

Task Offloading for Post-Disaster Rescue in Unmanned Aerial Vehicles Networks

Yuntao Wang^{ID}, Weiwei Chen, Tom H. Luan^{ID}, Senior Member, IEEE, Zhou Su^{ID}, Senior Member, IEEE,
Qichao Xu^{ID}, Ruidong Li^{ID}, Senior Member, IEEE, and Nan Chen^{ID}, Member, IEEE

Abstract—Natural disasters often cause huge and unpredictable losses to human lives and properties. In such an emergency post-disaster rescue situation, unmanned aerial vehicles (UAVs) are effective tools to enter the damaged areas to perform immediate disaster recovery missions, owing to their flexible mobilities and fast deployment. However, UAVs typically have very limited battery and computational capacities, which makes them harder to perform heavy computation tasks during the complicated disaster recovery process. This paper addresses the issue of the battery and computation resource limitation with a fog computing based UAV system. Specifically, we first introduce the vehicular fog computing (VFC) system in which the unmanned ground vehicles (UGVs) perform the computation tasks offloaded from UAVs. To avoid the transmission competitions yet enable cooperations among UAVs and UGVs, a stable matching algorithm is developed to transform the computation task offloading problem into a two-sided matching problem. An iterative algorithm is then developed which matches each UAV with the most suitable UGV for offloading. Finally, extensive simulations are carried out to demonstrate that the proposed scheme can effectively improve utilities of UAVs and reduce average delay through comparison with conventional schemes.

Index Terms—Unmanned aerial vehicles (UAVs), computation offloading, post-disaster rescue, matching.

I. INTRODUCTION

NATURAL disasters, such as earthquakes, mudslides and wildfires, could cause loss to thousands of lives and properties, while also incurring economic damage to the affected areas. As reported in [1], during the past 50 years,

Manuscript received 15 February 2021; revised 29 July 2021 and 14 November 2021; accepted 27 December 2021; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor K. Lee. Date of publication 28 January 2022; date of current version 18 August 2022. This work was supported in part by the NSFC under Grant U20A20175 and Grant U1808207 and in part by the Fundamental Research Funds for the Central Universities. An earlier version of this paper was presented in part at IEEE INFOCOM2020 [DOI: 10.1109/INFOCOM41043.2020.9155397]. (Corresponding author: Tom H. Luan.)

Yuntao Wang and Zhou Su are with the School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: yuntao.wang@stu.xjtu.edu.cn; zhousu@ieee.org).

Weiwei Chen and Qichao Xu are with the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China (e-mail: chenww@shu.edu.cn; xqc690926910@shu.edu.cn).

Tom H. Luan is with the School of Cyber Information Engineering, Xidian University, Xi'an 710071, China (e-mail: tom.luan@xidian.edu.cn).

Ruidong Li is with the Institute of Science and Engineering, Kanazawa University, Kanazawa 920-1192, Japan (e-mail: liruidong@ieee.org).

Nan Chen is with the Department of Electrical and Computer Engineering, Tennessee Tech University, Cookeville, TN 38505 USA (e-mail: nchen@tnstate.edu).

Digital Object Identifier 10.1109/TNET.2022.3140796

the frequency of recorded natural disasters has surged almost five-fold globally. While the pre-disaster planning is crucial, effective and fast-response post-disaster rescue is also very significant to the disaster recovery [2], [3]. The emerging UAVs combined with artificial intelligence (AI) technology have revolutionized the methodology of natural disaster rescue. Due to their low cost, high mobility and deployment flexibility, UAVs can easily enter the affected areas that are otherwise difficult to reach [4]. As such, utilizing the onboard sensors and wireless communications of UAVs, the digital map of the post-disaster area can be drawn. Furthermore, with their flexibility and mobility, UAVs can be utilized as communication relays whereby the dispatching and commands for post-disaster rescue can be efficiently forwarded to the rescue equipment on the ground for work [5]. With its mobility, flexibility, and communication capability, the UAV will become increasingly promising to play a major role in the post-disaster rescue [6]–[8].

However, the post-disaster rescue carried out by UAVs faces fundamental engineering challenges. On one hand, the capacity of UAV's on-board battery is limited due to its stringent space and weight requirements [9], which indicates that UAV can only serve rescue tasks for a limited time and coverage. On the other hand, due to the limited computational capability of a UAV, real-time applications such as image processing and video streaming often exceed UAVs' local data processing capabilities, resulting in computation delay [10], [11]. In addition to the computation capacity limitation, performing intensive computation tasks, such as video preprocessing, pattern recognition, and feature extraction, would be detrimental to their battery lifetime [12]. Therefore, a large number of intensive computation missions undertaken by UAVs during post-disaster rescue need to be offloaded to nearby computationally powerful devices for timely rescue response and enabling sustainable UAV life cycles.

A number of works have attempted to address the issue of UAVs' computation offloading, which can be mainly divided into two thrusts: offloading to the mobile edge computing (MEC) server and offloading to the cloud [9], [13]–[15]. MEC server cooperates with UAVs to perform the offloaded computation tasks from the UAVs timely [16], [17]. In specific, the computation missions produced by UAVs can be divided into two parts: one is computed on UAVs and the other is computed by the MEC server which receives it through the gateway or the access point (AP) [10]. While this approach guarantees low-latency UAV services, the computational capability of

MEC server is also limited and may not accomplish [18]. Offloading UAVs' computation tasks to remote cloud servers represents another effective method [19]. The remote cloud servers abstract the complexity of the infrastructures, providing unlimited computation resources and storage space through online access. However, it is difficult to guarantee UAVs' stringent quality of experience (QoE) considering the long-distance transmission between UAVs and the remote cloud and corresponding delay. Due to the emergent nature of disaster responding, current approaches cannot solve the UAV's offloading problem while well balancing the trade-off between resource shortage and computation task demand. Thereby, it is still an open and vital issue to design an efficient computation offloading scheme for UAVs.

Unmanned ground vehicles (UGVs) are smart vehicles that integrate a variety of novel technologies. Without human intervention, UGVs can perceive the environment information of the disaster area through various sensors and convert it into a language that the computer can recognize. The on-board computer parses the information and instructs the vehicle to conduct rescue tasks such as environmental perception and material delivery in the disaster area. To address the above issues, we propose a practical and efficient computation offloading scheme for UAVs, which utilizes the computation resources of UGVs to perform computation offloading. This computing paradigm that utilizes the idle computation resources of UGVs is known as the vehicular fog computing (VFC) [20], [21]. Specifically, a number of UGVs that have sufficient computation resources in total are deployed in disaster areas to meet the overall computing requirements of UAVs. Nevertheless, each UGV can only serve a number of UAVs due to the limitation of computation capacity. When UGVs are within the communication range of UAVs, UAVs can directly exchange data with UGVs through the high-rate short-range air-to-ground (A2G) communication which is based on IEEE 802.11n [22]. In this case, UAVs preferably offload the computation tasks to UGVs with idle computation resources. Due to the competitions and cooperations among UAVs and UGVs, a stable matching algorithm is proposed to match each UAV with the UGV that benefits it most. For a small number of UAVs that fail to match, their computation tasks are performed by base stations that survived during the disaster. Furthermore, the computation tasks of UAVs can be offloaded directly to UGVs without passing through base stations to further reduce the transmission delay. The main contributions of this paper are summarized below.

- *Framework:* We propose a tripartite dynamic cooperation framework among UAVs, UGVs and base stations during the post-disaster rescue. Based on the aerial subnetwork, multiple UAVs form a cooperative service platform, offloading computation tasks to and receiving the result from UGVs. A UAV first offloads the computation tasks to UGVs, which then uploads the execution results to the UAV within the communication range, and the UAV transmits the execution results to the UAV which initiates the offloading in the first place. The cooperation between UAVs and UGVs and that between UAVs and base sta-

tions are dynamic and static computation task offloading, respectively. Based on the A2G communications, UAVs prioritize the offloading of computation tasks to UGVs directly to reduce the transmission delay, whereas base stations are employed to receive the offloaded tasks when UGVs have no sufficient computing resources.

- *Scheme:* We devise a distributed stable matching algorithm that makes full use of the dynamic characteristics of UAVs and UGVs. UAVs select UGVs to match and offload computation tasks according to the trajectory and speed of UGVs, and then obtain execution results based on the cooperation among UAVs. To maximize the utility of UAVs, the proposed algorithm transforms UAVs' computation task offloading problem into a two-sided matching problem based on the structure of the problem. One side is UAVs and the other is UGVs, and both sides prefer to select the object which can maximize their own profits. The preference lists are built based on the UAVs' and UGVs' operation preferences, e.g., the maximum achievable benefits. After that, the algorithm solves the problem by matching each UAV with the UGV which benefits it most in an iterative manner.
- *Validation:* From the perspective of time, the finite time horizon is discretized into equal time slots. In terms of space, the locations and communication distances of UAVs and UGVs in each time slot are different. In this paper, UAVs and UGVs in each time slot are considered to be approximately static to select appropriate matching objects to ensure the smooth execution of the task. The feasibility and effectiveness of the proposed scheme are evaluated by extensive simulations. Numerical results demonstrate that the proposed scheme can not only maximize the utility of UAVs, but also reduce the average delay, compared with conventional schemes.

The remainder of this paper is summarized below. Section II reviews the related works. The system model is introduced in Section III. The analysis of utilities of UAVs, UGVs and base stations is presented in Section IV. Section V describes the matching-based computation task offloading scheme and Section VI provides the performance evaluation. The paper is closed with the conclusion and future work in Section VII.

II. RELATED WORK

In this section, we review the related works including applications of UAVs in wireless networks, computation offloading in MEC and cloud, and coordination between UAVs and UGVs.

A. Applications of UAVs in Wireless Networks

Extensive works have been conducted on applications of UAVs in wireless networks. Due to the high mobility, flexible deployment and strong adaptability of UAVs, UAVs have been considered as an important component for the successful development of various applications. On one hand, UAVs can act as aerial base stations for improving the connectivity of wireless equipments on the ground. On the other hand, UAVs

can act as airborne wireless relays, extending the coverage of wireless networks to provide stable wireless network communication for areas with inconvenient communication [23], [24]. Mozaffari *et al.* [25] study multiple UAVs acting as aerial base stations to collect the data from ground Internet of things (IoT) devices to realize reliable uplink communications with the minimum transmit power. The security of the cache-based UAV relay wireless networks is investigated by Cheng *et al.* [26], which maximizes the minimum average confidentiality rate among all users by jointly optimizing the time scheduling and trajectory of UAVs. A UAV-enabled wireless network in which relay UAVs are used for two-way communication between a group of remote user equipments and ground base stations is proposed by Li *et al.* [27] to maximize the overall rate of downlink and uplink. In both static and dynamic scenarios, Mozaffari *et al.* [28] analyze the feasibility of deploying UAVs as aerial base stations to provide wireless communication for a given geographical area to minimize the transmit power.

