Journal of Intelligent & Robotic Systems
https://doi.org/10.1007/s10846-018-0806-8

Real-time UAV Rerouting for Traffic Monitoring with Decomposition Based Multi-objective Optimization

Xiaofeng Liu[1] · **Zhong-Ren Peng**[2] · **Li-Ye Zhang**[3]

Received: 22 February 2017 / Accepted: 28 February 2018

Abstract

This paper introduces unmanned aerial vehicle (UAV) to monitor traffic situation, and considers the UAV real-time rerouting problem. Firstly, critical target is introduced at the time of UAV route re-planning, which is used to identify the existing visited targets and the remaining unvisited targets. Meanwhile, a real-time UAV rerouting model is proposed with the consideration of time window and multi-objective optimization. Then, a target insertion method is used to generate feasible UAV routes, and a decomposition based multi-objective evolutionary algorithm is proposed. Next, a case study and algorithm sensitivity analysis are implemented, and the results show that compared with the initial optimal solutions, the optimized optimal solutions are improved significantly. In addition, the proposed algorithm is compared with the non-dominated sorting genetic algorithm II (NSGA-II), the case study shows that the proposed algorithm outperforms NSGA-II in terms of computational time, the percentage of finding optimal UAV routes and solution quality. It suggests that the proposed algorithm is promising in planning UAV cruise routes.

Keywords Traffic monitoring · Unmanned aerial vehicle · Real-time rerouting · Multi-objective optimization

1 Introduction

In recent years, many countries issued different flight policies to promote the applications of unmanned aerial vehicle (UAV) in the civil market. In 2016, the US Federal Aviation Administration (FAA) allowed small UAVs weighing less than 55 pounds (25 kg) to fly without FAA airworthiness certification [1]. The Civil Aviation Administration of China (CAAC) approved that UAV indoor flights, UAV flights within visual line of sight, and UAV flights in areas of few population are exempted from CAAC regulation and approval [2]. In addition, due to the maturing UAV sensing technology, the easy operation, the lowing cost and its unique advantages of flexibility, mobility and wide-view, UAV has used broadly for traffic monitoring, such as traffic flow parameter estimation [3], traffic incident detection [4], road condition monitoring [5], bridge crack inspection [6], and so on. Before deploying UAVs to conduct traffic monitoring, it's necessary to find the optimal UAV cruise routes.

Many research studied UAV cruise route planning problem in the past decades. HUTCHISON [7] deployed UAVs to monitor targets of an area and formulated the UAV cruise route optimization as a traveling salesman problem (TSP), which aimed to minimize the total cruise distance. Then, the simulated algorithm was used to find the shortest cruise routes. YAN et al. [8] used UAVs to collect traffic information of road sections in a sparse road network, and the UAV routing problem was also formulated as a TSP problem, and then genetic algorithm was adopted to search

This work was supported in part by the National Natural Science Foundation of China award 51408417, 61503284, and the Science and Technology Project of Tianjin award 17KPXMSF00010, 16PTGCCX00150,17ZXRGGX00070, and 16JCZDJC38200.

✉ Xiaofeng Liu
microbreeze@126.com

Zhong-Ren Peng
zpeng@ufl.edu

Li-Ye Zhang
zhangly@ihpc.astar.edu.sg

[1] School of Transportation and Automotive, Tianjin University of Technology and Education, Tianjin 300222, China

[2] Department of Urban and Regional Planning, University of Florida, Gainesville 32611, USA

[3] Institute of High Performance Computing, A*STAR, Singapore 138632, Singapore

Published online: 12 March 2018

the shortest cruise routes. PITRE et al. [9] used UAVs for joint search and track missions, aimed to maximize the target detection, target tracking and vehicle survivability, and then the particle swarm optimization algorithm was adopted to optimize UAV cruise routes. PENG et al. [10] considered a UAV three-dimensional multi-constraint route planning problem, which aimed to minimize the weighted sum of voyage cost, altitude cost and threat cost, then a coevolutionary multi-agent genetic algorithm was proposed to find the UAV feasible flight routes. LIU et al. [11] formulated the UAV cruise route optimization as a multi-objective vehicle routing problem, which aimed to minimize the total UAV cruise distance and the number of unmonitored targets, and then the non-dominated sorting genetic algorithm II (NSGA-II) was adopted to find the optimal routes. The above-mentioned studies formulate the UAV cruise route planning into an optimization problem, and heuristic algorithms are used to find the optimal solutions. In addition, artificial potential field method [12], partially observable Markov decision process method [13], and Voronoi diagram method [14] were used to plan UAV routes, but time window (e.g., traffic peak hours) is not considered in these studies, which is crucial for traffic monitoring.

In summary, the previous studies focus on the static or deterministic UAV route planning problem, i.e., all information is known at the time of planning UAV routes. However, in real-life traffic monitoring, UAV monitoring demand may be dynamic and time-dependent. For example, there is an accident occurring in a road section in morning peak hours, it's necessary to deploy UAVs to monitor the new emerging road section in a given time period. In this condition, it is essential to adjust the UAV cruise routes immediately. The dynamism of this real-time UAV rerouting problem needs to consider the existing visited targets, new targets, the remaining unvisited targets and the target time window constraint together. In addition, optimal UAV cruise route planning heavily relies on the time-consuming optimization techniques (e.g., numerical computation). As to the real-time UAV cruise route planning problem, it's essential to find the optimal routes as soon as possible, which is the focus of this study.

Therefore, this study aims at the real-time UAV rerouting problem, and establishes a multi-objective UAV route optimization model to minimize the number of UAVs used and the total cruise cost, and then proposes a decomposition based multi-objective evolutionary algorithm. The major contributions of this study include the modeling of the real-time UAV rerouting problem, and the improvement of computational time and solution quality for UAV cruise route planning.

2 Model

2.1 Problem Description

The traditional procedure of UAV route planning for traffic monitoring can be described as follows. First, based on the traffic monitoring demand (e.g., traffic data collection, congestion surveillance, and accident investigation), traffic control center selects monitoring targets, and acquires their locations and monitoring service time. Then, the UAV cruise routes are optimized. Finally, according to the optimized routes, UAVs take off from the base, cruise the targets, take pictures/videos of the targets, transmit these pictures/videos back to the traffic control center via wireless communication, and return to the base.

However, the real-life traffic monitoring demand is dynamic. For example, UAVs conduct routine traffic monitoring according to the pre-set cruise routes, suddenly, a traffic accident occurs and leads to severe traffic congestion at an intersection in the morning peak hours, traffic control center needs to know more information of the accident in a given time period, thus how to adjust the UAV cruise routes in real time becomes an emerging problem.

