Efficiency-Aware Dynamic Service Pricing Strategy for Geo-Distributed Fog Computing

Aerial Edge Computing: Flying Attitude-aware Collaboration for Multi-UAV

Jianwen Xu, *Member, IEEE,* Kaoru Ota, *Member, IEEE,* Mianxiong Dong, *Member, IEEE*

Abstract—With the continuous innovation in manufacturing, Unmanned Aerial Vehicles (UAVs) have gradually become commodities from just professional equipment. As a universal type, quadrotor UAV allows us to see its potential for applications in multiple fields. Moreover, in the brand new field of aerial computing, UAVs have started to play a leading role in providing computing services to mobile users. However, limited by the performance of onboard equipment, we often cannot rely on one UAV to complete complex computing tasks. This paper first carries out a real-world case study and discovers the importance of flying attitude in applying quadcopter UAVs to achieve aerial edge computing. Then in designing the collaboration algorithms, we apply Monte Carlo Tree Search (MCTS) to realize the independent operations of UAVs while assisting each other in accomplishing the common goals. In performance evaluation, we compare the performance of our proposed solution with the existing methods. Finally, the results show that our MCTS-based algorithm can implement efficient collaboration among UAVs while reducing energy consumption and time cost in providing AEC services.

Index Terms—Mobile Edge Computing, Aerial Computing, Multiple UAV Collaboration, Energy Efficiency, Monte Carlo Tree Search

1 Introduction

NOWADAYS, quadcopter Unmanned Aerial Vehicles (UAVs) and related technologies are gradually occupying our field of vision. Thanks to the rapid innovation of manufacturers such as DJI, from household (photographing, video recording, etc.) to industry usages (monitoring, equipment inspecting, etc.), we all can obtain reliable solutions with controllable investment in manpower/resources. Furthermore, in the area of Information and Communication Technology (ICT), the application of quadcopter UAVs is also widely noticed in fulfilling the demands of coping with obstacles as well as bringing breakthroughs.

Until now, after several years of development, we are no longer satisfied with letting quadcopters only act as a supporting role in ICT research. That is, we expect UAVs to take on major roles such as flying service nodes providing distributed services to massive users within coverage. Furthermore, here we name this UAV-based framework as aerial edge computing (AEC).

As shown in Fig. 1, we use a typical three-tier structure to indicate the essential functions of AEC. First, for Ground Tier at the bottom, not only individual users are regarded as targets, but subnetworks, including Wireless Sensor Network (WSN), smart home, Industrial Internet of Things (IIoT), are also considered. Second, for Aerial Tier consisting chiefly of multiple UAVs, we make full use of the high mobility to enhance the flexibility and reliability of computing services being provided. Lastly, in the third tier, UAVs will be remotely supported, including energy management, backhaul data storage, and global monitoring to avoid the occurrence of local faults from affecting the overall structure. Moreover, to realize this AEC framework, we have to face two challenges in the technical aspect.

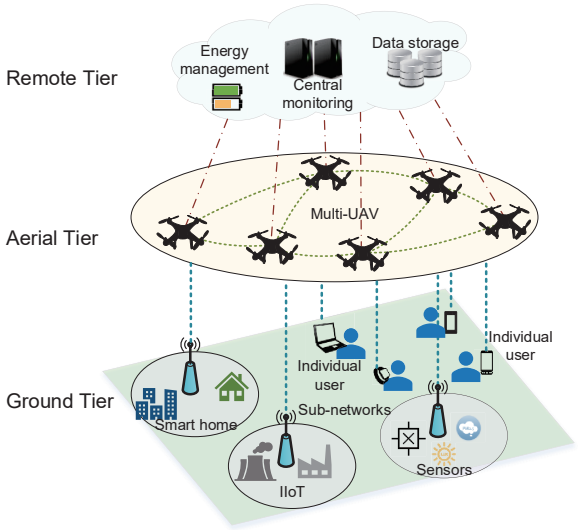


Fig. 1. Vision of aerial edge computing

The first one, in a real-world case, miniature quadcopter UAVs are still mainly powered by lithium-ion batteries. Even after ignoring the minor issue of charging time, it is difficult for UAVs to complete long-term independent operations without a continuous and stable power supply. In fact, this bottleneck is not just in the area of quadcopter UAVs. While using electric vehicles (EVs) and bicycles, we always have to deal with the trade-off between battery volume and usability (overall weight, driving distance on a single charge, battery lifetime, etc.) [1]. Then for UAVs taking off and leaving the ground, the shortcomings of battery power supply are magnified. We take the latest enterprise product of DJI, Matrice 300 RTK, as an example. Although with a 5000 mAh lithium-ion battery, M300 RTK can fly for no more than one hour after charging for one

- *J. Xu, K. Ota, and M. Dong are with the Department of Sciences and Informatics, Muroran Institute of Technology, Muroran, Hokkaido, Japan. E-mail: {jwxu, ota, mxdong}@mmm.muroran-it.ac.jp.*

Manuscript received XX XX, 2021; revised XX XX, 2022.

hour. Once the power is low, we may have to interrupt the service.

The second one comes from the limited workability of a single quadcopter UAV. This not only refers to the payload, which accounts for how many devices can be carried but also the maximum coverage. High mobility is the most considerable advantage of UAVs, but we can not expect a single UAV to expand service coverage by constantly moving. Especially for cases such as a given UAV receiving a new user request while performing a computing task, in order not to affect the processing of the current task, our solution usually can only be re-allocation of the request to a newly dispatched UAV.

Facing these obstacles above, we come up with the idea of cooperation. In our idea, if one UAV can not work for enough time to finish given tasks, why not assign more UAVs to help each other. We consider three main aspects in enabling the job division. The first one is timely communications among UAVs. This aspect is the basis of UAV cooperation. That is, UAVs interact with each other as a prerequisite for initiating cooperation. Based on the first aspect, secondly, we aim to minimize the time and energy cost of unnecessary UAV movements. This will be the main challenge in this research. The third and most essential aspect is the form of cooperation itself. Compared to the dispatching of multiple UAVs in performing given tasks, the cooperation we are going to achieve comes up with a teamwork level. Namely, UAVs in the mission are able to independently undertake tasks while helping each other in complicated jobs. Our objective is to explore the possibility of making individual UAVs realize effective teamwork.

In existing research, this topic already appears in the area of the ground robot. With limited mobility and range of perception, robots always have to cooperate to complete tasks such as inspections of facilities and search & rescue in hazardous environments. Then UAVs, which can be regarded as flying robots, from our point of view, can fill the gap in a similar way. Moreover, there also exist extra differences compared with research on ground robots. For instance, UAVs are able to overcome potential difficulties from terrain factors. In the urban area, which is plain with buildings that may obstruct communication itself, UAVs have flexible responses due to their mobility [2]. In addition, for quadcopter UAVs in flight, it is also different from ground robots in the measurement of motion trajectory and energy consumption. Here we use flying attitude to indicate the movements of UAVs. Moreover, we aim to get a more detailed grasp of the real-time energy consumption value of any given UAV when it is scheduled to complete a flight mission.

Back to the problem in AEC, quadcopter UAVs have a specific energy consumption model. To refine the cost calculation, we need to start with the force analysis of quadcopters in flying. Moreover, this flying attitude-aware collaboration strategy requires an independent operational capacity of quadcopter UAVs. That is, each UAV involved can make decisions on sending/receiving messages from nearby ones as well as changing fly status while considering limited battery power. In this paper, we design solutions based on Monte Carlo Tree Search to simulate finishing tasks by UAV collaboration.

Monte Carlo Tree Search (MCTS) is a decision-making algorithm widely applied in designing software programs for board games playing such as Go, chess, etc. MCTS possesses good flexibility and universality in accomplishing multi-state multi-branch selection processes. As a result, each UAV here will act individually as one player making self-decision according to the current state (position, left power, etc.). Existing solutions in virtue of ground terminals [3] or base stations [4] still rely on the centralized networks monitoring and controlling UAVs' movements. And in this paper, we try to let UAVs share more roles than devices under remote control. We consider that in the era of 5G and beyond, quadcopter UAVs have the capacity to participate in cooperation as individuals.

In summary, to solve the problem in multiple quadcopter UAV collaborative AEC, we apply MCTS in designing heuristic methods. Moreover, we consider the channel characteristics, including path loss and weighted sum rate, to evaluate the performance of the UAV-assisted wireless network being built. Compared with prior works, the novelty and advantages of this paper are as follows.

- We model the multiple UAVs system based on the flying attitudes of the quadcopter. Consequently, it is able to calculate and evaluate the real-time consumption of both time and energy given a fixed number of UAVs;
- On the basis of our proposed AEC platform, we design cooperation algorithms using Monte Carlo Tree Search to allocate UAVs in searching and providing services to users on the ground;
- In the performance evaluation part, we simulate the process of multiple UAVs covering the given area and estimate the channel characteristics of the edge network being built.

The rest of this paper is organized as follows. Section 2 lists the related studies on UAV technology, multiple individuals cooperation in robotics. Section 3 models the multiple UAVs system and formulates the problem to solve. Section 4 designs algorithms for multiple UAV cooperation in providing aerial edge computing services based on Monte Carlo Tree Search. Section 5 evaluates the performance of the proposed multiple UAV system while considering the energy cost of UAV flying as well as channel characteristics in wireless communications and mobile computing. Section 6 summarizes this work.