Different from the above works where the applications of UAVs are focused on navigation, communication and civilian use, our work discusses the feasibility of flying, communicating and performing missions during post-disaster rescue from perspectives of time and space.

B. Computation Offloading in MEC and Cloud

A number of works have been developed to offload the computation tasks to the MEC server and cloud. A transportation optimization model driven by data to distribute the collected data to the MEC server to minimize the computational effort is proposed by Garg *et al.* [16]. Liu *et al.* [18] investigate the joint routing optimization problem and computation offloading problem for UAVs under the computing architecture of the UAV-edge-cloud. Tran and Pompili [29] propose a heuristic algorithm to obtain the optimal solution of the large-scale network and study the joint computation task offloading at the MEC server. Based on the sequential game, Messous *et al.* [30] adopt a theoretical methodology to improve the performance of the energy overhead and decrease the execution delay to provide feasible solutions for the computation offloading in a UAV network. A game-theoretic approach to achieve efficient computation offloading for the MEC server in a distributed manner is developed by Chen *et al.* [31]. Gao *et al.* [32] utilize the offloading rate differentiation to guarantee a better QoE of the high-priority data in mobile cloud computing. A distributed resource allocation and computation offloading optimization scheme in the heterogeneous network with the MEC server is proposed by Zhang *et al.* [33]. Ti and Le [34] use the distributed and centralized computing architecture of fog-cloud computing to study the computation offloading problem with UAVs. Information-centric network [35] is also a promising approach for offloading computation tasks.

Unlike these studies on offloading computation tasks to the MEC server or the cloud, our work designs a feasible scheme which utilizes the idle computation resources shared by UGVs to offload the computation tasks of UAVs.

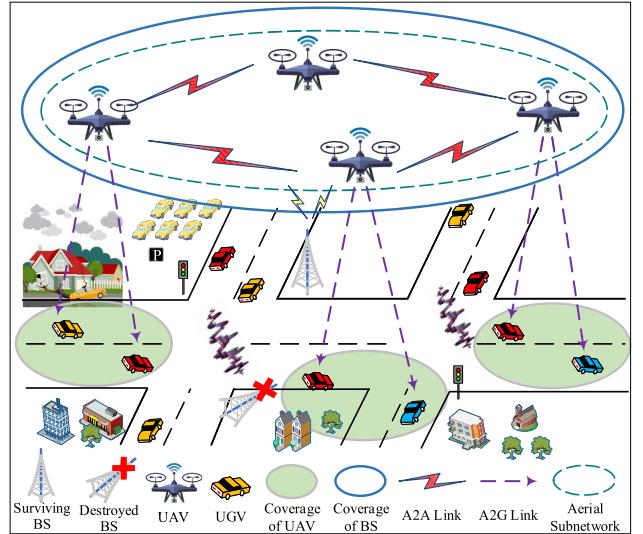


Fig. 1. Network structure for UAVs' computation offloading.

C. Coordination Between UAVs and UGVs

Extensive research has been conducted on the coordination between UAVs and UGVs. The performance gain of downlink transmission by utilizing multiple UAVs in cooperation with vehicles is investigated by Zhang and Liu [36] to recover wireless communication in areas where the network is paralyzed. Shang *et al.* [37] incorporate UAV into vehicle-to-everything (V2X) communications to build a physically secured platform for V2X communication. Liu *et al.* [38] study the cooperative routing problem of a new type of hybrid platform of ground vehicles and UAVs for intelligence, surveillance and reconnaissance missions. Zhang *et al.* [39] propose an imitation-enhanced deep reinforcement learning model, which allows UGVs and UAVs to form a complementary and cooperative alliance to accomplish tasks that they cannot accomplish alone. Li *et al.* [40] study the construction of effective path planning and automatic ground maps in the UAV-UGV collaboration system. A novel integrated vehicle system that utilizes cooperative UAVs and UGVs to conduct navigation, autonomous exploration and mapping in a 3-dimensional unknown environment denied by GPS is developed by Qin *et al.* [41].

Our work is different from the above works which simply utilize UAVs and UGVs as two separate individuals to collaborate, but merges the two into a whole to share resource.

III. SYSTEM MODEL

The system model is introduced in this section, which consists of motion model of UAVs, mobility model of UGVs, and communication model. Table I summarizes the notations utilized in this paper.

A. Motion Model of UAVs

The system model introduced in this paper consists of three types of entities, i.e., UAVs, UGVs and base stations, as shown in Fig. 1. The set of UAVs within the coverage of base stations is denoted as $\mathcal{M} = \{1, \dots, m, \dots, M\}$.

TABLE I
SUMMARY OF NOTATIONS

Symbols	Descriptions
$\mathcal{M}, \mathcal{N}, \mathcal{J}$	The set of UAVs, UGVs, and base stations for post-disaster rescue, respectively.
H, I	The fixed altitude of UAVs and number of time slots.
T	The finite time horizon during the process of performing tasks by UAVs.
δ, t_i	The equal length of each time slot and i th time slot.
$[x_m(t_i), y_m(t_i)]$	The instantaneous horizontal plane coordinates of UAV m in i th time slot.
$[x_n(t_i), y_n(t_i), 0]$	The instantaneous coordinate of UGV n in i th time slot.
$[x_j, y_j, 0]$	The coordinate of base station j .
$q_m(t_i), s_m(t_i)$	The instantaneous location and status of UAV m in i th time slot, respectively.
$q_m^{\text{start}}, q_m^{\text{end}}$	The start and end location of UAV m .
Q_m	The flight trajectory of UAV m within the finite time horizon T .
V_m^{\max}, L_m^{\max}	The maximum velocity and maximum flight distance of UAV m , respectively.
$\frac{\varepsilon}{\Omega_m(t_i)}, D$	The number of times the flight direction has changed and safe flight distance between every two UAVs.
$\Omega_m(t_i)$	The flight direction of UAV m within i th time slot.
ϕ	The maximum turning angle at which the flight direction changes between time slots.
\bar{v}, v, ρ	The average velocity, velocity of UGVs, and traffic density, respectively.
o_m	The number of UGVs entering the communication range of UAV m in each time slot.
g_m	The percentage of UGVs leaving the communication range of UAV m .
$F_m(t_i)$	The total number of UGVs within the communication range of UAV m in i th time slot.
$h_{m,n}(t_i), h_{m,j}(t_i)$	The channel gain between UAV m and UGV n or base station j in i th time slot, respectively.
η_0, φ	The channel power gain associated with a reference distance and path loss exponent.
$b_{m,n}(t_i), b_{m,j}(t_i)$	The instantaneous transmit power from UAV m to UGV n or base station j in i th time slot, respectively.
$\tilde{B}_{m,n}(t_i), \tilde{B}_{m,j}(t_i)$	The maximum power threshold between UAV m and UGV n or base station j in i th time slot, respectively.
$\gamma_{m,n}, \gamma_{m,j}$	The available data transmission rate between UAV m and UGV n or base station j in i th time slot.
$\omega_{m,n}, \omega_{m,j}$	The allocated spectrum bandwidth between UAV m and UGV n or base station j , respectively.
$\sigma^2, S_{m,n}(T_{m,n})$	The power of the additive white Gaussian noise and satisfaction function of UAV m .
D_m, C_m, T_m	The data size of the task, computation resource required for the task, and delay constraint, respectively.
$T_{m,n}^t, T_{m,n}^d, T_{m,n}$	The transmission time from UAV m to UGV n , dwell time of UGV n , and total delay, respectively.
$p_n, P_{m,n}(p_n)$	The price of unit computation resource charged by UGV n and payment from UAV m to UGV n .
ϖ	The unit sensing energy consumption of UAV m for the threat source z .
$\tau_m^z(t_i)$	The threat cost in sensing the threat source z in the flight of UAV m within i th time slot.
ℓ_m, Γ_m, E_m	The flight loss, overall threat cost of UAV m , and comprehensive energy cost, respectively.
$T_{m,n}^c, T_{m',j}^c$	The computation time of UGV n and base station j , respectively.
Υ_n, Υ_j	The number of CPU cycles of UGV n and base station j that is required per bit of data size, respectively.
f_n, Δ_j	The idle computation resources shared by UGV n and total computation resource of base station j , respectively.
Λ_n, Λ_j	The energy cost of UGV n and base station j in task computation, respectively.
ξ_n, ξ_j	The cost per unit of local computation resource of UGV n and base station j , respectively.
Ψ_n, Ψ_j	The energy cost of UGV n and base station j in data transmission, respectively.
$\chi_{m,n}, \chi_{m',j}$	The cost of data transmission from UAV m to UGV n or from UAV m to base station j per time slot, respectively.
$U_m(T_{m,n}, p_n), U_n(p_n), U_j(p_j)$	The utility of UAV m , UGV n , and base station j , respectively.

Each UAV is responsible for one area and these areas do not overlap each other. Let $\mathcal{N} = \{1, \dots, n, \dots, N\}$ and $\mathcal{J} = \{1, \dots, j, \dots, J\}$ denote the set of UGVs and base stations, respectively. A three-dimensional Cartesian coordinate system [42] for locations of the above three entities is adopted. In this system, we focus on the scenario where UAVs fly at a fixed hovering altitude H ($H > 0$), aiming to keep continuous flying over the air. The fixed hovering altitude of UAVs means the minimum altitude, which contributes to the avoidance of frequent aircraft ascending and descending owing to terrain or building blockage, so as to minimize the movement energy consumption of UAVs [4], [42], [43].

For the sake of simplicity, we discretize the finite time horizon T into I time slots, each with the equal length δ , i.e., $T = I\delta$ [43]. Within the coverage of base station j , the numbers of UAVs and UGVs remain constant in each time slot and change in different time slots. Let t_0 denote the initial time of time horizon T and t_i be the instantaneous time within i th time slot, for $t_i \in [t_0 + (i-1)\delta, t_0 + i\delta]$. Then, in i th time slot, the instantaneous location of UAV m is

$$q_m(t_i) = [x_m(t_i), y_m(t_i), H], \quad (1)$$

where $[x_m(t_i), y_m(t_i)]$ denotes the instantaneous horizontal plane coordinates of UAV m in i th time slot.

The start and end locations of UAV m are pre-determined, which are denoted as q_m^{start} and q_m^{end} , i.e.,

$$q_m^{\text{start}} = [x_m^{\text{start}}, y_m^{\text{start}}, H], \quad (2)$$

$$q_m^{\text{end}} = [x_m^{\text{end}}, y_m^{\text{end}}, H], \quad (3)$$

where $[x_m^{\text{start}}, y_m^{\text{start}}]$ and $[x_m^{\text{end}}, y_m^{\text{end}}]$ are the starting and ending horizontal plane coordinates of UAV m , respectively.