The goal of this real-time UAV rerouting problem is to obtain a set of UAV routes with minimum cruise cost, and find the optimal routes as soon as possible. In order to demonstrate the problem clearly, Fig. 1 is presented and the concept of critical target is introduced. A critical target is defined as a target which a UAV is currently monitoring, or a target which a UAV is heading to [15]. Critical targets should be identified immediately so that the new routes can be found.

As Fig. 1 shows, there are thirteen targets and a UAV base in the road network. At the initial time t_0, there are three optimized UAV routes, i.e., 0-8-7-11-0, 0-2-12-6-4-3-0, and 0-9-10-13-1-5-0. At the time t_1, traffic control center knows that a new target 14 appears, at this time, the route 0-8-7-11-0 is finished, and the remaining routes are not finished. At this time, one UAV is flying from target 12 to target 6, and one UAV is monitoring target 9. At the time t_1, target 6 and 9 are critical targets. In this context, the real-time UAV rerouting should consider the visited targets (i.e., target 2 and 12), unvisited targets (i.e., target 3, 4, 10, 13, 1, 5, and the new target 14), and critical targets (i.e., target 6 and 9). As to the new planned routes, they must originate from the UAV base or the critical targets, monitor each unvisited target and new target(s), and return to the UAV base. Then, the real-time UAV cruise routes are optimized within the computational time Δt. At the time t_2, new UAV cruise routes are generated.

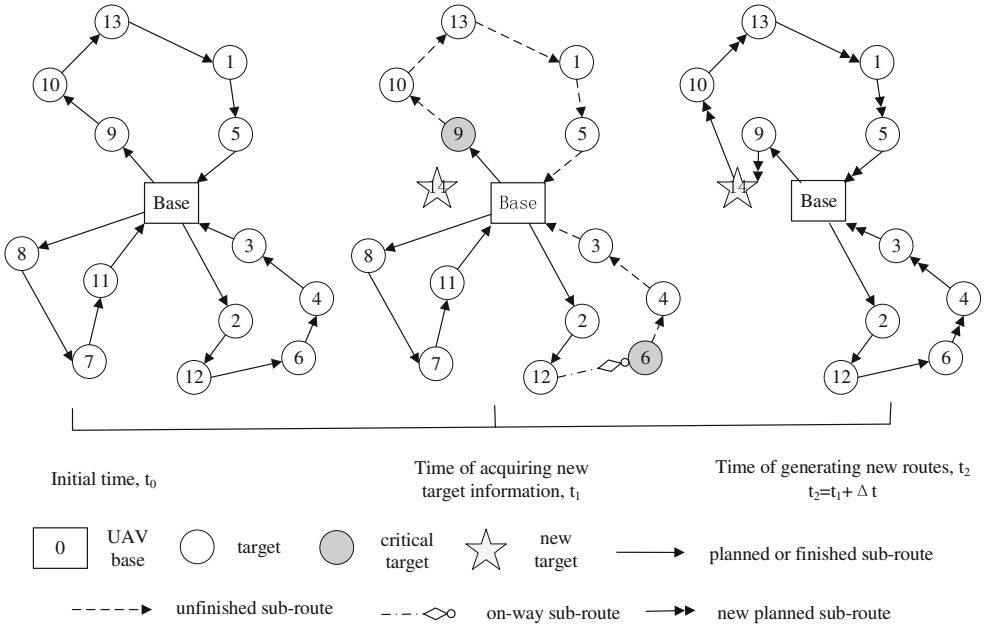

Fig. 1 Illustration of real-time UAV rerouting

2.2 Model Establishment

It is noted that at the time t_1, the targets of finished routes must be deleted before modeling (e.g., target 8, 7, and 11 in Fig. 1). And the remaining targets are grouped as three categories: visited, unvisited, and critical targets. Suppose that the sets of unvisited and critical targets are denoted as N_u and N_c respectively; the set of critical and unvisited targets is denoted as N_{cu}; the set of UAV base and unvisited targets is denoted as N_{0u}; the set of UAV base, critical and unvisited targets is denoted as N_{0cu}; the set of critical, unvisited, and visited targets is denoted as N_{cuv}; the set of UAV base, critical, unvisited, and visited targets is denoted as N_{0cuv}; and the set of UAVs is denoted as K.

Each UAV route of target pair (i, j) has its Euclidean distance and flight time, which are denoted as d_{ij} and t_{ij} respectively; the time window of target i is denoted as $[a_i, b_i]$, where a_i and b_i are the start time and the end time. If a UAV arrives at the target before the start time, it needs to wait until the start time begins, and if a UAV has monitored target i in the time window, it can leave to another target. The waiting time, monitoring service time and start monitoring time of target i are denoted as w_i, s_i and T_i respectively. As to the UAV base, $w_0 = s_0 = T_0 = 0$. The number of UAVs available is denoted as N, and the UAV maximum flight distance is denoted as D.

The optimization objectives are listed as follows.

$$\min f_1 = \sum_{k\in K}\sum_{j\in N_{cuv}} x_{k0j} \tag{1}$$

$$\min f_2 = \sum_{k\in K}\sum_{i\in N_{0cuv}}\sum_{j\in N_{0cuv}} x_{kij}\times(d_{ij}+\alpha\times s_i+\beta\times w_j) \tag{2}$$

The first objective is to minimize the number of UAVs used, and the second objective is to minimize the total cruise cost, which includes the UAV flight cost, monitoring cost, and waiting cost. In Eq. 2, α and β are the distance conversion coefficients of monitoring service time and waiting time respectively. Moreover, α is greater than β because monitoring service will cause more cost than waiting. The constraints are shown as follows.

$$\sum_{k\in K}\sum_{j\in N_{0u}} x_{kij} = 1, \forall i \in N_{cu} \tag{3}$$

$$\sum_{k\in K}\sum_{i\in N_{0cu}} x_{kij} = 1, \forall j \in N_u \tag{4}$$

$$\sum_{j\in N_{cuv}} x_{k0j} = 1, \forall k \in K \tag{5}$$

$$\sum_{i\in N_{cu}} x_{ki0} = 1, \forall k \in K \tag{6}$$

$$\sum_{k\in K}\sum_{j\in N_{0u}} x_{kij} = 1, \forall i \in N_c \tag{7}$$

$$\sum_{k\in K}\sum_{i\in N_{0cu}} x_{kih} = \sum_{k\in K}\sum_{j\in N_{0u}} x_{khj}, \forall h \in N_u \tag{8}$$

$$a_i \leqslant T_i \leqslant b_i, \forall i \in N_{cu} \tag{9}$$

$$x_{kij} = 1 \Rightarrow T_i + s_i + t_{ij} + w_{ij} = T_j, \forall i, j \in N_{0cu} \tag{10}$$

$$w_i = \max\{0, a_i - T_i\}, \forall i \in N_{cu} \tag{11}$$

$$\sum_{k\in K}\sum_{j\in N_{cuv}} x_{k0j} \leqslant N \tag{12}$$

$$\alpha\cdot\sum_{i\in N_{0cuv}}\sum_{j\in N_{0cuv}} x_{kij}\times s_i+\sum_{i\in N_{0cuv}}\sum_{j\in N_{0cuv}} x_{kij}\times d_{ij}\leqslant D,\forall k\in K \tag{13}$$