2 Related Work

The development of industrial manufacturing is self-evident in the advancement of scientific research. Miniature quadrotor, or we used to refer to all drones as a whole, is one of the most eye-catching categories. Today, quadcopter UAV already has the attribute of an aerial mobile development platform. In the area of UAV-assisted mobile computing, the existing researches mainly focus on topics such as UAVs as auxiliary equipment to expand the communication range. Yang *et al.* studied the energy consumption in terminals on the ground and UAVs and came up with the idea of UAV trajectory adjustment in solving the trade-off brought by fight propulsion [5]. Wu *et al.* optimized power control

in transmitting by jointly designing UAV trajectories and multiple-user wireless communications [6]. Dong *et al.* designed a UAV-based data gathering platform to improve the process of data collecting and processing in wireless sensor networks [7]. Sun *et al.* also paid attention to trajectory design in UAV-based systems. They put forward the idea of introducing solar harvesting as the energy source [8]. Li *et al.* regarded UAVs as the vital point in achieving low-cost information-centric Internet of Things [9]. Shakeri *et al.* surveyed the challenges of multiple UAV systems, which include the coverage problem, trajectory planning, and vision-based applications [10]. Nasir *et al.* employed non-orthogonal multiaccess (NOMA) in multiple user systems supported by UAVs as flying base stations [11]. Huang *et al.* focused on the spectrum sharing problem among UAVs and ground infrastructure [12].

In recent years, UAV technology also has been playing an important role as technical support. Zeng *et al.* summarized the fundamentals of UAV communications for next-generation wireless systems, including models of the channel, antenna, power consumption, performance metrics, etc [13]. Li *et al.* integrated current research topics, including software-defined networking (SDN) and virtual network management with radio access network [14] [15]. Ullah *et al.* surveyed the potential research issues in 5G-based UAV applications with deep reinforcement learning and cognitive channel modeling [16]. Nomikos *et al.* designed a mobile radio access network architecture for massive users by integrating ultra-dense networking, network virtualization as well as multiaccess edge computing [17]. Zhang *et al.* explored the possibilities for applying deep Q-learning in resource management in 5G networks [18]. Bithas *et al.* focused on the problem of shadowing environment caused by line-of-sight (LoS) propagation in UAV-enabled 5G communications. They designed a new channel model considering the influence of mobility and shadowing [19].

Furthermore, for the cooperation of multiple individuals, there also exist many instances in the area of ground robotics. Best *et al.* first applied MCTS in deciding when to communicate with other robots in teamwork. They divided the procedure into three steps each robot needs to repeat, planning, deciding, and communicating [20] [21], which greatly inspired the research ideas of this paper. Sahin *et al.* chose Counting Temporal Logic (CTL) as a solution in the expression of task specifications in multiple robots research [22]. Fung *et al.* designed algorithms for collecting information in a brand new environment. In this way, robots can decrease mutual interference at work [23]. Alonso-Mora *et al.* focused on the robot navigation facing obstacles, both static and dynamic, while each robot needs to share the visual module together [24]. Rizk *et al.* surveyed the recent research on heterogeneous multiple agent systems enabled by ground robots. Although we still require manpower in assigning a complicated task, under the unstoppable trend of intelligent automation, the necessity for human intervention is constantly decreasing [25].

In summary, existing research on both quadcopter UAV-assisted mobile computing and multiple robots cooperation has got into nearly all aspects. However, there are still few studies noticing the subject of multiple individual cooperation on UAVs, which, from our point of view, will be an essential research direction in the near future.

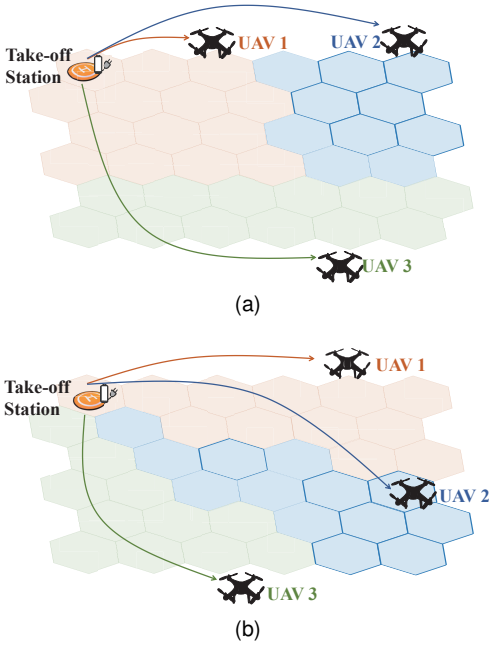


Fig. 2. Job division in multiple UAVs based AEC

3 Problem Formulation

In this section, we formulate the problems to solve and build the mathematical model for our proposed multiple quadcopter UAVs-based system.

3.1 Motivation

In the research of applying quadcopter UAVs in wireless communications and mobile computing, facing large-scale and complicated environments, when a single UAV can not meet our requirements, it is imperative to design a system based on multiple UAVs. Then in our assumption, the following scenario appears. Here we use a case study to introduce our motivation for this research.

While we are adopting UAVs in providing AEC services to multiple users within a relatively large area, an unavoidable problem to solve is the job division. We assume that UAVs will start by taking off and then head to the work area. After then, UAVs are in patrol mode to collect user requests within their signal coverage. At this time, to let UAVs play their roles without interfering with each other, we used to make a plan in advance to assign each UAV to a specific area.

As shown in Fig. 2, we give two examples of job division among multiple UAVs. Here we model the work area as cells (6*6). Three UAVs will work together to patrol all the cells. Fig. 2a is a common approach to cutting into three sub-areas. To make sure each UAV is equal in workload as much as possible, we consider the moving distance from the take-off station. However, this approach may still cause unnecessary waste. During the movement to assigned sub-areas, UAV 2 and UAV 3 in Fig. 2a have to come across the UAV 1's red part. Since for patrol mode, UAV 2 or UAV 3 may encounter the case that user requests in UAV 1's assigned area being received.

TABLE 1
Notations

Notation	Description
$u_i, n_u, \hat{nu}$	UAV node, number of UAVs, and number of UAVs in mission
n_{req}	Number of requests in the current cell
k_P, Λ_P	Number of currencies, and expected value of n_{req} in Poisson distribution
k_E, Λ_E	Shape and rate parameters in Erlang distribution
$t_{serv}, t_{fly}, t_{total}$	Time costs on UAVs hovering for service, flying, and in total
W_{total}	Total energy consumption generated by UAV motors
T_r, ω_r	Torque and angular velocity of UAV motors
$I_m, \dot{\omega_r}$	Moment of inertia and derivate of ω_r respect to t
c_t, c_l, c_e	Torque coefficient and lift coefficient in UAV force analysis, and exploration constant in UCT
ρ, c	Density of air, and speed of light
b, S, R	Width and area of motor blade, and radius of UAV propeller
$\vec{F_r^l}$	Lift force at vertical direction of propeller plane generated by UAV motors
$\vec{F_r^p}, \vec{F_r^{l'}}$	Push force and lift force on z axis generated by UAV motors
m_{total}, m_r, g	Total mass and shared part by motor r of UAV body with payload, and gravitational acceleration
θ, ϕ	Angle of tilt and elevation
$Pl^{LoS}, Pl^{NLoS}, Pl_f, Pl^{a2g}$	Line-of-sight (LoS), non-line-of-sight (NLoS), free space and air-to-ground path losses
η_{LoS}, η_{NLoS}	Path loss exponents of LoS and NLoS
$\chi_0^{LoS}, \chi_0^{NLoS}$	Gaussian random variables of LoS and NLoS
$d_i^{req}, \Delta h^{req_i}$	Distance and height difference between u_i and users
d_f, λ	Free space reference distance, and radio frequency
α, β	Parameters in Sigmmoid function
p_{LoS}, p_{NLoS}	Probability of LoS and NLoS
$\mathbb{P}_i^{req}, P^{tran}, \sigma^2$	Transmission power of u_i, receive power of device sending request, and power of Gaussian noise

Then to deal with the situation in Fig. 2a, we also design a second approach to assigning UAVs. Fig. 2b gives an ideal example in which three UAVs can achieve non-overlapping job division. As a result, the problem in this case study becomes how to divide the workload in a given work area into multiple UAVs, just like Fig. 2b. There exist three factors that may increase the difficulty in making a decision on an assignment. First, work areas in AEC may not all be regular graphs like the ones shown in Fig. 2. Even the example in Fig. 2b can not be easily re-applied when the number of cells changes. Second, since our research is to design and provide AEC services to users in cells, the mission of UAVs also includes the task processing from user requests. In a word, the same number of cells is usually not equal to the same workload. Third, we also have to consider the power management in multiple UAVs assignment.

Especially for the third factor, the energy consumption of quadcopter UAVs in different flying attitudes (hover, move, accelerate, etc.) should change significantly. Since the battery power of UAVs is limited in AEC, how to reduce unnecessary energy consumption will be the top priority of our research. Moreover, to verify this assumption, we design a preliminary experiment using DJI M210 RTK V2.