The location of UAV m within i th time slot is fixed when δ is chosen to be small enough. In this case, combining the location of UAV m in I time slots, the flight trajectory of UAV m within the finite time horizon T can be modeled as

$$Q_m = \{q_m^{\text{start}}, \dots, q_m(t_i), \dots, q_m^{\text{end}}\}. \quad (4)$$

The mobility constraint of UAV m can be denoted as

$$\delta \cdot d_m^{\min}/T \leq \|q_m(t_i) - q_m(t_{i-1})\| \leq \delta \cdot V_m^{\max}, \quad (5)$$

where V_m^{\max} is the maximum velocity of UAV m . The shortest distance between the start location and end location of UAV m is defined as

$$d_m^{\min} = \sqrt{(x_m^{\text{end}} - x_m^{\text{start}})^2 + (y_m^{\text{end}} - y_m^{\text{start}})^2}. \quad (6)$$

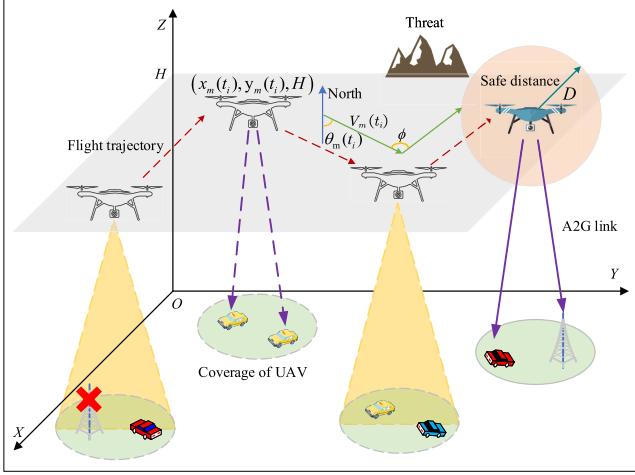


Fig. 2. Flight trajectory of UAVs.

B. Trajectory Planning Model of UAVs

The flight trajectory of UAVs is the set of flight segments in each time slot, which can be seen in Fig. 2. In i th time slot, the status of the instantaneous location of UAV m is expressed as a 5-tuple, i.e.,

$$s_m(t_i) = (x_m(t_i), y_m(t_i), H, \theta_m(t_i), V_m(t_i)), \quad (7)$$

where θ_m is the angle of UAV m between the tangent direction of flight and the reference heading, i.e., due north direction. $V_m(t_i)$ is the current velocity of UAV m which meets the limitation of maneuverability. When UAV m flies from the start location q_m^{start} to the end location q_m^{end} , there are multiple optimal or suboptimal flight trajectories. The specific form of the flight trajectory planning process of UAV m can be defined as

$$s_m^{\text{start}} \xrightarrow{v_{\text{route}}} s_m^{\text{end}}, \quad (8)$$

where $\mathcal{U}_{\text{route}}$ represents the set of all feasible flight trajectories that meet the constraints.

The flight distance of UAV m within the i th time slot can be expressed as $\|q_m(t_i) - q_m(t_{i-1})\|$, so the flight trajectory of UAV m is composed of I segments of different lengths passing through each time slot. Let L_m^{\max} represent the maximum flight distance, we have

$$\sum_{i=1}^I \|q_m(t_i) - q_m(t_{i-1})\| \leq L_m^{\max}. \quad (9)$$

The instantaneous location of the transition between the current time slot and the next time slot can be regarded as a waypoint. When planning the flight trajectory of UAV m , the shortest flight distance between the waypoints in the planned trajectory after the change of the flight direction is expressed as the minimum flight trajectory segment length l_m^{\min} . Therefore, the flight distance of UAV m is constrained by

$$(\varepsilon + 1) l_m^{\min} \leq \sum_{i=1}^I \|q_m(t_i) - q_m(t_{i-1})\| \leq L_m^{\max}, \quad (10)$$

where ε represents the number of times that the flight direction has changed. The distance that UAV m keeps flying in a

straight line before the flight direction changes is l_d , and the constraint condition can be expressed as

$$l_d \leq l_m^{\min}. \quad (11)$$

When the flight direction of UAV m changes between the current flight segment and the next segment, it should be ensured that the turning angle of UAV m is within a safe angle range. Let $\vec{\Omega}_m(t_i)$ be the flight direction vector of UAV m within i th time slot. The maximum turning angle at which the flight direction changes between i th time slot and $i+1$ th time slot is ϕ ($\phi < 90^\circ$). We have

$$\frac{\vec{\Omega}_m(t_i) \times \vec{\Omega}_m(t_{i+1})}{|\vec{\Omega}_m(t_i)| \times |\vec{\Omega}_m(t_{i+1})|} \geq \cos(\phi). \quad (12)$$

In the course of planning the flight trajectory of multiple UAVs, in addition to meeting the maneuverability constraints of UAVs, it is also necessary to consider the space constraint of each UAV. To ensure the flight safety, i.e., there is no collision between each other, the space constraint condition of UAV m and UAV m' performing missions in i th time slot is

$$\sqrt{(x_m(t_i) - x_{m'}(t_i))^2 + (y_m(t_i) - y_{m'}(t_i))^2} \geq D, \quad (13)$$

where D is the safe flight distance between every two UAVs.

C. Mobility Model of UGVs

According to [44], considering the mobile traffic model, the average velocity of UGVs is related to the traffic density ρ , which can be defined as

$$\bar{v} = \max \left\{ v_{\min}, v_{\max} \left(1 - \frac{\rho}{\rho_{\max}} \right) \right\}, \quad (14)$$

where v_{\min} and v_{\max} are the minimum and maximum velocity of UGVs, respectively. ρ_{\max} is the maximum traffic density. We consider that δ is sufficiently small and the velocity of UGVs in each time slot is equal to the average velocity, i.e., $v = \bar{v}$. The number of UGVs entering the communication range of UAV m in each time slot can be denoted as

$$o_m = \rho v \cdot \delta = \rho \bar{v} \cdot \delta. \quad (15)$$

The percentage of UGVs leaving the communication range of UAV m is denoted as g_m . Based on [45], the total number of UGVs in the communication range of UAV m in i th time slot can be obtained by

$$F_m(t_i) = \begin{cases} (o_m(t_i) + F_m(t_{i-1})) (1 - g_m(t_i)), & i > 1; \\ o_m(t_1) (1 - g_m(t_1)), & i = 1. \end{cases} \quad (16)$$

D. Communication Model

When $0 \leq t \leq T$, the instantaneous coordinate of UGV n is denoted as $q_n(t_i) = [x_n(t_i), y_n(t_i), 0]$, which can be obtained by GPS. For simplicity, the coordinate of base station j is denoted as $q_j = [x_j, y_j, 0]$. The wireless channel from UAVs to UGVs and base station j is dominated by a line-of-sight (LoS) transmission link and can be modeled by the quasi-static block fading channel model [42]. The channel remains constant within each fading block and is affected

by the distance-dependent power attenuation in such case. Therefore, within i th time slot, the channel gain between UAV m and UGV n can be formulated as

$$\begin{aligned} h_{m,n}(t_i) &= \eta_0 d_{m,n}(t_i)^{-\varphi} \\ &= \frac{\eta_0}{((x_m(t_i) - x_n(t_i))^2 + (y_m(t_i) - y_n(t_i))^2 + H^2)^{\varphi}}, \end{aligned} \quad (17)$$

where η_0 is the channel power gain associated with a reference distance. $d_{m,n}(t_i)$ denotes the distance between UAV m and UGV n in i th time slot. φ is the path loss exponent which is greater than 1. Similarly, the distance between UAV m and base station j in i th time slot is $d_{m,j}(t_i)$, and the channel gain between them can be expressed as

$$\begin{aligned} h_{m,j}(t_i) &= \eta_0 d_{m,j}(t_i)^{-\varphi} \\ &= \frac{\eta_0}{((x_m(t_i) - x_j)^2 + (y_m(t_i) - y_j)^2 + H^2)^{\varphi}}. \end{aligned} \quad (18)$$

During each time slot, UAVs sequentially offload their computation tasks to UGVs or base stations [46]. The instantaneous transmit power (measured in mW) from UAV m to UGV n and from UAV m to base station j in i th time slot is denoted as $b_{m,n}(t_i)$ and $b_{m,j}(t_i)$, which is constrained by

$$0 < b_{m,n}(t_i) \leq \bar{B}_{m,n}, \quad (19)$$

$$0 < b_{m,j}(t_i) \leq \bar{B}_{m,j}, \quad (20)$$

where $\bar{B}_{m,n}$ and $\bar{B}_{m,j}$ are the power threshold, i.e., the maximum transmit power in data transmission, from UAV m to UGV n and from UAV m to base station j , respectively.

Based on the quasi-static block fading channel model, the power threshold between UAV m and UGV n in i th time slot can be approximated as

$$\bar{B}_{m,n}(t_i) = B_0 - 10\varphi \lg \frac{d_{m,n}(t_i)}{d_0}, \quad (21)$$

where B_0 is the reference power received by UAV m at the reference distance d_0 . Similarly, the power threshold between UAV m and base station j in i th time slot can be approximated as

$$\bar{B}_{m,j}(t_i) = B_0 - 10\varphi \lg \frac{d_{m,j}(t_i)}{d_0}. \quad (22)$$

According to the Shannon bound, the available data transmission rate (measured in bits/second) between UAV m and UGV n in i th time slot can be denoted as

$$\gamma_{m,n} = \omega_{m,n} \log_2 \left(1 + \frac{b_{m,n}(t_i) h_{m,n}(t_i)}{\sum_{l \in \mathcal{M}, l \neq m} b_{l,n}(t_i) h_{l,n}(t_i) + \sigma^2} \right), \quad (23)$$

where $\omega_{m,n}$ denotes the allocated spectrum bandwidth between UAV m and UGV n . $b_{m,n}(t_i)$ is the instantaneous transmit power from UAV m to UGV n . σ^2 is the power of the additive white Gaussian noise. $\sum_{l \in \mathcal{M}, l \neq m} b_{l,n}(t_i) h_{l,n}(t_i)$ indicates the sum of the interference from other UAVs to UGV n in i th time slot.

Similarly, the available data transmission rate between UAV m and base station j in i th time slot can be denoted as

$$\gamma_{m,j} = \omega_{m,j} \log_2 \left(1 + \frac{b_{m,j}(t_i) h_{m,j}(t_i)}{\sum_{l \in \mathcal{M}, l \neq m} b_{l,j}(t_i) h_{l,j}(t_i) + \sigma^2} \right), \quad (24)$$

where $\omega_{m,j}$ denotes the allocated spectrum bandwidth between UAV m and base station j . $\sum_{l \in \mathcal{M}, l \neq m} b_{l,j}(t_i) h_{l,j}(t_i)$ is the sum of the interference from other UAVs to base station j in i th time slot.

