Multi-UAV Trajectory Planning for Energy-Efficient Content Coverage: A Decentralized Learning-Based Approach

Chenxi Zhao, Junyu Liu, *Member, IEEE*, Min Sheng, *Senior Member, IEEE*, Wei Teng, Yang Zheng, and Jiandong Li, *Fellow, IEEE*

***Abstract*— In next-generation wireless networks, high-mobility unmanned aerial vehicles (UAVs) are promising to provide content coverage, where users can receive sufficient requested content within a given time. However, trajectory planning for multiple UAVs to provide content coverage is challenging since 1) UAVs cannot provide content coverage for all users due to the limited energy and caching storage, and 2) the trajectory planning of UAV is coupled with each other. Moreover, the complete information based trajectory planning methods are unusable since UAVs cannot obtain prior information on the rapidly changing environment. In this paper, we investigate the multi-UAV trajectory planning for energy-efficient content coverage. We first formulate an energy efficiency maximization problem considering recharging scheduling, which aims to reduce the total length of trajectories of UAVs under the quality of service (QoS) constraints. To settle environment uncertainty, the trajectory planning problem is modeled as two coupled multi-agent stochastic games, whose equilibrium constitute the optimal trajectory. To obtain the equilibrium, we propose a decentralized reinforcement learning algorithm, which can decouple the two games. We prove that the proposed algorithm can converge to the optimal solution of the Bellman equation with a higher rate compared to the centralized one. Moreover, simulation results show that the energy efficiency of the proposed algorithm is smaller than 5% compared the optimal, which is obtained with the prior information of environments.**

***Index Terms*— UAV networks, multi-UAV trajectory planning, content coverage, decentralized learning, energy efficiency.**

I. Introduction

AERIAL base stations (ABSs) based on unmanned aerial vehicles (UAVs) have become an important approach to provide content coverage for next-generation wireless networks [1]–[3]. Specifically, the content coverage means that users should not only be served with a predefined transmission rate but also be served with sufficient requested contents within a given time. However, multi-UAV trajectory planning is challenging to provide long-term content coverage. First, UAVs fail to provide effective content coverage for all users. The differentiated content coverage service is provided according to caching strategies of UAVs and content requests of users. Thus, it is critical to leverage the differentiated content services provided by UAVs for multi-UAV trajectory planning. Second, the limitation of energy on UAVs may degenerate the performance of content coverage. Hence, high energy-efficient multi-UAV trajectory planning and recharging scheduling are indispensable for long-term content coverage. Third, in practical deployment, the prior environment information including the positions and content requests of users and channel state information (CSI) is unavailable to UAVs. Thus, it is difficult to apply the complete environment information based trajectory planning methods. In this paper, we focus on the decentralized multi-UAV trajectory planning to provide long-term energy-efficient content coverage for users in an unknown environment.

A. Related Work

In practical applications of UAVs, the deployment and trajectory planning of UAVs are two fundamental problems. For the deployment problem, the impact of the altitude of UAV and distances among UAVs on the downlink coverage performance of UAV-enabled small cells is investigated in [4]. In [5], an algorithm for three-dimensional deployment and dynamic movement of UAVs is formulated for maximizing the sum mean opinion score of ground users (GUs). For the deployment problem, the UAV trajectory planning and time scheduling are optimized to guarantee the security of UAV-relayed wireless networks with caching in [6]. Furthermore, the UAV-enabled wireless power transfer system is studied in [7], where the UAV-mounted mobile energy transmitter is dispatched to deliver wireless energy to energy receivers. For the deployment and trajectory planning via optimization technique, the nearly complete information is needed, where the information includes the locations of GUs, the content requests of GUs, and the state information of channels from UAVs to GUs. However, it is hard to collect the complete information of environments in practice due to the following reasons: 1) In typical applications of UAV communication networks, e.g.,

Manuscript received October 23, 2020; revised March 1, 2021; accepted April 12, 2021. Date of publication June 14, 2021; date of current version September 16, 2021. This work was supported in part by the Natural Science Foundation of China under Grant 61931005, Grant 61725103, and Grant U19B2025 and in part by the Young Elite Scientists Sponsorship Program by the China Association for Science and Technology (CAST). *(Corresponding author: Junyu Liu.)*

The authors are with the State Key Laboratory of Integrated Service Networks, Institute of Information Science, Xidian University, Xi'an 710071, China (e-mail: chenxizhao@stu.xidian.edu.cn; yangzheng_1@stu.xidian.edu.cn; junyuliu@xidian.edu.cn; msheng@mail.xidian.edu.cn; tengweitw@foxmail.com; jdli@mail.xidian.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSAC.2021.3088669.

Digital Object Identifier 10.1109/JSAC.2021.3088669

the search for remote areas or emergency communications, it is impractical for UAVs to obtain the prior information about the locations of GUs. In addition, UAVs generally do not have knowledge of the popularity of the cached contents, and, moreover, the content preferences might change across different GUs. 2) Due to the high mobility of UAVs, it is difficult to either exactly obtain CSI in time or explicitly derive CSI by following a specific radio propagation model. The lack of complete information of environment deactivates the conventional algorithms. Therefore, more effective methods should be tailored for the trajectory planning of UAVs in the unknown environment.

In unknown environments, the reinforcement learning (RL) based intelligent methods are proposed to enhance the performance for UAV networks [8]–[10], which can refine the control policies by interacting with environments. In [8], the trajectory planning of UAVs is conducted to coordinate real-time sensing tasks of UAVs over the cellular network. The utilities for UAV networks are maximized by solving the decentralized trajectory planning problem via an enhanced Q-learning algorithm. A flow-level model is proposed in [9], which is used to evaluate the performance of UAV networks, and a RL based approach is used to optimize the UAV trajectory and further maximize flow- and system-level throughput. In [10], the energy efficiency is maximized with joint consideration for communications coverage, fairness, energy consumption and connectivity. In the above literature, researchers design the trajectory within one cruising duration of UAVs. However, long task duration as well as the limited flight duration of the UAV may cause intermittent service. Hence, it is indispensable to design the recharging schedule of UAVs to perform long-term tasks. In [11], a closed chain constructed by multiple rechargeable UAVs is designed to provide seamless coverage and long-term information services for IoT nodes. In [12], an online UAV-assisted wireless caching design is proposed, where a single cache-enabled UAV is used to continuously enhance the wireless network capacity with regular recharging and content update. However, when UAVs provide service to GUs located in a larger region, a single UAV cannot satisfy the quality of service (QoS) of GUs due to the limited coverage range and energy capacity. Although multiple UAVs can be cooperatively deployed to provide service, centralized trajectory planning would result in high computational complexity. Hence, decentralized multi-UAV trajectory planning with consideration of recharging scheduling and energy efficiency is critical to provide service in a larger region.

UAVs are mostly used to complement cellular networks in hotspots or build up a wireless access network for emergency communications. All of these applications rely on the seamless coverage for the target region, including throughput coverage, information coverage, and communication coverage [13]–[15]. In [13], multiple aerial BSs are employed to provide fair coverage for GUs. To serve GUs in a fair pattern, the minimum throughput of GUs is maximized by optimizing the communication scheduling, UAV's trajectory, and power control. In [14], the coverage performance of UAV-aided cellular networks is analyzed, where a novel cooperative UAV clustering scheme is proposed to offload traffic from ground cellular BSs to cooperative UAV clusters. In [15], a deployment algorithm for the UAV airborne network is proposed to provide the on-demand coverage, which can provide service for GUs at temporary events. In most applications of UAVs, there is no enough capability of backhaul link to core networks. Hence, UAVs generally need to precache contents to decrease content delivery delay. In this context, the coverage for GUs means that GUs can receive sufficient desired contents rather than simply meeting the transmission rate requirement, which is called *content coverage*. Different from the communication coverage and throughput coverage, each UAV can only provide effective content coverage for a proportion of GUs according to caching strategies of UAVs and content requests of GUs. Hence, the multi-UAV trajectory planning should take the differentiated content coverage of UAV into account to provide content service for GUs.

As discussed above, decentralized multi-UAV trajectory planning is challenging to provide long-term content coverage in unknown environments. First, the prior environment information including the CSI, positions of GUs and content requests of GUs is unavailable for UAVs in unknown environments, where the information is important to estimate the effectivity of a multi-UAV trajectory planning. Second, due to the limited energy of UAV, it is vital to minimize the total energy cost of UAVs under the QoS constraint and take the recharging scheduling into account to provide long-term content coverage. Third, the content coverage for GUs means that each GU should be effectively served by specified UAVs, which depends on the GU's content requests and UAV's caching strategy. It is critical to efficiently leverage the differentiated content services provided by UAVs for the multi-UAV trajectory planning.

