The following publication 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 is available at https://doi.org/10.1016/j.cie.2018.05.039.

Unmanned Aerial Vehicle Scheduling Problem for Traffic Monitoring

Abstract: For more accurate multiple periods real-time monitoring of road traffic, this paper studies unmanned aerial vehicles scheduling problem under uncertain demands. A mixed integer programming model is designed for this problem by combining capacitated arc routing problem with inventory routing problem. A local branching based solution method is developed for solving the model. A case study on applying this model to Shanghai road traffic is performed. In addition, some numerical experiments are conducted to validate the effectiveness of the proposed model and the efficiency of the proposed solution method.

Keywords: UAV routing problem, CARP with time window, Inventory routing problem, Traffic monitoring

1. Introduction

With the rapid development of urban traffic, the surge of car ownership, traffic carrying capacity of road network is growing; further the corresponding traffic congestion and traffic safety issues are also highlighted. As of 2016, the motor car ownership of China has reached 285 million. According to the data of China Highway Network, 15 cities of China lost nearly 10 billion RMB a day due to traffic congestion and management problems. It is clear that traffic jams are frequent in many cities in China. Beijing, Shanghai and other big cities are becoming "blocked" cities. Adverse road conditions in a sense makes the car the most time-consuming way to travel; moreover, this trend is spreading to second and third tier cities of China.

How to avoid the emergence of various types of traffic problems is the focus of this study. Road traffic information data is the implementation of traffic planning, traffic control and other work basis. Existing urban road traffic monitors, such as vehicle induction coils, traffic cameras, infrared monitors, can collect the required traffic flow, speed and other data. However, as the result of a number of fast roads, trunk roads, sub trunk roads and other road segments are not installed traffic inspection equipment, the traffic information cannot be obtained. In addition, although some urban road segments have been installed traditional traffic monitors, the monitoring accuracy and parameters still need other means to verify and supplement. Unmanned aerial vehicles (UAVs, for short), commonly known as drones, as new traffic monitoring equipment, by loading different imaging sensors, can capture the target image, and monitored images can be transmitted in real time to control station through the wireless transmission system. After decades of development, the performance of UAV technology is improved continuously, and the function of UAVs is perfected. Especially in recent years, aviation, microelectronics, computers, navigation, communications, sensors and other related technologies are booming, making UAV technology accessed to application from research and development stage. Information collection is one of the most important applications of UAVs (Xia et al., 2017), and has been widely used in meteorological exploration, disaster monitoring, geological survey, environmental monitoring, avalanche detection and other fields. Moreover, the use of UAVs as a means of collecting traffic data has a number of advantages, such as wide monitoring areas (covering 300-500 meters at a height of 150 to 300 meters from the ground), low cost and excellent flexibility (Harwin and Lucieer, 2012).

In this paper, UAVs as the traditional traffic monitor auxiliary or supplementary way to collect road traffic information. In above case, faced with a certain number of UAVs, monitoring requirements and various constraints, the rational scheduling of UAVs has become one of the practical problems that need to be solved. The remainder of this article is organized as follows: Section 2 reviews the related works; Section 3 elaborates on the problem background; the model is formulated in Section 4. Local branching based solution methods are developed in Section 5. Experiments are illustrated in Section 6, and closing remarks are outlined in the last section.

2. Related works

In recent years, with the rapid advancements in the technology of UAVs, the number of researches on UAV monitoring and scheduling has increased. Salvo et al. (2014) presented a method to evaluate the real traffic flow conditions in urban areas by using UAVs in order to realize an accurate traffic study. Guerriero et al. (2014) proposed a distributed system of autonomous UAVs and gave a mathematical formulation of the problem as a multi-criteria optimization model based on vehicle routing problem with soft time window constraints, in which the total distances traveled by the UAVs, the customer satisfaction and the numbers of used UAVs were considered simultaneously to solve distributed dynamic scheduling problems. Murray et al. (2015) provided two mixed integer linear programming formulations aimed at optimal routing and scheduling of unmanned aircraft and delivery trucks to solve last-mile delivery in logistics operations. The classical vehicle routing problem (VRP) was introduced to study a UAV works in collaboration with a traditional delivery truck to distribute parcels in the scenario. Cho et al. (2015) designed a mathematical model based on VRP for UAV aided security operations in the oil and gas industry. They stated that UAVs could provide the information on possible emergency situations such as oil spills. The primary goal of their model was to generate an optimal UAV operational schedule to meet surveillance needs in the areas of interest in each time period by considering the minutely charging cost and operating cost. Yakıcı et al. (2016) addressed the problem of locating and routing of UAVs at tactical level and formulated this problem as an integer linear program with the aim of maximization of the total score collected from visited interest points by flight routes of UAVs. They also developed a novel ant colony optimization metaheuristic approach which can find the best known or close to the best known solution in a short time.

However, the monitoring demand points were considered as nodes in the network in these above studies. In fact, as mobile sensors, UAVs are more appropriate to monitor demand arcs than demand nodes, since they can monitor during flight. Although the UAV can travel arbitrarily in the air rather than being limited to a particular network, flying over an unknown space may cause UAVs to experience unknown obstacles. Therefore, in the deployment of UAVs, monitoring based on demand arcs is meaningful. Then the related works are mainly reviewed through the perspective of arc routing problem. The Chinese postman problem (CPP), presented by Mei-Ko (1962), was a typical representative of the basic arc routing optimization problem. Golden and Wong (1981) proposed capacitated arc routing problem and formulated it as a mixed integer programming problem firstly. This problem was getting more and more attention because it could be used to solve many problems in real life. Thus many scholars have done a lot of modeling analysis, some of which have been put into specific applications. For example, Eglese and Li (1992) designed a simulated annealing solution to solve the problem of deicing in Lancashire; Li and Eglese (1996) studied how to design routes for gritters to minimize costs including consideration of multiple depot locations, limited vehicle capacities, roads with different priorities for gritting and so on. In order to make the CARP more in line with the actual situation, many researchers added the restrictions on the basis of the problem or combined it with other variants. Lacomme et al. (2005) proposed that the trips must be planned over a multi-period horizon in many applications and gave a new problem called periodic CARP (PCARP). Monroy et al. (2013) introduced the periodic capacitated arc routing problem with irregular services. The problem consisted of determining a set of routes to cover a given network over a time horizon. Also, they presented a mathematical model and a heuristic solution approach. Salazar-Aguilar et al. (2013) introduced the synchronized arc and node routing problem, inspired from a real application arising in road marking operations. The aim of their problem was to determine the routes and schedules for the painting and replenishment vehicles in road marking operations so that the pavement marking is completed within the least possible time. Lopes et al. (2014) proposed the location-arc routing problem (LARP) and considered scenarios where the demand was on the edges rather than being on the nodes of a network. Riquelme-Rodríguez et al. (2016) presented and compared two methods for locating water depots along the road network so that penalty costs for the lack of humidity in roads and routing costs are minimized. Their problem belonged to the periodic capacitated arc routing domain from the background of periodically spraying water for dusting suppression; the demands were located on the arcs of the network and the arcs required service more than once in a time horizon.