Decision variable is given by

$$x_{kij}=\{0,1\},\forall i\in N_{0cuv}, j\in N_{0cuv} \tag{14}$$

Where, Eq. 3 ensures that for each critical or unvisited target, only one UAV leaves from it. Equation 4 ensures that for each unvisited target, only one UAV arrives at it. Equation 5 states that UAVs depart from the base. Equation 6 states that UAVs return to the base. Equation 7 ensures that for each critical target, UAV must leave it when UAV has arrived at or is approaching this target. Equation 8 ensures that for each unvisited target h, the entering UAV must leave this target. Equation 9 denotes the time window constraint of target i. Equation 10 denotes the time precedence of route (i, j). Equation 11 denotes the waiting time of target i. Equation 12 ensures that the number of UAVs available should not be exceeded. Equation 13 ensures that the UAV maximum flight distance should not be exceeded. Equation 14 states that x_{kij} equals to 1 when a UAV flies from target i to target j, otherwise, it equals to 0. It is worth noting that, as to the unfinished UAV route, the x_{kij} of its finished sub-route equals to 1. For example, in Fig. 1, 0-2-12-6-4-3-0 is an unfinished UAV route, sub-route 0-2-12-6 is finished (target 2 and 12 are the visited targets, target 6 is the critical target), thus $x_{k0\text{-}2}$, $x_{k2\text{-}12}$, and $x_{k12\text{-}6}$ equal to 1 respectively.

3 Algorithm

Weighted sum is the most commonly used method to deal with multi-objective optimization problem, which assigns different weights to different objective functions and converts the multi-objective problem into a single objective problem. However, it's difficult to determine the weights, and the impact of multi-objective optimization on the UAV cruise route planning. In this context, Pareto optimality is introduced to solve the multi-objective optimization problem, and several algorithms are developed in the past two decades, such as the nondominated sorting genetic algorithm (NSGA) [16], the strength Pareto evolutionary algorithm (SPEA) [17], the Pareto archived evolutionary strategy (PAES) [18], and the NSGA-II [19]. In recent years, multi-objective evolutionary algorithm based on decomposition (MOEA/D) [20–24], cellular genetic algorithm [25], sub-population genetic algorithm II (SPGA II) [26], hypervolume-based optimization [27, 28] are developed. In addition, some swarm intelligence based algorithms are developed, such as the improved archive-based micro genetic algorithm [29], hybrid particle optimization algorithm [30], hybrid ant colony algorithm [31], and flower pollination algorithm [32]. As to the multi-objective optimization, NSGA-II is the most popular and frequently used algorithm, which uses fast nondominated sorting and crowded distance estimation to select elitist individuals, and establishes tournament population to keep algorithm diversity. MOEA/D algorithm decomposes the optimization problem into a number of single objective optimization subproblems, and optimizes them simultaneously. Additionally, MOEA/D has low computational complexity and can deal with complicated Pareto set shapes [33], which is suitable for real-time UAV route planning, therefore, the decomposition strategy is adopted in this study. In addition, integer-based chromosome generation, OX crossover and multiple-exchange mutation are adopted to construct a decomposition based multi-objective optimization algorithm.

3.1 Tchebycheff Decomposition Approach

Three decomposition methods, i.e., weighted sum approach, Tchebycheff approach, and boundary intersection approach, are commonly used for multi-objective decomposition. Tchebycheff approach can deal with nonconcave Pareto Fronts by altering the weight vector [20]. Therefore, it is used in this study. In this approach, a single objective optimization subproblem is presented as follows.

$$\begin{aligned} Minimize\ \ & g(x|\lambda,z^*)=\max_{1\leqslant i\leqslant m}\{\lambda_i|f_i(x)-z_i^*|\} \\ Subject\ to\ \ & x\in\Omega \end{aligned} \tag{15}$$

Where, $\lambda=(\lambda_1,\ldots,\lambda_m)$ is a weight vector, z^* is the reference point, m is the number of objective functions, Ω is the decision space, and $f(x)$ is the objective function. For all

$$i=1,\ldots,m,\lambda_i\geq 0,\sum_{i=1}^{m}\lambda_i=1,\ \text{and}\ z_i^*=\min\{f_i(x)|x\in\Omega\}.$$

Let $\lambda^1,\ldots,\lambda^N$ be the weight vectors, correspondingly, there are N single objective optimization subproblems. When N is large and the weight vectors are selected reasonably, a good approximation to the Pareto Front of optimization problem, would be provided by the optimal solutions of these subproblems.

3.2 Algorithm Framework

The proposed algorithm includes parameter setting, integer-based chromosome generation, crossover, mutation, and neighborhood chromosome update. The algorithm framework is shown in Fig. 2.