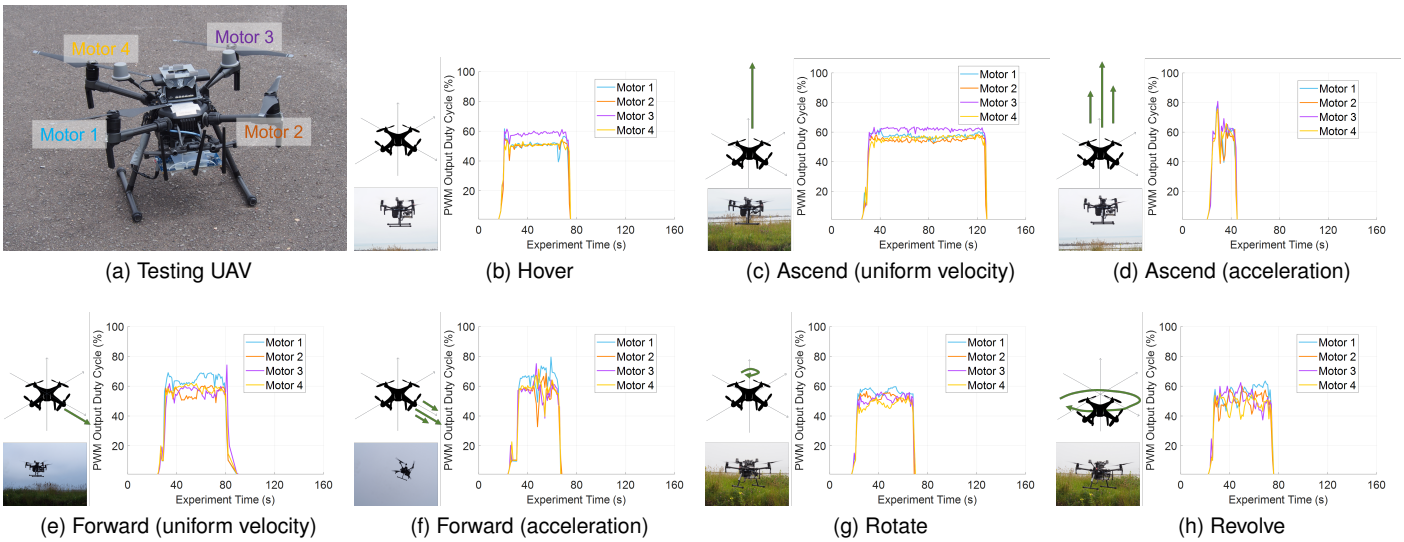
As shown in Fig. 3, we compare the PWM (Pulse Width Modulation) outputs of UAVs in different flying attitudes (hovering, ascending, forwarding, and turning). PWM output is positively correlated with the rotation velocity of the motor. The unit here is a percentage (duty cycle). For instance, 80% duty cycle refers to the case that in 4/5 of the time, voltage is applied to the motor. The order of four motors is given in 3a. We unify the x-axis and y-axis to facilitate comparison and add sketches of corresponding flying attitudes and photographs.

From Fig. 3b we first can mention that the results of the four motors are not the same, and motor 3 is always higher than the other motors. The reason is that we concentrate all the wiring on the position of the UAV body close to motor 3. As a result, the flight controller adjusts the actual speed of each motor to ensure the longitudinal force balance.

Next, for the comparison of PWM outputs in Fig. 3b to 3h, we can obtain that: First, there exist significant differences among the rotation velocities of motors in different flying attitudes, which means that the energy consumption also dramatically changes. Second, compared to hovering and forwarding in uniform velocity, acceleration operations cost more energy. Third, we also notice that the gaps among the four motors have shown some variation. Even the orders of four motors in value are reset. We consider this situation an essential issue in this research and will analyze it in the following chapters to reach a reasonable explanation.

Back to the problem in providing AEC services, in order to maximize the usage of a given number of UAVs in finishing computing tasks, what we need is not only the job division for multiple UAVs but also the teamwork ability. That is, quadcopter UAVs are preferred to work as partners. Thus, multiple UAVs can cooperate in solving user requests within the work area.

Fig. 4 gives an example of our idea on multiple quadcopter UAVs cooperation. As shown in Fig. 4, facing this task of collecting and coping with requests in cells, we are going to send multiple UAVs to respond. Each UAV will start from a take-off station to reach the corresponding sub-area and fly around. We used to plan to let UAVs be responsible for a specific area. However, in this way, several



(a) Testing UAV (b) Hover (c) Ascend (uniform velocity) (d) Ascend (acceleration)

(e) Forward (uniform velocity) (f) Forward (acceleration) (g) Rotate (h) Revolve

Fig. 3. PWM outputs of quadcopter UAV motor in different flying attitudes

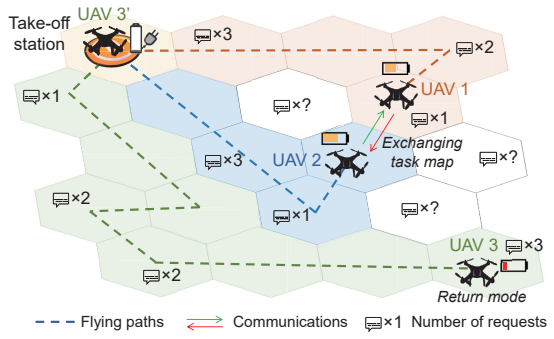


Fig. 4. An example of multiple quadcopter UAVs cooperation

problems may appear to increase energy consumption and reduce overall work efficiency, etc. For example, limited by battery power, even if we can allocate the size of the area for each UAV considering travel distance and the possible performance difference among UAVs (load, configuration, battery level, etc.), it is almost impossible to be fair and reasonable in a prior decision.

Moreover, the UAVs themselves or the equipment onboard are malfunctioning, or occasional requests from users in the responsible area all can bring uncertainty related to the success of the task and the overall stability of the multiple UAV system. To sum up, in order to achieve the goal of applying multiple UAVs to cover the given area, we need UAVs to own the ability to adapt to changes. To make them work together more efficiently, each UAV has to participate as an independent individual. That is, each one continuously interacts with others within the communication range to make clear the overall work progress in real-time. Especially when the given area is not a regular shape or even an unknown area, our multiple UAV systems can respond with flexible and changeable strategies. And for battery power as a bottleneck, after completing the handover of the overall progress, charged newcomers can seamlessly join in and continue to work.

3.2 System Overview

To fulfill the target elaborated in motivation, we will model the multiple quadcopter UAV systems. Suppose we are going to finish a task as shown in Fig. 4, $\{u_i \in U | i = 1, 2, ...n_u\}$ are used to indicate the UAVs being assigned for work. According to the demand of the task, we first send $\hat{n_u}$ UAVs. Until the task is finished, we need to keep $\hat{n_u}$ in work. That is, once a UAV u_i in work has to return for charging, we will send a full-charged one from the take-off station to replace it. Moreover, $wait$ is used to indicate the time cost of providing service in one of the cells or flying towards the target in the plan. u_i may not make a new movement until $wait = 0$.

Next, for the number of requests in a given cell, we use the Poisson distribution to denote the probability mass function.

$$Pr(n_{req} = k_P) = \frac{\Lambda_P^{k_P} \cdot e^{-\Lambda_P}}{k_P!} \quad (1)$$

where n_{req} stands for the number of requests from users in the current cell. Λ_P is the expect value of n_{req}. For the service time of one request, we use Erlang distribution to denote. Erlang distribution is a kind of probability distribution. It is widely applied in indicating the time interval between independent random events. Compared with the exponential distribution, which is also used in describing independent random events, Erlang can better fit the situation of real-world data. That is, Erlang is more suitable for multiple serial processes when a memoryless property is not significant. Then we have the probability density function (PDF)

$$Pr(x, k_E, \Lambda_E) = \frac{\Lambda_E^{k_E} \cdot x^{k_E - 1} \cdot e^{-\Lambda_E x}}{(k_E - 1)!} \quad (2)$$

where positive integer k_E and the positive real number Λ_E are the shape and rate of Erlang distribution, respectively. Thus, we have the total service time of a given cell in which there are n_{req} requests.

$$t_{serv}(n_{req}) \quad = -\frac{1}{\Lambda_E}\sum_{i=1}^{n_{req}} ln \prod_{k_E}^{j=1} U_{i,j},\ U_{i,j} \in (0,1] \tag{3}$$

where $U_{i,j}$ is a uniformly distributed random number [26]. As a result, we apply Erlang distribution in calculating the time cost while UAVs are hovering and providing AEC service to users.

In this research, both time and energy costs will be considered as basic metrics in judging the performance of finishing given tasks. First, for time cost, we focus on multiple UAVs as a whole instead of counting the time of all single ones. Besides the time t_{serv} while UAVs are hovering, we also need t^{fly} as time cost on UAV flying to calculate energy cost later.

Second, in this research, our motivation is to cope with the issue brought by the relatively low continuous work capacity of individual UAVs. Because of limited battery power, we come up with the idea of enabling the cooperation of multiple UAVs. For the mobile users' transmission energy, in the proposed scenario, mobile users do not need to increase the times of communications or continuous time in connections except for the original acquisition of the service. All possible adjustments on time or energy cost in the method itself are undertaken by the edge devices (UAVs). Moreover, for energy consumption on UAV batteries, we focus on the power source of the UAV motors, which accounts for the vast majority and always stays in line with our motivation. In fact, there exists a magnitude gap between the powers of the two parts. For example, as the equipment we are using, DJI Matrice 210 RTK V2 needs power ranging from about 317 to 437 watts, while NVIDIA Jetson TX2 only needs 7.5 watts to run. However, although the proportion of UAV transmission energy is too small to be concluded in a separate comparison, we did not give up the consideration of the energy consumption on transmission. Instead, we develop a more intuitive metric in the performance evaluation, the times of UAV communications are proportional to the UAV transmission energy. We believe that this is a better way to statistically compare the possible consumption of the entire multi-UAV system.