B. Contributions

In this work, we focus on the multi-UAV trajectory planning to provide long-term energy-efficient content coverage. To be specific, a decentralized RL based framework for the multi-UAV trajectory planning is developed. In the proposed framework, each UAV locally learns to conduct the trajectory planning, and shares learning results with others over a time-varying communication network. For clarity, the main contributions of this work are listed as follows.

- We formulate a multi-UAV trajectory planning problem to achieve the long-term energy-efficient content coverage. To combat the environment uncertainty, we model the multi-UAV trajectory planning problem as two coupled multi-agent cooperative stochastic games, which corresponds to the cruising planning and recharging scheduling. The equilibrium of the games constitute the optimal trajectory.
- To obtain equilibrium of the two games above, we propose a full decentralized multi-UAV cooperative RL (DMUCRL) algorithm, where we set two related Q-learning based learners in each UAV to decouple the two games. Besides, we design a novel reward function to ensure QoS constraints with minimal energy cost of UAVs.

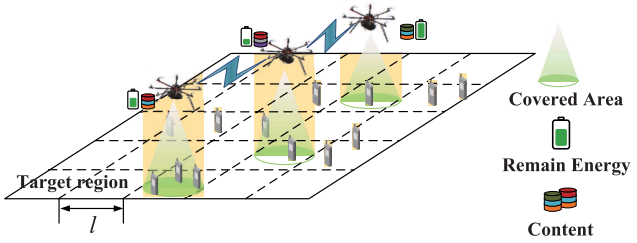


Fig. 1. The UAV network providing content service for GUs in a target region.

• We prove that the proposed DMUCRL algorithm can converge to the same solution compared to the centralized algorithm with a higher convergence rate. Simulation results show that the proposed algorithm can increase energy efficiency by 27.3% compared to the state-of-art straight flight trajectory planning. Furthermore, larger coverage area of UAV can further improve the energy efficiency.

The rest of this paper is organized as follows. In Section II, the system model of cache-enabled UAV network for content coverage is presented. The multi-UAV trajectory planning problem is formulated and a decentralized multi-UAV cooperative RL framework is proposed in Section III. In Section IV, the Q-learning based DMUCRL algorithm for the multi-UAV trajectory planning is given. Simulation results are presented in Section V, which is followed by the conclusions in Section VI.

II. System Model

A. Network Model

Table I lists notation in this paper. We consider a cache-enabled UAV network consisting of M UAVs and K GUs, which are located in a squared area (see Fig. 1). Let $\mathcal{M} = \{1, \ldots, M\}$ and $\mathcal{K} = \{1, \ldots, K\}$ denote the sets of UAVs and GUs, respectively. In the network, UAVs move horizontally at a certain altitude H. Moreover, UAVs deliver contents to GUs in multicast mode, and the azimuth angle of UAV's beam is α. Divide the target region into small squared grids with side length l, and set $l = \sqrt{2}H\tan(\alpha/2)$. Thus, UAVs can cover the entire small grid. Denote $\mathcal{G}$ as the set of small grids. In addition, a UAV can deliver contents to GUs only when they are located in the same grid. We assume that all UAVs operate in a synchronized manner. The time axis is partitioned into equal non-overlapping time slots. At each time slot, UAVs first determine whether returning to charging station. If a UAV does not return to charging station, it will orderly execute three operations as follows, 1) selecting a direction to move along for this time slot, 2) moving from the center of a grid to the center of another adjacent grid, and 3) hovering over the center of grid to deliver contents. The selection-movement-delivery cycle is termed as the unit of the content service process, whose duration is defined as the duration of a time slot (See Fig. 2). We consider that all UAVs move with a constant velocity. Thus, the cruising time equals l/V with V denoting the velocity of UAVs. In addition, the hovering time is set as T_{H}. Since the selection time is greatly smaller than the cruising time and hovering time, the duration of each time slot is given by $(T_{\mathrm{H}} + l/V)$.

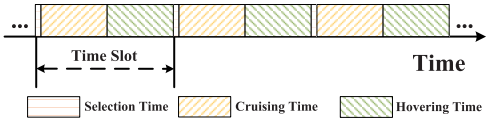


Fig. 2. The structure of time slot.

Let $\boldsymbol{U}_k(t)$ and $\boldsymbol{L}_m(t)$ denote the positions of GU k and UAV m at time slot t, respectively. The distance between GU k and UAV m at time slot t can be written as $D_{k,m}(t) = \sqrt{\|\boldsymbol{U}_k(t) - \boldsymbol{L}_m(t)\|^2 + H^2}$. We assume that the link between UAV and GU is dominated by LoS links. Hence, the channel gain of the link between UAV m and GU k is given by [13], [16]

$$G_{k,m}(t) = \rho_0 \left(D_{k,m}(t)\right)^{-2} = \frac{\rho_0}{\|\boldsymbol{U}_k(t) - \boldsymbol{L}_m(t)\|^2 + H^2}, \tag{1}$$

where ρ_0 denotes the path gain at reference distance. Hence, the signal-to-interference-plus-noise ratio (SINR) of the channel between UAV m and GU k at time slot t can be written as

$$\Gamma_{m,k}(t) = \frac{P_m^{\mathrm{T}}(t)\, G_{k,m}(t)}{I_{k,m}(t) + B_W N_0}, \tag{2}$$

where $P_m^{\mathrm{T}}(t)$ is the transmit power of UAV m at time slot t. $I_{k,m}(t) = \sum_{i \in \mathcal{I}_k(t) \setminus \{m\}} P_i^{\mathrm{T}}(t) G_{k,i}(t)$ is the interference power on UAV m at GU k, where $\mathcal{I}_k(t)$ is the set of UAVs covering GU k at time slot t. N_0 is the power spectral density of noise and B_W is the bandwidth of UAV-GU wireless channel in Hz. Moreover, we consider that UAVs are located in a time-varying communication network. Specifically, the topology of a UAV network may change due to the movement of UAVs, and each UAV can only exchange information with other UAVs located in the communication range.

In this paper, we adopt the content receiving rate as the QoS of GUs. To be specific, the QoS of GU k is satisfied if GU k successfully receives at least N_k^{Q} requested contents within T^{Q} time slots. In the considered UAV network, we assume that the private information related to GUs is not accessible for other UAVs, where the private information of GUs includes the positions and content requests of GUs, the CSI and reward of each UAV. On the contrary, the public information of UAVs is accessible for all UAVs, including positions of UAVs, the historical trajectory and actions of UAVs. This is a commonly-used assumption for the decentralized multi-UAV trajectory planning in the unknown environment.

B. Content Caching and Delivery Model

Let $\mathcal{C} = \{1, 2, \ldots, C\}$ denote the set of contents. For simplicity, we assume that the size of each content equals D bits. Furthermore, denote $\mathbf{p}_m = \{p_{m1}, p_{m1}, \ldots, p_{mC}\}$ as the vector of the caching strategy of UAV m, where $p_{ms} = 1$ denotes that UAV m caches content s, and $p_{ms} = 0$ otherwise.

TABLE I
NOTATION

Symbol	Definition	Parameter	Descriptions
$\mathcal{M}$,$\mathcal{K}$,$\mathcal{C}$,$\mathcal{G}$	Sets of UAVs, ground users (GUs), contents, and grids, respectively.	T_{H}	Hovering time of UAV.
B_W	Bandwidth of UAV-GU wireless channel in Hz	$\boldsymbol{U}_k(t)$,$\boldsymbol{L}_m(t)$	Positions of GU k and UAV m at time slot t.
$D_{k,m}(t)$	Distance between GU k and UAV m at time slot t.	l	Side length of small grids.
$G_{k,m}(t)$	Channel gain of the link between UAV m and GU k at time slot t.	$P_m^{\mathrm{T}}(t)$	Transmit power of UAV m at time slot t.
$\Gamma_{m,k}(t)$	SINR of the channel between UAV m and GU k at time slot t	N_k^{Q}	QoS constraints of GU k.
$z_{k,m}(t)$	UAV m and GU k are located in the same grid at time t or not.	E_m^{T}	Total energy capacity of UAV m.
$E_m(t)$	Remaining energy of UAV m at the end of time slot t.	$c_{ks}(t)$	GU k requests content s at time slot t or not.
$b_m(t)$	UAV m returns to charging station at time slot t or not.	p_{ms}	UAV m caches content s or not.
S_m	Caching storage of UAV m.	T_{C}	Time cost of recharging.

