International Journal of Cooperative Information Systems
Vol. 29, Nos. 1 & 2 (2020) 2040002 (22 pages)

DOI: 10.1142/S021884302040002X

Distributing UAVs as Wireless Repeaters in Disaster Relief via Group Role Assignment

Dongning Liu[*,‡], Qian Jiang[*,§], Haibin Zhu[†,¶] and Baoying Huang[*,‖]

[*]*School of Computer Science and Technology*
Guangdong University of Technology, Wai Huan Xi Road No. 100
Guangzhou 510006, P. R. China
[†]*Department of Computer Science and Mathematics*
Nipissing University, 100 College Drive
North Bay, P1B 8L7, Canada
[‡]*liudn@gdut.edu.cn*
[§]*jiangqian@mail2.gdut.edu.cn*
[¶]*haibinz@nipissingu.ca*
[‖]*baoyingh@qq.com*

Received 28 July 2019
Accepted 6 December 2019
Published 25 February 2020

When an earthquake occurs, disaster relief is an urgent, complex and critical mission. High on the list is communication network recovery within the disaster area. Unmanned aerial vehicles (UAVs) are often used in this regard. Some of them are used as collective repeaters to provide the required network coverage. Their timely, efficient, and collaborative deployment to specific locations is a big challenge. To meet this challenge, this paper formalizes and solves the problem of UAV deployment for signal relays via group role assignment (GRA). The minimum spanning tree algorithm is used to model a rapidly deployed optimal relay network. It can help establish the minimum number of relay points necessary to ensure communication stability. In this scenario, UAVs (agents) adopt roles as communication relays. The task of distributing UAVs to relay points can be solved quickly via the assignment process of GRA, which can solve the x-ILP problem with the help of the PuLP package of Python. Results from thousands of experimental simulations indicate that our solutions are effective, robust and practical. The process can be used to establish an optimal, efficient, and collaborative relay network using UAVs. Their rapid deployment can be a significant contribution to earthquake disaster relief.

Keywords: Disaster relief in earthquake; signal relay; collaborative deployment; unmanned aerial vehicle (UAV); role-based collaboration; group role assignment (GRA).

1. Introduction

China is a country that experiences many earthquake disasters. When such an event occurs, communication recovery within the disaster area plays a vital role in disaster

[¶]Corresponding author.

relief. Unmanned aerial vehicles (UAVs) can be rapidly deployed to re-establish communications when physical infrastructure has been damaged or destroyed.

UAVs can perform various roles in disaster relief such as patrolling, detecting, and relaying. Decision makers face the challenge of quickly determining the numbers of and roles adopted by such equipment. In communication recovery, it is necessary to have efficient tools to send the minimum number of UAVs to locations that provide stable coverage over the disaster area.

The key of deploying UAVs as communication relays is an awareness of the number and locations of the collection of relay points. Next comes the task of role and location assignment of drones from different base stations into a successfully coordinated network. There are many combination methods which can do this job. However, the specific scenario and scale of an earthquake often change. A comprehensive model is needed to deal with different but equally urgent incidents. It is critical for us to distribute a minimum number of mobile communication devices into a robust, economical, and collaborative network in the shortest time possible. Consideration must also be given to other important tasks that a UAV fleet must perform disaster relief and reconstruction.

This paper formalizes and solves the problem of establishing UAV signal relay networks via group role assignment (GRA). GRA[1,2] is the kernel of role-based collaboration (RBC). It has been revealed as a complex process throughout the life cycle of RBC, i.e. agent evaluation, role assignment, role-playing, and role transfer.[3–5] RBC has been developed into a practical methodology for decision makers relative to task distribution. GRA seeks an optimal role-to-agent assignment based on the results of agent evaluations and greatly affects collaboration efficiency and the degree of qualification of members involved in RBC.

Signal relay networks over a disaster area can be established by considering the collection of relay points as roles and UAVs as agents. The primary work of this paper is to provide a practical, effective and comprehensive solution to the problem via GRA. In modeling the rapid reestablishment of communications over a disaster area, we initially use the minimum spanning tree algorithm to determine an optimal collaborative relay network. This network offers stable, full coverage of the affected area with the minimum number of UAVs at the designated locations. Using the assignment process of GRA, relay locations are considered roles and UAVs are agents.

The contributions of this paper include the following:

- This paper concisely formulizes the problem of establishing signal relay networks using UAVs over a disaster area via GRA, which can provide a uniform method.
- An efficient, comprehensive method is proposed for deploying UAVs as collective repeaters to the right places within an acceptable time.
- This paper creates timely a collaborative yet stable communication network that uses the least number of UAVs as a significant contribution to earthquake disaster relief.

This paper is organized as follows: It discusses related works in Sec. 2. Section 3 describes a real-world scenario to illustrate the problem. Then, it formally defines the problem via RBC and its Environments — Classes, Agents, Roles, Groups, and Objects (E-CARGO) model in Sec. 4. Section 5 proposes a solution to the problem via GRA, with simulation experiments. This paper concludes by pointing out possible future work in Sec. 6.

2. Related Work

2.1. *Unmanned aerial vehicle and rescue*

The UAV has become a popular tool for search and rescue because it is fast and not affected by obstacles like heavy traffic congestion or road damage. There are many types of research involving the use of drones as a rescue tool.[6–9] Such uses are communication recovery, emergency response, search, and rescue, i.e. Refs. 6–9.

For example, Liu *et al.* propose to adopt the Unmanned aerial vehicle mounted Base Stations (UBS) to assist Machine to Machine (M2M) communications for disaster rescue.[6] In order to enable the maximum number of Human portable/wearable Machine-Type-Devices (HMTDS) to transmit data, they also propose the UBS Network Access and Resource Allocation (UNARA) scheme to wisely select the HMTDs and allocate them with network resources accordingly. Their research focuses on maximizing the communication numbers of the portable mobiles within the coverage of a drone based on their communication knowledge.

Wankmüller *et al.* analyze a scenario that a drone equipped with portable Automated External Defibrillators (AED) can fly from a base station to the patients' site where a bystander receives it and starts treatment.[7] They propose an integer linear program to determine the optimal allocation of drone base stations in a given geographical region. In detail, they develop an optimization model that follows the objectives to minimize the number of used drones and to minimize the average travel times of defibrillator drones to sudden cardiac arrest (SCA) events.

As for our research, we focus on using drones to recover global communication while a disaster has had happened. We propose the use of properly equipped UAVs to construct a practically optimal and collaborative relay network with the least communication cost regardless of the scale or the distribution of the tasks.

2.2. *Data-based rescue and agent-based rescue*

As a matter of fact, the above-mentioned restoration of local network communication in a disaster area, emergency rescue, and our proposed global communication recovery in disaster relief are essentially collaborative assignment problems.

In the research of assignment with constraints that were widely used in various domains, many investigations focus on assignment problems in different applications.[10–12] Simply, assignment methods are used by many researchers to achieve task distribution.[13–15]

Many researchers allocate their tasks based on data-related or agent-related collaborative methods, many of which first analyze the properties and distribution of the data,[8,16–18] the relationship between or among tasks,[19–22] or the specific platform and background,[18,23–25] and then select a specific algorithm[16,17,26] which is suitable for the tasks.