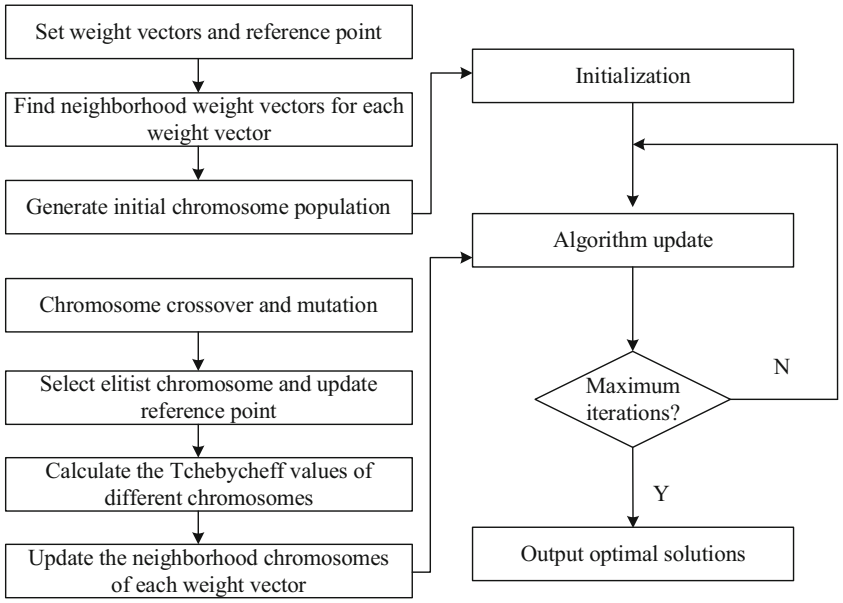

Fig. 2 Framework of the proposed algorithm

The procedure of the proposed algorithm is described as follows. First, the algorithm initializes the weight vectors, determines the T closet weight vectors for each weight vector, generates an initial chromosome population (i.e., each chromosome corresponds to a weight vector, and each vector has T closet weight vectors or T neighborhood chromosomes), and initializes the reference point. Then, the algorithm conducts crossover and mutation operation, in order to keep population diversity and improve the ability of searching local space. In addition, the algorithm selects the elitist chromosome, and updates the new reference point. Next, Tchebycheff approach is used to calculate the Tchebycheff values of different chromosomes, and the Tchebycheff values of the elitist chromosome and neighborhood chromosome are compared, if the former is smaller than the latter, the neighborhood chromosome is substituted by the elitist chromosome, thus to renew the chromosome population (see Step 2 below). Finally, the algorithm outputs the optimal solutions when the maximum iterations are satisfied.

More detailed algorithm steps are described as follows.

Step 1 Initialization

Step 1.1 Initialize the weight vectors $\{\lambda^1,\ldots,\lambda^i,\ldots,\lambda^N\}$, calculate the Euclidean distances between any two of them. Then, find out the T closest weight vectors for each weight vector. The neighborhood set of weight vector λ^i is denoted as $B(i) = \{i_1,\ldots,i_T\}$, the T closest weight vectors of λ^i is denoted as $\{\lambda^{i_1},\ldots,\lambda^{i_T}\}$.

Step 1.2 Generate an initial chromosome population $\{\lambda^1,\ldots,x^N\}$ by using integer-based chromosome generation (see Section 3.3), and the objective value of any chromosome x^i in the population is denoted as $FV^i = F(x^i)$.

Step 1.3 Initialize the reference point $z = (z_1,\ldots,z_m)$, where $z_j = \min_{1\leqslant i\leqslant N} f_j(x^i)$.

Step 2 Algorithm update

For $i = 1,\ldots,N$,do

Step 2.1 Generate a random number ranging from 0 to 1, and conduct crossover operation when the crossover probability is greater than the random number. Select two numbers from the neighborhood set $B(i) = \{i_1,\ldots,i_T\}$ randomly, which correspond to two chromosomes. Then, the two selected chromosomes are used to conduct OX crossover to generate two new chromosomes (see Section 3.5).

Step 2.2 Generate a random number ranging from 0 to 1, and conduct mutation operation when the mutation probability is greater than the random number. Select one number from the neighborhood set $B(i) = \{i_1,\ldots,i_T\}$ randomly, which corresponds to one chromosome. Then, the selected chromosome is used to conduct multiple-exchange mutation to generate a new chromosome (see Section 3.5).

Step 2.3 Update the objective values of the new chromosome population, and the chromosome with minimum objective values is denoted as y, and y is used as the elitist chromosome. Then, update the reference point Z.

For each $j = 1,\ldots,m$, if $z_j > f_j(y)$, then $z_j = f_j(y)$.

Step 2.4 Update neighborhood chromosomes

For each $j \in B(i)$, if $g(y|\lambda^j, z) \leq g(x^j|\lambda^j, z)$, then set $x^j = y$, and $FV^j = F(y)$.

Step 3 Stop and output the optimal solution

If the maximum iterations are satisfied, then stop and output the best solution $\{x^1, \ldots, x^N\}$ and their objective values $\{F(x^1), \ldots, F(x^N)\}$. Otherwise, go to step 2.

3.3 Integer-based Chromosome Generation

In traditional mathematical optimization, the variable is always the real number. However, in the UAV route optimization, a UAV route is consisted of different targets, and each target has an integer number, i.e., a UAV route is an integer number sequence. Therefore, it's important to generate feasible UAV route chromosomes; otherwise, large amount of infeasible route chromosomes will cause more computational time and meaningless work. The integer-based target arrangement method [11] is used to represent the UAV route chromosome, in which each target has an integer number and each UAV route chromosome is consisted of different integer numbers. For example, in Fig. 1, there are thirteen targets in the road network, and one UAV route chromosome can be represented as 5-7-13-2-4-8-9-11-1-3-6-10-12.

As to the real-time UAV routing problem, UAV route chromosome representation should consider the visited and unvisited targets at the time of UAV route planning. Figure 1 is used to represent the feasible chromosome generation. At the time t_0, UAVs begin to take flights according to the pre-set cruise routes. At the time t_1, a new target 14 appears, and at this time, the route 0-8-7-11-0 is finished. Meanwhile, there are two unfinished UAV routes, i.e., 0-2-12-6-4-3-0 and 0-9-10-13-1-5-0. In these two routes, target 2, 12, 6 and 9 are visited, and others are not visited. Therefore, the visited target set is { 2,6,7,8,9,11,12} , and the unvisited target set is { 1,3,4,5,10,13,14} . In this situation, the UAV route chromosome is only decided by the unvisited target set. Then, the above-mentioned target arrangement method is used to generate feasible chromosomes from the random number sequence { 1,3,4,5,10,13,14} . For example, 13-4-5-1-10-14-3 and 10-5-4-13-3-1-14 can represent two feasible UAV route chromosomes respectively.

3.4 UAV Sub-route Division

Based on the aforementioned feasible UAV route chromosome, it's necessary to divide it into different sub-routes so as to meet the model constraints and calculate the values of objective functions. A target insertion method is proposed to conduct the UAV sub-route division. The steps of this method are described as follows.

Step 1 Delete the targets of the finished UAV routes and generate feasible UAV route chromosome.

Step 2 Find out the unfinished UAV routes, insert the chromosome genes into an unfinished UAV route one by one, and generate a UAV sub-route. The operation should meet the constraints of UAV maximum flight distance and target time window. If the constraints are not satisfied, insert the chromosome genes into another unfinished UAV route until all unfinished UAV routes are used.

Step 3 If there are some genes remaining in the chromosome, these genes are used to generate new UAV sub-routes. The new UAV sub-route generation should satisfy the constraints of UAV maximum flight distance and target time window.