Although multi-period extensions have been involved in CARP, adopting a frequency variable (i.e. visiting an arc every k periods) was not time-dependent. Thus inventory-based CARP will be the core problem in this paper. Beltrami and Bodin (1974) studied how to invest the least number of vehicles to solve the problem of garbage removal in New York and Washington from a long-term consideration. Russell and Igo (1979) introduced the transport strategy of how to distribute to a customer in a day within a week in order to make transportation more efficient. Federgruen and Zipkin (1984) were the first to clearly propose an integrated inventory and vehicle routing problem (IIVRP). In the same year, Golden et al. (1984) did a similar study that the urgency of demands was described by the ratio of the current inventory level to the inventory capacity; then a heuristic algorithm was designed to solve this problem. Recently, Li et al. (2014) considered an inventory routing problem (IRP) in a large petroleum enterprise group. In their paper, it was more important to avoid stock out for any station, rather than purely focusing on transportation cost minimization. For this, they presented a tabu search algorithm and Lagrangian relaxation technique to tackle the problem. Schuijbroek et al. (2017) studied a major operational cost in bike sharing systems, which was rebalancing the bikes over time such that the reasonable number of bikes and open docks are available to users. Moreover, they determined service level requirements at each bike sharing station and designed optimal vehicle routes to rebalance the inventory. Azadeh et al. (2017) presented a model of inventory routing problem with transshipment in the presence of perishable product. Vehicle routing and inventory decisions were made simultaneously over the planning horizon to meet customer's demand under maximum level policy. Also, they proposed a genetic algorithm based approach to solve the problem, and a numerical example was used to illustrate the validity of the model.

By comparing with the existing related works, this study differs from them in the following aspects. First, considering uncertain monitoring demands are distributed on arcs rather than nodes, arc routing problem is introduced to solve the UAVs traffic flow monitoring problem. Further, for the purpose of real-time traffic flow monitoring, adding the variant of time window and inventory routing problem on the basic optimization model. There are no relevant studies which cover IRP and CARPTW[full name???] simultaneously in the existing works, therefore this paper makes an explorative study on this new problem in specific background.

3. Problem description

The problem in this study relates with dynamically allocating a limited set of UAVs to a traffic network with monitoring demands in multiple periods. Figure 1 shows the main process of using UAVs for road traffic monitoring. Firstly, determine the location and number of road sections to be monitored and use them as a monitoring target for UAVs. Then, enter the monitoring target space and the location of the monitoring order in the control platform to decide the UAV route trajectory. After that, airborne high-definition cameras monitor the road traffic conditions during the process of flight, and through the wireless transmission system, the monitoring video real-time back to the UAV base station. Finally, control personnel extracts the traffic information from the video and reports the results to the drivers traveling on the corresponding road in time to assist them to make informed decisions in the choice of driving routes.

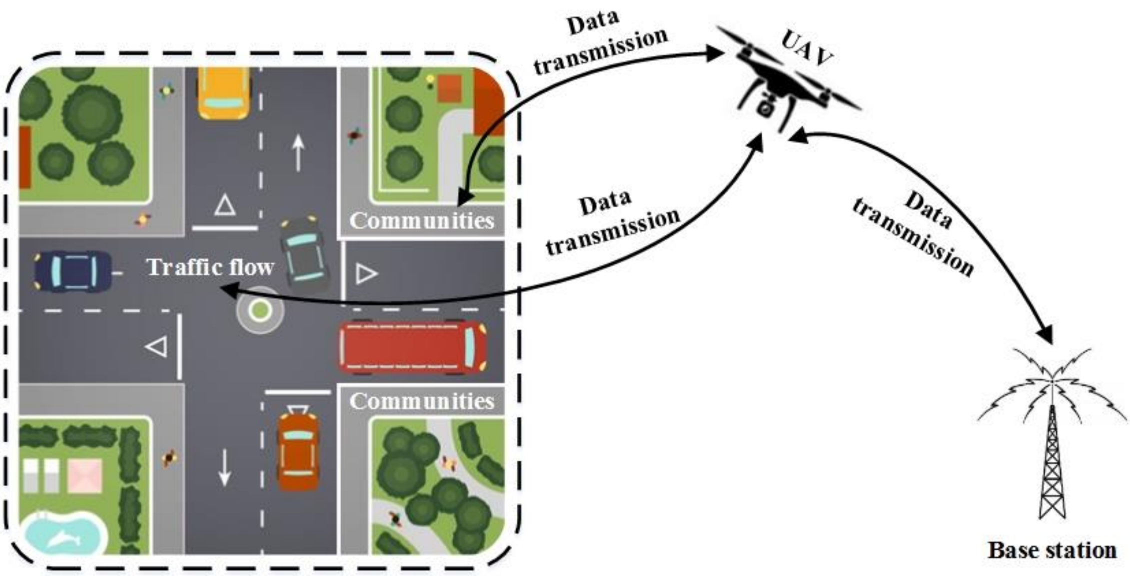

Figure 1: The air survey process of unmanned aerial vehicle

This paper combines the proposed problem with the CARPTW & IRP model, as shown in Figure 2. The basic CARP problem is described as follows: In a network, each monitoring demand is represented by the arc, each arc has a cost, and part of the arcs hold the service demand. UAVs service all demand arcs from a depot. The problem is making the UAV total operating costs minimum under the conditions of meeting the UAV capacity constraints. We assume that the road traffic segments, which are regularly monitored by UAVs, cost less. In UAV scheduling, each UAV usually has a fixed takeoff time. In addition, roads with monitoring demand should be served by UAVs in a certain time; the earliest and latest service time of each road is different. Therefore, in view of the above problems, the time window is introduced in the CARP base model.

In reality, the occurrence of traffic accidents on a road may lead to a significant increase in monitoring demand of the relevant arcs in the network over the next few periods. Not only real-time information of traffic data is usually collected more frequently, but also the current commercial UAVs can run within half an hour; so the research on allocating UAVs in multiple time periods within an hour is meaningful. Thus, the CARPTW and IRP frameworks are used to build such scenarios. As a supplementary method of traditional traffic monitors, the repeating monitoring of UAVs and traditional monitoring equipment may lead to over-monitoring and bring some cost. In addition, as shown in Figure 1, if the UAVs invade private communities, a number of monitoring can cause distress to residents in the community. Then the cost of over-frequent monitoring is simulated as the cost of inventory holding, and this value is described with 'monitoring satisfaction'. Whereas in some cases, this cost may be set to zero.