IV. UTILITY ANALYSIS

In this section, we introduce the utilities of UAVs, UGVs and base stations, respectively.

A. Utility of UAVs

The computation task generated by UAV m can be described as a tuple $\{D_m, C_m, T_m\}$, where D_m represents the data size of the computation task, C_m is the required computation resource for the computation task, and T_m is the task completion delay constraint.

1) Transmission delay function. The transmission delay is a critical quality of service (QoS) criteria for UAVs. When UAV m offloads the computation task to UGV n , the transmission time $T_{m,n}^t$ can be denoted as

$$T_{m,n}^t = \frac{D_m}{\gamma_{m,n}}. \quad (25)$$

In our work, UGVs drive on the two-way dual carriageway, and UAVs are located directly above the road. In addition, the communication range of UAVs is a circle. Due to the mobility of UGVs, UGV n may leave the communication range of UAV m during data transmission. The dwell time of UGV n in the communication range of UAV m is defined as $T_{m,n}^d$. We have

$$T_{m,n}^d = \frac{d_{m,n}}{\bar{v}}, \quad (26)$$

where $d_{m,n}$ is the distance between the location of UGV n receiving the data at the beginning and the end point of the diameter of the circle in the forward direction of UGV n , which can be obtained by GPS. Therefore, the data transmission will occur if and only if $T_{m,n}^d \geq T_{m,n}^t$, where $T_{m,n}^d$ is the delay constraint of data transmission.

2) Computation delay function. The benefits of UAVs are also associated with the computation delay. Under the premise of offloading the computation task from UAV m to UGV n , once UGV n completely receives the task data, it begins to compute the task, and the computation time is denoted as $T_{m,n}^c$ (defined in Eq. (35)). Therefore, the total delay $T_{m,n}$ can be obtained as

$$\begin{aligned} T_{m,n} &= T_{m,n}^t + T_{m,n}^c \\ &= D_m \cdot \left[\omega_{m,n} \log_2 \left(1 + \frac{b_{m,n} h_{m,n}}{\sum_{l \in \mathcal{M}, l \neq m} b_{l,n} h_{l,n} + \sigma^2} \right) \right]^{-1} \\ &\quad + T_{m,n}^c. \end{aligned} \quad (27)$$

3) **Satisfaction function.** In the practical post-disaster rescue scenario, rescue tasks of UAVs are urgent to be completed, and the QoE of UAVs is affected by the completion time. If the $T_{m,n}$ is shorter, UAV m can gain higher satisfaction with the completed task. When $T_{m,n}$ is longer, the satisfaction of UAV m decreases rapidly. As a whole, the satisfaction function of UAV m decreases with $T_{m,n}$ and is a concave function. Based on [45], the satisfaction function when UAV m offloads the computation task to UGV n can be denoted as

$$S_{m,n}(T_{m,n}) = \alpha \log(1 + \beta - T_{m,n}), \quad (28)$$

where α is the satisfaction coefficient. The larger α , the higher the satisfaction. β is a positive constant making the satisfaction function non-negative.

4) **Payment function.** The price of unit computation resource charged by UGV n is defined as p_n . Once the computation task is completed, the payment from UAV m to UGV n for the shared C_m amount of computation resource can be denoted as

$$P_{m,n}(p_n) = C_m \cdot p_n. \quad (29)$$

5) **Flight loss function.** The flight loss, i.e., energy cost in flight, of UAVs is closely related to their maneuverability [47], [48]. When the flight altitude H is higher and the flight distance $\sum_{i=1}^I \|q_m(t_i) - q_m(t_{i-1})\|$ of UAV m is longer, the flight loss of UAV m will be greater, as more electric energy is required to overcome the gravitational potential energy and generate kinetic energy to maintain a fixed flight altitude. In other words, the flight loss of UAV m is positively related to its flight altitude and flight distance. The flight loss function of UAV m whose flight trajectory is composed of I segments of different lengths passing through each time slot can be expressed as

$$\ell_m = \iota_1 \sum_{i=1}^I \|q_m(t_i) - q_m(t_{i-1})\| + \iota_2 H, \quad (30)$$

where ι_1 and ι_2 are the unit energy cost related to flight distance and flight altitude, respectively.

6) **Threat cost function.** During the whole flight process of UAV m , there are inevitable threats such as obstacles, complex terrain and severe weather [49]. To approximate the threat cost of UAV m , we denote the sources of threat in flight as a set $\mathbb{Z} = \{1, \dots, z, \dots, Z\}$. Here, the flight distance within i th time slot is divided into u segments of equal distance path, and the linear distance from the threat source $z \in \mathbb{Z}$ to the midpoint of each segment is $\{d_{z_1}, d_{z_2}, \dots, d_{z_u}\}$. When UAV m flies from the start location q_m^{start} to the end location q_m^{end} , the flight trajectory of UAV m should be collision-free. In other words, UAV m should sense the specific location of the threat source z within each segment to maintain a certain distance from it. Therefore, the threat cost of UAV m within i th time slot can be measured by the energy consumption in sensing the location of threat source z , which can be defined as

$$\tau_m^z(t_i) = \varpi \sum_{k=1}^u d_{z_k}, \quad (31)$$

where ϖ is the unit energy consumption of UAV m for sensing the threat source $z \in \mathbb{Z}$.

When UAV m is affected by a set of threat sources in \mathbb{Z} in flight, the total threat cost of UAV m affected by the external

environment can be expressed as

$$\Gamma_m = \sum_{i=1}^I \sum_{z \in \mathbb{Z}} \tau_m^z(t_i). \quad (32)$$

As UAV m is jointly affected by its own maneuverability constraints and external environmental threats during the flight, the comprehensive energy cost of UAV m is

$$E_m = \lambda \ell_m + (1 - \lambda) \Gamma_m, \quad (33)$$

where λ is a coefficient related to the comprehensive energy loss.

When UAV m offloads the computation task to UGV n , the utility of UAV m can be defined as the difference between the revenue of the time to complete the computation task and the cost including the payment and energy loss. We have

$$\begin{aligned} U_m(T_{m,n}, p_n) &= S_{m,n}(T_{m,n}) - P_{m,n}(p_n) - E_m \\ &= \alpha \log(1 + \beta - T_{m,n}) - p_n C_m - (\lambda \ell_m + (1 - \lambda) \Gamma_m). \end{aligned} \quad (34)$$

B. Utility of UGVs

In this work, one UGV can only share its idle computation resources with one UAV to obtain revenue, and the strategy of the UGV is to adjust the price to obtain the maximum revenue. Similar to UAV m , when UAV m offloads the computation task to UGV n , the utility of UGV n can be defined as the difference between its obtained revenue and the cost.

1) **Computation delay function.** In the case that UAV m decides to offload the computation task to UGV n , once the computation task of UAV m is completely offloaded, UGV n will immediately execute the computation task, and the computation time $T_{m,n}^c$ of UGV n can be denoted as

$$T_{m,n}^c = \frac{C_m \Upsilon_n}{f_n}, \quad (35)$$

where f_n is the idle computation resources shared by UGV n . Υ_n represents the number of CPU cycles of UGV n that is required to process per bit of data.

2) **Computation cost function.** After UGV n receives the data transmitted from UAV m , UGV n utilizes its idle computation resources to perform task processing. The computation cost of UGV n incurred during this period is measured by the energy consumption in task computing, which is defined as

$$\Lambda_n = f_n \cdot \xi_n, \quad (36)$$

where ξ_n denotes the unit cost of UGV n in task computation.

3) **Transmission cost function.** The transmission cost of UGV n is positively correlated with the transmission delay. UGV n needs to receive the computation task offloaded by UAV m and return the computation results after the implementation of the computation task. Since the processed data is relatively small, the time for returning the computation results is omitted for simplicity. Under such a premise, the transmission cost of UGV n can be expressed as

$$\Psi_n = T_{m,n}^t \cdot \chi_{m,n} = \frac{D_m}{\gamma_n} \cdot \chi_{m,n}, \quad (37)$$

where $\chi_{m,n}$ is the cost of task transmission from UAV m to UGV n per time slot.

Therefore, the utility of UGV n can be denoted as

$$\begin{aligned} U_n(p_n) &= P_{m,n}(p_n) - T_{m,n}^c - \Lambda_n - \Psi_n \\ &= C_m \cdot p_n - \frac{C_m \Upsilon_n}{f_n} - f_n \cdot \xi_n - \frac{D_m}{\gamma_{m,n}} \cdot \chi_{m,n}. \end{aligned} \quad (38)$$

C. Utility of Base Stations

Although UAVs prioritize the offloading of computation tasks to UGVs, there may still exist a UAV m' belonging to \mathcal{M} that fails to offload its computation task to any UGV n belonging to \mathcal{N} . In this case, the computation task generated by UAV m' will be performed by the base station j .

1) **Computation delay function.** When base station j completely receives the computation task offloaded from UAV m' , the computation task is immediately executed. The computation time $T_{m',j}^c$ of base station j is

$$T_{m',j}^c = \frac{C_{m'} \Upsilon_j}{\varrho \Delta_j}, \quad (39)$$

where Υ_j is the number of CPU cycles of base station j that is required per bit of data size. ϱ represents the percentage of computation resource allocated by base station j and the total computation resource of base station j is defined as Δ_j .

2) **Computation cost function.** When base station j takes advantage of local computation resources to carry out the computation task of UAV m' , a certain amount of computation overhead will be generated, and the computation cost of base station j can be denoted as

$$\Lambda_j = \varrho \cdot \Delta_j \cdot \xi_j, \quad (40)$$

where ξ_j denotes the cost per unit of local computation resource of base station j to execute the computation task.

3) **Transmission cost function.** The longer the transmission delay, the higher the transmission cost of base station j . Similarly, the time of the backhaul process after base station j has performed the computation task is not considered for simplicity. As a result, the transmission cost of base station j can be regarded as the overhead of receiving the computation task offloaded from UAV m' , which can be expressed as

$$\Psi_j = \frac{D_{m'}}{\gamma_{m',j}} \cdot \chi_{m',j}, \quad (41)$$

where $\chi_{m',j}$ is the cost of task transmission from UAV m' to base station j per time slot.

To summarize, the utility of base station j is expressed as

$$\begin{aligned} U_j(p_j) &= C_{m'} \cdot p_j - T_{m',j}^c - \Lambda_j - \Psi_j \\ &= C_{m'} \cdot p_j - \frac{C_{m'} \Upsilon_j}{\varrho \Delta_j} - \varrho \cdot \Delta_j \cdot \xi_j - \frac{D_{m'}}{\gamma_{m',j}} \cdot \chi_{m',j}, \end{aligned} \quad (42)$$

where p_j is the price per unit computation resource charged by the base station j .