TABLE II
UAV FLIGHT ENERGY MODEL PARAMETERS

Parameter	Descriptions	Parameter	Descriptions
δ_e	Blade drag coefficient	R_{S}	Rotor solidity
Ω_e	Blade angular velocity	g	Gravity acceleration
R_e	Rotor radius	A	Rotor disc area
ρ	Air density	M_{UAV}	UAV mass
κ_p	Induced power factor	V	UAV velocity

Due to the limited UAV caching storage, the cache strategy of UAV m should satisfy

$$\sum_{s\in\mathcal{C}} p_{ms} = S_m, \tag{3}$$

where S_m denotes the caching storage of UAV m. Denote $\mathbf{c}_k(t) = \{c_{k1}(t), c_{k2}(t), \ldots, c_{kC}(t)\}$ as the vector of content requested by GU k at time slot t. In particular, $c_{ks}(t) = 1$ indicates that GU k requests content s at time slot t, and $c_{ks}(t) = 0$ otherwise. UAV m can deliver content s to GU k if UAV m and GU k are located in the same grid and $p_{ms}c_{ks}(t) = 1$. Therefore, the number of contents that can be delivered from UAV m to GU k at time slot t is given by

$$N_{k,m}(t) = z_{k,m}(t)\min\left\{\mathbf{c}_k(t)(\mathbf{p}_m)^{\mathrm{T}}, \left\lfloor\frac{T_{\mathrm{H}}C_{k,m}}{D}\right\rfloor\right\}, \tag{4}$$

where $\lfloor\cdot\rfloor$ denotes the floor function and $C_{k,m} = B_W\log_2(1+\Gamma_{m,k})$ denotes the achievable data rate between GU k and UAV m, respectively. In (4), $z_{k,m}(t) \in \{0,1\}$ denotes whether UAV m and GU k are located in the same grid at time t. Specifically, $z_{k,m}(t) = 1$ indicates that UAV m and GU k are located in the same grid at time slot t, and $z_{k,m}(t) = 0$ otherwise. In addition, $\mathbf{c}_k(t)(\mathbf{p}_m)^{\mathrm{T}}$ is the number of contents, which are cached in UAV m and requested by GU k at time slot t.

C. UAV Energy Model

Denote E_m^{T} and $E_m(t)$ as the total energy capacity of UAV m and the remaining energy of UAV m at the end of time slot t, respectively. The energy consumption of UAVs consists of computing energy, flight energy, and transmission energy, which are elaborated as follows.

- *Computing Energy:* Denote P_{C} as the computation power of all UAVs. Hence, the computing energy consumption of UAVs at time slot t is given by $E_{\mathrm{C}} = P_{\mathrm{C}}(T_{\mathrm{H}} + l/V)$.
- *Flight Energy:* In this paper, we consider a rotary-wing UAV propulsion power model which only depends on the velocity factor

$$P_{\mathrm{F}}(V) = \frac{1}{2}d_0\rho R_{\mathrm{S}}AV^3 + P_0\left(1+\frac{3V^2}{(\Omega_e R_e)^2}\right) + P_1\left(\sqrt{1+\frac{V^4}{4v_0^4}} - \frac{V^2}{2v_0^2}\right)^{\frac{1}{2}} \tag{5}$$

  where $P_0 = \frac{\delta_e}{8}\rho R_{\mathrm{S}}A\Omega_e^3R_e^3$, $P_1 = (1+\kappa_p)\frac{(gM_{\mathrm{UAV}})^{3/2}}{\sqrt{2\rho A}}$, $v_0 = \sqrt{\frac{gM_{\mathrm{UAV}}}{2\rho A}}$, and $d_0 = \frac{S_{\mathrm{F}}}{sA}$, with S_{F} denoting the fuselage equivalent flat plate area [17]. The parameters of the UAV flight energy model are listed in Table II. Thus, the rotary-wing UAV propulsion energy consumption at each time slot can be given by $E_{\mathrm{F}}(V) = P_{\mathrm{F}}(V)\frac{l}{V} + P_{\mathrm{F}}(0)T_{\mathrm{H}}$.
- *Transmission Energy:* When one UAV arrives the center of a grid, the UAV will detect the CSI of all channels from it to GUs in the same grid. According to the CSI, UAV m chooses a transmit power at time slot t satisfying

$$B_W\log_2\left(1+\frac{P_m^{\mathrm{T}}(t)G_{k,m}(t)}{B_WN_0}\right)T_{\mathrm{H}} \geq S_mD, \quad \forall k\in\mathcal{U}_m \tag{6}$$

  where $\mathcal{U}_m$ is the set of GUs located in the same grid with UAV m. Hence, the transmission energy consumption of UAV m at time slot t is given by $E_{\mathrm{T},m}(t) = P_m^{\mathrm{T}}(t)T_{\mathrm{H}}$.

Divide the energy of UAVs into many levels with E^{U} as the minimum energy unit. Thus, the number of energy levels of UAV m equals $\lceil E_m^{\mathrm{T}}/E^{\mathrm{U}}\rceil$, where $\lceil\cdot\rceil$ denotes the ceiling function. We have $E^{\mathrm{U}} = E^{\mathrm{R}} + E^{\mathrm{M}}$, where E^{R} and E^{M} denote the energy consumption that UAV returns to charging station and the maximum energy consumption in each time slot, respectively. Especially, E^{R} is not smaller than the energy consumption that UAV returns to charging station from any position in the target region. In the following, we give the clear explanation for E^{M}. The energy consumption of UAVs consists of computing energy, flight energy, and transmission energy. Since all UAVs move with a constant velocity, the flight energy consumption of each UAV in each time slot is equal. In addition, the computing energy consumption and transmission energy consumption are generally greatly smaller than the flight energy consumption. Therefore, even the computing

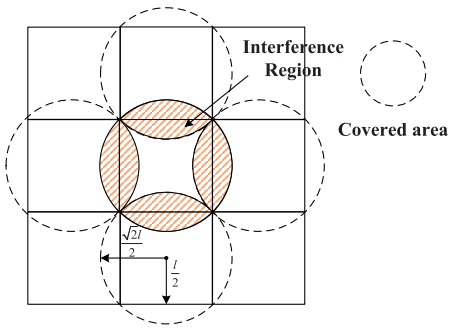


Fig. 3. Illustration of interference region.

energy consumption and transmission energy consumption of each UAV is different, the energy consumption of each UAV in each time slot is approximately same. E^{M} is the sum of flight energy consumption, the maximum computing energy consumption, and the maximum transmission energy consumption. As mentioned above, UAV determines whether returning to charging station at the beginning of each time slot. Thus, to ensure UAV can return to charging station after next determination, the minimum energy unit should satisfy $E^{\mathrm{U}} = E^{\mathrm{R}} + E^{\mathrm{M}}$. Meanwhile, UAV has to return to charging station when the remain energy is smaller than the second-lowest level.

Note that UAV m does not consider the interference from other UAVs when it chooses $P_m^{\mathrm{T}}(t)$. However, there exists interference from other UAVs hovering over the adjacent grids when UAV m delivers contents to GUs. To be specific, the cover range of UAVs is a circle, which is larger than the grid. GUs located at the edge region of grid may be interfered by other UAVs hovering over the adjacent grids, where the edge region is called as interfered region (see Fig. 3). To avoid interference, UAVs should avoid passing through adjacent grids at the same time slot.

As discussion above, the remain energy of UAV m at the end of time slot t is given by

$$E_m(t) = E_m(t-1) - \left(E_{\mathrm{T},m}(t) + E_{\mathrm{C}} + E_{\mathrm{F}}\right), \quad \forall t \geq 1. \tag{7}$$

Moreover, we have $E_m(0) = E_m^{\mathrm{T}}$.