Here, we use $\hat{n_u}$ to indicate the number of UAVs assigned to patrol around the target area. Starting from $t = 0$, $\hat{n_u}$ UAVs departure from a take-off station to take charge of the given area. In flight, each UAV will keep in touch with each other to ensure a clear division of labor. Their common goal is to complete all the tasks being collected during flight as quickly as possible. As mentioned above, the energy cost of flying is the main consideration in our research since once any UAV only has enough battery power left to return to the take-off station, it must return in time, and a newly charged one will then join the mission as handover. Thus, next, we calculate the energy consumption in flight [27]

$$W_{total} = \int_0^{t^{total}} \sum_{r=1}^{4} T_r(t) \cdot \omega_r(t)\ dt \tag{4}$$

W_{total} is the total energy consumed in motor-driven systems. Since we focus on the type of quadcopter UAV, $r \in \{1,2,3,4\}$ in Equation (4) refers to the four motors of UAV. t_{total} is the time cost on UAVs from taking off to landing. That is, t_{total} contains t_{serv} and t_{fly}. $T_r(t)$ and $\omega(t)$ are torque and angular velocity of motors, respectively. For torque, we have

$$T_r = c_t \cdot \omega_r^2 + I_m \cdot \dot{\omega}_r \tag{5}$$

where c_t is the torque coefficient. I_m stands for the moment of inertia. $\dot{\omega}_r$ is the derivative of ω_r with respect to t. While the effect from $\dot{\omega}_r$ is considered small, we usually ignore this term [28]. Then Equation (4) can be expressed as

$$W_{total} = c_t \int_0^{t^{total}} \sum_{r=1}^{4} \omega_r(t)^3 dt \tag{6}$$

Once we know the ω_r, we also can get the lift force generated by the motor. First, we consider the situation in which the UAV is hovering unmoved.

$$\begin{aligned} d|\vec{F_r^l}| \quad &= \frac{1}{2} c_l \cdot \rho \cdot v_r^2 \cdot dS \\ &= \frac{1}{2} c_l \cdot \rho \cdot (\omega_r \cdot R)^2 \cdot b \cdot dR \end{aligned} \tag{7}$$

where c_l stands for the lift coefficient. ρ is the air density. v_r refers to the linear velocity of the propeller calculated by angular velocity ω_r and radius R. dS is the small area of the motor blade calculated by radius dR and width of blade b. Thus we obtain the lift force.

$$|\vec{F_r^l}| \quad = \frac{1}{6} c_l \cdot \rho \cdot \omega_r^2 \cdot R^3 \cdot b \tag{8}$$

As a result, to generate enough lift force for hovering motionless, we need to ensure that [29]

$$\sum_{r=1}^{4} |\vec{F_r^l}| = m_{total} \cdot g \tag{9}$$

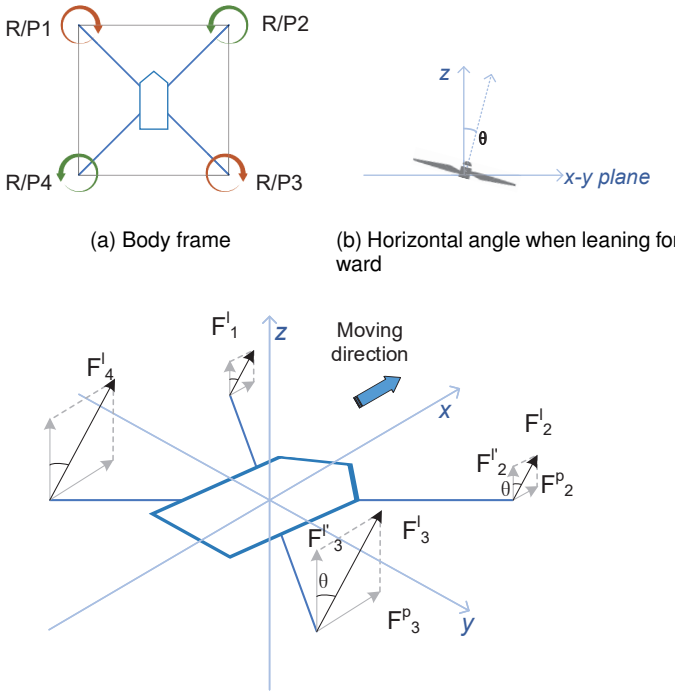
where m_{total} is the total mass of the quadcopter UAV and all the payload (avionics system, communication module, etc.). g is the gravitational acceleration.

Next, we turn to the situation of the UAVs moving. For quadcopter UAVs, the movement is achieved by tilting the body to generate the push force. We use figures to give some examples.

Fig. 5 gives an instance to build the relationship between motor rotating and UAV moving. As shown in Fig. 5b, the four motors/propellers (R/P1$\sim$ R/P4) are at both sides of the quadcopter UAV body. To fly forward, a quadcopter UAV needs to change the angular velocity of the motors. That is, to break the balance give in Equation (9), we just increase ω_3^r, ω_4^r and decrease ω_1^r, ω_2^r. UAV body then may tilt like Fig. 5b. We use θ to denote this angle of tilt, then we have

$$\vec{F_r^l} = \vec{F_r^p} + \vec{F_r^{l'}} \tag{10}$$

where $\vec{F_r^l},\ r \in \{1,2,3,4\}$ refers to the lift force at the vertical direction of the propeller plane. $\vec{F_r^p}$ denotes the push force generated by the angle of tilt on the x axis. Thus, after summing up the lift forces from four motors, we have



(a) Body frame

(b) Horizontal angle when leaning forward

(c) Principles of quadcopter UAV movement

Fig. 5. The case of quadcopter UAV movement

$$F^{p}_{total} = \sum_{r=1}^{4} |\vec{F}^{p}_{r}| = m_{total} \cdot ac$$
$$F^{l'}_{total} = \sum_{r=1}^{4} |\vec{F}^{l'}_{r}| = m_{total} \cdot g \tag{11}$$

where $\vec{F}^{l}_{r}$ refers to the lift force generated by motor r. ac is the acceleration of the UAV moving on the $x-y$ plane. ac with a negative value refers to the case of deceleration. $\vec{F}^{l'}_{r}$ is the component on the z axis. Since we have

$$\theta = atan(\frac{|\vec{F}^{p}_{r}|}{|\vec{F}^{l'}_{r}|}) \tag{12}$$

Thus, we obtain the expression of $|\vec{F}^{l}_{r}|$

$$|\vec{F}^{l}_{r}| = \frac{|\vec{F}^{l'}_{r}|}{cos\theta} = \frac{m_r g}{cos[atan(\frac{ac}{g})]} \tag{13}$$

where m_r stands for the part of mass taken by motor r. With Equation (7), we have the angular velocity of one motor r (ω_r) as

$$\omega_r = \sqrt{\frac{6|\vec{F}^{l}_{r}|}{c_l \rho R^3 \cdot b}} = \sqrt{\frac{6m_r \cdot g}{c_l \rho R^3 b \cdot cos[atan(\frac{ac}{g})]}} \tag{14}$$

Together with Equation (6), we can obtain the total energy cost of quadcopter UAV in flight respect to t^{total}

$$W_{total} = c_t \int_0^{t^{total}} \sum_{r=1}^{4} (\sqrt{\frac{6m_r(t)g}{c_l \rho R^3 b \cdot cos[atan(\frac{ac(t)}{g})]}})^3 dt \tag{15}$$

$$min \quad c_t \int_0^{t^{total}} \sum_{r=1}^{4} (\sqrt{\frac{6m_r(t)g}{c_l \rho R^3 b \cdot cos[atan(\frac{ac(t)}{g})]}})^3 dt$$
$$s.t. \quad \sum_{r=1}^{4} m_r = m_{total} \tag{16}$$

where c_t and c_l are torque and lift coefficient, respectively. ρ is air density. g is gravitational acceleration. R is the radius of UAV propeller. b is the width of propeller blade. ac is the acceleration of UAV moving on the $x-y$ plane. Total time after a given UAV taking off t^{total} is given in Equation (4). The order of motors r is given in Fig. 5a. The mass of UAV with payload undertaken by motor r m_r is given in Equation (13).

Besides the energy consumption undertaken by UAV batteries, we also consider time cost as a second optimization problem in this research.

$$min \quad t^{total} = t^{fly}(tvl\_dis) + t^{serv}(n_{req}) \tag{17}$$

where latency constraints of task processing in total t^{total} are given by two parts. The time cost of a UAV flying t^{fly} at speed refers to the travel distance between departure and destination location. The rest part on UAV hovering for service t^{serv} refers to how long the user requests can be solved.

After building the wireless network with multiple UAVs, we also consider the transmission impediments in channels of wireless communications to evaluate the performance of our proposed multiple UAV system. The channel characteristics we consider are path loss, signal-to-interference-plus-noise ratio (SINR), and weighted sum rate (WSR) in multi-user multiple-input and multiple-output (MIMO) systems which are crucial in designing UAV based systems [30].