Remark 1: For each UAV $m \in \mathcal{M}$, it matches UGVs with higher priority than the base station j . However, when a UAV (belong to \mathcal{M}) fails to match, its computation task will be performed by the base station j alternatively.

V. MATCHING-BASED COMPUTATION TASK OFFLOADING

In this section, the computation task offloading problem of UAVs is first transformed into a two-sided matching problem [50], [51]. Next, we structure the preference lists of both sides based on their preferences for profit. Finally, the stable matching algorithm is proposed to solve the problem iteratively.

A. Problem Formulation

Due to the combinatorial nature of the computation task offloading problem, we transform the problem into a two-sided matching problem based on the following definition.

Definition 1 (Computation Task Offloading Decisions):

The computation task offloading decisions of UAVs and UGVs can be described as a $M \times N$ matrix $\mathbf{X}_{M \times N}$ and a $N \times M$ matrix $\mathbf{Y}_{N \times M}$, respectively [52]. All elements of $\mathbf{X}_{M \times N}$, i.e., the (m, n) -th element $x_{m,n}$, are binary values which are 0 or 1. When only UGV n can bring benefits to UAV m , or UGV n brings more benefits to UAV m than other UGVs, UAV m prefers UGV n to perform its computing task, i.e., $x_{m,n} = 1$. Otherwise, $x_{m,n} = 0$. We have

$$\begin{cases} x_{m,n} = 1, & \text{if } m \text{ prefers } n; \\ x_{m,n} = 0, & \text{otherwise.} \end{cases} \quad (43)$$

Similarly, all elements of $\mathbf{Y}_{N \times M}$, i.e., the (n, m) -th element $y_{n,m}$, are binary values which are 0 or 1. $y_{n,m} = 1$ denotes that UGV n prefers UAV m to obtain higher benefit, and otherwise, $y_{n,m} = 0$. We can obtain

$$\begin{cases} y_{n,m} = 1, & \text{if } n \text{ prefers } m; \\ y_{n,m} = 0, & \text{otherwise.} \end{cases} \quad (44)$$

Remark 2: As participating UAVs and UGVs are selfish and they seek to maximize their benefits, there is a risk that UAVs and UGVs may have completely conflicting decisions, e.g., $x_{m,n} = 1$ and $y_{n,m} = 0$. Thereby, only if a mutual agreement has been reached on both parties, i.e., $x_{m,n} = y_{n,m} = 1$, the one-to-one matching between UAV m and UGV n is successful, and UGV n will only provide the computation offloading service for the successfully matched UAV m .

Combining with the utility of UAV m analyzed in Section IV-A, the utility of UAV m can be rewritten as

$$G_m(\vec{x}_m) = \sum_{n=1}^N x_{m,n} \cdot y_{n,m} \cdot U_m(T_{m,n}, p_n), \quad (45)$$

where \vec{x}_m is the m th row element of the matrix $\mathbf{X}_{M \times N}$, which can be described as $\vec{x}_m = \{x_{m,1}, \dots, x_{m,n}, \dots, x_{m,N}\}$. Similarly, the utility of UGV n can be rewritten as

$$G_n(\vec{y}_n) = \sum_{m=1}^M x_{m,n} \cdot y_{n,m} \cdot U_n(p_n), \quad (46)$$

where \vec{y}_n is the n th row element of the matrix $\mathbf{Y}_{N \times M}$, which can be denoted as $\vec{y}_n = \{y_{n,1}, \dots, y_{n,m}, \dots, y_{n,M}\}$.

The purpose of UAVs is to maximize their utilities defined in Eq. (45) while considering the total offloading delay. Hence,

the optimization problem of UAVs can be formulated as

$$\begin{aligned} P1 : \max_{\mathbf{X}_{M \times N}} & \sum_{m=1}^M G_m(\vec{x}_m) \\ & = \max \sum_{m=1}^M \sum_{n=1}^N x_{m,n} \cdot y_{n,m} \cdot U_m(T_{m,n}, p_n) \\ \text{s.t. } & C_1 : T_{m,n} \leq T_m, \quad \forall m \in \mathcal{M}, \forall n \in \mathcal{N} \\ & C_2 : \sum_{m=1}^M x_{m,n} \leq 1, \quad \forall n \in \mathcal{N} \\ & C_3 : \sum_{n=1}^N x_{m,n} \leq 1, \quad \forall m \in \mathcal{M} \end{aligned} \quad (47)$$

where C_1 describes that the total delay $T_{m,n}$ should be less than the delay constraint T_m . In addition, C_2 and C_3 guarantee that each UAV can only be matched with at most one UGV, and each UGV can only perform at most one computation task for a UAV, respectively.

For UGVs, their purposes are to maximize their utilities defined in Eq. (46) while considering the computation cost. The optimization problem of UGVs can be formulated as

$$\begin{aligned} P2 : \max_{\mathbf{Y}_{N \times M}} & \sum_{n=1}^N G_n(\vec{y}_n) \\ & = \max \sum_{n=1}^N \sum_{m=1}^M x_{m,n} \cdot y_{n,m} \cdot U_n(p_n) \\ \text{s.t. } & C_4 : T_{m,n}^t \leq T_{m,n}^d, \quad \forall m \in \mathcal{M}, \forall n \in \mathcal{N} \\ & C_5 : T_{m,n} \leq T_m, \quad \forall m \in \mathcal{M}, \forall n \in \mathcal{N} \\ & C_6 : \sum_{m=1}^M y_{n,m} \leq 1, \quad \forall n \in \mathcal{N} \\ & C_7 : \sum_{n=1}^N y_{n,m} \leq 1, \quad \forall m \in \mathcal{M} \end{aligned} \quad (48)$$

where C_4 and C_5 denote the delay constraints of the problem. Based on C_6 and C_7 , there will only be a one-to-one correspondence among UAVs and UGVs.

Definition 2 (Matching): Based on the structure of the problem, the tuple $(\mathcal{M}, \mathcal{N}, \mathcal{L})$ is used to formulate the combinatorial matching problem, in which \mathcal{M} and \mathcal{N} represent the finite sets of UAVs and UGVs, respectively. \mathcal{L} is the set of matching preferences common to both UAVs and UGVs. Particularly, $\mathcal{L}(m)$ indicates the preference of UAV m on the set of $\mathcal{N} \cup \{m\}$, $\forall m \in \mathcal{M}$, and $\mathcal{L}(n)$ represents the preference of UGV n on the set of $\mathcal{M} \cup \{n\}$, $\forall n \in \mathcal{N}$. In addition, μ is a one-to-one match between UAVs and UGVs based on $(\mathcal{M}, \mathcal{N}, \mathcal{L})$. Here, $\mu(m) = n$ represents the successful match between UAV m and UGV n , which can be established if and only if $x_{m,n} = y_{n,m} = 1$. Besides, $\mu(m) = m$ indicates that UAV m does not match with any UGV and its computation task will be performed by base station j .

B. Preference List Structure

The maximum achievable benefits of UAVs and UGVs are quantified as the preferences between UAVs and UGVs. If a UAV successfully matches with a UGV which is prioritized in the preference list of the UAV, the UAV will gain more benefits, and it is the same for UGVs. According to the preference for the profit, each side (i.e., UAVs or UGVs) of the matching problem has to structure a list of its preference

by ranking the other side in descending order. In the case of $\mu(m) = n$, the benefit of UAV m can be denoted as

$$\begin{aligned} G_m^*|_{\mu(m)=n} & = U_m(T_{m,n}, p_n) \\ & = S_{m,n}(T_{m,n}) - P_{m,n}(p_n) - E_m. \end{aligned} \quad (49)$$

The benefit obtained by UGV n can be described as

$$\begin{aligned} G_n^*|_{\mu(n)=m} & = U_n(p_n) \\ & = C_m \cdot p_n - f_n \cdot \xi_n - D_{m,n}. \end{aligned} \quad (50)$$

Based on [53], a preference relation (i.e., \succ) is introduced to represent the comparison of the preference. For example, $n \succ_m n'$ and $n \succeq_m n'$ describe that UAV m matches UGV n with higher preference than UGV n' and the preference of UAV m for UGV n is at least as much as UGV n' , respectively. We have the following formulas for case $n \succ_m n'$ and case $n \succeq_m n'$, respectively,

$$n \succ_m n' \Leftrightarrow G_m^*|_{\mu(m)=n} > G_m^*|_{\mu(m)=n'}, \quad (51)$$

$$n \succeq_m n' \Leftrightarrow G_m^*|_{\mu(m)=n} \geq G_m^*|_{\mu(m)=n'}. \quad (52)$$

Similarly, $m \succ_n m'$ and $m \succeq_n m'$ represent that UGV n matches UAV m with higher preference than UAV m' and the preference of UGV n for UAV m is no less than that for UAV m' , respectively. We also have the following formulas for case $m \succ_n m'$ and case $m \succeq_n m'$, respectively,

$$m \succ_n m' \Leftrightarrow G_n^*|_{\mu(n)=m} > G_n^*|_{\mu(n)=m'}, \quad (53)$$

$$m \succeq_n m' \Leftrightarrow G_n^*|_{\mu(n)=m} \geq G_n^*|_{\mu(n)=m'}. \quad (54)$$

To obtain the specific preference lists of UAVs and UGVs, each participant on one side will match all members on the other side to calculate the preference for each combination. The preference lists $\mathcal{L}(m)$ and $\mathcal{L}(n)$ are sorted in descending order according to the results of $G_m^*|_{\mu(m)=n}$ and $G_n^*|_{\mu(n)=m}$, respectively. Therefore, the preference lists for UAVs and UGVs can be structured as

$$\mathcal{L}_M = \{\mathcal{L}(1), \dots, \mathcal{L}(m), \dots, \mathcal{L}(M)\}, \quad (55)$$

$$\mathcal{L}_N = \{\mathcal{L}(1), \dots, \mathcal{L}(n), \dots, \mathcal{L}(N)\}. \quad (56)$$

C. Profit-Based Computation Task Offloading

After each side has structured its own preference list, the matching problem can be solved by the proposed stable matching algorithm based on the Gale-Shapley algorithm [54], which is shown in Algorithm 1.

Definition 3 (Apply Rule): Each UAV m , $m \in \mathcal{M}$, sends a matching request to the UGV n which ranks first in its preference list.

Definition 4 (Select Rule): Each UGV n , $n \in \mathcal{N}$, which has received only one matching request, will select the applicant as the matching candidate. If multiple matching requests have been received, the applicant with the highest ranking in the preference list of UGV n will be selected as the matching candidate, and the remaining applicants will be rejected.