For instance, Mouradian *et al.* propose a coalition formation algorithm for multi-robot systems.[8] It is a novel heuristic system since the population of robots in large-scale disaster settings is very large. They rely on Quantum Multi-Objective Particle Swarm Optimization (QMOPSO). They conduct extensive simulation experiments and compared their algorithm with other existing algorithms.

Geng *et al.* also use a modified centralized algorithm based on particle swarm optimization (MCPSO) to solve the task allocation problem in the search and rescue domain.[16] They provide a benchmark against distributed algorithms in search and rescue application area. They suppose that a centralized algorithm should perform better than distributed algorithms because it has all the available information at hand to solve the problem.

Besides, Okan *et al.* create an approximate Multi-agent Markov Decision Process (MMDP) model with single-agent, static task, shortest path pruning, task clustering, and online planning horizon approximations in our online planning framework.[17] This model is also an agent-based algorithm. Although the performances of their models are better than their benchmark algorithms, some questions remain. First, their experimental results show that the proposed solutions are time-consuming. Rescue-related problems are time-critical and must be solved quickly to obtain optimal solutions. Second, some of these solutions like the MDP model may not have optimal results.[27]

It is difficult to compare our solution with these algorithms due to differences in experimental scenarios and objectives. Our approach can seek a practically global optimal solution by defining a formalization model with an assignment algorithm to deal with the x-ILP problem in our scenario. Thousands of simulation experiments demonstrate that our solution is scale-independent; it can swiftly determine the practically optimal result even though the task's scale becomes large.

2.3. *Role-based rescue*

Allocating tasks based on the role instead of the agent may help build a model regardless of the distribution of the tasks. It helps formalize the problem and build a robust model when solving collaborative issues. There are some role-based assignment researches,[28,29] i.e. RBC[2,3] and RBAC.[30,31]

For example, Dastani *et al.* investigate the determination of conditions under which an agent can adopt a role and the meanings for the agent to play a role.[29] They define relationships between roles and agents and then discuss architectural and functional changes that an agent must deal with when the agent enters an open system. Their work proposes a framework to support role negotiation and agent evaluation.

On the other hand, GRA,[4,28] a complex task in the life cycle of RBC,[1,4,5] has arisen as an important methodology. It significantly improves collaboration efficiency by seeking an optimal team execution based on agent evaluations.

With RBC, E-CARGO and GRA, Zhu *et al.* contributed an efficient way to solve the collaborative assignment problem.[4] They clarify the GRA problem (GRAP), describe a general assignment problem (GAP), convert a GRAP to a GAP, propose RBC and its E-CARGO model based on the Kuhn–Munkres (K–M) algorithm, conduct numerical experiments, and analyze the solutions' performance. Their solution is highly practical for the role assignment of large groups. Because this solution is established on the evaluation of agents, it reveals the importance of agent evaluation. Possibly, with this proposed solution, the complexity (being NP-hard) of constrained assignment problems can be reduced by pertinent domain-oriented agent evaluations.

Liu *et al.* investigate the role assignment that is an important part of the RBC process.[3] Considering that decision maker's preferences would affect the team's performance in many cases, they propose a series of methods to solve the GRA with Balance (GRAB) problem that considers both leader's preferences and team's performance by using a logic-based association rule mining method called the One Clause at a Time Approach (OCAT). Their method helps decision makers easily mine fine-grain preferences from coarse-grain preferences.

Zhu *et al.* solve the M–M assignment problem, an important open problem related to the well-known K–M algorithm by proposing the KM_B algorithm based on the RBC and its E-CARGO model.[32] They first formalize the problem in a second-order bipartite graph. Next, they solve it by improving the K–M algorithm with backtracking, i.e. KM_B, which is verified to be practical through simulation experiments. They eventually prove that the proposed algorithm is valid and the worst time complexity of the KM_B algorithm is $O((\sum L^a[i])^3)$, where L^a denotes the maximum number of tasks that can be assigned to agent i.

The primary steps to use the GRA and its E-CARGO model include defining the roles, the agents, and the qualification matrix for the agents to be assigned with a specific role. In our scenario, the role is not clear. Consequently, we improve the RBC by appending the role awareness procedure so as to transform our problem into an x-ILP problem and use GRA to get the practically optimal solution for our research.

In summary, RBC is the fundamental theory and methodology of this paper. Our research describes the relay points problem using its E-CARGO model. The kernel of RBC is GRA. It focuses on constructing a practically optimal and collaborative relay network while minimizing communication cost using a minimum number of UAVs.

3. A Real-World Scenario

The County of Jiuzhaigou is situated in the depths of the mountains in the border area of three counties of Napping, Songpan, and Pingwu in the Aba Tibetan and Qiang Autonomous Prefecture, Northwestern Sichuan, China. This scenic location is also known as Nine-Village Gully. Until recent years, it had remained undisturbed in these mountains due to its inaccessibility. Travel services are provided via horse trails or mountain paths. Scattered along the gully, among alpine lakes, are nine Tibetan settlements, hence the name Jiuzhaigou.

Sichuan is a quake-prone region. In May 2008, an 8.0-magnitude earthquake struck Wenchuan and killed more than 80,000 people. In 2013, a 7.0-magnitude quake hit Lushan, killing 196 people. Unfortunately, another 7.0-magnitude earthquake jolted Jiuzhaigou County on August 8th, 2017. According to the China Earthquake Networks Center (CENC), the quake struck at a depth of 20 km. More than 90 emergency vehicles and 1,200 personnel were dispatched to participate in the rescue work.

The communication system was affected by the quake. This essential component of search and rescue required the deployment of several mobile communication vehicles to resume communication. Because the earthquake destroyed most of the roads in Jiuzhaigou, the vehicles were only capable of communication within a 2 km radius (see Fig. 1(a)).

To restore communication over the disaster area, the communication department decided to deploy UAVs carrying communication equipment. These UAVs departed from various bases in Jiuzhaigou simultaneously. We assume that each UAV hover stably in the air with a small radius so that we treat it as a stationary state. Due to the load limitations of the drones, the on-board communication equipment had to be within 3 km of the mobile communication vehicle (see Fig. 1(b)). The communication distance between drones was limited to 6 km.

To ensure the communication quality, the altitude of each drone should be much smaller than its coverage radius; hence the administrators can consider that a drone's coverage shape is also a circle (see Fig. 1(c)). In general, for robust

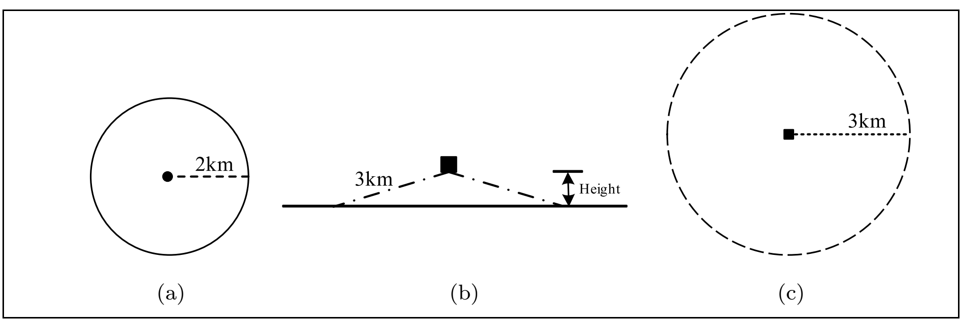

Fig. 1. The covered shape of the communication vehicle and the drone.