In this paper, the energy efficiency of the considered UAV network is defined as [11]

$$\zeta = \frac{D\sum\limits_{t}\sum\limits_{m\in\mathcal{M}}\sum\limits_{k\in\mathcal{K}} N_{k,m}(t)}{\sum\limits_{t}\sum\limits_{m\in\mathcal{M}}\left(E_m(t-1) - E_m(t)\right)}\,(\text{bit/J}), \tag{8}$$

where the numerator indicates the total transmitted bits and the denominator is the total energy consumption. Note that the energy efficiency in (8) is based on the QoS constraints.

D. UAV Recharging Model

To support continuous content coverage, UAVs should be recharged regularly due to their limited energy. Let $b_m(t)$ denote whether UAV m returns to charging station or not at time slot t. In particular, $b_m(t) = 0$ indicates that UAV m returns to charging station at time slot t, and $b_m(t) = 1$ otherwise. Set the time cost of recharging as T_{C} time slots and all UAVs arrive at a fixed point in the target region when they return from charging station. The position of the fixed point is denoted as $\boldsymbol{L}_{\mathrm{F}}$. Thus, if UAV m returns to charging station at time slot t, we have

$$b_m(\tau) = \begin{cases} 0, & \tau \in [t, t+T_{\mathrm{C}}-1], \text{ and } b_m(t) = 0 \\ 1, & \tau = t+T_{\mathrm{C}}, \text{ and } b_m(t) = 0 \end{cases} \tag{9}$$

$$\boldsymbol{L}_m(t+T_{\mathrm{C}}) = \boldsymbol{L}_{\mathrm{F}}, \tag{10}$$

and

$$E_m(t+T_{\mathrm{C}}) = E_m^{\mathrm{T}}. \tag{11}$$

In addition, the difference of the UAV content caching strategy of UAVs should be taken into account when we design the recharging scheduling of UAVs. To guarantee the QoS of GUs, each content should be cached in at least one UAV hovering over the target region for each time slot. Hence, we have

$$\sum_{m\in\mathcal{H}(t)} p_{ms} b_m(t) \geq 1, \quad \forall s \in \mathcal{S},\ t \geq 0 \tag{12}$$

where $\mathcal{H}(t)$ denotes the set of UAVs hovering over the target region at time slot t.

III. Decentralized Multi-UAV Cooperative RL Framework for Multi-UAV Trajectory Planning

In this section, we first formulate the multi-UAV trajectory planning problem as an energy consumption minimization problem. Then, the statement of the multi-agent cooperative stochastic game is presented, which is used to solve the multi-UAV trajectory planning optimization in the unknown stochastic environment.

The multi-UAV trajectory planning consists of two parts: 1) the schedule of UAVs returning to charging station, and 2) the cruise policy of UAVs at each time slot. In the selection time of each time slot, each UAV first independently decides whether it returns to charging station or not at this time slot. Then, each UAV will independently choose a direction to move along at this time slot if it does not return to charging station. Note that a UAV could return to charging station even if its energy is not smaller than the second lowest energy level. The optimal multi-UAV trajectory planning aims to maximize the energy efficiency under the QoS constraints in a long-term content service process.

A. Problem Formulation

We aim to find a multi-UAV trajectory planning, which can ensure the QoS of each GU with minimal energy consumption.

Denoting $\boldsymbol{L}_m(t) = [x_m(t), y_m(t)]$ as the position of UAV m at the end of time slot t, we have

$$\left|\boldsymbol{L}_m(t) - \boldsymbol{L}_m(t-1)\right|^2 b_m(t) = l^2, \tag{13}$$

$$\left(x_m(t) - x_m(t-1)\right)\left(y_m(t) - y_m(t-1)\right) b_m(t) = 0. \tag{14}$$

Constraints (13) and (14) indicate that UAVs can only move from the center of a grid to the center of another adjacent grid at each time slot if it does not return to charging station.

Particularly, in constraint (13), the distance UAV flown at each time slot is limited as l if the UAV does not return to charging station. In constraint (14), at least one of the abscissa and ordinate of UAV m is not changed, which means that the UAV only has five types of movements, i.e., 1) moving forward, 2) moving backward, 3) moving left, 4) moving right, and 5) staying put. Hence, constraints (13) and (14) can ensure the UAV moves l along x-axis or y-axis in each time slot. Moreover, the start position of UAV m is the center of a grid. Thus, constraints (13) and (14) can ensure that UAVs only move from the center of a grid to the center of another adjacent grid at each time slot.

The multi-UAV trajectory planning problem can be formulated as

$$\begin{aligned}
(\mathrm{P0}) \quad & \min_{b_m(t),\, \boldsymbol{L}_m(t)} \sum_{t\geq 0} \sum_{m\in\mathcal{M}} b_m(t)\left(E_m(t) - E_m(t+1)\right) \\
& s.t.\ (6)-(14) \\
& \sum_{\tau=1}^{T^Q} \sum_{m\in\mathcal{M}} N_{k,m}(t+\tau) \geq N_k^{\mathrm{Q}}, \\
& \forall k \in \mathcal{K},\ t = 0, T^{\mathrm{Q}}, 2T^{\mathrm{Q}} \ldots \qquad (15) \\
& \boldsymbol{L}_m(0) = \boldsymbol{L}_m^0, \quad E_m(0) = E_m^{\mathrm{T}}, \quad \forall m \in \mathcal{M} \qquad (16) \\
& b_m(t) \in \{0,1\}, \quad \forall m \in \mathcal{M},\ t \geq 0. \qquad (17)
\end{aligned}$$

where $\boldsymbol{L}_m^0, \forall m \in \mathcal{M}$ is the start position of UAV m. Constraint (15) indicates that GU k should successfully receive at least N_k^{Q} contents within T^{Q} time slots. Hence, problem (P0) aims to minimize the total energy consumption of UAVs under the QoS constraints.

In the considered UAV network, UAVs have no prior information on the locations and content requests of GUs. Hence, $b_m(t)$ and $N_{k,m}(t)$ in problem (P0) are unknown. The complete information based trajectory planning methods fail to effectively solve problem (P0) under an unknown environment. To this end, we formulate the multi-UAV trajectory planning as two multi-agent cooperative stochastic games in the following.

B. Game Statement

In multi-UAV trajectory planning, UAVs cooperatively conduct the trajectory planning to maximize the global long-term reward. At each time slot, the reward of each UAV relies on the current state of environment and actions of all UAVs. The state transition of the environment is complicated, resulting from a joint action containing actions of all UAVs taking at the time slot. The environment turns into a new stochastic state at the start of each time slot, which is influenced by the previous state and actions of all UAVs taking at the previous time slot. Thus, the transition of state-action satisfies the Markov property. Moreover, there are two characteristics in the considered multi-UAV trajectory planning problem. One is the cooperation among UAVs. UAVs are organized as a team to provide content service to ground users (GUs) in the target region. Hence, the trajectory of UAVs should be cooperatively designed considering the difference of content service of UAVs to achieve high energy efficiency. The other one is the competition among UAVs. The competition is mainly introduced by the lack of a central controller. In decentralized system, each UAV wants to greedily maximize the its long-term reward, where the system performance may be inversely decreased. In this case, the framework of multi-agent Markov game which provides a suitable paradigm to analyze the interrelationship between decision makers, can be naturally applied. The multi-agent Markov game framework applies to the setting with both collaborative and competitive relationships among agents. There are some researches have been made based on the multi-agent Markov game framework [18]–[22]. However, in [18], [19], the reward functions of all agents are set to be identical, which simplifies the problem since the difference of agents are ignored. Moreover, in [20]–[22], researchers ignore the time-varying communication among agents, which may greatly impact the system performance. As mentioned above, in this paper, we consider adopting a multi-agent cooperative stochastic game to model multi-UAV trajectory planning, where the time-varying communication among UAVs is also taken into account.

In view of the discussion above, we define the multi-agent cooperative stochastic game over time-varying UAV communication networks as a tuple $\langle \mathcal{M}, \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \mathcal{G} \rangle$ [23],

- $\mathcal{M}$ is the set of agents.
- $\mathcal{S}$ is the set of environment states. At time slot t, the environment state is denoted as $s(t)$.
- $\mathcal{A} = \{\mathcal{A}_1, \mathcal{A}_2, \ldots \mathcal{A}_M\}$ is the set of joint actions, where $\mathcal{A}_m$ is the set of individual actions of agent m. The joint action at time slot t is denoted as $\boldsymbol{a}(t) \in \mathcal{A}$, while the individual action of agent m is denoted as $a_m(t) \in \mathcal{A}_m$. Hence, the joint action can be written as $\boldsymbol{a}(t) = (a_1(t), \ldots, a_M(t))$.
- $\mathcal{P}$ is the set of state transition probabilities. $P_{ss'}(\boldsymbol{a}(t))$ is the state transition probability from state s to state s' by taking the joint action $\boldsymbol{a}(t) \in \mathcal{A}$.
- $\mathcal{R} = \{\mathcal{R}_1, \ldots \mathcal{R}_M\}$ is the set of reward function, where $\mathcal{R}_m$ is the set of immediate reward of agent m.
- $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_t, \ldots\}$, where $\mathcal{T}_t$ is the undirected topology graph of a time-varying network at time slot t.