Definition 5 (Deferred-Acceptance Rule): If UAV m is already the matching candidate of UGV n and UGV n receives new matching requests in the next round, UGV n will compare

Algorithm 1 The Stable Matching Algorithm

```

1: Input:  $\mathcal{M}, \mathcal{N}$ 
2: Output:  $\mu$ 
3: Initialize  $\Phi = \mathcal{M}$ ,  $N_n = 0$  and  $G_n = 0$ ;
4: Each  $m \in \mathcal{M}$  structures the preference list  $\mathcal{L}(m)$  based
   on Eq. (49),  $\mathcal{L}(m) \neq \emptyset$ ;
5: Each  $n \in \mathcal{N}$  structures the preference list  $\mathcal{L}(n)$  based on
   Eq. (50),  $\mathcal{L}(n) \neq \emptyset$ ;
6: while  $\Phi \neq \emptyset$  and  $\mathcal{L}(m) \neq \emptyset$  do
7:   for  $m \in \Phi$  do
8:     Each  $m$  sends a matching request to the  $n$  which ranks
        first in the updated preference list  $\mathcal{L}(m)$ ;
9:   end for
10:  for  $n \in \mathcal{N}$  do
11:    if  $\mu(n) = m$  then
12:      if  $N_n \geq 1$  and  $G_{n \max}^* = G_n^*|_{\mu(n)=m'}$  then
13:         $\mu(n) = m'$ ;
14:      else
15:         $\mu(n) = m$ ;
16:      end if
17:    else
18:      if  $N_n \geq 1$  and  $G_{n \ max}^* = G_n^*|_{\mu(n)=m}$  then
19:         $\mu(n) = m$ ;
20:      else
21:         $\mu(n) = n$ ;
22:      end if
23:    end if
24:    Update: Remove  $m$  from  $\Phi$  if it has become a candidate;
       Add the rejected  $m$  which used to be a candidate to  $\Phi$ ;
       Remove  $n$  which has sent out the rejection to  $m$ 
       from  $\mathcal{L}(m)$ ;
25:  end for
26: end while

```

the benefits generated by all applicants, including the previous candidate. If there is an alternative applicant UAV m' which satisfies $m' \succeq_m m$, UAV m' will be selected as the new matching candidate and UAV m will be rejected.

Therefore, the detailed process of using the iterative approach to implement the entire matching process in each time slot is as follows.

Phase 1 (Matching Preference Initialization):

- Structure the preference lists $\mathcal{L}(m)$ and $\mathcal{L}(n)$ for UAV m , $\forall m \in \mathcal{M}$, and UGV n , $\forall n \in \mathcal{N}$.
- Initialize μ and G_n . Besides, Φ and N_n are defined as the set of unsuccessfully matched UAVs and the number of received matching requests of UGV n , respectively, i.e., $\Phi = \mathcal{M}$, $N_n = 0$ at the beginning.

Phase 2 (Iterative Matching): Repeat the following process in an iterative manner.

- Perform the apply rule for UAVs.
 - Each UAV m , $m \in \Phi$ sends a matching request to the UGV n which ranks first in its preference list $\mathcal{L}(m)$.
- Perform the select rule for UGVs.
 - Each UGV n , $n \in \mathcal{N}$, which has received matching requests, regardless of the quantity, selects the applicant

with the highest ranking in its preference list $\mathcal{L}(n)$ as the matching candidate and rejects the remaining.

- Perform the deferred-acceptance rule for UGVs.
 - If there is a better matching candidate, UGV n , $n \in \mathcal{N}$ will reject the applicant. Otherwise, the applicant will be selected as the new candidate.
- The UAV which has become a candidate will be removed from Φ and the candidate which is rejected will be added to Φ to update Φ . Meanwhile, the UGV sending out the rejection will be removed from the preference list of the rejected UAV.

This iteration continues until UAV m , $\forall m \in \mathcal{M}$ has been successfully matched with UGV n , $\forall n \in \mathcal{N}$, or rejected by all UGVs.

Phase 3 (Computation Task Offloading): After obtaining the matching results in **Phase 2**, each UGV n , $n \in \mathcal{N}$ will perform the computation task sent by the matched UAV. As for UAVs that fail to match, their tasks will be performed by the base station j .

Remark 3: Any UAV m , $m \in \mathcal{M}$, or UGV n , $n \in \mathcal{N}$, that does not satisfy the constraints will be removed from the preference lists regardless of its position in the lists.

Remark 4: For any UAV m , $m \in \mathcal{M}$ which has been matched with UGV n , $n \in \mathcal{N}$, if there exists a UGV n' , $n' \in \mathcal{N}$ which satisfies $n' \succeq_m n$, UGV n' should have rejected UAV m before. In other words, there should exist a UAV m' , $m' \in \mathcal{M}$ satisfying $m' \succeq_{n'} m$. It means that UGV n' selects UAV m' as the matching candidate and even if UAV m applies to match with UGV n' again, it will still be rejected. Therefore, UAV m can only accept the current matching result without remorse and the whole system is stable.

VI. PERFORMANCE EVALUATION

A. Simulation Setup

In the simulation scenario, we consider a signal cell consisting of multiple UAVs, multiple UGVs and a base station [20]. The finite time horizon T is divided into $I = 50$ time slots, each with the equal length $\delta = 0.05s$, and the initial time is set to $t_0 = 0$. The data size D_m and the required computation resource C_m of UAV's computation tasks range from 1 to 3Mb and from 2 to 3 CPU cycles/Mb, respectively. In addition, the value of shared idle computation resources f_n is between 1 and 4 MC/s ($MC = 10^6$ cycles) and the price of unit computation resource p_n is between 1.5 and 4. The x-axis and y-axis distances between UAVs and UGVs are both randomly distributed in $[0, 5]m$, and UAVs fly at an altitude $H = 5m$ [43]. Based on [42], the channel power gain η_0 is 9.7×10^{-4} at the reference distance $d_0 = 3m$. Other parameters in the simulation are summarized in Table II [55], [56].

Based on the above conditions, the proposed stable matching algorithm can better optimize the utility of UAVs by comparing with the random and fixed matching algorithms [54], respectively. The random matching algorithm randomly matches between one member of UAVs and the other of UGVs. The fixed matching algorithm matches members of UAVs with members of UGVs in a certain order. In this paper, they are matched once there is a UAV

TABLE II
SIMULATION PARAMETERS

Parameters	Values	Parameters	Values
M	[100, 500]	N	[100, 500]
D_m	[1, 3] Mb	C_m	[2, 3] cycles/Mb
f_n	[1, 4] MC/s	p_n	[1.5, 4]
H	5m	d_0	3m
η_0	9.7×10^{-4}	t_0	0
φ	1	σ^2	10
α	3.5	β	1000
I	50	u	5
ξ_n	0.5	E_m	1 J
$D_{m,n}$	1	p_m	40 dBm
$\omega_{m,n}$	50 MHz	δ	0.05 second
v_{\min}, v_{\max}	{50, 110} Km/h	ρ, ρ_{\max}	{50, 110} Veh/Km

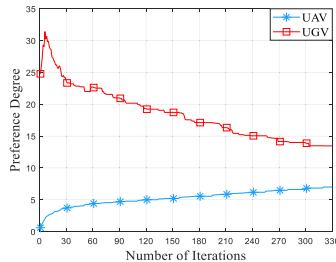


Fig. 3. Preference degree of UAVs and UGVs vs. number of iterations.

which generates the task and a UGV with idle computation resource.

B. Performance Comparison

In this subsection, we first evaluate the preference degrees of UAVs and UGVs for matched objects when the number of iterations increases. The preference degree of a UAV (or UGV) is defined as the sum of its ranking in the preference lists of all UGVs (or UAVs). Fig. 3 shows the preference degrees for both UAVs and UGVs with different number of iterations, where the smaller the y-axis value, the higher the ranking in the preference lists, and the higher chance to be matched. We can see that the preference degree of UAVs decreases with the increase of the number of iterations, while the preference degree of UGVs increases. Both eventually show a convergence trend. Moreover, although the preference degree of UGVs increases, it is still smaller than that of UAVs. The reasons are as follows: at the beginning of the iteration, UAVs prefer to match UGVs ranking first in their preference lists. In other words, UAVs prefer UGVs which minimize their cost, while the profits of the corresponding UGVs are minimum. At this time, UAVs have the highest preference degree for the matched UGVs, while the matched UGVs have the lowest preference degree. However, according to the rules defined in Section V-C, as the number of iterations increases, UGVs also select UAVs that make themselves more profitable. Therefore, with the increase of the number of iterations, the preference degrees of UAVs and UGVs for matched objects gradually decreases and increases, respectively, and both eventually converge.

Next, we evaluate the average preference degree of UAVs for matched UGVs when the number of UGVs increases.

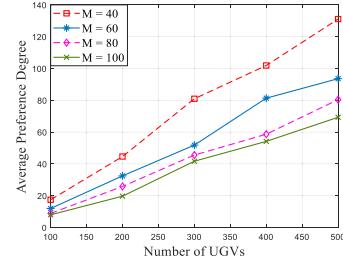


Fig. 4. Average preference degree of UAVs vs. number of UGVs.

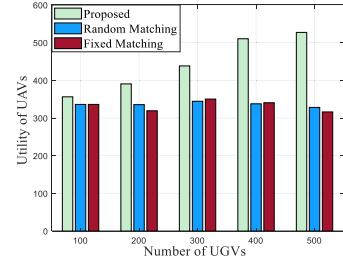


Fig. 5. Utility of UAVs vs. number of UGVs.

In Fig. 4, it can be seen that the average preference degree of UAVs in the proposed algorithm is positively related to the number of UGVs. Moreover, the average preference degree of UAVs decreases when the number of UAVs increases from 40 to 100. The reason is that when the number of UAVs is less than that of UGVs, with more UGVs, each UAV is more likely to be matched with the UGV at the top of its preference list. As a result, the corresponding average preference degree of UAVs is higher. Similarly, when the number of UGVs remains constant, as the number of UAVs increases, the possibility of UAVs matching their preferred UGVs continues to decrease, and the corresponding average preference degree of UAVs becomes lower. Therefore, with the increase of the number of UGVs, the average preference degree of UAVs for matched UGVs gradually increases, and the average preference degree of UAVs is inversely correlated with the number of UAVs.

Then, we evaluate the utilities of UAVs and UGVs of the proposed algorithm by comparing them with the random and fixed matching algorithms. The UAV's utility variation with different number of UGVs is shown in Fig. 5. It can be seen that the utility of UAVs in random and fixed matching algorithms remain in a stable state with the increase of the number of UGVs. Moreover, the utility of UAVs in the proposed algorithm is positively related to the number of UGVs and outperforms the other two algorithms. The change of UGVs' utility with different number of UGVs is shown in Fig. 6. We can see that the utility of UGVs in the proposed algorithm is inversely related to the number of UGVs and lower than the other two algorithms. Besides, the utilities of UGVs in the other two algorithms also remain in a stable state. The reason for Figs. 5 and 6 is that the probability of UAVs matching their preferred UGVs increases with the increase of the number of UGVs in the proposed algorithm. In other words, UAVs are more likely to match the UGVs which make them cost less, and the corresponding matched

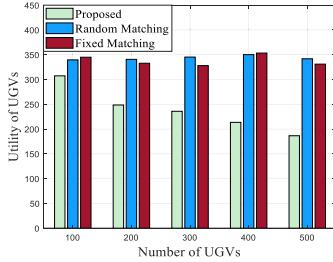


Fig. 6. Utility of UGVs vs. number of UGVs.