To better illustrate the target insertion method, an example is given. As mentioned above, a feasible UAV route chromosome is 13-4-5-1-10-14-3, meanwhile, two unfinished UAV routes are 0-2-12-6-4-3-0 and 0-9-10-13-1-5-0, in which the visited targets are {2,12,6,9} . Then, target insertion method is conducted. First, target 13 is inserted into 0-2-12-6, a new route 0-2-12-6-13-0 is generated, if the new route satisfies the constraints of UAV maximum flight distance and target time window, then target 4 is inserted, a new route 0-2-12-6-13-4-0 is generated. If this route satisfies the two constraints, target 5 is inserted, a new route 0-2-12-6-13-4-5-0 is generated, and if this route doesn't satisfy the two constraints, the feasible route is 0-2-12-6-13-4-0. Second, it begins to insert the remaining targets {5,1,10,14,3} . Target 5 is inserted into 0-9, a new route 0-9-5-0 is generated, and the constraints of UAV maximum flight distance and target time window are analyzed. Doing this repeatedly, and suppose that the other feasible route is 0-9-5-1-10-0. Third, the remaining targets are {14,3} , a new UAV route 0-14-0 is generated, if the new route satisfies the two constraints, target 3 is inserted, and the route 0-14-3-0 is generated. If this route satisfies the two constraints, it is another feasible UAV route. In other words, as to the feasible UAV route chromosome, 13-4-5-1-10-14-3, three UAV sub-routes can be generated, i.e., 0-2-12-6-13-4-0, 0-9-5-1-10-0, and 0-14-3-0, which means that three UAVs are needed to conduct traffic monitoring, and the third UAV monitors the new target 14. Based on the divided sub-routes, both the number of UAVs used and the total cruise cost can be determined.

3.5 Crossover and Mutation

OX crossover and multiple-exchange mutation are used in this study, to keep the population diversity and improve the algorithm search ability. Examples are given to demonstrate the crossover and mutation operation. Suppose that there are

two chromosomes, 13-4-5-1-10-14-3 and 1-3-4-5-10-13-14. OX crossover designates two matched sections randomly, e.g., 5-1-10-14 and 4-5-10-13. Then, the former matched section is put in the front of the latter chromosome, i.e., 5-1-10-14 ‖ 1-3-4-5-10-13-14. Next, the repeated numbers of the rear part are deleted, and the new chromosome is represented as 5-1-10-14-3-4-13. Similarly, 4-5-10-13 and 13-4-5-1-10-14-3 are recombined as the new chromosome 4-5-10-13-1-14-3.

Suppose that there is a chromosome 13-4-5-1-10-14-3, multiple-exchange mutation designates two exchange positions randomly, e.g., the third and fourth positions. Then, 5 and 1 exchange their positions, and the new chromosome is 13-4-1-5-10-14-3. Then, the mutation operator continues to designate two exchange positions at random, e.g., the first and the seventh positions, then 13 and 3 exchange their positions. After the operation, the new chromosome is represented as 3-4-1-5-10-14-13.

4 Case Study

Suppose that there are twenty targets and a UAV base in a road network, and the locations and time windows of targets are the same with those in Ref. [34]. The monitoring service time of target 4, 6, 12, 19 and 20 is set at 0.15h, and the monitoring service time of other targets is set at 0.1 h. The UAV maximum flight distance is set at 50km, the UAV flight speed is set at 20 km/h, the distance conversion coefficients of monitoring service time and waiting time are set at 20 km/h and 5 km/h respectively. The static optimized routes are the same with those in Ref. [34], then the UAV arrival time of each target can be calculated, which are shown in Fig. 3. In Fig. 3, there are four UAV routes and each target has two numbers, i.e., target serial number and UAV arrival time. In addition, at the time t=0, UAVs begin to take off and monitor targets according to the planned cruise routes.

Suppose that at the time t=3.4h, the traffic control center knows that there is an accident occurring at target 21, its location and time window are (9.6 km, 12 km) and (3.5 h, 8.5h) respectively, and its monitoring service time is set at 0.1h. Therefore, the real-time UAV rerouting for monitoring new target should be considered. Based on the target time window and its UAV arrival time, we can know that, at the time t=3.4 h, the route 0-13-15-8-0 is finished, and the remaining three routes are unfinished. In the route 0-6-16-19-7-0, one UAV is waiting to monitor target 6; in the route 0-18-20-11-17-3-4-14-0, one UAV is waiting to monitor target 18; in the route 0-10-9-12-2-5-1-0, one UAV is flying from target 12 to target 2. At this time, the critical target set is {2, 6, 18} , and the unvisited target set is {1, 3, 4, 5, 7, 11, 14, 16, 17, 19, 20, 21}.

Then, optimization is conducted at the MATLAB R2014b platform based on the following basic parameters. 1) The UAV route chromosome population is set at 100; 2) the maximum iterations are set at 300; 3) the crossover and mutation rate are set at 0.8 and 0.1 respectively, and the number of multi-exchange is set at 3; and 4) the number of neighborhood weight vectors is set at 6.

In order to avoid algorithm randomness, the optimization is conducted for twenty times to obtain twenty optimized solutions, in which there are seven optimal solutions. The optimal total cruise cost is 223.25 km, and the optimal number of UAVs used is 3. In addition, an initial chromosome population is used to generate 100 initial solutions, and the best 20 solutions are selected to compare with the optimized solutions. The comparison of initial and optimized solutions is shown in Fig. 4.

The result comparison of initial and optimized solutions is listed in Table 1. It can be seen from Table 1 that compared with the initial solutions, the maximum, average,