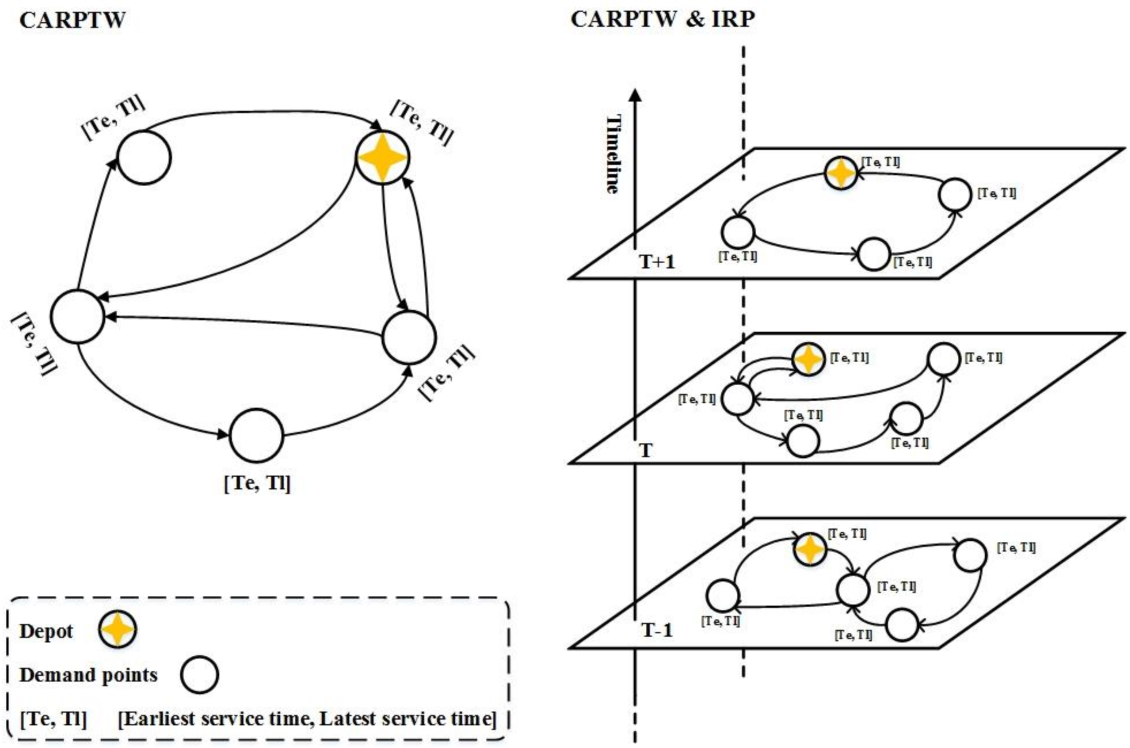

Figure 2: CARPTW and CARPTW & IRP

For this problem, an important decision is how to allocate UAVs so as to minimize the total cost in the monitoring process. Deploying more UAVs to more routes will cost more, but can more continuously monitor more demand arcs to reduce the risk of non-monitoring. The challenges embedded in the above decision problem contain the following aspects. In the first place, each UAV works for several monitoring demand arcs; each arc is served by several UAVs.

The many-to-many mapping relationships between them as well as the capacity limitation of each UAV bring a lot of challenges to model formulation for this problem. In the next place, from the perspective of each monitoring arc, their demands are uncertain and may be related to the density of traffic flow throughout the day. In the end, to real-time traffic monitoring for the purpose, arc routing problems need to be addressed in multiple periods of inventory adjustment. Meanwhile, the time windows of each demand arc should be taken into account. They are another type of decision embedded in the problem.

4. Model formulation

This section formulates mix integer programming models for UAVs scheduling in traffic monitoring. First of all, CARP model of Golden and Wong (1981) is modified to adapt to the UAVs monitoring traffic background, and the time window is added to make it more close to the reality situation. We then gradually extend the problem to multiple periods. Besides, from the level of UAV technology study, researchers are working to improve the endurance of UAVs. Limited battery capacity will affect the UAV flight endurance, which may also further have an effect on the monitoring capability. Because the most of UAVs that are suitable for traffic flow monitoring are battery-powered, relevant factors of battery energy are taken into account in the model.

4.1 Notions

Indices and sets:

i, j index of monitoring demand nodes.

(i, j) index of monitoring demand arcs.

$G(N, E, C)$ network with a set of nodes N, $i, j \in N, i \neq j, N = \{0,1, \cdots, |N|\}$; a set of undirected arcs E, $(i, j) \in E$; and a matrix of arc costs C.

k index of UAVs.

K set of all the UAVs, $K = \{1,2 \cdots, k, \cdots, |K|\}$.

t index of monitoring periods.

T set of all the monitoring periods, $T = \{0,1 \cdots, t, \cdots, |T|\}$.

w index of periods that the UAV needs to be charged.

W	set of all periods that the UAV needs to be charged, $W = \{1,2 \cdots, w, \cdots, |W|\}$.

Parameters:

c_{ij}	fixed operating cost for traversing the arc $(i,j) \in E$.
$c_{i,j}^{a}$	additional operating cost for monitoring the arc $(i,j) \in E$.
$d_{i,j}$	equals one if there is monitoring demand at the arc $(i,j) \in E$; otherwise, equal zero.
$\bar{Q}_k$	operating capacity of a UAV k in units of cost.
$r_{i,j}$	time for traversing the arc $(i,j) \in E$.
$r_{i,j}^{s}$	time for serving the arc $(i,j) \in E$.
e_i	the earliest serving time at the node i.
l_i	the latest serving time at the node i.
$h_{i,j}$	inventory holding cost for frequent monitoring the arc $(i,j) \in E$ per period.
f_k	charging cost of the UAV k per period.
$q_{i,j}$	the increase in unmet demand for monitoring the arc $(i,j) \in E$ per period.
$\bar{S}_{i,j}$	the maximum 'monitoring satisfaction' of the arc $(i,j) \in E$ per period.
$\bar{V}_k$	battery capacity of the UAV k in units of time length.
M	a sufficiently large positive number.

Decision variables:

$\gamma_{i,j,k,t}$	a binary variable, equals one if the arc $(i,j) \in E$ is traversed by the UAV k at the period t.
$\theta_{i,j,k,t}$	a binary variable, equals one if the arc $(i,j) \in E$ is served by the UAV k at the period t.
$\sigma_{i,j,k,t}$	a flow variable, eliminates sub-tours.
$\varphi_{i,k}$	the arrival time at the node i of the UAV per period.
$\tau_{i,j,t}$	'monitoring satisfaction' of the arc $(i,j) \in E$ at the period t.
β_k	charging quantity of the UAV k per period.