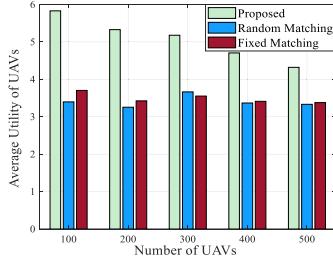


Fig. 7. Average utility of UAVs vs. number of UAVs.

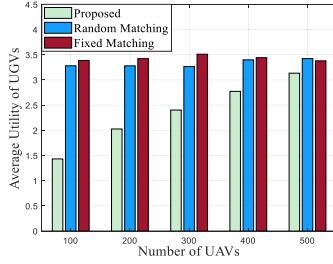


Fig. 8. Average utility of UGVs vs. number of UAVs.

UGVs gain less. Therefore, the utilities of UAVs and UGVs are positively and inversely related to the number of UGVs, respectively. For the other two algorithms, since the matching order among UAVs and UGVs is fixed (or random) in the fixed matching (or random matching) algorithm, each UAV is less likely to match its preferred UGV. As such, the change of the number of UGVs only has little effect on the utilities of both sides in the fixed matching and random matching algorithms, causing stable utilities of both sides. Moreover, in the proposed algorithm, according to the apply rule, select rule, and deferred acceptance rule in the two-sided matching, the UAVs can iteratively match their more preferred UGVs and form stable matching outcomes when the matching process ends. Therefore, the utilities of UAVs and UGVs are higher and lower in the proposed algorithm, respectively, while those in the other two algorithms keep stable.

Afterwards, we evaluate the average utilities of UAVs and UGVs by comparing the proposed algorithms with two benchmarks. The performance of UAVs' average utility with different number of UAVs is depicted in Fig. 7. It can be seen that the average utility of UAVs in the proposed algorithm decreases with the increase of the number of UAVs, while the average utilities of UAVs in the random and fixed matching algorithms remain in a stable state. In Fig. 7, the average utility of UAVs in the proposed algorithm is higher than that in the other two algorithms. Fig. 8 shows the change of UGVs'

TABLE III

COMPARISON OF RUNNING TIME IN THE PROPOSED ALGORITHM, FIXED MATCHING ALGORITHM, AND RANDOM MATCHING ALGORITHM

		Proposed	Fixed matching	Random matching
Running time (millisecond)	M=100, N=400	431	272	187
	M=200, N=200	521	304	224
	M=400, N=100	803	587	313

average utility under different number of UAVs. It can be seen in Fig. 8 that the average utility of UGVs in the proposed algorithm is positively related to the number of UAVs and is lower than that in the other two algorithms. The average utilities of UGVs in the other two algorithms also remain to be stable. The reason for the above two figures is that there are more optional UGVs can be selected by UAVs, when the number of UGVs keeps increasing. As a result, the probability that UAVs match their preferred UGVs is higher. Moreover, this probability decreases with the increasing number of UAVs, as more UAVs means higher competition among them. As a result, each UAV can only select the relatively preferred UGV, and its profit will be reduced, thereby the profit of the matched UGV will increase. Besides, in the two benchmarks, the matching process among UAVs and UGVs is either fixed or randomly selected. As a consequence, the chance of every UAV to match its preferred UGV is lower than that in our proposed algorithm, resulting in a lower average utility of UAVs. Moreover, similar to the above analysis in Figs. 5 and 6, the change of the number of UAVs only has little effect on the utilities of both sides in the two benchmarks. Thereby, the average utilities of UAVs and UGVs remain relatively stable with the increase of the number of UAVs.

Table III shows the comparison of running time of the matching process between UAVs and UGVs in three algorithms. As seen in Table III, the running time in three algorithms increases with the number of UAVs and decreases with the number of UGVs. The reason is that in the two-sided matching process, more UGVs indicates a higher chance of UAVs matching their preferred UGVs. Meanwhile, more UAVs implies higher competition in matching and a lower chance to match the preferred UGV. As such, with the higher number of UAVs or lower number of UGVs, each UAV may take a longer time to match its preferred UGVs and may only match the relatively preferred UGV. Furthermore, in the two benchmarks, as the matching pairs between UAVs and UGVs are either fixed or randomly established, the execution time for the matching procedure is smaller than that in the proposed algorithm.

Finally, we evaluate the impact of data size (i.e., D_m) and required computation resource (i.e., C_m) on the average task offloading delay. Fig. 9 shows the average delay under different number of UGVs. From Fig. 9, we can see that the average delay decreases with the increase of the number of UGVs. Moreover, the larger the values of data size and computation resource, the larger the average delay. Besides, in Fig. 9, the possibility of UAVs matching their preferred

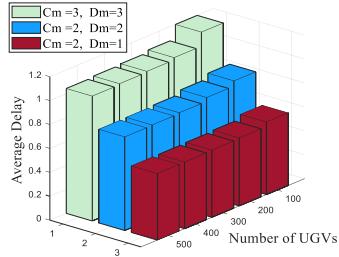


Fig. 9. Average delay vs. number of UGVs.

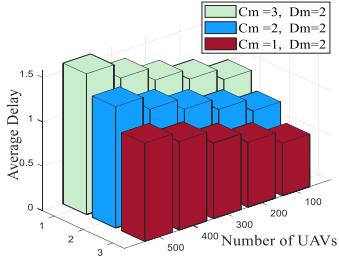


Fig. 10. Average delay vs. number of UAVs.

UGVs is positively related to the number of UGVs in the proposed algorithm. Fig. 10 shows the average task offloading delay under different number of UAVs. From Fig. 10, we can see that the average delay is positively related to the number of UAVs. Moreover, with more required computation resource, the larger the average delay. It can be explained as follows. According to Eqs. (25), (27) and (35), the average task delay is positively correlated with the values of data size and required computation resource when other conditions are constant. From Eqs. (27) and (49), we can see that the task execution latency affects UAVs' preferences for profit, and UAVs prefer to match UGVs which contributes to the minimum average delay. Besides, with the increasing number of UAVs and the decreasing number of UGVs, the probability of matching their preferred UGVs with minimum average delay is in decline, resulting in that each UAV can only select the relatively preferred UGV and its average task delay will increase. Therefore, the average delay decreases with the number of UGVs and increases with the number of UAVs.

VII. CONCLUSION

In this paper, we have proposed a cooperative UAV computation task offloading scheme for post-disaster rescue based on VFC. To address the battery limitation and long computation delays of UAVs, the VFC has been introduced by utilizing the idle computation resources of UGVs to perform tasks. Considering that both UAVs and UGVs seek their own maximum profits, the stable matching algorithm has been proposed to transform the computation task offloading problem into a two-sided matching problem. The proposed algorithm has solved the problem in an iterative manner while maximizing the utility of UAVs. Finally, numerical results have demonstrated that the proposed scheme can effectively improve the utility of UAVs and reduce the average delay. For the future work, we plan to carry out empirical studies to address the practical barriers in real-world implementation

including UAVs' flying altitude change for obstacle avoidance, unaccessible disaster terrains for UGVs, and automatic UAV recycling control.

REFERENCES

- [1] Y.-J. Zheng, Q.-Z. Chen, H.-F. Ling, and J.-Y. Xue, "Rescue wings: Mobile computing and active services support for disaster rescue," *IEEE Trans. Services Comput.*, vol. 9, no. 4, pp. 594–607, Jul. 2016.
- [2] Y. Wang, Z. Su, N. Zhang, and D. Fang, "Disaster relief wireless networks: Challenges and solutions," *IEEE Wireless Commun.*, vol. 28, no. 5, pp. 148–155, Oct. 2021, doi: [10.1109/WMC.101.2000518](https://doi.org/10.1109/WMC.101.2000518).
- [3] M. Aljehani and M. Inoue, "Performance evaluation of multi-UAV system in post-disaster application: Validated by HITL simulator," *IEEE Access*, vol. 7, pp. 64386–64400, 2019.
- [4] S. Yin, Y. Zhao, and L. Li, "UAV-assisted cooperative communications with time-sharing SWIPT," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
- [5] Z. Su, Y. Wang, Q. Xu, and N. Zhang, "LVBS: Lightweight vehicular blockchain for secure data sharing in disaster rescue," *IEEE Trans. Dependable Secure Comput.*, early access, Mar. 13, 2020, doi: [10.1109/TDSC.2020.2980255](https://doi.org/10.1109/TDSC.2020.2980255).
- [6] H. Shakhatreh, A. Khreichah, and B. Ji, "UAVs to the rescue: Prolonging the lifetime of wireless devices under disaster situations," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 4, pp. 942–954, Dec. 2019.
- [7] G. Aiello, F. Hopps, D. Santisi, and M. Venticinque, "The employment of unmanned aerial vehicles for analyzing and mitigating disaster risks in industrial sites," *IEEE Trans. Eng. Manag.*, vol. 67, no. 3, pp. 519–530, Aug. 2020.
- [8] Y. Wang, Z. Su, Q. Xu, R. Li, and T. H. Luan, "Lifesaving with rescuechain: Energy-efficient and partition-tolerant blockchain based secure information sharing for UAV-aided disaster rescue," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, May 2021, pp. 1–10.
- [9] Y. Zeng and R. Zhang, "Energy-efficient UAV communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747–3760, Jun. 2017.
- [10] T. Bai, J. Wang, Y. Ren, and L. Hanzo, "Energy-efficient computation offloading for secure UAV-edge-computing systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 6074–6087, Jun. 2019.
- [11] O. Munoz, A. Pascual-Iserte, and J. Vidal, "Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4738–4755, Oct. 2015.
- [12] M.-A. Messous, S.-M. Senouci, H. Sedjelmaci, and S. Cherkaoui, "A game theory based efficient computation offloading in an UAV network," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4964–4974, May 2019.
- [13] F. Zhou, Y. Wu, H. Sun, and Z. Chu, "UAV-enabled mobile edge computing: Offloading optimization and trajectory design," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
- [14] W. Chen, B. Liu, H. Huang, S. Guo, and Z. Zheng, "When UAV swarm meets edge-cloud computing: The QoS perspective," *IEEE Netw.*, vol. 33, no. 2, pp. 36–43, Mar./Apr. 2019.
- [15] W. Fan, Y. Liu, B. Tang, F. Wu, and Z. Wang, "Computation offloading based on cooperations of mobile edge computing-enabled base stations," *IEEE Access*, vol. 6, pp. 22622–22633, 2017.
- [16] S. Garg, A. Singh, S. Batra, N. Kumar, and L. T. Yang, "UAV-empowered edge computing environment for cyber-threat detection in smart vehicles," *IEEE Netw.*, vol. 32, no. 3, pp. 42–51, May/Jun. 2018.
- [17] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV-based IoT platform: A crowd surveillance use case," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 128–134, Feb. 2017.
- [18] B. Liu, H. Huang, S. Guo, W. Chen, and Z. Zheng, "Joint computation offloading and routing optimization for UAV-edge-cloud computing environments," in *Proc. IEEE SmartWorld, Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People Smart City Innov. (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI)*, Oct. 2018, pp. 1745–1752.
- [19] H. Guo, J. Liu, and J. Zhang, "Efficient computation offloading for multi-access edge computing in 5G HetNets," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.
- [20] Z. Zhou, P. Liu, J. Feng, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Computation resource allocation and task assignment optimization in vehicular fog computing: A contract-matching approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3113–3125, Apr. 2019.