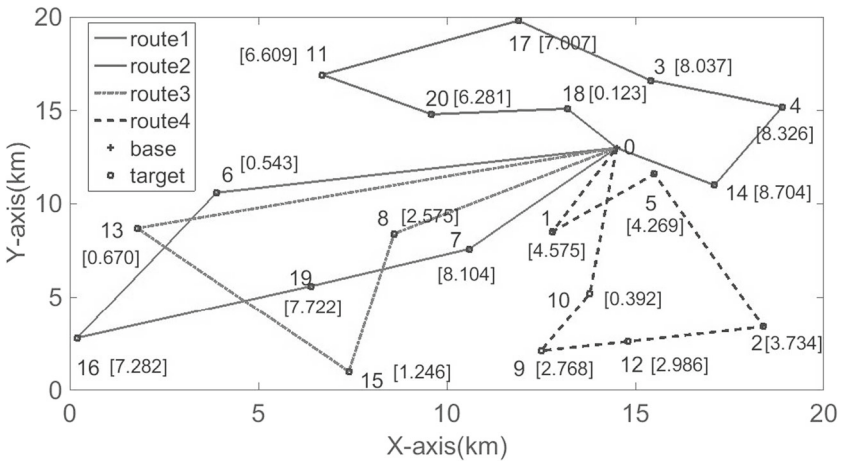

Fig. 3 Target distribution and its UAV arrival time

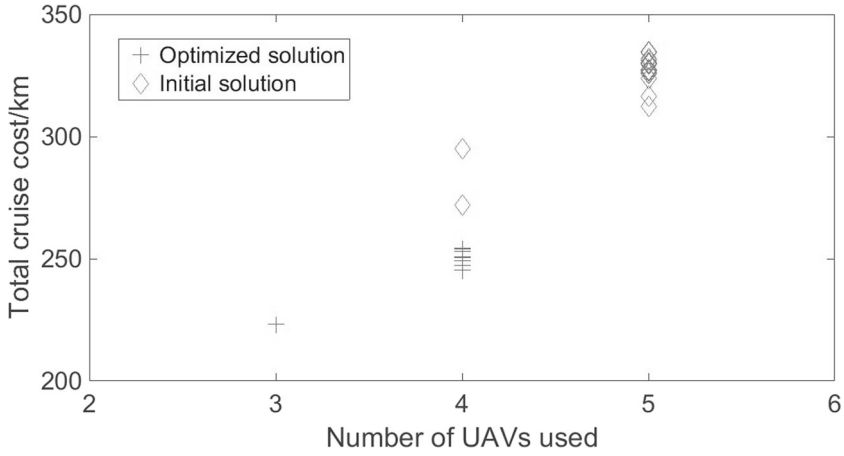

Fig. 4 Comparison of initial and optimized solutions

and optimal values of the optimized solutions decrease by 20.00%, 24.03%; 25.51%, 25.19%; and 25.00%, 17.87% respectively. This demonstrates that the optimization is feasible and effective.

Corresponding to the optimal solution, the optimal UAV routes are shown in Fig. 5. As Fig. 5 shows, there are three critical targets, i.e., target 2, 6 and 18, and three UAV routes, i.e., UAV 1 takes the monitoring task for target 21, and its cruise route is 0-18-21-20-11-17-3-4-0; the cruise route of UAV 2 is 0-6-16-19-7-0, which is the same with that of Fig. 3; and the cruise route of UAV 3 is 0-10-9-12-2-1-5-14-0. This means that no additional UAVs are needed to take traffic monitoring task.

5 Result and Discussion

In order to evaluate the algorithm performance, the proposed decomposition based multi-objective optimization algorithm is compared with the most frequently used NSGA-II, and the algorithm parameters are the same with the aforementioned basic ones. The computer configuration parameters are: windows 8 operation system, Intel Core i7-4870HQ CPU 2.50GHz, and RAM 16.0GB. Optimization is conducted at the MATLAB R2014b platform for twenty times. Twenty optimized solutions of NSGA-II and the proposed algorithm are shown in Fig. 6. In Fig. 6, there are three optimal solutions by using NSGA-II, and there are seven optimal solutions by using the proposed algorithm.

Moreover, the result comparison of NSGA-II and the proposed algorithm is shown in Table 2. It can be seen from Table 2 that compared with NSGA-II, the average computational time of the proposed algorithm is 25.90 seconds, which decreases by 48.65%; the percentage of finding optimal solution of the proposed algorithm is 35%, which increases by 133.33%. In addition, the mean number of UAVs used and the mean total cruise cost decrease by 5.19% and 2.53% respectively. This demonstrates that the proposed algorithm outperforms NSGA-II in terms of computational time, the percentage of finding optimal solution, and solution quality.

There are two reasons accounting for the algorithm improvement. The first is that the proposed algorithm has lower computational complexity, and the second is that the proposed algorithm uses neighborhood strategy to select elitist chromosome. The computational complexities of NSGA-II and the proposed algorithm are $O(mN^2)$ and $O(mNT)$ respectively, in which m is the number of optimization objective functions, N is the size of the population, and T is the number of neighborhood weight vectors for each weight vector. If the number of optimization objective functions and the population size are the same, the computational complexity ratio of the proposed algorithm and NSGA-II is $O(T)/O(N)$. Generally, T is much smaller

Table 1 Result comparison of initial and optimized solutions

Item	Maximum number of UAVs used	Maximum total cruise cost (km)	Average number of UAVs used	Average total cruise cost (km)	Optimal number of UAVs used	Optimal total cruise cost (km)
Initial solutions	5	334.80	4.90	323.07	4	271.81
Optimized solutions	4	254.36	3.65	241.69	3	223.25
Decrease degree (%)	−20.00	−24.03	−25.51	−25.19	−25.00	−17.87