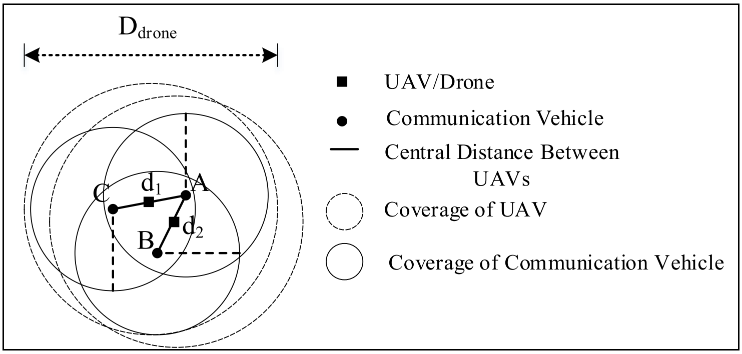

Fig. 2. An example of stable communication.
Notes: A square represents a UAV, a circle represents a communication vehicle, a solid line represents the central distance between different UAVS, a slash circle represents the coverage of UAV, and a solid circle represents the coverage of communication vehicle.

performance, the rescue center assumed that one UAV could only guarantee stable communication of one link (like AB or AC in Fig. 2) between two different communication vehicles. However, it can be used as a back-up communication drone for other links if some of the intersected UAVs fail to work. For example, when the UAV over the link AB breaks down by accident, the UAV in AC can be a temporary communication medium for AB.

In order to guarantee the effectiveness of the proposed solution, partial real-world elevation data of Jiuzhaigou was used to carry out all of the simulation experiments. Figure 3 is the contour map drawn from the partial elevation data of Jiuzhaigou. The square shape in Fig. 3 represents the base station, and the circle shape is the communication vehicle. To keep our visualization distribution result clearly displayed, we set the covered radius of a UAV as 3 km, and the number of bases is 8. The number of communication vehicles is set as 15.

As each drone has a maximum flight time, it needs to be replaced by other drones from the same base station when it is approaching its maximum flight time. Also, each base station needs to dedicate some drones to carry out patrolling and detecting tasks in the disaster area. Therefore, each station has a limited number of assigned drones for communication (see Table 1).

This signal relay task is a great challenge in the deployment of UAVs in Jiuzhaigou. It is important to quickly determine the number of signal relaying UAVs and send them to the right locations within an acceptable time, while having other UAVs available to do other relief jobs.

From the above scenario, the problem can be dealt with by following the initial steps of RBC and a related GRA problem. The first part is role negotiation, which needs to be aware of and propose the fewest relay points and related roles. After that, the primary objective is to minimize the total cost of the assigned UAVs from the different base stations to the relay points while ensuring global communication in Jiuzhaigou.

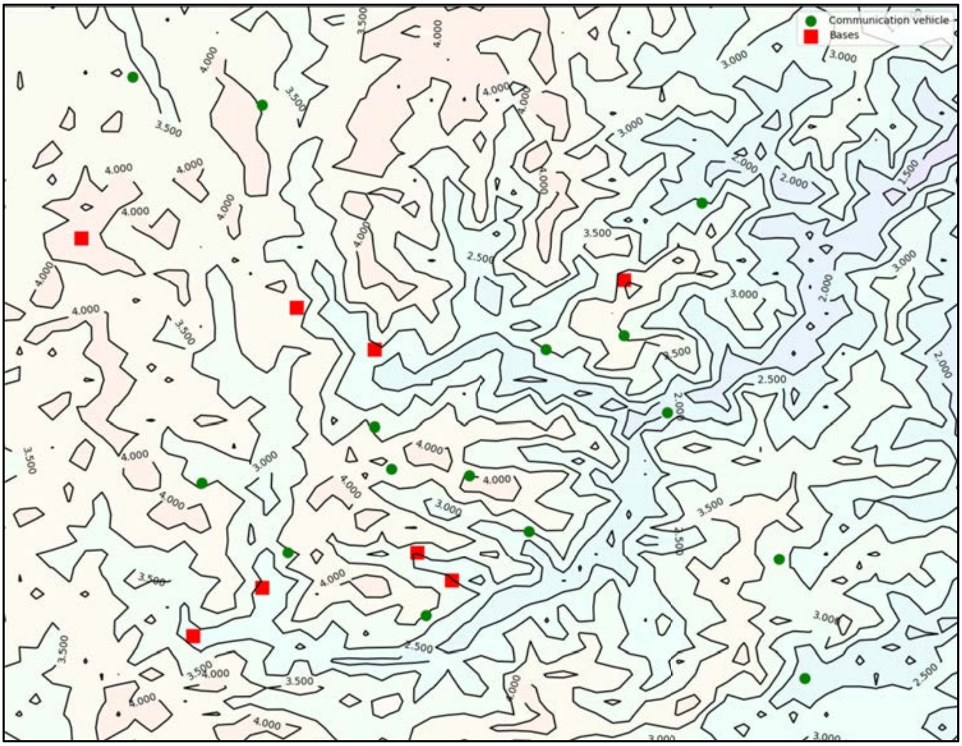

Fig. 3. (Color online) The contour map of the partial real-world elevation data of Jiuzhaigou. *Notes*: A red square represents a base station, and a green circle represents a communication vehicle.

4. Problem Formulization with GRA

To solve the problem, we initially describe it using the GRA method, which is the kernel of Role-Based Collaboration (RBC) and its formalization model E-CARGO.

4.1. *The brief of E-CARGO model*

In order to well define the problem, to be self-contained, we first concisely describe the required concepts and definitions of the E-CARGO model. With the E-CARGO model, a system $\sum$ can be described as a 9-tuple $\sum ::= \langle C, O, A, M, R, E, G, s_0, H\rangle$, where C is a set of classes, O is a set of objects, A is a set of agents, M is a set of messages, R is a set of roles, E is a set of environments, G is a set of groups, s_0 is the initial state of the system, and H is a set of users. In such a system, A and H, E and G are tightly coupled sets. A human user and his/her agent perform a role

Table 1. The limited numbers of UAV.

Base Station	1	2	3	4	5	6	7	8
The limited numbers of UAV	10	13	16	9	11	14	12	15

together. Every group should work in an environment. An environment regulates a group.

In discussing role assignment problems, environments (e) and groups (g) are simplified into vectors and matrices, respectively. Furthermore, we use nonnegative integers $m(= |A|$, where $|A|$ is the cardinality of set A) to express the size of the agent set A, $n(= |R|)$ the size of the role set R, $i, i_1, i_2, \ldots$ the indices of agents, and $j, j_1, j_2, \ldots$ the indices of roles.

Before formalizing our problem by RBC and E-CARGO it is important to classify the role, the agent and the qualification matrix that evaluates the agent when assigning it to a specific role. In our scenario, considering UAVs as agents is straightforward, but considering the relay points as roles poses quite challenging. For this reason, role negotiation was dealt with first, because it was needed prior to using the GRA method.