- [21] W. Chen, Z. Su, Q. Xu, T. H. Luan, and R. Li, "VFC-based cooperative UAV computation task offloading for post-disaster rescue," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Jul. 2020, pp. 228–236.
- [22] Y. Zhou, N. Cheng, N. Lu, and X. S. Shen, "Multi-UAV-aided networks: Aerial-ground cooperative vehicular networking architecture," *IEEE Veh. Technol. Mag.*, vol. 10, no. 4, pp. 36–44, Dec. 2015.
- [23] Y. Wang, Z. Su, J. Ni, N. Zhang, and X. Shen, "Blockchain-empowered space-air-ground integrated networks: Opportunities, challenges, and solutions," *IEEE Commun. Surveys Tuts.*, early access, Dec. 1, 2021, doi: [10.1109/COMST.2021.3131711](https://doi.org/10.1109/COMST.2021.3131711).
- [24] Y. Wang, Z. Su, N. Zhang, and A. Benslimane, "Learning in the air: Secure federated learning for UAV-assisted crowdsensing," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1055–1069, Apr. 2021.
- [25] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient Internet of Things communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, Nov. 2017.
- [26] F. Cheng, G. Gui, N. Zhao, Y. Chen, J. Tang, and H. Sari, "UAV-relaying-assisted secure transmission with caching," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3140–3153, May 2019.
- [27] L. Li, T. Chang, and S. Cai, "UAV positioning and power control for two-way wireless relaying," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 1008–1024, Feb. 2020.
- [28] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 3949–3963, Jun. 2016.
- [29] T. X. Tran and D. Pompili, "Joint task offloading and resource allocation for multi-server mobile-edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 856–868, Jan. 2019.
- [30] M.-A. Messous, A. Arfaoui, A. Alioua, and S.-M. Senouci, "A sequential game approach for computation-offloading in an UAV network," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2017, pp. 1–7.
- [31] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.
- [32] A. Gao, Y. Hu, W. Liang, Y. Lin, L. Li, and X. Li, "A QoE-oriented scheduling scheme for energy-efficient computation offloading in UAV cloud system," *IEEE Access*, vol. 7, pp. 68656–68668, 2019.
- [33] J. Zhang, W. Xia, F. Yan, and L. Shen, "Joint computation offloading and resource allocation optimization in heterogeneous networks with mobile edge computing," *IEEE Access*, vol. 6, pp. 19324–19337, 2018.
- [34] N. T. Ti and L. Bao Le, "Joint resource allocation, computation offloading, and path planning for UAV based hierarchical fog-cloud mobile systems," in *Proc. IEEE 7th Int. Conf. Commun. Electron. (ICCE)*, Jul. 2018, pp. 373–378.
- [35] R. Li and H. Asaeda, "A blockchain-based data life cycle protection framework for information-centric networks," *IEEE Commun. Mag.*, vol. 57, no. 6, pp. 20–25, Jun. 2019.
- [36] S. Zhang and J. Liu, "Analysis and optimization of multiple unmanned aerial vehicle-assisted communications in post-disaster areas," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12049–12060, Dec. 2018.
- [37] B. Shang, L. Liu, J. Ma, and P. Fan, "Unmanned aerial vehicle meets vehicle-to-everything in secure communications," *IEEE Commun. Mag.*, vol. 57, no. 10, pp. 98–103, Oct. 2019.
- [38] Y. Liu, Z. Luo, Z. Liu, J. Shi, and G. Cheng, "Cooperative routing problem for ground vehicle and unmanned aerial vehicle: The application on intelligence, surveillance, and reconnaissance missions," *IEEE Access*, vol. 7, pp. 63504–63518, 2019.
- [39] J. Zhang, Z. Yu, S. Mao, S. C. G. Periaswamy, J. Patton, and X. Xia, "IADRL: Imitation augmented deep reinforcement learning enabled UGV-UAV coalition for tasking in complex environments," *IEEE Access*, vol. 8, pp. 102335–102347, 2020.
- [40] J. Li, G. Deng, C. Luo, Q. Lin, Q. Yan, and Z. Ming, "A hybrid path planning method in unmanned air/ground vehicle (UAV/UGV) cooperative systems," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9585–9596, Dec. 2016.
- [41] H. Qin *et al.*, "Autonomous exploration and mapping system using heterogeneous UAVs and UGVs in GPS-denied environments," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1339–1350, Feb. 2019.
- [42] L. Zhang, Z. Zhao, Q. Wu, H. Zhao, H. Xu, and X. Wu, "Energy-aware dynamic resource allocation in UAV assisted mobile edge computing over social internet of vehicles," *IEEE Access*, vol. 6, pp. 56700–56715, 2018.
- [43] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in uav-enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, Sep. 2018.
- [44] Z. Su, Y. Hui, and T. H. Luan, "Distributed task allocation to enable collaborative autonomous driving with network softwarization," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 2175–2189, Oct. 2018.
- [45] Z. Su, Q. Xu, Y. Hui, M. Wen, and S. Guo, "A game theoretic approach to parked vehicle assisted content delivery in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 6461–6474, Jul. 2017.
- [46] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.
- [47] Y. Wu, K. H. Low, and C. Lv, "Cooperative path planning for heterogeneous unmanned vehicles in a search-and-track mission aiming at an underwater target," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 6782–6787, Jun. 2020.
- [48] Q. Liu, L. Shi, L. Sun, J. Li, M. Ding, and F. S. Shu, "Path planning for UAV-mounted mobile edge computing with deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5723–5728, May 2020.
- [49] S. Wan, J. Lu, P. Fan, and K. B. Letaief, "Toward big data processing in IoT: Path planning and resource management of UAV base stations in mobile-edge computing system," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5995–6009, Jul. 2020.
- [50] S. Sekander, H. Tabassum, and E. Hossain, "Decoupled uplink-downlink user association in multi-tier full-duplex cellular networks: A two-sided matching game," *IEEE Trans. Mobile Comput.*, vol. 16, no. 10, pp. 2778–2791, Oct. 2017.
- [51] J. Ding and J. Cai, "Two-side coalitional matching approach for joint MIMO-NOMA clustering and BS selection in multi-cell MIMO-NOMA systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2006–2021, Dec. 2019.
- [52] M. Zeng, S. Leng, Y. Zhang, and J. He, "QoE-aware power management in vehicle-to-grid networks: A matching-theoretic approach," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 2468–2477, Jul. 2018.
- [53] Z. Zhou *et al.*, "When mobile crowd sensing meets UAV: Energy-efficient task assignment and route planning," *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5526–5538, Nov. 2018.
- [54] Z. Zhou, C. Gao, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Social big-data-based content dissemination in internet of vehicles," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 768–777, Feb. 2018.
- [55] Y. Liu, K. Xiong, Q. Ni, P. Fan, and K. B. Letaief, "UAV-assisted wireless powered cooperative mobile edge computing: Joint offloading, CPU control, and trajectory optimization," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2777–2790, Apr. 2020.
- [56] Y. Liu, Q. Wang, H. Hu, and Y. He, "A novel real-time moving target tracking and path planning system for a quadrotor UAV in unknown unstructured outdoor scenes," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 11, pp. 2362–2372, Nov. 2019.



Yuntao Wang is currently pursuing the Ph.D. degree with the School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an, China. His research interests include security and privacy protection in wireless networks, vehicular networks, and UAV networks.



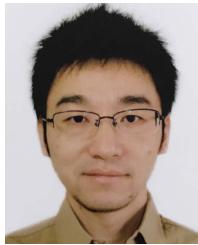
Weiwei Chen received the master's degree from the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai, China. His research interests are in the general areas of wireless network architecture and vehicular networks.



Tom H. Luan (Senior Member, IEEE) received the Ph.D. degree from the University of Waterloo, Canada, in 2012. He is currently a Professor with the School of Cyber Engineering, Xidian University, China. He has authored/coauthored more than 40 journal articles and 30 technical articles in conference proceedings. He has one U.S. patent. His research mainly focuses on content distribution and media streaming in vehicular *ad-hoc* networks and peer-to-peer networking and the protocol design and performance evaluation of wireless cloud computing and edge computing. He has served as a TPC Member for IEEE GLOBECOM, ICC, and PIMRC.



Ruidong Li (Senior Member, IEEE) received the D.Eng. degree from the University of Tsukuba in 2008. He is currently an Associate Professor with the Institute of Science and Engineering, Kanazawa University, Japan. His current research interests include future networks, big data networking, blockchain, the Internet of Things, network security, and wireless networks. He is the Secretary of IEEE ComSoC Internet Technical Committee and the Founder and the Chair of the IEEE SIG on Big Data Intelligent Networking and IEEE SIG on Intelligent Internet Edge. He is a Guest Editor of prestigious journals, such as *IEEE Communications Magazine*, *IEEE Network*, and *IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING*.



Zhou Su (Senior Member, IEEE) research interests include wireless networking, mobile computing, and network security. He has received the Best Paper Award of IEEE ICC2020, IEEE BigdataSE2019, IEEE CyberSciTech2017, and so on. He has served as the Track/Symposium Chair for several international conferences, including IEEE VTC, IEEE/CIC ICCC, WCSP, and so on. He is currently an Associate Editor of IEEE INTERNET OF THINGS JOURNAL, IEEE OPEN JOURNAL OF THE COMPUTER SOCIETY, and *IET Communications*.



Qichao Xu received the Ph.D. degree from the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai, China, in 2019. He is currently an Assistant Professor with Shanghai University. His research interests include wireless network architecture and vehicular networks.



Nan Chen (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Canada, in 2019. She is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Tennessee Tech University, USA. Her research interests include electric vehicle charging/discharging scheme design in smart grid, next-generation wireless networks, and machine learning application in vehicular cyber-physical systems.