For the communications between UAVs in the air and devices on the ground, namely air-to-ground communications, we consider both line-of-sight (LoS) and non-line-of-sight (NLoS) models of path loss [31].

$$Pl_i^{LoS} = Pl_f + 10\eta_{LoS} \cdot log_{10} d_i^{req} + \chi_0^{LoS}$$
$$Pl_i^{NLoS} = Pl_f + 10\eta_{NLoS} \cdot log_{10} d_i^{req} + \chi_0^{NLoS} \tag{18}$$

where η and χ stand for path loss exponent and Gaussian random variable, respectively. d_i^{req} is the distance between u_i and the position of the request. And for the free space path loss $Pl_f(d_f)$ we have

$$Pl_f = 20 \cdot log_{10}(\frac{4\pi \cdot d_f \cdot f}{c}) \tag{19}$$

where d_f is the free space reference distance. And f denotes the frequency (Hz), and c denotes the speed of light.

$$d_i^{req} \gg \lambda = \frac{c}{f} \tag{20}$$

λ denotes the signal wavelength. That is, d_a must be large enough that the antennas are in the far-field of each other [32]. Then we have the path loss of the air-to-ground part,

$$\begin{aligned} Pl_i^{a2g} = & \quad p_{LoS} \cdot Pl_i^{LoS} + p_{NLoS} \cdot Pl_i^{NLoS} \\ = & \quad \frac{Pl_i^{LoS}}{1+\alpha \cdot e^{-\beta \cdot (\phi - \alpha)}} + \frac{Pl_i^{NLoS} \cdot \alpha \cdot e^{-\beta \cdot (\phi - \alpha)}}{1+\alpha \cdot e^{-\beta \cdot (\phi - \alpha)}} \end{aligned} \tag{21}$$

where p_{LoS} and p_{NLoS} are probability of LoS and NLoS. α and β are parameters in Sigmmoid function. And we have

$$p_{LoS} + p_{NLoS} = 1 \tag{22}$$

ϕ is the elevation angle given by

$$\phi_i = asin(\frac{\Delta h_i^{req}}{d_i^{req}}) \tag{23}$$

where Δh_i^{req} and d_i^{req} are height difference and distance between UAVs and ground devices, respectively. Here we apply uniform distribution in determining the positions of devices sending out requests.

Then to calculate SINR, we need to obtain the power of the incoming, the power of the other signals as interference in the network, as well as noise term. First, we have the received power of the device sending out a request to UAV u_i [33]

$$\mathbb{P}_i^{req} = \frac{P^{tran}}{10^{0.1 Pl_i^{a2g}}} \tag{24}$$

where P^{tran} denotes the transmission power. Next, to calculate the interference power, since, for u_i, any other signals can be regarded as interfering sources; thus, we obtain the SINR.

$$SINR_i^{a2g} = \frac{P}{I+N} = \frac{\mathbb{P}_j^{req}}{\sum\limits_{k}^{n_{req}} \mathbb{P}_{k,k\neq j}^{req} + \sigma_g^2} \tag{25}$$

where σ_g^2 is the power of Gaussian noise. At last, we also can estimate the weighted sum rate of communications between UAVs in the air and users on the ground.

$$\begin{aligned} WSR_i^{a2g} &= \sum_{j}^{n_{req}} log_2(1 + SINR_i^{a2g}) \\ &= \sum_{j}^{n_{req}} log_2(1 + \frac{\mathbb{P}_j^{req}}{\sum\limits_{k}^{n_{req}} \mathbb{P}_{k,k\neq j}^{req} + \sigma_g^2}) \end{aligned} \tag{26}$$

4 Algorithm Design

In this section, we are going to design the algorithms to solve the problem formulated in Section 3.

We assume that in a given area, there exist n_C cells. $\hat{n_u}$ quadcopter UAVs are sent out to fly across the whole area. For each cell, there exist n_{req} requests to be coped with by $\hat{n_u}$ UAVs. And we are going to design algorithms based on Monte Carlo Tree Search (MCTS) in setting UAVs as individual players in playing a collaborative game.

In MCTS, the basic principle is repeating a four-step (Selection, expansion, simulation, and back-propagation) rollout process in searching for optimal solutions, which is widely used in playing board games such as chess and Go. However, there exist some differences between existing MCTS and our design. First, the number of players is not fixed. In other words, our approach has to satisfy the case of multiple players, which are quadcopter UAVs in AEC. Second, the terminal condition in our approach is different from the existing ones. That is, rather than win or lose for each player, what we pursue is the result that all cells being reversed. Third, for the reward after any rollout, we also have to design a dedicated reward function to make sure we can accomplish the set goal.

We divide the algorithm design into three parts. First, Algorithm 1 is responsible for finding out all the options available in the current rollout.

Algorithm 1 PACR: Possible Actions in Current Rollout

Require: $u_i.Cell(\tau)$
Ensure: $u_i.Cell(\tau).PossActs$
1: **for** $i \leftarrow 1$ to $length(u_i.Cell(\tau).neighbor)$ **do**
2: **if** $u_i.Cell(\tau).neighbor[i].visited = 0$ **then**
3: $u_i.Cell(\tau).PossActs.add(u_i.Cell(\tau).neighbor[i])$;
4: **end if**
5: **end for**
6: **if** $u_i.Cell(\tau).PossActs = \emptyset$ **then**
7: find the nearest cell whose $visited = 0$;
8: **end if**

As shown in Algorithm 1, in current τ, to choose the next destination to approach, the preferred options are the neighbors of current cell $CurrCell$. $visited$ denotes whether a cell is visited by any UAV or not. An exceptional case exists (Line 6,7) while all neighbors are already visited by other UAVs. At this time, u_i can add the nearest non-neighbor cell as a supplement. Since the number of neighbors is an integer between 1 and 6, the time complexity of Algorithm 1 is $O(1)$.

To solve the main difficulty in balancing between exploitation and exporation [34], we follow the Upper Confidence Bound 1 applied to Trees (UCT) [35] while adding our modification to fit into the application scenario here. For the criterion of taking actions between deep excavation into movements with known considerable average reward (exploitation) and a brand new attempt at unknown territory (exploration), we have

$$\begin{aligned} UCT(ChdCell, Cell) = & \quad \frac{Rwd(ChdCell)}{Num(ChdCell)} \\ & + 2c_e \cdot \sqrt{\frac{2ln(Num(Cell))}{Num(ChdCell)}} \end{aligned} \tag{27}$$

Equation (27) gives our UCT expression, where $Num()$ denotes the number of $Cell/ChdCell$. And we have the child of current cell $ChdCell.parent = Cell$. c_e refers to the exploration constant [36]. $Rwd(ChdCell) \setminus Num(ChdCell) \in [0,1]$. The reward function $Rwd()$ is given in Line 4-10 in Fig. 6. Here we consider two cases of reward being obtained in the recursive loop. First, once

there exist request(s) at the position of current $ChdCell$ ($n_{req} > 0$), a full reward will be added, which is the same as a win in traditional MCTS. Second, although there may exist no request at the current $ChdCell$, it still can earn a reward by the proportion of unvisited neighbors (Line 7-9). However, only when all the neighbors are not visited at this time-slot (τ) can $ChdCell$ get the same full reward as the first case.

Second, we design a collaborative MCTS for multiple UAV-based AEC.

Fig. 6 gives the four steps of our collaborative MCTS. We assume that there are three UAVs (U_1, U_2, and U_3) in the mission. First for Selection step in Fig. 6, our collaborative MCTS starts from U_1. The fractions refer to the reward being obtained in rollouts. The denominators are the numbers of neighbors when a given UAV is visiting the current cell. According to the example shown in Fig. 2 and 4, the maximum is 6. However, once the current cell is at the edge of the work area, the denominator will be less than 6.

Similarly, the numerators are unvisited neighbors. In this way, UAVs are encouraged to explore the area with less-visited cells. Moreover, we are deliberately making efforts to avoid the overlapping of the areas where UAVs are responsible. Second, for the Expansion step, we choose a child node of the current node. This node is one of the possible movements of U_1 after U_3. Third, for the Simulation step, we simulate the process of rollout until the terminal condition is reached (all cells being reversed). Lastly, in the Back-propagation step, we add the reward of rollout result to each of the ancestors.

Next we will design a multiple UAV cooperation algorithm based on Algorithm 1 and Fig. 6.

Algorithm 2 gives our main design on cooperated multiple UAVs communications (MUCA). According to metrics defined in Table 1, in MUCA, we also pay attention to time and energy consumption in UAV communication. At τ, first we check if the current u_i is busy ($wait > 0$) or not as shown in Line 3. Once u_i is not in service or flight, we need the state of u_i while making a decision at τ. And we obtain that the initial velocity (v_{start}) of the child cell equals the final velocity (v_{end}) of the current cell (Line 5).