4.2 Mathematical model

Minimize $Z = \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} \sum_{t \in T} c_{ij}\, \gamma_{i,j,k,t}$

$$+\sum_{i\in N}\sum_{j\in N}\sum_{t\in T} h_{i,j}\tau_{i,j,t} + \sum_{k\in K} W f_k \beta_k$$

(1)

Subject to:

$$\sum_{p\in N}\gamma_{p,i,k,t} - \sum_{p\in N}\gamma_{i,p,k,t} = 0 \qquad \forall i\in N, k\in K, t\in T$$

(2)

$$\sum_{k\in K}\left(\theta_{i,j,k,t} + \theta_{j,i,k,t}\right) \le 1 \qquad \forall (i,j)\in E, t\in T \tag{3}$$

$$\gamma_{i,j,k,t} \ge \theta_{i,j,k,t} \qquad \forall (i,j)\in E, k\in K, t\in T \tag{4}$$

$$\sum_{i\in N}\sum_{j\in N}\left(c_{ij}\gamma_{i,j,k,t} + c_{i,j}^{a}\theta_{i,j,k,t}\right) \le \bar{Q}_k \qquad \forall k\in K, t\in T \tag{5}$$

$$\sum_{p\in N}\sigma_{i,p,k,t} - \sum_{p\in N}\sigma_{p,i,k,t} = \sum_{j\in N}\theta_{i,j,k,t} \qquad \forall i\in N\backslash\{0\}, k\in K, t\in T$$

(6)

$$\sigma_{i,j,k,t} \le n^2\gamma_{i,j,k,t} \qquad \forall (i,j)\in E, k\in K, t\in T \tag{7}$$

$$\varphi_{i,k} + t_{i,j} + t_{i,j}^{s} + M\left(1-\theta_{i,j,k,t}\right) \le \varphi_{j,k} \qquad \forall (i,j)\in E, k\in K, t\in T \tag{8}$$

$$\tau_{i,j,t-1} \ge q_{i,j} + \tau_{i,j,t} - M\sum_{k\in K}\left(\theta_{i,j,k,t} + \theta_{j,i,k,t}\right) \qquad \forall (i,j)\in E, t\in T \tag{9}$$

$$\tau_{i,j,t-1} \le q_{i,j} + \tau_{i,j,t} + M\sum_{k\in K}\left(\theta_{i,j,k,t} + \theta_{j,i,k,t}\right) \qquad \forall (i,j)\in E, t\in T \tag{10}$$

$$\tau_{i,j,t} \ge \bar{S}_{i,j} - M\left(1-\sum_{k\in K}\left(\theta_{i,j,k,t} + \theta_{j,i,k,t}\right)\right) \qquad \forall (i,j)\in E, t\in T \tag{11}$$

$$\sum_{j\in N\backslash\{0\}}\gamma_{1,j,k,t+w} \le M\left(1-\sum_{j\in N\backslash\{0\}}\gamma_{1,j,k,t}\right) \qquad \forall k\in K, t\in T, w\in W \tag{12}$$

$$\sum_{i\in N}\sum_{j\in N}\sum_{k\in K}\left(r_{i,j}\gamma_{i,j,k,t} + r_{i,j}^{s}\theta_{i,j,k,t}\right) \le W\beta_k \qquad \forall t\in T \tag{13}$$

$$e_i \le \varphi_{i,k,t} \le l_i \qquad \forall i\in N, k\in K, t\in T \tag{14}$$

$$q_{i,j} \le \tau_{i,j,t} \le \bar{S}_{i,j} \qquad \forall (i,j)\in E, t\in T \tag{15}$$

$$0 \le \beta_k \le \bar{V}_k \qquad \forall k\in K \tag{16}$$

$$\gamma_{i,j,k,t} \in \{0,1\} \qquad \forall (i,j)\in E, k\in K, t\in T$$

(17)

$$\theta_{i,j,k,t} \in \{0,1\} \qquad \forall (i,j)\in E, k\in K, t\in T \tag{18}$$

$$\sigma_{i,j,k,t} \ge 0 \qquad \forall (i,j)\in E, k\in K, t\in T \tag{19}$$

The objective of the model is to minimize the total traversal costs of UAVs. Constraints (2) are the flow conservation constraints. Constraints (3) impose that each arc with monitoring demand will be served by the UAV exactly once. Constraints (4) ensure that arc (i, j) can be served only if UAV k covers the arc. Constraints (5) indicate the operating capacity of UAVs. Constraints (6) and (7) eliminate illegal sub-tours. Constraints (8) are the time constraint for the UAVs to reach different demand arcs. Constraints (9)-(11) ensure that either the 'monitoring satisfaction' will be reduced in the next period if there is no monitoring requirement during one period, or be reached to the maximum if there is monitored in one period. Constraints (12) impose that UAVs cannot work when charging. Constraints (13) ensure that the total power consumption of UAVs cannot exceed the total quantity of charging. Constraints (14) are service time window constraints of UAVs. Constraints (15) state the upper and lower bounds of 'monitoring satisfaction'. Constraints (16) are the battery capacity constraints. Constraints (17)-(19) are nonnegative and integer constraints.

5. Algorithmic strategies

The established model can be solved immediately by CPLEX solver when facing small-scale problem instances. However for some large-scale instances, the direct solving process by the CPLEX solver is time consuming, and even the problem scale is intractable for the CPLEX. Thus this paper designs the local branching based method to solve the established model. The core idea of the local branching strategy is to use the CPLEX solver as a black-box 'tactical' tool to explore suitable solution subspaces defined and controlled at a 'strategic' level by a simple external branching framework (Fischetti and Lodi, 2003).

In this study, two sets of binary variables $\gamma_{i,j,k,t}$ and $\theta_{i,j,k,t}$ are included in the mixed integer programming model. It is obvious that the kernel variable $\theta_{i,j,k,t}$ is important for searching the solution space of the problem. And the solving speed of the model is mainly limited by the branching process of the binary variable $\theta_{i,j,k,t}$. In consequence, local branching is proposed to solve the model. The local branching strategy is exact in nature; the procedure is in the spirit of well-known local search metaheuristics, but the neighborhoods are obtained by introducing local branching cuts. This method can be regarded as a two-level branching strategy aimed at favoring early updating of the incumbent solution, hence producing improved solutions at early stages of the computation. It is noteworthy that setting a time limit for solving each sub-problem embedded in the procedure in order to control the total solving time within a reasonable range. If the time reaches the limit, the CPLEX solver will stop and return a feasible but non-optimal solution; otherwise the optimal solution is gained by the CPLEX solver. Furthermore, it is rather remarkable that the parameter u should be set as a proper value so as to contain a large number of feasible solutions. The basic scheme of the local branching is displayed in Figure 3.