4.2. *Role negotiation*

The role negotiation process in this paper, called role awareness, aims to set up a relay network by specifying the specific properties of relay points, i.e. roles.

Our target is to achieve global communication in the disaster area through the establishment of a practical, optimal relay network formed by UAVs from different base stations. It is clear that drones (UAVs) from different bases can be regarded as agents. As for the relay network, we need to decompose it into the relay points so as to assign the UAVs specifically and evaluate the qualification of the drones from different base stations. There exist a mutually exclusive relationship and a collaborative relationship among these relay points, which are suited to become roles. Furthermore, the distance between the base stations and the relay points can be used to evaluate the performance of an agent relative to a specific role.

Although the locations of base stations are given and the distance between one base and one relay point can be calculated, the number and locations of the relay points are uncertain. Consequently, the primary problem here is how to become aware of the roles, i.e. the replay points. That means to find a relay network which consists of a series of proper relay points with the least communication cost and the fewest UAVs.

Based on the relay network's property, we decide to use Prim's algorithm to set up the relay network. It is a kind of minimum spanning tree algorithm which is widely used in computer network.[33] We use the minimum spanning tree algorithm here because it has the property of the least global communication cost if the positions of communication vehicles are determined before the relay points are selected and the number of UAVs is sufficient for fully covering all of the communication vehicles. As shown in Fig. 4(a), if two or more communication vehicles share the same UAV for fully covered communication, their coverage must intersect in our scenario. However, they can communicate directly, and in a real-world disaster, most of the UAVs are usually distributed for those communication vehicles whose

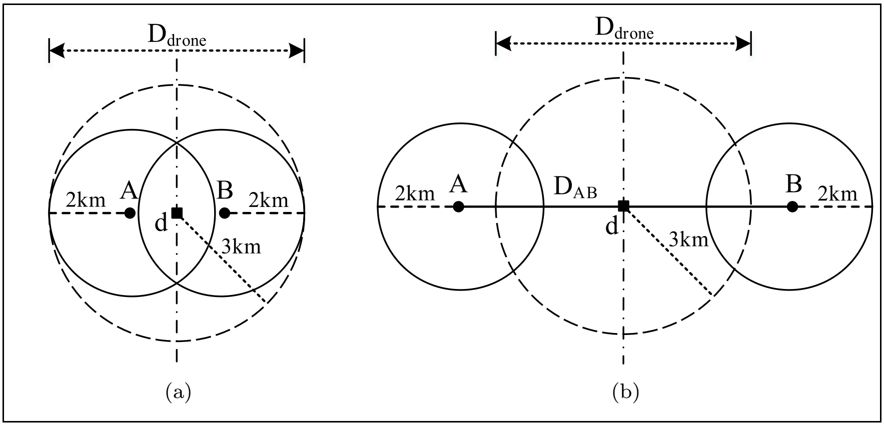

Fig. 4. Analyzing the reasons for using the minimum spanning tree.
Notes: Two solid circles represent the coverage areas of two communication vehicles, named A and B. A bigger slash circle is the coverage area of a UAV, named d. D_{AB} represents the central distance between vehicle A and vehicle B. D_{drone} represents the coverage diameter of UAV d.

coverage does not overlap. For this reason, using the minimum spanning tree is practical and it can achieve global communication with the least communication cost. Besides, using the minimum spanning tree also makes the relay network easier to expand.

After getting the practically optimal relay network, the next task is to determine the location of relay points. Considering that all of the communication vehicles are equally important when a UAV is placed between communication vehicles or extends the line, the coverage of the UAV is maximized, see Fig. 4(b). Consequently, we propose to distribute the UAVs to the line or the extended line between the communication vehicles. Here, we propose a method as follows to find the proper relay points from the minimum spanning tree.

First, according to the minimum spanning tree in Fig. 5, we can consider that the minimum number of relay points T_d that fully cover two local communication areas can be formalized by

$$T_d = \sum_{A,B \in V_c} \left\lceil \frac{D_{AB} + 2 \times r_t}{D_{\text{drone}}} \right\rceil, \tag{1}$$

where

- V_c is the set of all the communication vehicles.
- D_{AB} means the central distance between two different communication vehicles A and B belonging to the V_c set (see Fig. 6(a)).
- r_t represents the covered radius of a communication vehicle.
- D_{drone} is the covered diameter of the drone.

Here we use a simple example to explain Eq. (1). We assume that (1) there are two communication vehicles (A and B) in a disaster area; (2) the sum of their

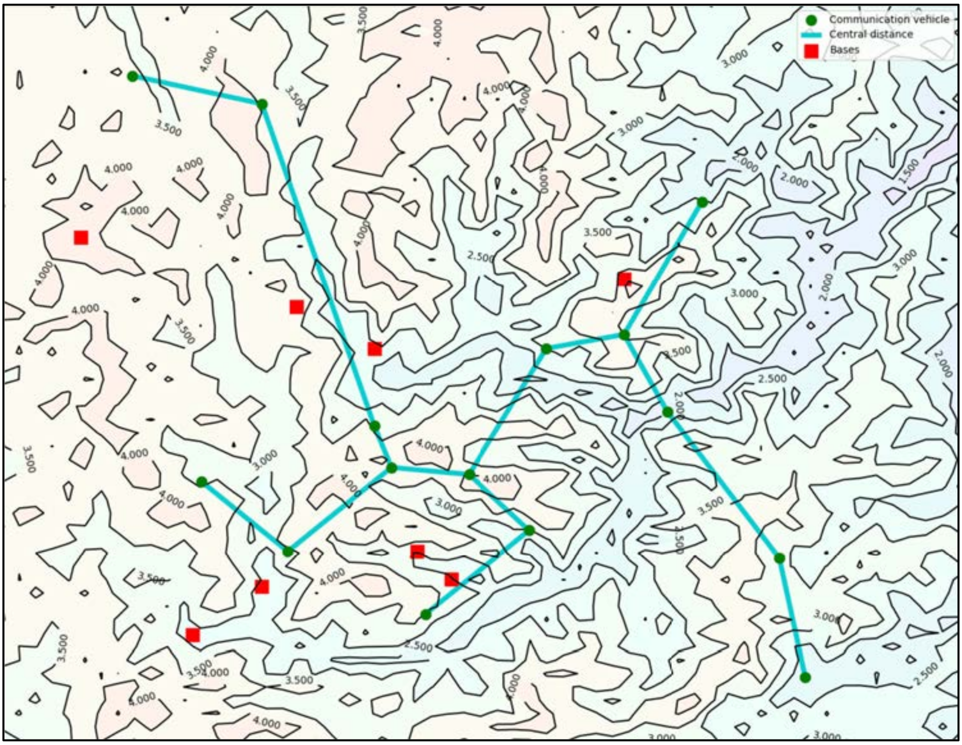

Fig. 5. (Color online) The minimum spanning tree of relay points.
Notes: A red square represents a base station, and a green circle represents a communication vehicle, and a cyan line represents the central distance between two communication vehicles.

central distance D_{AB} and their coverage radius r_t is longer than a single drone's coverage diameter D_{drone} but shorter than twice the length of D_{drone}. Hence, we propose using two drones (d_1 and d_2) to achieve full coverage of the area being served by *vehicle*s A *and* B (see Fig. 6(b)). Then, if there exists one of the items in Eq. (1) that is not divisible before being rounded, there is some coverage overlap between drones. Under this condition, to maximize the coverage of drones, we move the relay points outward along the extension of the center distance line until the maximum communication distance between drones is reached (see Fig. 6(c)).

