Environmental Protection* 160: 13–24.

Xia, Y., Batta, R., Nagi, R. (2017) Controlling a fleet of unmanned aerial vehicles to collect uncertain Information in a Threat Environment. *Operations Research* 65(3): 674–692.

Xu, C., Xu, M., Yin, C. (2020) Optimized multi-UAV cooperative path planning under the complex confrontation environment. *Computer Communications* 162: 196–203.

Zhen, L., Gao, J., Tan, Z., Wang, S., Baldacci, R. (2022) Branch-price-and-cut for trucks and drones cooperative delivery. *IIE Transactions* 55(3): 271–287.

Zhen, L., Li, M., Laporte, G., Wang, W. (2019a) A vehicle routing problem arising in unmanned aerial monitoring. *Computers & Operations Research* 105: 1–11.

Zhen, L., Wang, B., Li, M., Wang, W., Huang, L. (2019b) Inspection routing problem for coal mine safety personnel in underground mines. *Computers & Industrial Engineering* 130: 526–536.

Zheng, Y., Du, Y., Sheng, W., Ling, H. (2019) Collaborative human–UAV search and rescue for missing tourists in nature reserves. *INFORMS Journal on Applied Analytics* 49(5): 371–383.