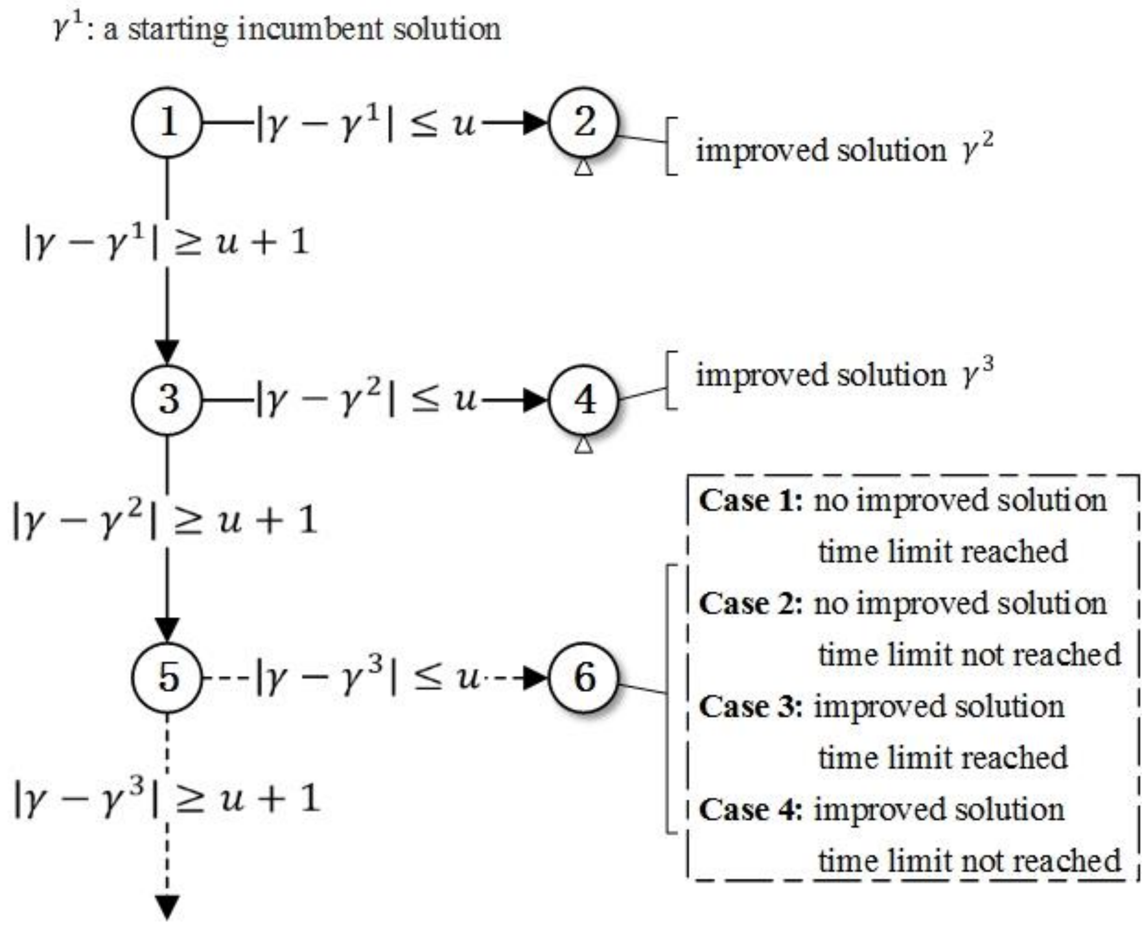

Figure 3: The basic procedure of local branching

In Figure 3, the nodes marked by the triangles correspond to the solution from the CPLEX solver. A starting incumbent solution γ^1 is gained by the CPLEX solver at the root node 1. The right-branch node 2 corresponds to the optimization within the neighborhood $|\gamma - \gamma^1|$, which is performed through CPLEX solver converging to an optimal solution in the neighborhood, say γ^2. Then this solution becomes the new incumbent solution and is re-applied to the left-branch node 3, where the exploration of $|\gamma - \gamma^2|, |\gamma - \gamma^1|$ at node 4 produces a new incumbent solution γ^3. Hereafter node 5 is addressed, which corresponds to the initial problem amended by the two additional constraints $|\gamma - \gamma^1| \geq u + 1$ and $|\gamma - \gamma^2| \geq u + 1$.

There can be four different cases when solving each sub-problem by calling to CPLEX， such as right-branch node 6. In the case 1, the set time limit is reached, but the target value is not improved. It means that the non-optimal new solution obtained through the CPLEX solver is worse than the existing best solution. In the case 2, the target value is improved within the set time limit; the new solution obtained through the CPLEX solver is worse than the existing best solution. However, it is an optimal one for the sub-problem under this situation. In the case 3, although the set time limit is reached, the target value is improved. It signifies that there is a non-optimal new solution obtained through the CPLEX solver, which is better than the existing best solution. In the case 4, the obtained target value through the CPLEX solver is improved within the set solving time limit. It implies the optimal solution of the special sub-problem is better than the existing best one.

The coping strategies for the above four cases are demonstrated in Figure 4. For the case 1, if the time limit is reached with no improved solution, we should reduce the size of the neighborhood in an attempt to accelerate its exploration. This is obtained by reducing the left-hand side term by $\lfloor u/2 \rfloor$. Node 5 has three child branches: node 6, for which the time limit is reached with no improved solution, node $6'$, for which the reduction of the neighborhood size for finding a better solution in the neighborhood, and node 7. For the case 2, node 6 will be cut off due to the new solution is worse than the incumbent best solution. For the case 3, we backtrack to the father node and create a new node associated with the new incumbent solution, without modifying the value of parameter u. In the lower left corner of Figure 4, where node 5 has three son branches: node 6, for which the time limit is reached with an improved solution γ^3, and the regular two branch nodes $6'$ and 7. For the case 4, we evolve the father node to the better one and continue to search the solution by the CPLEX solver. This situation denotes the standard flow of the local branching procedure, as shown in Figure 3.

All possible situations throughout the process are embodies in the above four cases. In this solving process, a quantity of branching cuts is defined to achieve the neighborhood advancement. If the solution has not been improved over a set number of consecutive iterations, the whole solving mentation will be terminated.