4.3. *Problem formulizations*

According to the analysis above, we define the relay points obtained by role awareness as roles and base stations as agents. To better describe RBC and its E-CARGO model, we use one small-scale experiment as an example here. The role number we found by using role awareness based on our proposed method is 58, and the value for the base station is 8; The covered radius of the drone is 3 km, and the max communication distance between the UAVs is 6 km; we also set the location of the

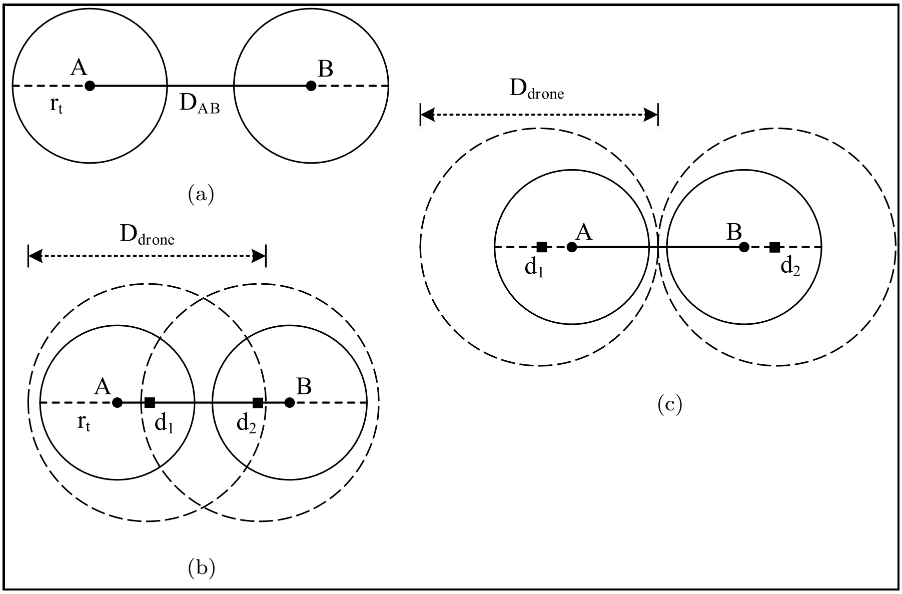

Fig. 6. Possible positions of the UAVs and the communication vehicles.
Notes: A and B are communication vehicles with the solid circle as the coverage areas while two bigger slash circles represent the coverage areas of UAV d_1 and the UAV d_2.

communication vehicles in a range from 3 km to 98 km. It is a suitable range for the partial real-world elevation data of Jiuzhaigou we selected.

Definition 1. A *role range vector* L is a vector of the lower bound of the ranges of roles in environment e of group g.

Note: L is a valuable component in the *E-CARGO* model. It provides an indication of the minimum number of agents required for each role within a properly functioning group in GRA. In our scenario, each role can only be assigned to one agent, so the L vector here can be formalized as: $L =$ [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1].

Definition 2. An *ability limit vector* L^a is an m-vector, where $L^a[i] (0 \le i < m)$ indicates how many roles can be assigned to agent i at most.

From Definition 2, we actually introduce a new constraint, i.e. $|a_i.R_p| + 1 \le L^a[i] (0 \le i < m)$, where $a_i.R_p$ means the set of potential roles. The number of roles assigned to agent i should be less than the number of roles allowed for agent i. For example, for the scenario here, $L^a = [10\ 13\ 16\ 9\ 11\ 14\ 12\ 15]$.

Definition 3. A *qualification matrix* Q is an $m \times n$ matrix, where $Q[i, j] \in [0, 1]$ expresses the qualification value of agent $i (0 \le i < m)$ for role $j (0 \le j < n)$. $Q[i, j] = 0$ indicates the lowest value and 1 the highest.

$$
\begin{pmatrix}
0.34 & 0.31 & 0.38 & 0.35 & 0.32 & 0.38 & \ldots & 0.73 & 0.35 & 0.75 & 0.73 & 0.77 & 0.72 \\
0.34 & 0.37 & 0.31 & 0.34 & 0.36 & 0.33 & \ldots & 0.46 & 0.34 & 0.56 & 0.51 & 0.62 & 0.46 \\
0.15 & 0.17 & 0.14 & 0.22 & 0.17 & 0.12 & \ldots & 0.68 & 0.22 & 0.74 & 0.70 & 0.78 & 0.67 \\
0.24 & 0.21 & 0.28 & 0.25 & 0.22 & 0.29 & \ldots & 0.66 & 0.25 & 0.69 & 0.67 & 0.71 & 0.65 \\
0.27 & 0.24 & 0.30 & 0.18 & 0.24 & 0.35 & \ldots & 0.28 & 0.18 & 0.33 & 0.30 & 0.37 & 0.27 \\
0.17 & 0.16 & 0.21 & 0.11 & 0.16 & 0.25 & \ldots & 0.36 & 0.11 & 0.42 & 0.38 & 0.46 & 0.35 \\
0.50 & 0.46 & 0.54 & 0.41 & 0.46 & 0.59 & \ldots & 0.27 & 0.41 & 0.23 & 0.25 & 0.23 & 0.27 \\
0.11 & 0.12 & 0.12 & 0.18 & 0.12 & 0.11 & \ldots & 0.63 & 0.18 & 0.69 & 0.66 & 0.72 & 0.62
\end{pmatrix}
$$

Fig. 7. Q matrix in our scenario.

Notes: The number of rows represents the number of agents and the number of columns is equal to the number of roles. In our scenario, the number of roles is 58, and the number of agents is 8. For display purposes, only the first and last six columns are shown.

Note: A Q matrix can be obtained by comparing all the qualifications of agents with all the requirements of roles. In this scenario, we define the qualification of one agent as the distance between the base which the agent belongs to a specific relay point. We now get the $Q'[i, j]$, which is not normalized. Here, we use max–min normalization method to normalize $Q[i, j]$ as the following:

$$Q[i, j] = \frac{Q'[i, j] - \min\{Q'[i][j]\}}{\max\{Q'[i, j]\} - \min\{Q'[i, j]\}}.$$

The Q matrix for the scenario here is shown in Fig. 7.

Definition 4. A *role assignment matrix* T is defined as an $m \times n$ matrix, where $T[i, j] \in \{0, 1\}(0 \leq i < m, 0 \leq j < n)$ indicates whether or not agent i is assigned to role $j.T[i, j] = 1$ means yes and 0 no.

Definition 5. The *group performance* σ of group g is defined as the sum of the assigned agents' qualifications, i.e.

$$\sigma = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} Q[i, j] \times T[i, j].$$

Definition 6. Role j is *workable* in group g if it has been assigned enough agents, i.e.

$$\sum_{i=0}^{m-1} T[i, j] \geq L[j].$$

Definition 7. T is *workable* if each role j is workable, i.e. $\sum_{i=0}^{m-1} T[i, j] = L[j](0 \leq j < n)$. Group g is *workable* if T is workable.

From the above definitions, group g can be expressed by a Q, L, T, and L^a.