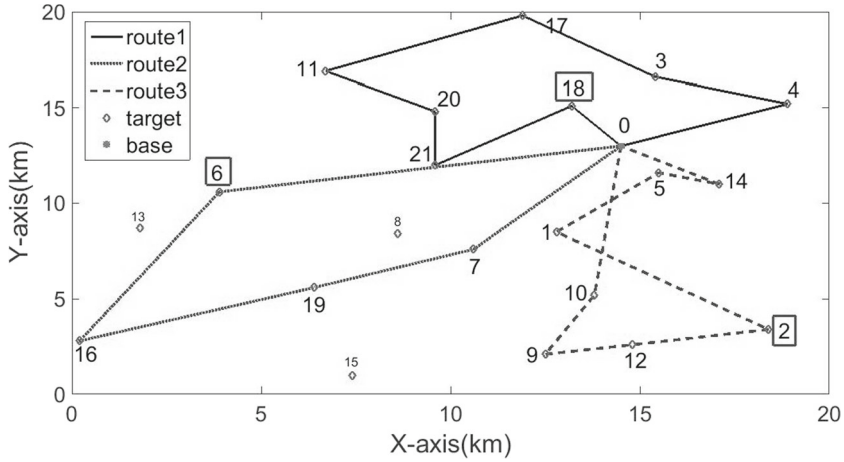

Fig. 5 Re-planned optimal UAV routes

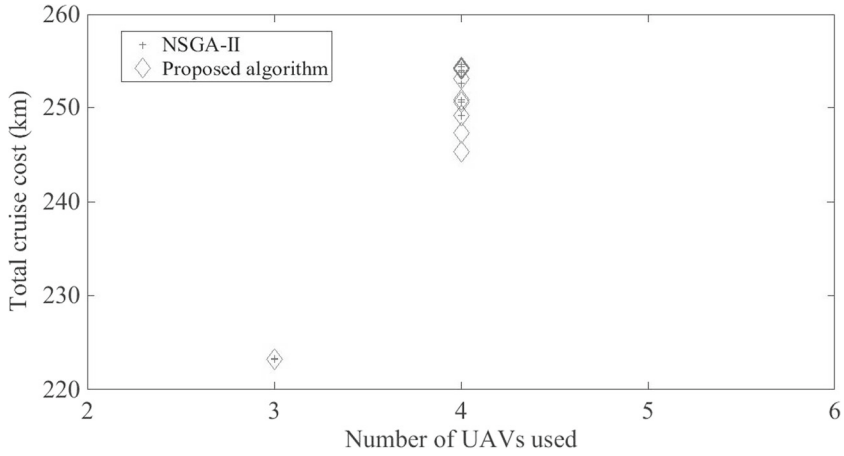

Fig. 6 Solutions of NSGA-II and the proposed algorithm

Table 2 Result comparison of NSGA-II and the proposed algorithm

Item	Average computation time (s)	Percentage of finding optimal solution (%)	Number of UAVs used			Total cruise cost (km)		
			Max	Min	Mean	Max	Min	Mean
NSGA-II	50.44	15	4	3	3.85	254.71	223.25	247.96
Proposed algorithm	25.90	35	4	3	3.65	254.25	223.25	241.69
Improvement (%)	−48.65	+133.33	0	0	−5.19	−0.18	0	−2.53

Table 3 Sensitivity analysis of the number of neighborhood weight vectors

Item	Average number of UAVs used	Average total cruise cost (km)	Average computation time (s)
T=3	3.85	250.87	24.76
T=6	3.65	241.91	25.90
T=9	3.85	247.25	25.25

Table 4 Sensitivity analysis of iteration times and population size

Scenario	Average computation time (s)	Percentage of finding optimal solution (%)	Number of UAVs used			Total cruise cost (km)		
			Max	Min	Mean	Max	Min	Mean
1	9.00	10.00%	4	3	3.85	269.95	223.25	250.43
2	17.27	10.00%	4	3	3.8	263.60	223.25	249.13
3	13.26	30.00%	4	3	3.7	263.29	223.25	243.93
4	25.90	35.00%	4	3	3.65	254.25	223.25	241.69

than N, thus $O(T)/O(N)$ is less than 1. Therefore, the proposed algorithm has faster computational speed. Additionally, NSGA-II adopts nondominated sorting and tournament selection to rank and select elitist chromosomes, and the proposed algorithm uses neighborhood strategy to update neighborhood chromosomes for each chromosome, thus any individual chromosome has more probabilities to be selected as the elitist one.

Furthermore, sensitivity analysis of the proposed algorithm parameters is conducted. The number of neighborhood weight vectors (T) is set at 3, 6, and 9 respectively, and other parameters are the same with the aforementioned basic ones. The optimization is conducted for twenty times, and the analysis result is shown in Table 3. It can be seen from Table 3 that the average number of UAVs used is 3.65, and the average total cruise cost is 241.91 km when T equals to 6, which are smaller than the objective values when T equals to 3 and 9. This means that the solution quality becomes better when T equals to 6. In addition, iteration times (G) are set at 200 and 300 respectively, the population size (N) is set at 50 and 100 respectively, and four scenarios are given as follows, i.e., scenario 1, G=200, N=50; scenario 2, G=200, N=100; scenario 3, G=300, N=50; scenario 4, G=300, N=100. Other parameters are the same with the aforementioned basic ones, and the optimization is also conducted for twenty times and the analysis result is shown in Table 4. Table 4 shows that the average computational time increases significantly with iteration times and population size increasing. Scenario 4 has the best solution quality, the mean number of UAVs used is 3.65, the mean total cruise cost is 241.69 km, and the percentage of finding optimal solution is 35%. However, the average computational time is the longest, i.e., 25.90 seconds, in other words, greater iteration times and population size are beneficial to improve the solution quality, but will increase computational time obviously.

6 Conclusions

This paper has proposed a real-time UAV rerouting model and a decomposition based multi-objective optimization algorithm. The model has considered the dynamic traffic monitoring demand, i.e., new monitoring target and time window, which is more suitable for real-life traffic monitoring. We have conducted a case study and compared the proposed algorithm with NSGA-II using the same chromosome generation, OX crossover, and multiple-exchange mutation operator. The case study results indicate that the proposed algorithm outperforms NSGA-II in terms of computational time, the percentage of finding optimal UAV routes, and solution quality. It implies that the proposed algorithm is promising to plan UAV cruise routes.

In addition, we have studied the impact of some algorithm parameters (i.e., the number of neighborhood weight vectors, iteration times, and population size) on the optimization performance. We find that the algorithm performance is influenced by the number of neighborhood weight vectors. Greater iteration times and population size are helpful to improve the solution quality, but increase computational time significantly.

In real-life traffic monitoring, the dynamism of UAV cruise route planning needs to consider the time-variant monitoring tasks, and the cooperation of multiple UAVs. In the future, we plan to study UAV cruise route planning in these uncertain conditions.

References

1. Federal Aviation Administration. Unmanned aircraft systems [EB/OL]. https://www.faa.gov/uas/. Accessed: Dec 22 2016 (2016)
2. Civil Aviation Administration of China. Air traffic control regulation of civil unmanned aerial vehicle systems [EB/OL]. http://www.caac.gov.cn/XXGK/XXGK/GFXWJ/201610/t20161010840016.html. Accessed: Dec 22 2016 (2016)
3. Xu, Y., Yu, G., Wu, X., Wang, Y., Ma, Y.: An enhanced viola-jones vehicle detection method from unmanned aerial vehicles imagery. IEEE Trans. Intell. Trans. Syst. **18**(7), 1–12 (2017)
4. Zhang, L., Peng, Z., Sun, D.J., Liu, X.: A Uav-Based Automatic Traffic Incident Detection System for Low Volume Roads. In: Transportation research board of the national academies, pp. 542–558. National Research Council, Washington (2013)
5. Dobson, R., Colling, T., Brooks, C., Roussi, C., Watkins, M., Dean, D.: Collecting decision support system data through remote