Next, from Line 6 to 9, we judge if the child cell can be selected as the next step by calculating whether the remaining battery power is enough for u_i to return to the take-off station from the position of the child cell. Once the power is enough, we sum up the time cost as $wait$. Two parts come from UAV moving (Line 8) and service providing (Line 10∼12). Once u_i's battery can not afford the trip from $Cell$, $ChdCell$ to the take-off station, the current UAV then enters return mode, and a full-charged one will take place (Line 18 and 19). Lastly, as the most critical part of achieving cooperation among multiple quadcopter UAVs, in our design, if u_i makes a decision on the next movement, the other $u_j, j \neq i$ can receive a message from u_i about the update in each other's map. Moreover, to avoid extra costs on communications among UAVs, we only consider one-time forwarding. If u_j is not in the communication range of u_i, it has to wait until they are nearby. In consideration of the worst case, the time complexity of Algorithm 2 is $O(\hat{n_u} n_M)$. n_M refers to the maximum number of MCTS iterations [37].

Algorithm 2 MUCA: Multiple UAVs Cooperation Algorithm

Require: $u_i, i \leftarrow 1, 2...\hat{n_u}$
Ensure: $\forall Cell \in map, Cell.visited \neq 0$
1: **procedure** LOOP ($\tau \leftarrow \tau + 1$)
2: **for** $i = 1$ to $\hat{n_u}$ **do**
3: **if** $u_i.wait = 0$ **then**
4: $u_i.ChdCell(\tau) \leftarrow$ **CoopMCTS**$(u_i.Cell(\tau))$;
5: $u_i.ChdCell(\tau).v_{start} \leftarrow u_i.Cell(\tau).v_{end}$
6: **if** $u_i.LeftBat \geq EnergyCst(u_i.ChdCell(\tau), u_i.Cell(\tau)) + EnergyCst(TakeoffSta, u_i.Cell(\tau))$ **then**
7: $u_i.LeftBat \leftarrow u_i.LeftBat - EnergyCst(u_i.ChdCell(\tau), u_i.Cell(\tau))$;
8: $u_i.wait \leftarrow TimeCst(u_i.ChdCell(\tau), u_i.Cell(\tau))$;
9: Calculate $u_i.ChdCell(\tau).v_{end}$ from $TimeCst()$
10: **if** $u_i.ChdCell(\tau), n_{req} > 0$ **then**
11: $u_i.wait \leftarrow u_i.wait + TimeServ(u_i.ChdCell(\tau))$;
12: **end if**
13: Record Pos at $u_i.map$
14: **if** any other $u_j, j \neq i$ witnin the comm range of u_i **then**
15: Record $u_i.ChdCell(\tau).Pos$ at $u_j.map$;
16: **end if**
17: **else**
18: $u_i.LeftBat \leftarrow FullCharge$;
19: $u_i.Cell(\tau).Pos \leftarrow TakeoffSta$;
20: **end if**
21: **else**
22: $u_i.wait \leftarrow u_i.wait - 1$;
23: **end if**
24: **end for**
25: **end procedure**

5 Performance Evaluation

In this section, we carry out simulations on our proposed cooperated multiple UAVs communications algorithm (MUCA) and evaluate the performance in finishing tasks with different numbers of quadcopter UAVs.

In the simulation, we assume that the free space is divided into hexagonal cells equal in size. All quadcopter UAVs departure from the take-off station to cope with unknown requests from uses on the ground. For multiple UAVs, they need to complete two tasks, searching and collecting requests while patrolling within the free space, then satisfy all the requests together. We repeat this whole process in several rounds. As the common goal and the end conditions of one round experiment, multiple UAVs will continue flying across all the cells to receive and process requests from users. Table 2 gives the constant experimental settings.

As shown in Table 2, we give the parameter values being used in this research. First, for the quadcopter UAV body, including the size of the propeller blade, mass, velocity, and acceleration/deceleration, we refer to the specifications of DJI Matrice 210 RTK V2 [38]. In deciding the frequency

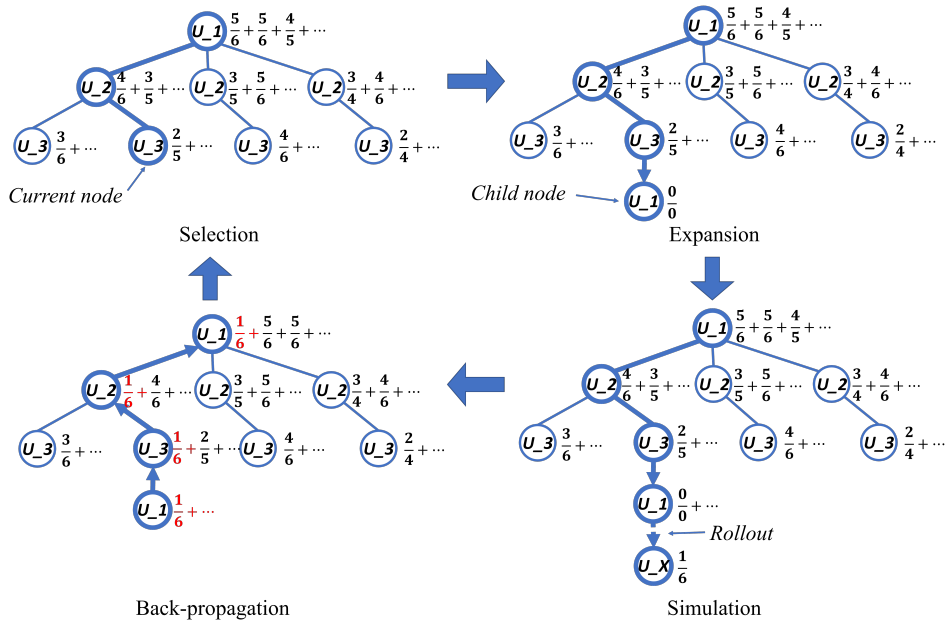


Fig. 6. An example of multiple quadcopter UAVs cooperation

TABLE 2
Experimental settings

Parameter	Value
Radius/width of propeller blade	21.5/2.5 cm
Total mass of UAV with full payload	6.14 kg
Maximum velocity of UAV	61.2 km/h
Acceleration/Deceleration of UAV	3/-2 m/s^2
Free space reference distance d_f	5 m
Flying height of UAVs	200 m
Frequency (FR1) f	4.9 GHz
Power of Gaussian noise	-100 dBm
Power of transmission	33 dBm
Torque coefficient c_t	1.14×10^{-7} $N \cdot m \cdot s^2$
Lift coeffieicnt c_l	1.7
Density of air ρ	1.225 kg/m^3
Path loss exponent η_{LoS}, η_{NLoS}	2, 2.5
Gaussian random variable $\chi_0^{LoS}, \chi_0^{NLoS}$	5, 20 dB
Parameters in Sigmmoid function α, β	12, 0.135
Exploration parameter c_e	$\sqrt{2}$

and transmission power, we refer to the configurations of Standard C Band devices. For the coefficients of torque and lift, we refer from [28] and [39]. The density of air is at 101.325 kPa (abs) and 15 $^\circ C$. We refer from [31] [32] and choose path loss exponent and parameters in Sigmoid function. We set $\hat{n_u}$ from 3 to 5, a number of cells from 25 (5×5) to 225 (15×15), and repeat ten rounds for each group. The side length of the hexagonal cell is 200 m. We compare the results of our solutions with three existing methods.

As shown in Fig. 7, we separately draw line charts and bar charts to display the results of energy consumption and time cost. RSU stands for a baseline method. That is, after finding out all the possible actions in Algorithm 1, a next step is randomly selected from $PossActs$. BFS is a conventional algorithm widely applied in networking and communication problems. BFS shows considerate efficiency in dealing with finite data structures. TAMCTS is based on a single-player MCTS method in [40]. TAMCTS prefers the average immediate cost in deciding the UCT function and shows superior performance in the task assignment of multiaccess edge computing.

First, we discuss the energy results of the four methods. Here we consider the two parts of main consumption when multiple quadcopter UAVs are traveling and hovering for service. In Fig. 7a, RSU consumes the most energy in general. Especially when the number of cells increases, the gap becomes larger and larger. TAMCTS costs about 10 % more than our method. BFS shows good performance in the energy of UAV traveling. Besides, in Fig. 7b, we also consider the situation when UAVs are hovering and providing AEC services to users in cells. Our method is better in terms of value and stability. And in Fig. 7c, we add up the two parts and obtain the total energy undertaken by batteries.

Second, we evaluate the results of time cost in AEC. Here we consider three parts of main consumption when multiple quadcopter UAVs are in different flying attitudes. From Fig. 7d and 7e, we can know that most of the time is spent on flying at uniform motion. Together with the results of UAV hovering, we can figure out that our method can achieve the reduction of both energy and time in providing AEC services to ground users.