Definition 8. The UAVs assignment problem via GRA it to find a T to

$$\min \sigma = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} Q[i, j] \times T[i, j]$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & \ldots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \ldots & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & \ldots & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Fig. 8. T matrix in our scenario.

Notes: The number of rows represents the number of agents and the number of columns is equal to the number of roles. In our scenario, the number of roles is 58, and the number of agents is 8. For display purposes, only the first and last six columns are shown.

subject to

$$T[i,j] \in \{0,1\} \quad (0 \le i < m, 0 \le j < n), \tag{2}$$

$$\sum_{i=0}^{m-1} T[i,j] = L[j] \quad (0 \le j < n), \tag{3}$$

$$\sum_{j=0}^{j-1} T[i,j] \le \left\lceil \frac{L^a[i]}{2} \right\rceil \quad (0 \le i < m), \tag{4}$$

where

- Equation (2) is a 0-1 constraint.
- Equation (3) indicates that each role must have enough agents to solve its task.
- Equation (4) is not similar to the original GRA problem and means that an agent can only be assigned to a limited number of roles; each base needs to keep at least half of the limited number of roles for replacing those assigned drones with low battery.

For instance, Fig. 8 is a T following the constraints $L = [1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]$ and $L^a = [10\ 13\ 16\ 9\ 11\ 14\ 12\ 15]$. The total cost of the drones from eight bases is 12.30, which is an optimal result via GRA.

Note that this problem is a linear programming problem if we transfer the Q and T matrix into vectors. Then, it can be solved using an industry standard optimization tool like the PuLP linear programming toolkit, which is an open source package of Python.

Figure 9 shows the assignment result for our solutions. From Fig. 9, we note that there are some intersected coverages from different drones. It can also help improve the stability of the relay network. From our scenario, we assume that one drone can only guarantee stable communication of one communication link. Although some intersected drones of the relay network may seem surplus if the covered radius of a drone is much bigger than the covered radius of the communication vehicle, we need to maintain the fault tolerance and stability of each communication link, especially

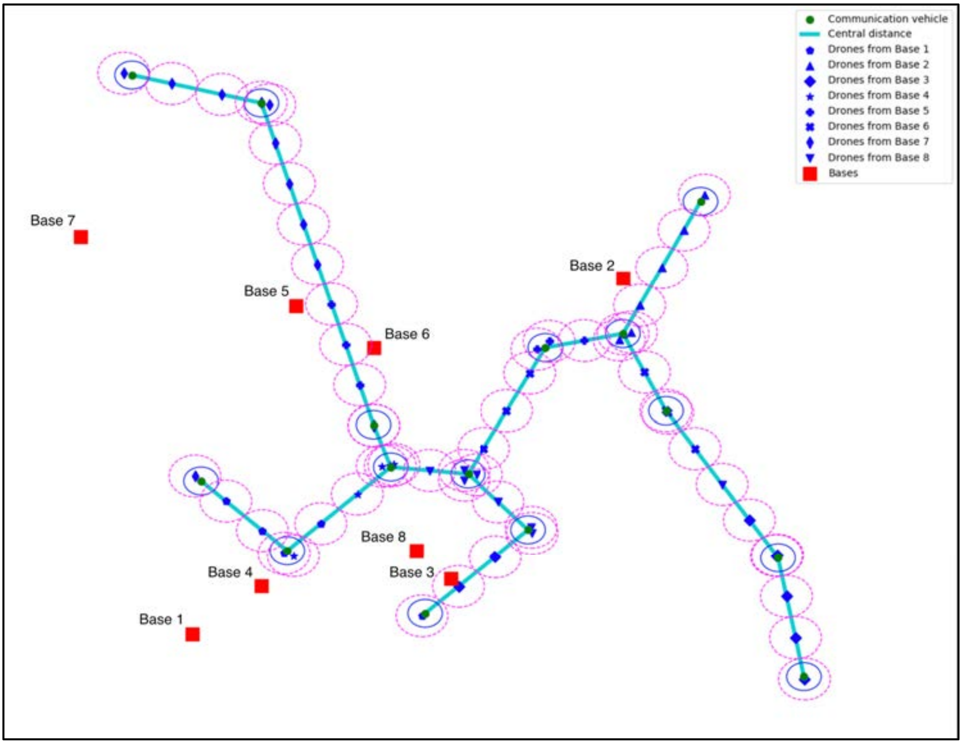

Fig. 9. The assignment result of our solution.

Notes: Different kinds of shapes of the minimum spanning tree or near the minimum spanning tree represent the UAVs from different base stations, and a solid circle represents the communication coverage radius of a communication vehicle while a dotted circle is the communication coverage radius of a UAV.

when swapping out relay network drones with low batteries. Besides, if all of the intersected UAVs are replaced by one shared UAV, it will greatly influence all of the location communication vehicles connected to the shared UAV should its service be unexpectedly interrupted. Thus, our solution tries to keep all of the intersected UAVs for the stability of local communications while increasing the number of back-up UAVs.

5. Solutions and Experiments

5.1. *The flowchart of our solution*

To recover global communication of the disaster area, we apply our solutions as follows. First, we use Prim's algorithm and our proposed method in Sec. 5 to specify the roles; we utilize Prim's algorithm to conditionally find out the practically optimal relay network with the least communication cost and then implement our proposed role awareness algorithm which can ensure minimal drone use to find the

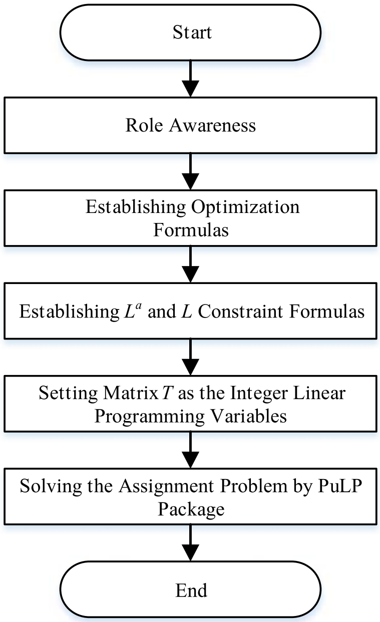

Fig. 10. The flowchart of solution.

proper relay points while ensuring the stability of local communication from the relay network. After becoming aware of the roles, define the agents, and calculate the qualification matrix that evaluates the agents for the role, we use the RBC and its model E-CARGO to formalize the problem as an x-ILP problem. The specific flowchart of our solution is given in Fig. 10.

5.2. *Role awareness*

Based on the analysis in Sec. 4, we propose a role awareness method. In fact, it is the step of modeling the RBC and E-CARGO, which is the so-called role negotiation. This method moves the original relay point along the extension line until the distance between drones reaches its max communication distance. In order to ensure the UAVs fully cover all of the communication vehicles, we need to judge the position of the boundary UAVs for each line. These processes aim to maximize the coverage of the drones as well as locally minimizing the communication cost. This method also keeps the relay network robust because there may be some overlapping coverage, see Fig. 2. Local communication stability can be maintained when changing out those UAVs with low batteries. We formalize our role awareness method as follows.