In the multi-agent cooperative stochastic game, each UAV individually chooses its own action at each time slot and all UAVs cooperate with other UAVs to achieve the same goal. To accomplish the same goal, all UAVs should reach a consensus in selecting actions [24]. In other words, the selection of joint action should be agreed by all UAVs, which is described as equilibrium.

In this paper, a finite-state Markov decision process (MDP) is introduced to characterize the game process of each UAV. As mentioned above, the transition of state-action satisfies the Markov property. We model the game process in each UAV as a MDP $(\mathcal{S}, \mathcal{A}_m, \mathcal{P}_m, \mathcal{R}_m)$, where $\mathcal{P}_m$ means the matrix of state transition probability of UAV m. Note that each UAV operates in an unknown environment and does not know the reward and transition functions in advance. Hence, we design a decentralized multi-UAV cooperative

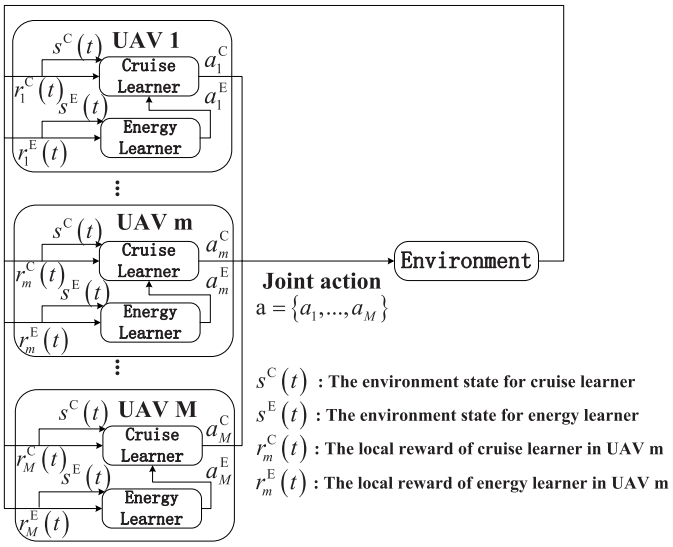


Fig. 4. Illustration of DMUCRL framework for multi-UAV trajectory planning.

RL (DMUCRL) framework to obtain the equilibrium of the multi-agent cooperative stochastic game above.

C. DMUCRL Framework

The proposed DMUCRL framework is characterized by strategic interactions between the UAVs. Specifically, UAVs are heterogeneous individual entities, which can independently choose actions. Moreover, UAVs have the same goal and are influenced by the actions of other UAV. Additionally, in DMUCRL framework, UAVs may not be able to observe the rewards and CSI of other UAVs, and the positions and content requests of GUs.

The key components of the DMUCRL framework for the multi-UAV trajectory planning studied in this paper is given in Fig. 4. In particular, there are two learners in each UAV. The energy learner learns to design the schedule of returning to charging station and the cruise learner learns to choose the direction to move along at each time slot. All energy learners in UAVs constitute a multi-agent cooperative stochastic game $\left\langle \mathcal{M}, \mathcal{S}^{\mathrm{E}}, \mathcal{A}^{\mathrm{E}}, \mathcal{P}_m^{\mathrm{E}}, \mathcal{R}^{\mathrm{E}}, \mathcal{T} \right\rangle$, which is termed as energy control game (ECG) with each energy learner being regarded as an agent. Similarly, all cruise learners in UAVs constitute a multi-agent cooperative stochastic game $\left\langle \mathcal{M}, \mathcal{S}^{\mathrm{N}}, \mathcal{A}^{\mathrm{N}}, \mathcal{P}_m^{\mathrm{N}}, \mathcal{R}^{\mathrm{N}}, \mathcal{T} \right\rangle$, which is termed as cruise control game (CCG) with each cruise learner being regarded as an agent. The joint energy control action and joint cruise control action are denoted by $\mathbf{a}^{\mathrm{E}} = \left(a_1^{\mathrm{E}}, \ldots, a_M^{\mathrm{E}}\right)$ and $\mathbf{a}^{\mathrm{N}} = \left(a_1^{\mathrm{N}}, \ldots, a_M^{\mathrm{N}}\right)$, respectively, where $a_m^{\mathrm{E}} \in \mathcal{A}_m^{\mathrm{E}}$ and $a_m^{\mathrm{N}} \in \mathcal{A}_m^{\mathrm{N}}$ are the energy control action and cruise control action of UAV m, respectively. The immediate reward of UAV m for the two games are denoted by $r_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, s^{\mathrm{E}\prime}\right) \in \mathcal{R}_m^{\mathrm{E}}$ and $r_m^{\mathrm{N}}\left(s^{\mathrm{N}}, \mathbf{a}^{\mathrm{N}}, s^{\mathrm{N}\prime}\right) \in \mathcal{R}_m^{\mathrm{N}}$, respectively. In particular, $r_m^{\mathrm{N}}\left(s^{\mathrm{N}}, \mathbf{a}^{\mathrm{N}}, s^{\mathrm{N}\prime}\right)$ is the reward when cruise learner chooses joint action $\mathbf{a}^{\mathrm{N}}$ in state s^{N} and arrives at state $s^{\mathrm{N}\prime}$. Accordingly, there are two MDPs in each UAV to model the CCG process and ECG process, where the two MDPs are denoted by $\left(\mathcal{S}^{\mathrm{E}}, \mathcal{A}_m^{\mathrm{E}}, \mathcal{P}_m^{\mathrm{E}}, \mathcal{R}_m^{\mathrm{E}}\right)$ and $\left(\mathcal{S}^{\mathrm{N}}, \mathcal{A}_m^{\mathrm{N}}, \mathcal{P}_m^{\mathrm{N}}, \mathcal{R}_m^{\mathrm{N}}\right)$, respectively. As shown in Fig. 4, the output of the energy learner, determining whether UAV returns to charging station or not, is regarded as one of the inputs of the cruise learner. The cruise learner will select a direction that the UAV moves along at each time slot only if the UAV does not return to charging station.

In the next section, we propose a decentralized RL algorithm based on Q-learning to obtain the equilibrium of above two games.

IV. Proposed Decentralized Multi-UAV Cooperative RL Algorithm

As mentioned above, we model the multi-UAV trajectory planning as two coupled multi-agent cooperative stochastic games, i.e., the ECG and CCG. In this section, we first describe the setting for ECG and CCG. Then, the Q-learning based DMUCRL algorithm is proposed for maximizing the expected long-term reward of the considered multi-UAV network, where each learner operates in an unknown stochastic environment and does not know the reward and transition functions in advance. Similarly as the centralized Q-learning, the proposed algorithm is based on a series of state-action. In addition, the one-step reward for each UAV needs to be designed to evaluate the action executed by UAV at a state. The DMUCRL algorithm aims to ensure that each UAV gradually learns the optimal policy based on the series of state-action and one-step reward. Hence, the setting for CCG and ECG focuses on designing the state transition and one-step reward.

A. Learner Setting for CCG

In CCG, we aim to find a multi-UAV trajectory planning composed of actions of all UAVs at each time slot. The multi-UAV trajectory planning should ensure the QoS of each GU and simultaneously maximize energy efficiency. Intuitively, it is reasonable to consider the energy efficiency when the QoS of each GU is satisfied due to the limited energy of UAV. However, setting the maximization of the average energy efficiency as the objective may lead to unfair service, which means that a proportion of GUs are served with high delay [10]. Hence, it is necessary to take the service fairness into account for the multi-UAV trajectory planning.

As mentioned above, the cruise learner setting for CCG can be described as follows.