Besides the main metrics, we also regard some supplementary results to further confirm our evaluation. Fig. 8a gives the times of communications among multiple quadcopter UAVs. In Fig. 7, we mention that BFS shows considerable performance beyond expectation. However, what needs to be paid is the extra many times' information exchanges. Moreover, we calculate the channel characteristics of the edge network built by multiple UAVs. The average path loss of air-to-ground communications is 141.23. SINR is 0.13. WSR is given in Fig. 8b. We separately calculate the results

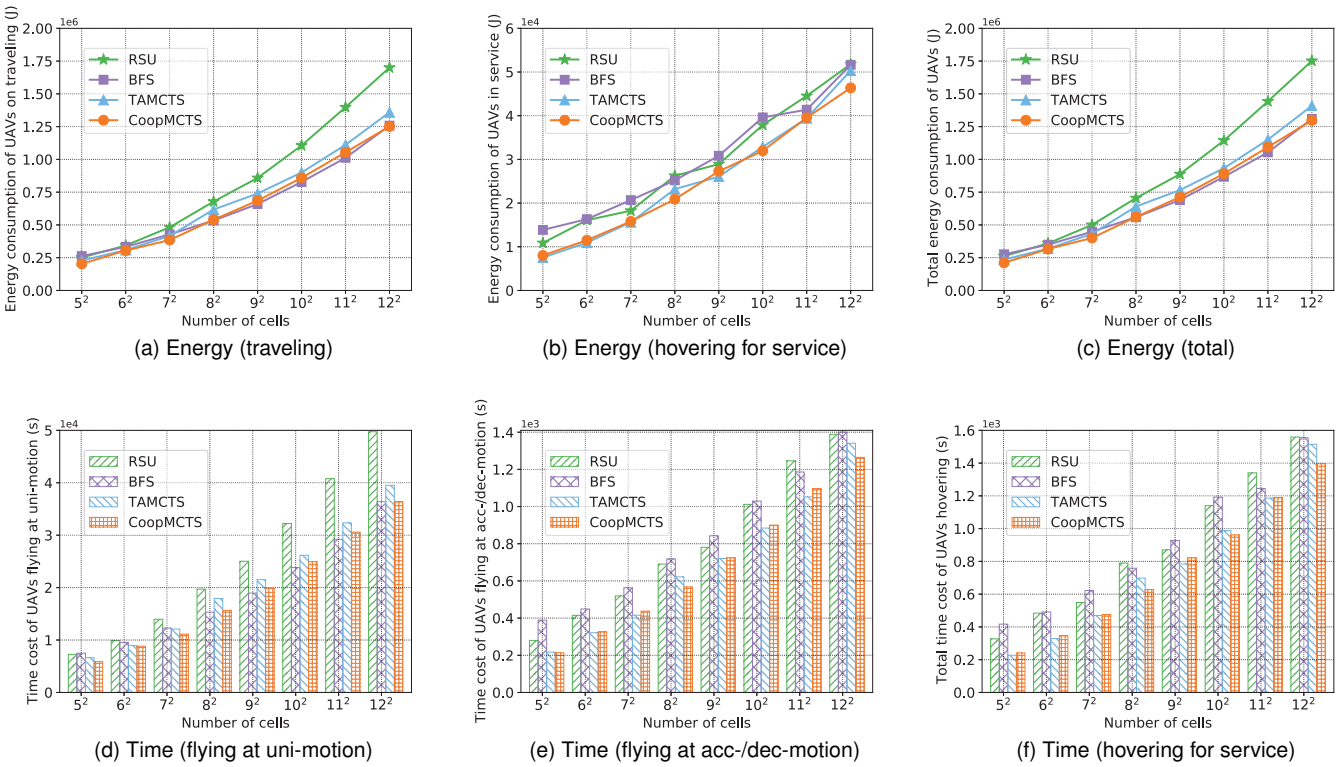


Fig. 7. Energy consumption and time cost of quadcopter UAVs

TABLE 3
Job division in Fig. 9

UAV No.	Num. of cells	Flying path
1 (red)	15	(1, 0)-▷(2, 0)-▷(3, 0)-▷(4, 0)-▷(5, 0)-▷(6, 0)-▷(7, 0)-▷(7, 1)-▷(7, 2)-▷(7, 3)-▷(7, 4)-▷(7, 5)-▷(7, 6)-▷(7, 7)-▷(2, 5)
2 (blue)	11	(0, 1)-▷(0, 3)-▷(0, 4)-▷(0, 5)-▷(0, 6)-▷(0, 7)-▷(2, 7)-▷(3, 6)-▷(3, 5)-▷(3, 4)-▷(2, 3)
3 (green)	13	(1, 1)-▷(2, 1)-▷(3, 1)-▷(4, 1)-▷(5, 2)-▷(5, 1)-▷(6, 1)-▷(6, 2)-▷(6, 3)-▷(6, 4)-▷(6, 5)-▷(5, 6)-▷(2, 4)
4 (yellow)	12	(0, 2)-▷(1, 3)-▷(1, 4)-▷(1, 5)-▷(1, 6)-▷(1, 7)-▷(2, 6)-▷(3, 7)-▷(4, 7)-▷(5, 7)-▷(6, 7)-▷(6, 6)
5 (grey)	12	(1, 2)-▷(2, 2)-▷(3, 2)-▷(3, 3)-▷(4, 2)-▷(5, 3)-▷(4, 3)-▷(5, 4)-▷(5, 5)-▷(4, 5)-▷(4, 6)-▷(4, 4)

with different numbers of UAVs. The variation shows that WSR does not change significantly with the number of UAVs working at the same time.

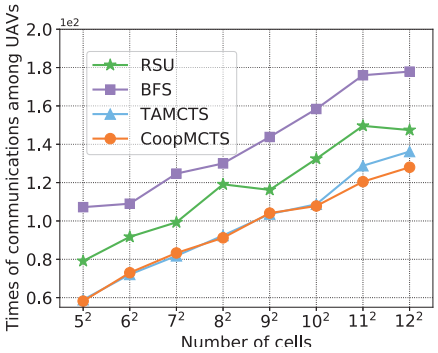
Lastly, we use an example to verify the utility of our proposed algorithms in solving the job division problem shown in Fig. 2. Fig. 9 is a simulation result when there are multiple quadcopter UAVs are assigned to patrol an 8×8 cells' area. The details of the flying paths of five UAVs are given in Table 3. The position of *Task-off Station* is (0, 0). Then we can know that our proposed algorithms achieve the job division as expected. There exist some special cases in which some UAVs make long-distance movements without hovering for service. However, these cases only happen when most of the cells are already visited. For instance, UAV 1 (flying paths in red hexagons) travels from (7, 7) to (2, 5). The reason why choosing UAV 1 is that UAVs in shorter distances to (2, 5) are all busy. After internal communications among each other, the mission is finally divided into UAV 1.

In summary, the comparison of multiple metrics shows that our proposed CoopMCTS method can reduce energy consumption on UAV batteries, especially in case of a large workload. The results of times of communications among UAVs also show the efficiency of CoopMCTS in guaranteeing up-to-date task progress. In this way, multiple quadcopter UAVs can perform independent operations in achieving AEC.

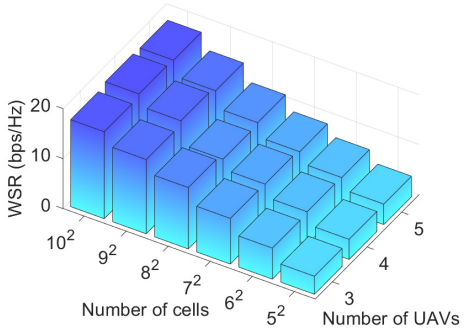
6 Conclusion

In this paper, we focus on the problem of reducing the energy cost of onboard batteries in multiple UAVs cooperated aerial edge computing. We first model the energy consumption based on the flying attitudes of the quadrotor UAVs. In designing the collaborative strategy of UAVs, we apply Monte Carlo Tree Search in achieving independent operations of single UAVs while working together. In the performance evaluation, the results show that our MCTS-based method owns efficiency and stability while reducing energy consumption and time cost.

In the future, our target is to advance the applications of quadcopter UAV collaboration under complex environments in 5G and beyond networks. Our current research is



(a) Times of UAV communications



(b) WSR (bps/Hz)

Fig. 8. Supplementary results

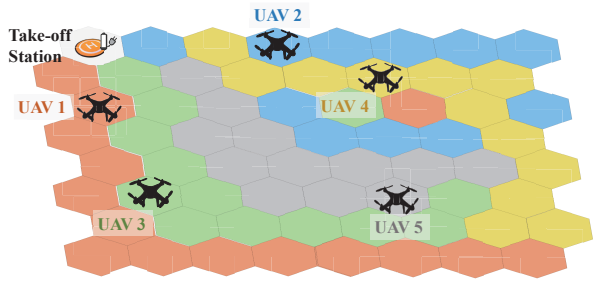


Fig. 9. An example of multiple UAVs cooperation in AEC: hexagons in different colors are flying paths of UAVs assigned in the current mission

still not paying attention to all the technical details that are needed in real-world experiments, which is also an essential topic in our next research phase.

Acknowledgments

This work is partially supported by JSPS KAKENHI Grant Numbers JP19K20250, JP20H04174, JP22K11989, JP22K17876, Leading Initiative for Excellent Young Researchers (LEADER), MEXT, Japan, and JST, PRESTO Grant Number JPMJPR21P3, Japan. Mianxiong Dong is the corresponding author.

References

[1] E. Redondo-Iglesias, E. Vinot, P. Venet, and S. Pelissier, "Electric vehicle range and battery lifetime: a trade-off," in *32nd Electric Vehicle Symposium (EVS32)*, May 2019. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02143273v2

[2] J. Xu, K. Ota, and M. Dong, "Big data on the fly: Uav-mounted mobile edge computing for disaster management," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2620–2630, 2020.

[3] Y. Zeng and R. Zhang, "Energy-efficient uav communication with trajectory optimization," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, 2017.

[4] S. Zhang, Y. Zeng, and R. Zhang, "Cellular-enabled uav communication: A connectivity-constrained trajectory optimization perspective," *IEEE Transactions on Communications*, vol. 67, no. 3, pp. 2580–2604, 2019.