Here, we analyze the time complexity of the algorithm above. When we choose one edge from the minimum spanning tree, all of the relay points of this edge are determined based on the ratio relationship between the edge length and the

The method of finding the relay points

Input:

- A list of edges of the communication vehicles of the minimum spanning tree **MST**
- The covered radius of the drone $\mathbf{r_t}$
- The maximum connection distance between drones $\mathbf{D_{drone}}$

Output:

- A list of the relay points $\mathbf{R_p}$

```
FindRelayPoints(MST)
{   Rp = initializeRelayPoints()//initialize the list of the relay points.
    minNumber = initializeMinNumber() //initialize the minimum drone's
      number.
    centralPoint = initializeCentralPoint(MST)//initialize the central
      point of the edge.
    while(MST){//when the spanning tree heap is not empty.
      E=popEdges(MST) //pop the smallest distance from the heap
      minNumber = getMinDrones(MST, E, rt) //add the edge into the
        MST tree
      if(minNumber % 2 =0){//if the number of drones is even
//add the points except for the boundary points of the edge if the distance
           //between the point and the central point is several times of the
             Ddrone
           addNotBoundaryPointsForEvenNumber(centralPoint,
             Ddrone, minNumber)
//add boundary points
addBoundaryPointsForEvenNumber(centralPoint, Ddrone,
      minNumber)
           }
      else {addCentralPoint(Rp, centralPoint) //add the central point to
             the list of relay points
           addNotBoundaryPointsForOddNumber(centralPoint,
             Ddrone, minNumber)
           //add boundary points
addBoundaryPointsForOddNumber(centralPoint, Ddrone,
      minNumber)
             }
         }
      Return Rp
}
```

coverage distance of a UAV. Thus, the time complexity of finding the relay points in each edge is $O(1)$. The total time complexity of this algorithm is $O(n)$, where n represents the number of edges of the minimum spanning tree, based on the *while* loop.

With the above algorithm, in the scenario discussed in Sec. 3, we can obtain the relay points shown as the blue shapes in Fig. 9. According to the assumption that one UAV can only guarantee stable communication with one other link, the number of relay points is 58.

5.3. *Experiments*

Here, we use our proposed real-world scenario as an example to carry out different scale of experiments. Just like we assumed in Sec. 3, the number of the UAV base stations is 8 and the number of communication vehicles is 15. With the help of the method of finding the minimum relay points mentioned in Sec. 5.2, we can find 58 relay points as roles to achieve full coverage communication for the communication vehicles. After gaining an awareness of the roles and calculating the qualification matrix that evaluates the agents for each role, we can transform the signal relay problem to a linear objective function with linear constraints based on Definition 8. To swiftly solve this linear programming problem, we decided to use the PuLP package of Python which is a mathematical programming package.[34] Figure 9 shows the final assignment result.

In order to ensure that our solutions are practical in a real-world scenario, we have conducted thousands of simulation experiments to test proposed algorithms at different scales. All experiments used the platform shown in Table 2.

The time cost on different scales of our simulation experiments is shown in Fig. 11. Figure 11 illustrates that our solution can quickly determine proper relay points and distribute UAVs from different bases to the relay points regardless of the different scale of tasks or the different distribution of tasks in our scenario. Rescue-related problems are time-critical and require us to obtain optimal solutions quickly. Therefore, our solution is suitable and practical for communication rescue in disaster relief. In addition, Fig. 11 indicates that our solutions are also effective, robust and practically global optimal when the positions of communication vehicles

Table 2. The configuration of the experimental platform.

Hardware	
CPU	2.6 GHz Intel Core i7
Memory	16 GB 2400 MHz DDR4
Software	
OS	macOS Mojave Version 10.14.4
Editor	Visual Studio Code Version 1.33.1
Python	Python 3.7.1

are determined before the relay points are selected and the number of UAVs is sufficient for the full coverage of all communication vehicles.

6. Conclusion

This paper formalizes a successful approach to solve the x-ILP problem with the help of PuLP package of Python. It designs a uniform solution concerning the signal relay problem of UAV in a disaster area. And thousands of random simulation experiments show that our solution can quickly find the collaborative relay points and assign the UAVs from different base stations to these points. It also demonstrates that our solution is practical, robust, and practically global optimal.

This paper formalizes and solves the signal relay problem of the UAVs in disaster relief via GRA. In modeling, it first uses the minimum spanning tree algorithm to design the practically collaborative optimal relay network. And our proposed method in Sec. 4 can establish an awareness of the relay points, which are taken as roles to be assigned with UAVs (as agents). Second, distributing the UAVs to these collaborative relay points can be solved quickly via the assignment process of GRA. Through these strategies, it can quickly recover global communication in the disaster area with the least communication cost regardless of the scale and the distribution.

From this paper, further investigations may be required in the following directions:

- There may be additional constraints that required further investigations like the mutually exclusive constraints between drones from different bases, and the multi-task assignments for each drone.
- Some UAV properties, such as stability, endurance, and maximum flight altitude, raise concerns worthy of additional study.

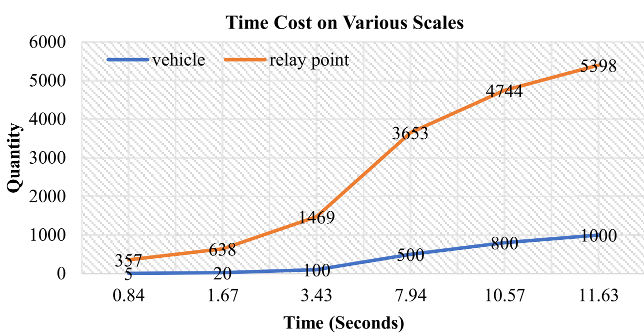

Fig. 11. Time cost on various scales.
Notes: To analyze the impact of the number of vehicles on time, we set the covered radius of the UAVs as 3 km, the number of bases is 8, and the location of the communication vehicles range from 3 km to 1,200 km for discretely generating them so as to cover the whole disaster area.

- Elevation differences in a disaster area suggest that the third dimension needs to be considered when assigning drones.

Acknowledgments

We appreciate Mike Brewes for his assistance in proofreading this paper. This paper is extended from our previous work, i.e. Q. Jiang, H. Zhu, M. Liao, B. Huang, X. Fang and D. Liu, "Solving the Signal Relay Problem of UAV in Disaster Relief via Group Role Assignment", which has been accepted by the *Chinese Conference on Computer Supported Cooperative Work and Social Computing 2019* (*ChineseCSCW 2019*) as Full Paper.

Thanks to National Natural Science Foundation of China for supporting part of this work under Grant No. 61402118, Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN2018-04818, S&T Project of Guangdong Province, Grant No. 2016B010108007 and S&T Project of Guangzhou, Grant No. 201604020145.