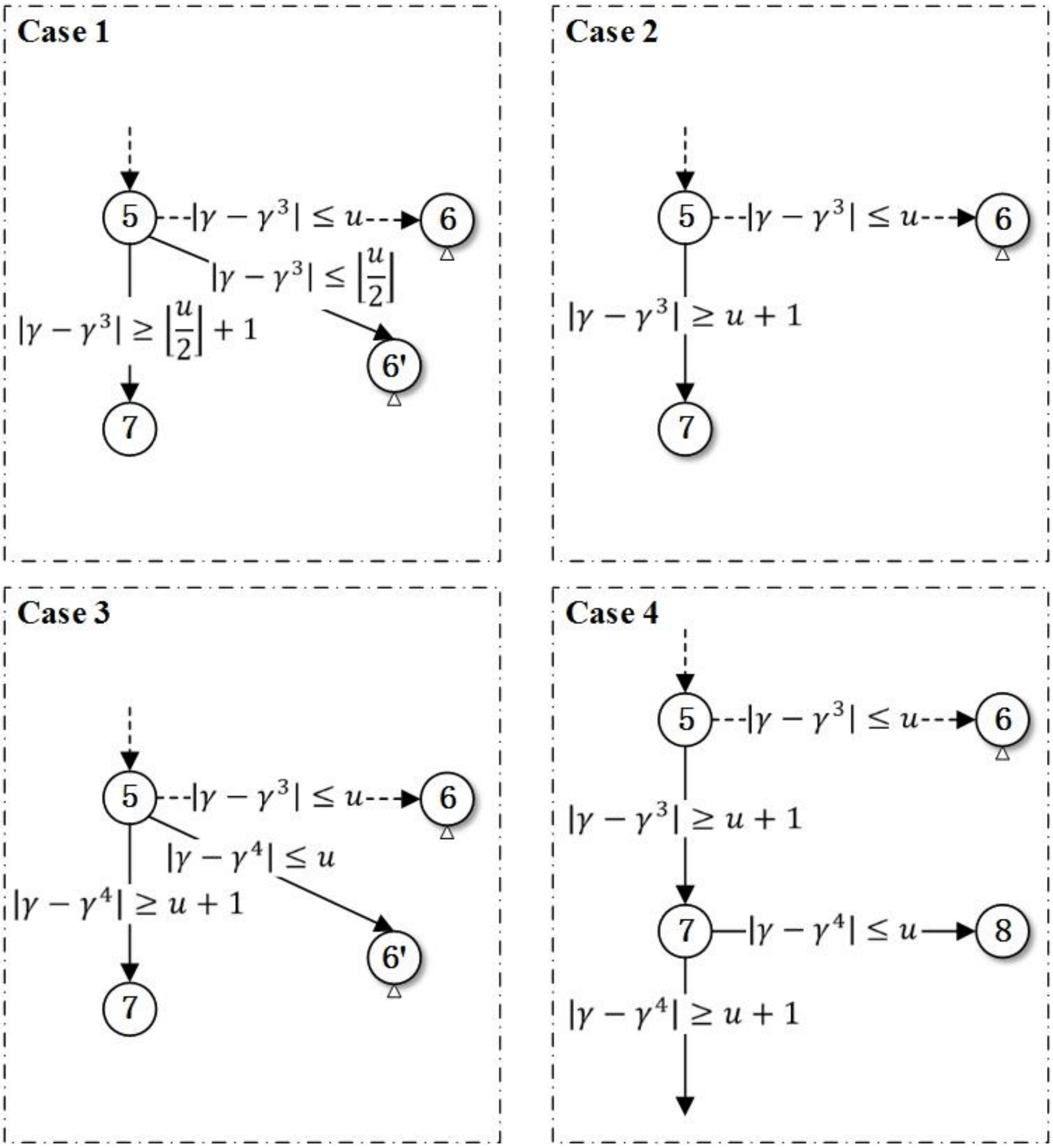

Figure 4: The coping strategies of four different cases

6. Model application and experiments

In this section, the proposed models and solution algorithms are applied to a practical problem of traffic flow monitoring in Shanghai, China. Besides, some numerical experiments are conducted on a PC (Intel Core i5, 1.70G Hz; Memory, 8G) by CPLEX12.6.1 (Visual Studio 2015, C#) to validate the feasibility of the proposed model and the effectiveness of the solution method.

6.1 Problem background

The model application was implemented in the traffic network of Shanghai. Shanghai, as a representative of large cities in China, has extremely serious traffic congestion and various periods of urban traffic congestion situations are not the same. According to the actual situation in Shanghai, morning and evening peak hours are usually 7:30 am ~ 9:30 am and 16:30 pm ~ 18: 30 pm; in these two time periods, the traffic flow of the whole network is extreme large and there will be a large area of congestion. During 12:00 pm ~ 14:00 pm in the working days, the traffic flow of the whole network is larger than usual[what does ‘usual’ mean?] and the local area congestion is more significant. The different time periods of Shanghai traffic congestion can be seen in Figure 5. The traffic congestion is divided into four levels, red is ‘Blocked’, yellow is ‘Crowded’, light green is ‘Smooth’, green is ‘Unblocked’. The above situation of traffic congestion is based on a large number of actual traffic history data.

Figure 5: The situation of traffic congestion in Shanghai [这两个图分别是指哪个时间段的？]

6.2 Sensitivity test of model parameters

In CARP network, the same UAVs or different UAVs can access a number of undirected arcs and there are capacity constraints. Table 1 shows several solution results of the model under different capacity quantity constraints.

From Table 1 we can see that when the scale is extreme small, capacity constraints have little effect on the solution time, but too small capacity may make the model cannot get the optimal solution. Then, when the scale increases slightly, the effect of capacity constraints is more obvious. Thus setting an appropriate operating capacity value will help find a better solution in a shorter time.

Table 1: Comparisons under different operating capacities

Instance ID	Operating Capacity	OBJ	CPU Time (s)
(3,3)-2-3-1	40	N.A.	0.5
(3,3)-2-3-2	50	1578.0	0.7
(3,3)-2-3-3	55	1578.0	1.0
(3,3)-2-3-4	80	1578.0	0.9
(5,5)-2-3-1	80	3755.0	414.7
(5,5)-2-3-2	100	3031.0	62.1
(5,5)-2-3-3	150	2751.0	12.2
(5,5)-2-3-4	160	2751.0	11.9
(6,6)-4-4-1	150	4562.0	1681.3
(6,6)-4-4-2	180	4562,0	2839.4
(6,6)-4-4-3	250	4562.0	823.7
(6,6)-4-4-4	300	4562.0	3404.3

Notes: (1) The numbers in each case id (e.g., '(4,4)-2-3-1') denote the number of arcs (i,j) '(4,4)', the number of vehicles '2', the number of periods '3', the index of the case '1st', respectively.

6.3 Sensitivity test of algorithm parameters

Before verifying the effectiveness of the algorithm, some parameters used in proposed method should be determined. On one hand, u is a critical parameter to control the neighborhood size of solving each node during the branching process. On the other hand, the time limits for solving each node by the CPLEX solver also need to be set to a proper value.

Table 2: Computation time under different parameter settings

CPU Times (s)	The Time Limit on Each Node 10 (s)	20 (s)	40 (s)	80 (s)	160 (s)
Par u = 5	584.6	414.0	242.9	447.2	870.0
Par u = 10	339.0	251.1	170.9	323.9	539.1
Par u = 25	543.7	427.5	348.5	522.2	716.7
Par u = 50	842.1	682.4	531.7	885.3	958.4
Par u = 100	1009.2	943.0	898.4	1024.0	1359.2

Some experiments are conducted to explore the influence of the above parameter settings on the computation time of solution result, which is illustrated in Table 2. The number of iterations mentioned in the local branching is set to 100. It is to impose that the whole solution procedure terminates if the incumbent best objective value has not been improved for 100 consecutive iterations.