1) State and Action for Cruise Learner: The environment state for cruise learner consists of the current positions and historical trajectories of UAVs and the time slot index. In particular, at time slot t, the environment state for cruise learner can be denoted as

$$s^{\mathrm{C}}(t) = \left[\left\{ \boldsymbol{G}_g(t) \right\}_{g \in \mathcal{G}}, \left\{ \boldsymbol{L}_m(t) \right\}_{m \in \mathcal{M}}, t \right]. \tag{18}$$

In (18), $\boldsymbol{G}_g(t)$ is a vector including M elements, where the m-th element is a binary variable denoting whether UAV m had hovered over grid g before time slot t. Hence, the number of state space of the cruise learner equals $2^{M(L/l)^2}(L/l)^{2M}T$. Without loss of generality, the set of actions for cruise learner $\mathcal{A}^{\mathrm{C}}$ consists of five actions, i.e., 1) moving forward, 2) moving

backward, 3) moving left, 4) moving right and 5) returning to charging station. Thus, we have $\left|\mathcal{A}_m^{\mathrm{C}}\right| = 5, \forall m \in \mathcal{M}$.

2) Reward for Cruise Learner: Considering the difference of QoS of each GU, the corresponding fairness index of the content coverage at time slot t is given by [25]

$$f(t) = \frac{N_k^{\mathrm{S}} \sum\limits_{k\in\mathcal{K}} \mu_k \beta_k(t) \xi_k(t)}{\sum\limits_{k\in\mathcal{K}} \mu_k N_k^{\mathrm{S}}}. \tag{19}$$

In (19), $\beta_k(t) = N_k(t) / \sum_{i\in\mathcal{K}} N_i(t)$ is the proportion of the number of contents received by GU k, where $N_k(t)$ denotes the number of contents received by GU k at time slot t. μ_k is the normalization constant to ensure that the maximum value of the weighted average fairness is unity. We set $\mu_k = \left(1 + 1/\ln\left(b_1/\sum_{i\in\mathcal{K}} b_i\right)\right) / \left(1 + 1/\ln\left(b_k/\sum_{i\in\mathcal{K}} b_i\right)\right)$, where $b_k = N_k^{\mathrm{S}} / \sum_{i\in\mathcal{K}} N_i^{\mathrm{S}}, \forall k \in \mathcal{K}$. Furthermore, in (19), the self-fairness of GU k is defined by $\xi_k(t) = \log \beta_k(t) / \log b_k$, which is a relative factor of the fairness. Note that the maximum value of $f(t)$ is achieved when all GUs obtain exactly their fair share of the resources according to their QoS, i.e., $\beta_k(t) = b_k$. The minimum value of $f(t)$ occurs when one GU obtains all resources while other $(K-1)$ GUs are starved.

Denoting $r_{k,m}(t)$ as the number of contents received by GU k from UAV m at time slot t, we can obtain

$$r_{k,m}(t) = \min\left(N_{k,m}(t), N_k^{\mathrm{S}} - \sum_{\tau=0}^{t-1} N_k(\tau)\right) \tag{20}$$

Accordingly, the immediate reward of the cruise learner in UAV m at time slot t is given by

$$r_m^{\mathrm{C}}(t) = \begin{cases} \frac{f(t) \sum\limits_{k\in\mathcal{K}} \frac{z_{k,m}(t)}{\sum\limits_{m\in\mathcal{M}} z_{k,m}(t)} r_{k,m}(t)}{E_m(t-1) - E_m(t)}, & b_m(t) = 1 \\ 0, & b_m(t) = 0 \end{cases} \tag{21}$$

where the numerator indicates the number of contents provided by UAV m at time slot t, and the denominator indicates that the energy consumption of UAV m at time slot t. Furthermore, $z_{k,m}(t) / \sum_{n\in\mathcal{M}} z_{k,n}(t)$ means that UAVs will equally divide the total reward if there are many UAVs located in the same grid at time slot t. Hence, $r_m^{\mathrm{C}}(t)$ gives the energy efficiency of UAV m at time slot t in bit/J, which is discounted due to the service fairness. The reward function in (21) can guide cruise learners to provide more content service for GUs received fewer contents.

3) Policy for Cruise Learner: The policy $\pi_m^{\mathrm{C}} : \mathcal{S}^{\mathrm{C}} \to \mathcal{A}_m^{\mathrm{C}}$ of the cruise learner in UAV m, denoting a mapping from the state set to the action set, is a probability distribution of the available actions $a_m^{\mathrm{C}} \in \mathcal{A}_m^{\mathrm{C}}$ in a given state s^{C}. In particular, for UAV m in state s^{C}, the cruise policy of the cruise learner in UAV m is denoted as $\pi_m^{\mathrm{C}}\left(s^{\mathrm{C}}\right) = \left\{\pi_m^{\mathrm{C}}\left(s^{\mathrm{C}}, a_m^{\mathrm{C}}\right) \mid a_m^{\mathrm{C}} \in \mathcal{A}_m^{\mathrm{C}}\right\}$, where each element $\pi_m^{\mathrm{C}}\left(s^{\mathrm{C}}, a_m^{\mathrm{C}}\right)$ is the probability of UAV m choosing action a_m^{C} in state s^{C} and we have $\pi_m^{\mathrm{C}}\left(s^{\mathrm{C}}, a_m^{\mathrm{C}}\right) \in [0,1]$. Accordingly, a joint cruise policy can be denoted by $\pi^{\mathrm{C}}\left(s^{\mathrm{C}}\right) = \left\{\pi_1^{\mathrm{C}}\left(s^{\mathrm{C}}\right), \ldots, \pi_M^{\mathrm{C}}\left(s^{\mathrm{C}}\right) \mid s^{\mathrm{C}} \in \mathcal{S}^{\mathrm{C}}\right\}$. Let $\pi_{-m}^{\mathrm{C}}\left(s^{\mathrm{C}}\right) = \left\{\pi_1^{\mathrm{C}}\left(s^{\mathrm{C}}\right), \ldots, \pi_{m-1}^{\mathrm{C}}\left(s^{\mathrm{C}}\right), \pi_{m+1}^{\mathrm{C}}\left(s^{\mathrm{C}}\right), \ldots, \pi_M^{\mathrm{C}}\left(s^{\mathrm{C}}\right)\right\}$ denote the same strategy profile but without the strategy π_m^{C} of UAV m. Similarly, we define the joint cruise action as a vector $\mathbf{a}^{\mathrm{C}} = \left\{a_1^{\mathrm{C}}, \ldots, a_M^{\mathrm{C}}\right\}$, where each element a_m^{C} is the action of the cruise learner in UAV m. Let $\mathbf{a}_{-m}^{\mathrm{C}} = \left\{a_1^{\mathrm{C}}, \ldots, a_{m-1}^{\mathrm{C}}, a_{m+1}^{\mathrm{C}}, \ldots, a_M^{\mathrm{C}}\right\}$ denote the same action profile but without the action a_m^{N}. Hence, the joint cruise policy can be written as

$$\pi^{\mathrm{C}} = \left\{\pi^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) \mid s^{\mathrm{C}} \in \mathcal{S}^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}} \in \mathcal{A}^{\mathrm{C}}\right\}. \tag{22}$$

Since cruise learners individually choose their actions at each time slot, it is reasonable to assume that chosen actions of all cruise learners are independent given in the current environment state [24]. Hence, the joint cruise policy can be defined as $\pi^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) = \prod_{m\in\mathcal{M}} \pi_m^{\mathrm{C}}\left(s^{\mathrm{C}}, a_m^{\mathrm{C}}\right)$, which is the probability of cruise learners choosing joint action $\mathbf{a}^{\mathrm{C}}$ in state s^{C}.