[5] D. Yang, Q. Wu, Y. Zeng, and R. Zhang, "Energy tradeoff in ground-to-uav communication via trajectory design," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 6721–6726, 2018.

[6] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-uav enabled wireless networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 2109–2121, 2018.

[7] M. Dong, K. Ota, M. Lin, Z. Tang, S. Du, and H. Zhu, "Uav-assisted data gathering in wireless sensor networks," *The Journal of Supercomputing*, vol. 70, no. 3, pp. 1142–1155, 2014. [Online]. Available: https://doi.org/10.1007/s11227-014-1161-6

[8] Y. Sun, D. Xu, D. W. K. Ng, L. Dai, and R. Schober, "Optimal 3d-trajectory design and resource allocation for solar-powered uav communication systems," *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4281–4298, 2019.

[9] T. Li, K. Ota, T. Wang, X. Li, Z. Cai, and A. Liu, "Optimizing the coverage via the uavs with lower costs for information-centric internet of things," *IEEE Access*, vol. 7, pp. 15 292–15 309, 2019.

[10] R. Shakeri, M. A. Al-Garadi, A. Badawy, A. Mohamed, T. Khattab, A. K. Al-Ali, K. A. Harras, and M. Guizani, "Design challenges of multi-uav systems in cyber-physical applications: A comprehensive survey and future directions," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3340–3385, 2019.

[11] A. A. Nasir, H. D. Tuan, T. Q. Duong, and H. V. Poor, "Uav-enabled communication using noma," *IEEE Transactions on Communications*, vol. 67, no. 7, pp. 5126–5138, 2019.

[12] Y. Huang, W. Mei, J. Xu, L. Qiu, and R. Zhang, "Cognitive uav communication via joint maneuver and power control," *IEEE Transactions on Communications*, vol. 67, no. 11, pp. 7872–7888, 2019.

[13] Y. Zeng, Q. Wu, and R. Zhang, "Accessing from the sky: A tutorial on uav communications for 5g and beyond," *Proceedings of the IEEE*, vol. 107, no. 12, pp. 2327–2375, 2019.

[14] H. Li, K. Ota, and M. Dong, "Eccn: Orchestration of edge-centric computing and content-centric networking in the 5g radio access network," vol. 25, pp. 88–93, 2018.

[15] ——, "Ls-sdv: Virtual network management in large-scale software-defined iot," vol. 37, pp. 1783–1793, 2019.

[16] Z. Ullah, F. Al-Turjman, and L. Mostarda, "Cognition in uav-aided 5g and beyond communications: A survey," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 3, pp. 872–891, 2020.

[17] N. Nomikos, E. T. Michailidis, P. Trakadas, D. Vouyioukas, H. Karl, J. Martrat, T. Zahariadis, K. Papadopoulos, and S. Voliotis, "A uav-based moving 5g ran for massive connectivity of mobile users and iot devices," *Vehicular Communications*, vol. 25, p. 100250, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2214209620300218

[18] C. Zhang, M. Dong, and K. Ota, "Fine-grained management in 5g: Dql based intelligent resource allocation for network function virtualization in c-ran," vol. 6, pp. 428–435, 2020.

[19] P. S. Bithas, V. Nikolaidis, A. G. Kanatas, and G. K. Karagiannidis, "Uav-to-ground communications: Channel modeling and uav selection," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 5135–5144, 2020.

[20] G. Best, M. Forrai, R. R. Mettu, and R. Fitch, "Planning-aware communication for decentralised multi-robot coordination," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1050–1057.

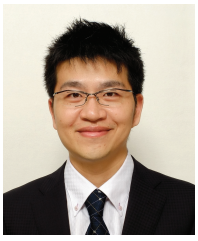
[21] F. Sukkar, G. Best, C. Yoo, and R. Fitch, "Multi-robot region-of-interest reconstruction with dec-mcts," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9101–9107.

[22] Y. E. Sahin, P. Nilsson, and N. Ozay, "Multirobot coordination with counting temporal logics," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1189–1206, 2020.

[23] N. Fung, J. Rogers, C. Nieto, H. I. Christensen, S. Kemna, and G. Sukhatme, "Coordinating multi-robot systems through environment partitioning for adaptive informative sampling," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3231–3237.
[24] J. Alonso-Mora, E. Montijano, T. Nägeli, O. Hilliges, M. Schwager, and D. Rus, "Distributed multi-robot formation control in dynamic environments," *Autonomous Robots*, vol. 43, no. 5, pp. 1079–1100, 2019.
[25] Y. Rizk, M. Awad, and E. W. Tunstel, "Cooperative heterogeneous multi-robot systems: A survey," *ACM Comput. Surv.*, vol. 52, no. 2, Apr. 2019. [Online]. Available: https://doi.org/10.1145/3303848
[26] Xycoon, "Statistics - econometrics - forecasting," Office for Research Development and Education, Mar. 2022. [Online]. Available: https://www.xycoon.com/
[27] F. Yacef, N. Rizoug, O. Bouhali, and M. Hamerlain, "Optimization of energy consumption for quadrotor uav," in *International Micro Air Vehicle Conference and Flight Competition (IMAV) 2017*, 2017, pp. 215–222.
[28] T. Luukkonen, "Modelling and control of quadcopter," *Independent research project in applied mathematics, Espoo*, vol. 22, 2011.
[29] A. Nemati and M. Kumar, "Modeling and control of a single axis tilting quadcopter," in *2014 American Control Conference*, 2014, pp. 3077–3082.
[30] A. A. Khuwaja, Y. Chen, N. Zhao, M. Alouini, and P. Dobbins, "A survey of channel modeling for uav communications," *IEEE Communications Surveys Tutorials*, vol. 20, no. 4, pp. 2804–2821, 2018.
[31] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1046–1061, 2017.
[32] J. L. Volakis, *Antenna Engineering Handbook*, 5th ed. McGraw Hill, Nov. 2018.
[33] S. Zhang, H. Zhang, B. Di, and L. Song, "Cellular uav-to-x communications: Design and optimization for multi-uav networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 2, pp. 1346–1359, 2019.
[34] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of monte carlo tree search methods," *IEEE Transactions on Computational Intelligence and AI in games*, vol. 4, no. 1, pp. 1–43, 2012.
[35] L. Kocsis and C. Szepesvári, "Bandit based monte-carlo planning," in *Machine Learning: ECML 2006*, J. Fürnkranz, T. Scheffer, and M. Spiliopoulou, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 282–293.
[36] L. Kocsis, C. Szepesvári, and J. Willemson, "Improved monte-carlo search," *Univ. Tartu, Estonia, Tech. Rep*, vol. 1, 2006.
[37] P. L. Lanzi, "Evaluating the complexity of players' strategies using mcts iterations," in *2019 IEEE Conference on Games (CoG)*, 2019, pp. 1–8.
[38] "Matrice 200 series," DJI, Jan. 2022. [Online]. Available: https://www.dji.com/matrice-200-series
[39] G. Leishman, *Principles of Helicopter Aerodynamics with CD Extra*, ser. Cambridge aerospace series. Cambridge University Press, 2006. [Online]. Available: https://books.google.co.jp/books?id=nMV-TkaX-9cC
[40] S. Yu, B. Dab, Z. Movahedi, R. Langar, and L. Wang, "A socially-aware hybrid computation offloading framework for multi-access edge computing," *IEEE Transactions on Mobile Computing*, vol. 19, no. 6, pp. 1247–1259, 2020.

Jianwen Xu received the B.Eng degree in Electronic and Information Engineering from Dalian University of Technology in 2014, M.Eng degree in Information and Communication Engineering from Shanghai Jiao Tong University in 2017, and Ph.D. degree in Engineering from Muroran Institute of Technology in 2020. He is currently an assistant professor with the Department of Sciences and Informatics, Muroran Institute of Technology. He received the JSPS Postdoctoral Fellowship for Research in Japan for 2020 Fiscal Year. He was selected as a Non-Japanese Researcher by NEC C&C Foundation for 2019 Fiscal Year. His main fields of research interest include edge computing, Internet of things.

Kaoru Ota received her M.S. degree in computer science from Oklahoma State University in 2008, and her B.S. and Ph.D. degrees in computer science and engineering from the University of Aizu in 2006 and 2012, respectively. She is currently an associate professor with the Department of Sciences and Informatics, Muroran Institute of Technology. She currently serves as an Editor for IEEE Transactions on Vehicular Technology (TVT), IEEE Internet of Things Journal, IEEE Communications Letters, and IEEE Wireless Communications Letters. She is Clarivate Analytics 2019, 2021 Highly Cited Researcher (Web of Science) and selected as JST-PRESTO researcher in 2021.



Mianxiong Dong was born in Shanghai, China. He received B.S., M.S. and Ph.D. in Computer Science and Engineering from The University of Aizu, Japan. He is the Vice President and youngest ever Professor of Muroran Institute of Technology, Japan. He is the recipient of IEEE TCSC Early Career Award 2016, IEEE SCSTC Outstanding Young Researcher Award 2017, The 12th IEEE ComSoc Asia-Pacific Young Researcher Award 2017, Funai Research Award 2018 and NISTEP Researcher 2018 (one of only 11 people in Japan) in recognition of significant contributions in science and technology by MEXT, Japan. He is Clarivate Analytics 2019, 2021 Highly Cited Researcher (Web of Science) and Foreign Fellow of EAJ.