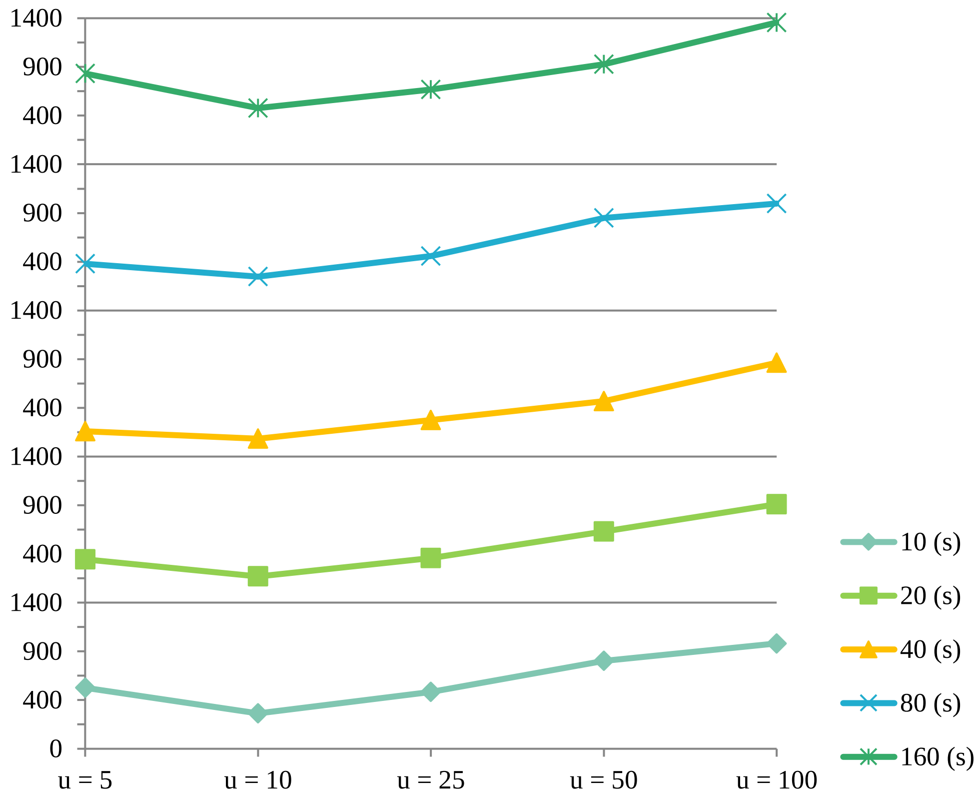

Figure 6: Computation time under different parameter settings

The results can be seen intuitively in Figure 6, the value of parameter u and the time limit on each node are not positively and negatively related to the computation time. Figure 6 indicates that the setting of u as 10 and the time limit on each node as 40 seconds is proper for this problem. Therefore the next experiments will use the above setting for the solution method.

6.4 Experiments on the proposed solution method

There are few related works that discuss problems that are similar as this paper; accordingly we mainly compare our proposed solution method with the widely used CPLEX solver so as to evaluate performance of our method.

Table 3: Comparison between the proposed method and the CPLEX solver for small scale

Instance ID	CPLEX solving directly		Local branching		OBJ Gap
	Z_{CPLEX}	T_{CPLEX} (s)	Z_{LoBr}	T_{LoBr} (s)	
(3,3)-2-3-1	1192.0	1.0	1192.0	1.8	0.00%
(3,3)-2-3-2	1122.0	0.9	1122.0	2.1	0.00%
(3,3)-2-3-3	1017.0	1.1	1017.0	2.0	0.00%
(3,3)-2-3-4	1094.7	1.0	1094.7	1.9	0.00%
(3,3)-2-3-5	868.0	0.9	868.0	2.0	0.00%
(4,4)-2-3-1	2169.0	1.7	2169.0	3.4	0.00%
(4,4)-2-3-2	2271.3	1.4	2271.3	3.6	0.00%
(4,4)-2-3-3	1834.0	1.5	1834.0	3.1	0.00%
(4,4)-2-3-4	1758.3	1.4	1758.3	2.7	0.00%
(4,4)-2-3-5	1863.7	1.2	1863.7	3.5	0.00%
(5,5)-3-4-1	4927.0	8.9	4927.0	37.6	0.00%
(5,5)-3-4-2	3102.0	4.8	3102.0	17.9	0.00%
(5,5)-3-4-3	4485.0	9.6	4485.0	41.9	0.00%
(5,5)-3-4-4	5284.0	4.6	5284.0	13.7	0.00%
(5,5)-3-4-5	4606.0	10.8	4606.0	41.2	0.00%
Avg.		3.4		11.9	0.00%

Notes: (1) 'Z' denotes the objective values; 'T' denotes the computation time, and the unit is the second. (2) $GAP_{OBJ} = (Z_{LoBr} - Z_{CPLEX}) / Z_{CPLEX}$

In general, the results solved by the CPLEX solver directly are regarded as the optimal solutions, and the proposed solution method can obtain feasible solutions. In order to investigate the efficiency of the proposed solution method, a series of experiments in different scales are designed. As shown in Table 3, the CPLEX solver can solve the model more quickly than the proposed method in small scales. However, both of them can get the optimal solutions. When the demand arcs exceed a certain scale, the CPLEX solver cannot catch the optimal solutions, but the proposed solution method can obtain the objective value more quickly than the CPLEX solver, which can be listed in Table 4. Although sometimes the CPLEX solver gets better results than the proposed solution method, the computing time of the CPLEX solver is longer. In addition, the average gap between the results obtained by the proposed solution method and the optimal result is about 0.105%, which indicates the results are accurate. It means the proposed solution method is appropriate for solving the formulated model, and may also be potentially applicable to solve more complicated problem.