References

1. H. Zhu and M. C. Zhou, Role-based collaboration and its kernel mechanisms, *IEEE Trans. Syst. Man Cybern.* **36**(4) (2006) 578–589.
2. D. Liu, B. Huang and H. Zhu, Solving the tree-structured task allocation problem via group multirole assignment, *IEEE Trans. Autom. Sci. Eng.* **17**(4) (2020) 41–55.
3. D. Liu, Y. Yuan, H. Zhu, S. Teng and C. Huang, Balance preferences with performance in group role assignment, *IEEE Trans. Cybern.* **48**(6) (2017) 1800–1813.
4. H. Zhu, M. C. Zhou and R. Alkins, Group role assignment via a Kuhn–Munkres algorithm-based solution, *IEEE Trans. Syst. Man Cybern.* **42**(3) (2012) 739–750.
5. H. Zhu, Avoiding conflicts by group role assignment, *IEEE Trans. Syst. Man Cybern. Syst.* **46**(4) (2016) 535–547.
6. X. Liu and N. Ansari, Resource allocation in UAV-assisted M2M communications for disaster rescue, *IEEE Wirel. Commun. Lett.* **8**(2) (2018) 580–583.
7. C. Wankmüller, C. Truden, C. Korzen, P. Hungerländer, E. Kolesnik and G. Reiner, Base station allocation of defibrillator drones in mountainous regions (2019), arXiv:1902.06685.
8. C. Mouradian, J. Sahoo, R. H. Glitho, M. J. Morrow and P. A. Polakos, A coalition formation algorithm for multi-robot task allocation in large-scale natural disasters, in *13th Int. Wireless Communications and Mobile Computing Conf.* (*IWCMC 2017*), Valencia (2017), pp. 1909–1914.
9. I. Maza, F. Caballero, J. Capitan, J. R. Martinez-De-Dios and A. Ollero, Firemen monitoring with multiple UAVs for search and rescue missions, in *8th IEEE Int. Workshop on Safety, Security, and Rescue Robotics* (*SSRR-2010*), Philadelphia, PA, USA (2010), doi: 10.1109/SSRR.2010.5981565.
10. S. Chopra, G. Notarstefano, M. Rice and M. Egerstedt, A distributed version of the Hungarian method for multirobot assignment, *IEEE Trans. Robot.* **33**(4) (2017) 932–947.
11. D. Li, M. Li, X. Meng and Y. Tian, A hyperheuristic approach for intercell scheduling with single processing machines and batch processing machines, *IEEE Trans. Syst. Man Cybern. Syst.* **45**(2) (2015) 315–325.
12. Z. Wang and C. Shen, Small cell transmit power assignment based on correlated bandit learning. *IEEE J. Sel. Areas Commun.* **35**(5) (2017) 1030–1045.

13. X. Zhang, J. Wang, Y. Fang and J. Yuan, Multilevel humanlike motion planning for mobile robots in complex indoor environments, *IEEE Trans. Autom. Sci. Eng.* **16**(3) (2018) 1244–1258.
14. Y. Li, X. Li and J. N. D. Gupta, Solving the multi-objective flowline manufacturing cell scheduling problem by hybrid harmony search, *Expert Syst. Appl.* **42**(3) (2015) 1409–1417.
15. J. Wang, Emergency healthcare workflow modeling and timeliness analysis, *IEEE Trans. Syst. Man Cybern. A*, *Syst. Humans* **42**(6) (2012) 1323–1331.
16. N. Geng, Q. Meng, D. Gong and P. W. H. Chung, How good are distributed allocation algorithms for solving urban search and rescue problems? A comparative study with centralized algorithms, *IEEE Trans. Autom. Sci. Eng.* **16**(1) (2019) 478–485.
17. O. Asik and H. L. Akin, Effective multi-robot spatial task allocation using model approximations, in *RoboCup 2016: Robot World Cup XX, LNCS,* Vol. 9776 (Springer, Heidelberg, 2017), pp. 243–255.
18. S. Singh and R. Singh, Earthquake disaster based efficient resource utilization technique in iaas cloud, *Int. J. Adv. Res. Comput. Eng. Technol.* **2**(6) (2013) 225–229.
19. G. Fortino, W. Russo, C. Savaglio, W. Shen and M. Zhou, Agent-oriented cooperative smart objects: From IoT system design to implementation, *IEEE Trans. Syst. Man Cybern. Syst.* **48**(11) (2018) 1939–1956.
20. M. Gomez, E. Plaza and C. Abasolo, Problem-solving methods and cooperative information agents, *Int. J. Coop. Inform. Syst.* **11**(3&4) (2002) 329–354.
21. Y. C. Jiang, J. Hu and D. Lin, Decision making of networked multiagent systems for interaction structures, *IEEE Trans. Syst. Man Cybern. A*, *Syst. Humans* **41**(6) (2011) 1107–1121.
22. L. Tian, J. Luo, Y. Zeng and H. Wu, Modeling and algorithms for multiagent communication through interactive dynamic influence diagrams, *Appl. Artif. Intell.* **30**(4) (2016) 352–377.
23. W. Y. Wang, Y. C. Jiang and W. W. Wu, Multiagent-based resource allocation for energy minimization in cloud computing systems, *IEEE Trans. Syst. Man Cybern. Syst.* **47**(2) (2017) 205–220.
24. N. R. Jennings, P. Faratin, M. J. Johnson, T. J. Norman, P. O'Brien and M. E. Wiegand, Agent-based business process management, *Int. J. Coop. Inform. Syst.* **5**(2&3) (1996) 105–130.
25. Z. Liu, T. Liu, T. Tun, L. Cai and G. Yang, Agent-based online quality measurement approach in cloud computing environment, in *Proceedings of 2010 IEEE/WIC/ACM Int. Conf. Web Intelligence and Intelligent Agent Technology* (*WI-IAT 2010*), Toronto (2010), pp. 686–690.
26. X. Zheng and H. Liu, A hybrid vertical mutation and self-adaptation based MOPSO, *Comput. Math. Appl.* **57**(11–12) (2009) 2030–2038.
27. R. Nair and M. Tambe, Hybrid BDI-POMDP framework for multiagent teaming, *J. Artif. Intell. Res.* (2015), https://doi.org/10.1613/jair.1549.
28. R. Burkard, M. DellAmico and S. Martello, *Assignment Problems*, revised reprint (SIAM, Philadelphia, PA, USA, 2012).
29. M. Dastani, V. Dignum and F. Dignum, Role-assignment in open agent societies, in *Proc. Second Int. Joint Conf. Autonomous Agents and Multiagent Systems* (ACM, New York, USA, 2003), pp. 489–496.
30. Y. Sun and P. Pan, PRES: a practical flexible RBAC workflow system, in *Proc. 7th Int. Conf. Electronic Commerce* (*ICEC 2005*), Xi'an, China (2005), pp. 653–658.
31. N. Mundbrod, and M. Reichert, Object-Specific Role-Based Access Control, *Int. J. Coop. Inform. Syst.* **28**(1) (2019) 1950003-1–30.

32. H. Zhu, D. Liu, S. Zhang, Y. Zhu, L. Teng and S. Teng, Solving the many to many assignment problem by improving the Kuhn–Munkres algorithm with backtracking, *Theore. Comput. Sci.* **618**(7) (2016) 30–41.
33. S. Ren, P. Yi, D. Hong, Y. W and T. Zhu, Distributed construction of connected dominating sets optimized by minimum-weight spanning tree in wireless Ad-hoc sensor networks, in *Proc. 2014 IEEE 17th Int. Conf. Computational Science and Engineering* (*CSE 2014*), Chengdu, China (2014).
34. S. Mitchell, M. O'Sullivan and I. Dunning, *PuLP: A Linear Programming Toolkit for Python* (Department of Engineering Science, The University of Auckland, Auckland, 2011).