According to the notation above, we give the definition of action value function, i.e., Q function, for the cruise learner. Specifically, Q function of the cruise learner in UAV m is the expected reward by executing action a_m^{C} in state s^{C} under a given policy π_m^{C}, which is given by

$$Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, \pi_m^{\mathrm{C}}\right) = \mathbb{E}\left(\sum_{\tau=0}^{\infty} \gamma^{\tau} r_m^{\mathrm{C}}(t+\tau+1) \mid s^{\mathrm{C}}(t) = s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}(t) = \mathbf{a}^{\mathrm{C}}, \pi_m^{\mathrm{C}}\right) \tag{23}$$

The values of (23) are called action values, i.e., Q values. In (23), we consider the long-term reward of UAV. Specifically, the long-term reward is the sum of immediate reward at the current time slot, and the future rewards that discounted by a constant discounted factor γ with $\gamma \in [0,1]$. Besides, $r_m^{\mathrm{C}}(t+\tau+1)$ is the immediate reward of the cruise learner in UAV m at time slot $(t+\tau+1)$. The value of γ reflects the effect of future rewards on the optimal decisions. If γ is close to 0, it means that the decision emphasizes the near-term gain. On the contrary, if γ is close to 1, it gives more weights to future rewards and the decisions are farsighted. In addition, it can be obtained that the Bellman optimality equation for Q value is given by

$$Q_m^{\mathrm{C}*}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) = \sum_{\mathbf{a}_{-m}^{\mathrm{C}}} \prod_{n\in\mathcal{M}\setminus\{m\}} \pi_n^{\mathrm{C}}\left(s^{\mathrm{C}}, a_n^{\mathrm{C}}\right) \sum_{s^{\mathrm{C}\prime}} P_{s^{\mathrm{C}} s^{\mathrm{C}\prime}}\left(\mathbf{a}^{\mathrm{C}}\right) \times \left[r_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, s^{\mathrm{C}\prime}\right) + \gamma \max_{\mathbf{a}^{\mathrm{C}\prime}\in\mathcal{A}^{\mathrm{C}}} Q_m^{\mathrm{C}*}\left(s^{\mathrm{C}\prime}, \mathbf{a}^{\mathrm{C}\prime}\right)\right] \tag{24}$$

In the selection time of each time slot, UAVs select actions according to their local Q functions. The action selection strategy should strike a balance between exploration and exploitation. The UAV could reinforce the evaluation it already knows to be good but also explore new actions. In this work, we consider an ϵ-greedy exploration strategy for the cruise learner. Specifically, the cruise learner in UAV m selects a random action $a_m^{\mathrm{C}} \in \mathcal{A}_m^{\mathrm{C}}$ in state s^{C} with probability ϵ and selects the best action $a_m^{\mathrm{C}*}$ with probability $(1-\epsilon)$, where the best action satisfies $Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}*}, \pi_m^{\mathrm{C}}\right) \geq Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, \pi_m^{\mathrm{C}}\right), \forall \mathbf{a}^{\mathrm{C}} \in \mathcal{A}^{\mathrm{C}}$ with $a_m^{\mathrm{C}*}$ being the m-th element of $\mathbf{a}^{\mathrm{C}*}$. Furthermore, as mentioned in Section III-C, the action selection for cruise learner is impacted by the one for energy learner. Particularly, if the energy learner in UAV m selects the action returning to

charging station, the cruise learner will not choose any action in set $\mathcal{A}_m^{\mathrm{C}}$. Otherwise, the cruise learner in UAV m will select an action according to the ϵ-greedy exploration strategy above. Hence, the probability of selecting action $a_m^{\mathrm{C}} \in \mathcal{A}_m^{\mathrm{C}}$ in state s^{C} is given by

$$\pi_m^{\mathrm{C}}\left(s^{\mathrm{C}}, a_m^{\mathrm{C}}\right) = \begin{cases} 0, & \text{if UAV } m \text{ returns} \\ 1-\epsilon, & \text{if UAV } m \text{ does not return and} \\ & Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \cdot, \cdot\right) \text{ of } a_m^{\mathrm{C}} \text{ is the highest} \\ \epsilon, & \text{otherwise} \end{cases} \tag{25}$$

where $\epsilon \in (0,1)$. In addition, if there are multiple actions corresponding to the highest Q value, the cruise learner will randomly choose an action from them.

B. Learner Setting for ECG

In ECG, we aim to design a multi-UAV schedule to control them return to charging station under the content coverage continuity, which means that the QoS of each GU should always be met in the whole process. Intuitively, more UAVs hovering over the target region can provide better content coverage for GUs. However, the content coverage continuity may not be guaranteed if all UAVs hover over the target region until the energy of them is exhausted. In addition, the difference among the content caching strategies of UAVs should be taken into account when we design the schedule of UAVs.

1) State and Action for Energy Learner: The environment state for energy learner consists of current state of each UAV (the current remain energy and position). In particular, the environment state at time slot t for energy learner is denoted by

$$s^{\mathrm{E}}(t) = \left[\left\{E_m(t)\right\}_{m\in\mathcal{M}}, \left\{b_m(t)\right\}_{m\in\mathcal{M}}\right]. \tag{26}$$

Hence, we have $\left|\mathcal{S}^{\mathrm{E}}\right| = \prod_{m\in\mathcal{M}}\left(\left\lceil E_m^{\mathrm{T}}/E^{\mathrm{U}}\right\rceil + 1\right)$. Moreover, the set of actions for energy learner consists of two actions, i.e., returning to charging station or not.

2) Reward for Energy Learner: Considering constraint (12), we define the one-step reward of energy learners in UAVs as

$$r_m^{\mathrm{E}}(t) = \begin{cases} -10, & \text{constraints violation} \\ b_m(t), & \text{otherwise.} \end{cases} \tag{27}$$

Since $b_m(t) \in \{0,1\}$, the reward of energy learner in the UAV that do not return to charging station equals one when constraint (12) is satisfied. On the contrary, the reward of energy learner in the UAV that returns to charging station equals 0, while the reward of each energy learner equals -10 if constraint (12) is violated. Hence, the reward in (27) could allow UAVs to hover over the target region as much as possible and concurrently avoid constraints violation.

3) Policy for Energy Learner: The joint energy policy and joint energy action for CCG are denoted by $\pi^{\mathrm{E}}\left(s^{\mathrm{E}}\right) = \left\{\pi_1^{\mathrm{E}}\left(s^{\mathrm{E}}\right), \ldots, \pi_M^{\mathrm{E}}\left(s^{\mathrm{E}}\right) \mid s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}\right\}$ and $\mathbf{a}^{\mathrm{E}} = \left\{a_1^{\mathrm{E}}, \ldots, a_M^{\mathrm{E}}\right\}$, respectively. Hence, the joint energy policy of UAVs can be written as

$$\pi^{\mathrm{E}} = \left\{\pi^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) \mid s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{\mathrm{E}}\right\}. \tag{30}$$

In addition, the Q function for energy learner in UAV m is the expected reward by executing action a_m^{E} in state s^{E} under a given energy policy π_m^{E}, which is given by

$$Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, \pi_m^{\mathrm{E}}\right) = \mathbb{E}\left(\sum_{\tau=0}^{\infty}\gamma^{\tau} r_m^{\mathrm{E}}(t+\tau+1) \mid s^{\mathrm{E}}(t) = s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}(t) = \mathbf{a}^{\mathrm{E}}, \pi_m^{\mathrm{E}}\right). \tag{31}$$

Similarly as the cruise learner, we consider an ϵ-greedy exploration strategy in the action selection for the energy learner. Specifically, the energy learner in UAV m selects a random action $a_m^{\mathrm{E}} \in \mathcal{A}_m^{\mathrm{E}}$ with probability ϵ and selects the best action $a_m^{\mathrm{E}*}$ with probability $(1-\epsilon)$, where the best action satisfies $Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}*}, \pi_m^{\mathrm{E}}\right) \geq Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, \pi_m^{\mathrm{E}}\right), \forall \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{\mathrm{E}}$ with $a_m^{\mathrm{E}*}$ being the m-th element of $\mathbf{a}^{\mathrm{E}*}$.

Hence, the probability of choosing action a_m^{E} in state s^{E} is given by

$$\pi_m^{\mathrm{E}}\left(s^{\mathrm{E}}, a_m^{\mathrm{E}}\right) = \begin{cases} 1-\epsilon, & \text{if } Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \cdot, \cdot\right) \text{ of } a_m^{\mathrm{E}} \text{is the highest} \\ \epsilon, & \text{otherwise} \end{cases} \tag{32}$$

where $\epsilon \in (0,1)$. In addition, if there are multiple actions corresponding to the highest Q value, the energy learner will randomly choose an action from them.

C. Q-Learning Based DMUCRL Algorithm for Multi-UAV Networks

In the following, we describe the Q-learning based DMUCRL algorithm. Each UAV runs two standard Q-learning algorithms to learn the optimal Q values of each state-action pair, and obtain the optimal local policies for the energy control and cruise control. Moreover, each UAV can share its local Q values with its neighbors over the time-varying communication network to reach a consensual estimate of Q values. In the Q value update step, both the energy learner and cruise learner in each UAV follow the update rules as (28) and (29), as shown at the bottom of the next page.