sensing of unpaved roads. Transp Res Rec.: J. Transp. Res. Board (2433), 108–115 (2014)
6. Zink, J., Lovelace, B.: Unmanned aerial vehicle bridge inspection demonstration project. Minnesota Department of Transportation, Minnesota (2015)
7. Hutchison, M.G.: A method for estimating range requirements of tactical reconnaissance UAVs. In: Proceedings of AIAA'S 1St technical conference and workshop on unmanned aerospace vehicles, pp. 1–12. AIAA, Virginia (2002)
8. Yan, Q., Peng, Z., Chang, Y.: Unmanned aerial vehicle cruise route optimization model for sparse road network. In: Transportation research board of the national academies, pp. 432–445. National Research Council, Washington (2011)
9. Pitre, R.R., Delbalzo, R.: Uav route planning for joint search and track missions–an information-value approach. IEEE Trans. Aerosp. Electron. Syst. **48**(3), 2551–2565 (2012)
10. Peng, Z., Wu, J., Chen, J.: Three-dimensional multi-constraint route planning of unmanned aerial vehicle low-altitude penetration based on coevolutionary multi-agent genetic algorithm. J. Cent. South Univ. Technol. **18**(5), 1502–1508 (2011)
11. Liu, X., Gao, L., Guan, Z., Song, Y.: A multi-objective optimization model for planning unmanned aerial vehicle cruise route. Intern. J. Adv. Robot. Syst. **13**, 1–8 (2016)
12. Chen, Y., Luo, G., Mei, Y., Yu, J., Su, X.: UAV Path planning using artificial potential field method updated by optimal control theory. Intern. J. Syst. Sci. **47**(6), 1407–1420 (2016)
13. Ragi, S., Chong, E.K.P.: UAV Path planning in a dynamic environment via partially observable Markov decision process. IEEE Trans. Aerosp. Electron. Syst. **49**(4), 2397–2412 (2013)
14. Pehlivanoglu, Y.V.: A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV. Aerospace Sci. Technol. **16**(1), 47–55 (2012)
15. Chen, H., Hsueh, C., Chang, M.: The real-time time-dependent vehicle routing problem. Transport. Res. Part E **42**, 383–408 (2006)
16. Srinivas, N., Deb, K.: Multiobjective optimization using nondominated sorting in genetic algorithms. Evol. Comput. **2**(3), 221–248 (1994)
17. Zitzler, E., Thiele, L.: Multi-objective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Trans. Evol. Comput. **4**(3), 257–271 (1999)
18. Knowles, J.D., Corne, D.W.: The Pareto archived evolution strategy: a new baseline algorithm for pareto multiobjective optimization. In: Proceedings of the 1999 congress on evolutionary computation, pp. 98–105. IEEE, Washington (1999)
19. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
20. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. Evol. Comput. **11**(6), 712–731 (2007)
21. Zhang, Q., Liu, W., Tsang, E., Virginas, B.: Expensive multiobjective optimization by MOEA/d with Gaussian process model. IEEE Trans. Evol. Comput. **14**(3), 456–474 (2010)
22. Tan, Y., Jiao, Y., Li, H., Wang, X.: MOEA/D plus uniform design: a new version of MOEA/d for optimization problems with many objectives. Comput. Oper. Res. **40**(6), 1648–1660 (2013)
23. Ma, X., Liu, F., Qing, Y., Gong, M., Yin, M., Li, L., Jiao, L., Wu, J.: MOEA/D with opposition-based learning for multiobjective optimization problem. Neurocomputing **146**(146), 48–64 (2014)
24. Wang, Z., Zhang, Q., Zhou, A., Gong, M., Jiao, L.: Adaptive replacement strategies for MOEA/d. IEEE Trans. Cybern. **46**(2), 474–486 (2016)
25. Nebro, A.J., Durillo, J.J., Luna, F., Dorronsoro, B., Alba, E.: MOCEll: a cellular genetic algorithm for multiobjective optimization. Intern. J. Intell. Syst. **24**(7), 726–746 (2009)
26. Chang, P.C., Chen, S.H.: The development of a sub-population genetic algorithm II (SPGA II) for multi-objective combinatorial problems. Appl. Soft Comput. **9**(1), 173–181 (2009)
27. Bader, J., Zitzler, E.: Hype: an algorithm for fast hypervolume-based many-objective optimization. Evol. Comput. **19**(1), 45–76 (2011)
28. While, L., Bradstreet, L., Barone, L.: A fast way of calculating exact hypervolumes. IEEE Trans. Evol. Comput. **16**(1), 86–95 (2012)
29. Tiwari, S., Fadel, G., Deb, K.: AMGA2: Improving the performance of the archive-based micro-genetic algorithm for multi-objective optimization. Eng. Optim. **43**(4), 377–401 (2011)
30. Cagnina, L.C., Esquivel, S.C., Coello, C.A.C.: Solving constrained optimization problems with a hybrid particle swarm optimization algorithm. Eng. Optim. **43**(8), 843–866 (2011)
31. Ke, L., Zhang, Q., Battiti, R.: MOEA/D-ACO: a multiobjective evolutionary algorithm using decomposition and ant colony. IEEE Trans. Cybern. **43**(6), 1845–1859 (2013)
32. Yang, X., Karamanoglu, M., He, X.: Flower pollination algorithm: a novel approach for multiobjective optimization. Eng. Optim. **46**(9), 1222–1237 (2014)
33. Li, H., Zhang, Q.: Multiobjective optimization problems with complicated pareto sets, MOEA/d and NSGA-II. IEEE Trans. Evol. Comput. **13**(2), 284–302 (2009)
34. Liu, X., Peng, Z., Chang, Y., Zhang, L.: Multi-objective evolutionary approach for UAV cruise route planning to collect traffic information. J. Cent. South Univ. **19**(12), 3614–3621 (2012)

Xiaofeng Liu received the B.S. and M.S. degrees from the Wuhan University of Technology, Wuhan, China, in 2004 and 2007, respectively, and received his Ph.D. degree in traffic information engineering and control from the Tongji University, Shanghai, China in 2013. He currently works in the School of Automotive and Transportation, Tianjin University of Technology and Education, China. His current research interest lies in the area of transportation applications using unmanned aerial vehicle.

Zhong-Ren Peng received the B.S. degree from the Central China Normal University, Wuhan, China in 1983, the M.S. degree from the Chinese Academy of Sciences, Beijing, China in 1986, and received his Ph.D. degree from the Portland State University, Oregon, USA in 1994. He currently works in the Department of Urban and Regional Planning, University of Florida, USA. His current research interest lies in the area of air quality analysis using unmanned aerial vehicle and urban transportation planning.

Li-Ye Zhang received the B.S. and M.S. degrees from the Changsha University of Science and Technology, Changsha, China, in 2002 and 2005, respectively, and received his Ph.D. degree in traffic information engineering and control from the Tongji University, Shanghai, China in 2014. He occupied the traffic safety post-doctoral research work from 2014 to 2016 in the National University of Singapore. He currently works in the Institute of High Performance Computing, Singapore. His current research interest lies in the area of transportation optimization.