Table 4: Comparison between the proposed method and the CPLEX solver for large scale

Instance ID	CPLEX solving directly		Local branching		OBJ Gap
	Z_{CPLEX}	T_{CPLEX} (s)	Z_{LoBr}	T_{LoBr} (s)	
(6,6)-4-4-1	6924.0	876.1	6924.0	411.8	0.00%
(6,6)-4-4-2	5907.9	802.9	5947.1	406.9	0.66%
(6,6)-4-4-3	6500.9	455.1	6530.0	212.4	0.45%
(6,6)-4-4-4	6362.1	1004.8	6362.1	513.1	0.00%
(6,6)-4-4-5	5120.4	1128.2	5122.0	506.9	0.03%
(8,8)-4-5-1	8896.6	1966.0	8896.6	609.7	0.00%
(8,8)-4-5-2	11767.2	3017.4	11767.2	731.0	0.00%
(8,8)-4-5-3	10902.1	2906.0	10919.5	827.2	0.16%
(8,8)-4-5-4	12692.0	2891.1	12696.8	720.0	0.04%
(8,8)-4-5-5	10088.5	2237.3	10131.5	880.4	0.43%
(10,10)-5-5-1	15120.0	3786.5	15129.7	916.9	0.06%
(10,10)-5-5-2	20892.2	4142.8	20970.4	983.2	0.37%
(10,10)-5-5-3	15135.9	4464.5	15135.9	1018.9	0.00%
(10,10)-5-5-4	19058.0	5318.9	19226.6	1242.1	0.88%
(10,10)-5-5-5	18628.6	4272.1	18640.5	1023.1	0.06%
(15,15)-6-5-1	N.A.	>7200.0	26366.0	1470.7	N.A.
(15,15)-6-5-2	N.A.	>7200.0	33181.6	1437.8	N.A.
(15,15)-6-5-3	N.A.	>7200.0	38035.5	1395.4	N.A.
(15,15)-6-5-4	N.A.	>7200.0	26348.5	1655.9	N.A.
(15,15)-6-5-5	N.A.	>7200.0	31300.5	1631.0	N.A.
Avg.					0.21%

7. Conclusions

This study proposes a mixed integer programming model for traffic monitoring, which optimizes the decisions of UAV scheduling with the objective of minimizing the total expected costs of operation under uncertain monitoring demands. Some advances and contributions of this study mainly include:

(1) The UAVs are introduced as a supplementary means of urban road traffic inspection equipment as to improve the accuracy of traffic monitoring. As mobile sensors, UAVs are more appropriate to monitor demand arcs than demand nodes. Thus in the deployment of UAVs, monitoring based on demand arcs is considered in this paper. The focus of most works is based

on demand node.

(2) The proposed UAVs route planning optimization model combines the basic model of capacitated arc routing problem with time window and inventory routing problem, which provide a framework for multiple-period real-time traffic monitoring by UAVs. Most of the existing related studies did not take into account this problem.

(3) The proposed local branching based method has good solution efficiency. Especially when the scale of the problem is expanded, local branching can search for a better solution in a shorter calculation time and approximate the exact solution. This provides an effective decision-making way to solve the UAVs scheduling problem.

However, there are limitations for the current study. For example, the traffic information collected by UAVs can be used as assistance and supplement of ground fixed traffic monitors, how to effectively integrate information from both of them is a worthy study. Furthermore, the local branching based solution method is a heuristic method in nature. How to design an exact solution method for solving the large-scale problem instance is still a challenging task for further studies. All of these limitations will form the research directions for our future studies.

References

Azadeh, A., Elahi, S., Farahani, M.H. & Nasirian, B. (2017). A genetic algorithm-Taguchi based approach to inventory routing problem of a single perishable product with transshipment. *Computers & Industrial Engineering*, 104, 124-133.

Beltrami, E.J. & Bodin, L.D. (1974). Networks and vehicle routing for municipal waste collection. *Networks*, 4, 65-94.

Cho, J., Lim, G., Biobaku, T., Kim, S. & Parsaei, H. (2015). Safety and Security Management with Unmanned Aerial Vehicle (UAV) in Oil and Gas Industry. *Procedia Manufacturing*, 3, 1343-1349.

Eglese, R. & Li, L. (1992). Efficient routeing for winter gritting. *Journal of the Operational Research Society*, 43, 1031-1034.

Federgruen, A. & Zipkin, P. (1984). A combined vehicle routing and inventory allocation problem. *Operations Research*, 32, 1019-1037.

Fischetti, M. & Lodi, A. (2003). Local branching. *Mathematical Programming*, 98, 23-47.

Golden, B., Assad, A. & Dahl, R. (1984). Analysis of a large scale vehicle routing problem with an inventory component. *Large scale systems*, 7, 181-190.

Golden, B.L. & Wong, R.T. (1981). Capacitated arc routing problems. *Networks*, 11, 305-315.

Guerriero, F., Surace, R., Loscrí, V. & Natalizio, E. (2014). A multi-objective approach for unmanned aerial vehicle routing problem with soft time windows constraints. *Applied Mathematical Modelling*, 38, 839-852.

Harwin, S. & Lucieer, A. (2012). Assessing the Accuracy of Georeferenced Point Clouds Produced via Multi-View Stereopsis from Unmanned Aerial Vehicle (UAV) Imagery. *Remote Sensing*, 4, 1573.

Lacomme, P., Prins, C. & Ramdane-Chérif, W. (2005). Evolutionary algorithms for periodic arc routing problems. *European Journal of Operational Research*, 165, 535-553.

Li, K., Chen, B., Sivakumar, A.I. & Wu, Y. (2014). An inventory–routing problem with the objective of travel time minimization. *European Journal of Operational Research*, 236, 936-945.

Li, L.Y.O. & Eglese, R.W. (1996). An Interactive Algorithm for Vehicle Routeing for Winter — Gritting. *Journal of the Operational Research Society*, 47, 217-228.

Lopes, R.B., Plastria, F., Ferreira, C. & Santos, B.S. (2014). Location-arc routing problem: Heuristic approaches and test instances. *Computers & Operations Research*, 43, 309-317.

Mei-Ko, K. (1962). Graphic programming using odd or even points. *Chinese Math.*, 1, 273-277.

Monroy, I.M., Amaya, C.A. & Langevin, A. (2013). The periodic capacitated arc routing problem with irregular services. *Discrete Applied Mathematics*, 161, 691-701.

Murray, C.C. & Chu, A.G. (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies*, 54, 86-109.

Riquelme-Rodríguez, J.-P., Gamache, M. & Langevin, A. (2016). Location arc routing problem with inventory constraints. *Computers & Operations Research*, 76, 84-94.

Russell, R. & Igo, W. (1979). An assignment routing problem. *Networks*, 9, 1-17.

Salazar-Aguilar, M.A., Langevin, A. & Laporte, G. (2013). The synchronized arc and node routing problem: Application to road marking. *Computers & Operations Research*, 40, 1708-1715.

Salvo, G., Caruso, L. & Scordo, A. (2014). Urban Traffic Analysis through an UAV. *Procedia - Social and Behavioral Sciences*, 111, 1083-1091.

Schuijbroek, J., Hampshire, R.C. & van Hoeve, W.J. (2017). Inventory rebalancing and vehicle routing in bike sharing systems. *European Journal of Operational Research*, 257, 992-1004.

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, 674-692.

Yakıcı, E. (2016). Solving location and routing problem for UAVs. *Computers & Industrial Engineering*, 102, 294-301.

Zhen, L., Zhuge, D. & Zhu, S.-L. (2016). Production stage allocation problem in large corporations. *Omega*, in press, doi: 10.1016/j.omega.2016.11.009