In (28) and (29), $s^{\mathrm{N}\prime}$ is the next environment state for CCG if cruise learners execute joint action $\mathbf{a}^{\mathrm{N}}$ in state s^{N}. Similarly, $s^{\mathrm{E}\prime}$ is the next environment state for ECG if energy learners execute joint action $\mathbf{a}^{\mathrm{E}}$ in state s^{E}. Furthermore, $\alpha^t(s,\mathbf{a})$ and $\beta^t(s,\mathbf{a})$ are learning rates, which are given by

$$\alpha^t(s,\mathbf{a}) = \begin{cases} \frac{A}{(N(s,\mathbf{a})+1)^{\delta_1}}, & s(t) = s, \mathbf{a}(t) = \mathbf{a} \\ 0, & \text{otherwise,} \end{cases} \tag{33}$$

$$\beta^t(s,\mathbf{a}) = \begin{cases} \frac{B}{(N(s,\mathbf{a})+1)^{\delta_2}}, & s(t) = s, \mathbf{a}(t) = \mathbf{a} \\ 0, & \text{otherwise,} \end{cases} \tag{34}$$

where $N(s,\mathbf{a})$ is the number of occurrences of state-action pair $(s,\mathbf{a})$ at time slot t with $s \in \mathcal{S}^{\mathrm{C}} \cup \mathcal{S}^{\mathrm{E}}$ and $\mathbf{a} \in \mathcal{A}^{\mathrm{C}} \cup \mathcal{A}^{\mathrm{E}}$. (33) and (34) indicate that the Q value of state-action pair $(s,\mathbf{a})$ in each UAV is updated if and only if the joint action $\mathbf{a}$ occurs in state s; otherwise, it remains unchanged. As discussed above, the Q-learning based DMUCRL algorithm for each UAV is written in Algorithm 1.

Algorithm 1 The Q-Learning Based DMUCRL Algorithm

Initialization:
Set the parameters A, B, δ_1, δ_2 and $N(s,\mathbf{a})=0$.
for all $m \in \mathcal{M}$ **do**
  Initialize the action-value $Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, a_m^{\mathrm{E}}\right) = 0, \forall s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, a_m^{\mathrm{E}} \in \mathcal{A}_m^{\mathrm{E}}$ and $Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, a_m^{\mathrm{C}}\right) = 0, \forall s^{\mathrm{C}} \in \mathcal{S}^{\mathrm{C}}, a_m^{\mathrm{C}} \in \mathcal{A}_m^{\mathrm{C}}$.
end for
Set the maximal iteration counter $LOOP$ and $loop = 0$.
repeat
  Set $s = s_0$ and $t = 0$.
  for all $m \in \mathcal{M}$ **do**
    Send Q_m^{E} and Q_m^{C} to the neighbors $\{n \in \mathcal{C}_m^t\}$.
  end for
  while $t < T$ **do**
    Observe state s.
    for all $m \in \mathcal{M}$ **do**
      UAV m selects a_m^{E} and a_m^{C} according to $\pi_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \cdot\right)$ and $\pi_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \cdot\right)$, respectively.
    end for
    Obtain the joint actions $\mathbf{a}^{\mathrm{E}}$ and $\mathbf{a}^{\mathrm{C}}$, and the rewards $r_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$ and $r_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right)$.
    Set $N\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) = N\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) + 1$ and $N\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) = N\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) + 1$.
    Update $\alpha^t(s,\mathbf{a})$, $\beta^t(s,\mathbf{a})$, $Q_m^{\mathrm{E}}\left(^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$ and $Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right)$.
    Send Q_m^{E} and Q_m^{C} to the neighbors $\{n \in \mathcal{C}_m^t\}$.
    Set $t = t + 1$.
  end while
  Set $loop = loop + 1$.
until $loop > LOOP$

1) The Convergence of Algorithm 1: In the following, we give the proof of Algorithm 1.

Theorem 1: Let $\left\{Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t+1\right)\right\}$ *be the successive iteration of* $Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$ *obtained at UAV* m *according to (28). We have* $\mathbb{P}\left(\lim_{t\to\infty} Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) = Q^{\mathrm{E}*}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)\right) = 1, \forall m \in \mathcal{M}, s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{E}$.

Proof: Please refer to Appendix A. □

Similarly, for all cruise learners, we can obtain $\mathbb{P}\left(\lim_{t\to\infty} Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) = Q^{\mathrm{C}*}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right)\right) = 1, \forall m \in \mathcal{M}, s^{\mathrm{C}} \in \mathcal{S}^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}} \in \mathcal{A}^{\mathrm{C}}$.

2) Complexity of the Proposed Trajectory Planning Algorithm: We analyze the complexity of the proposed algorithm from two aspects, i.e., the control complexity and convergence rate. First, the control complexity indicates the size of information exchanged among UAVs to achieve consistent control strategy. The control complexity is critical especially when UAVs provide service to dense ground users (GUs) in a large region. In the proposed reinforcement learning framework, the information exchanged among UAVs consists of the local learning result of each UAV, whose size is determined by the size of state space and action space of learners. As mentioned above, the size of the state space and action space of cruise learner is impacted by the number of UAVs and the number of small squared grids. Moreover, the size of the state space and action space of energy learner is impacted by the number of energy levels of UAVs. Therefore, for given UAV communication network, the control complex of the proposed algorithm can keep constant as the increase of the density of GUs in the target region. Furthermore, we can change the method of dividing small grids to decrease the control complexity of the proposed algorithm when the target region is large. As discussed above, the control complexity of the proposed algorithm can be kept in a low level even if the contents are requested by densely deployed ground users located in large areas. This is important for the practical application of UAV communication networks. For the convergence rate, we compare the convergence rate of the proposed algorithm and the central algorithm, and the simulation results is shown in Fig. R3-4. As we can see, the proposed algorithm can converge to the same solution with a higher rate.

Note that although the deep Q network (DQN) and deep deterministic policy gradient (DDPG) based algorithm are strong in practice, the theoretic analysis of their mechanism, especially convergence, is not sufficient. However, in UAV communication networks, the mobility of UAVs results in the high-dynamic characteristic of the environment, which needs the stable convergence of the algorithm. Moreover, UAVs can be used in a variety of scenarios, which have different environment characteristics. In this case, the parameters of DQN

$$Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t+1\right) = Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right) - \beta^t\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) \sum_{n\in\mathcal{N}_m^t} \left(Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right) - Q_n^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right)\right) + \alpha^t\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)\left(r_m\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) + \gamma \max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_m^{\mathrm{E}}\left(s^{\mathrm{E}\prime}, \mathbf{a}^{\mathrm{E}\prime}, t\right) - Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right)\right) \tag{28}$$

$$Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, t+1\right) = Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, t\right) + b_m(t)\left(-\beta^t\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) \sum_{n\in\mathcal{N}_m^t} \left(Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, t\right) - Q_n^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, t\right)\right) + \alpha^t\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right)\left(r_m\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}\right) + \gamma \max_{\mathbf{a}^{\mathrm{C}\prime}\in\mathcal{A}^{\mathrm{C}}} Q_m^{\mathrm{C}}\left(s^{\mathrm{C}\prime}, \mathbf{a}^{\mathrm{C}\prime}, t\right) - Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, t\right)\right)\right). \tag{29}$$

where $Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right)$ and $Q_m^{\mathrm{C}}\left(s^{\mathrm{C}}, \mathbf{a}^{\mathrm{C}}, t\right)$ are Q values for ECG and CCG, respectively. In addition, $\mathcal{N}_m^t$ is the set of neighbors of UAV m at time slot t.

and DDPG based algorithms should be carefully designed to adopt the different environments. Inversely, the proposed RL framework has great universality, which can be directly used in similar scenarios. Furthermore, in this paper, we use the contraction properties of random time-varying graph and stochastic approximation to prove that not only the trajectory planning strategy in each UAV can converge to the local optimal strategy but also the local optimal strategies in all UAV are same.

V. Simulation Results

For our simulations, we consider a 1000 m $\times$ 1000 m square target region and 20 GUs randomly located in the target region. The pathloss exponent and the noise power spectral density are set as $\alpha = 2$ and $N_0 = -174$ dBm/Hz, respectively [26]. The bandwidth of UAV-GU wireless channel is set as $B_W = 2$ MHz. The size of each content is set as $D = 2$ Mbits and the total number of contents equals $C = 30$. The caching storage of each UAV is set as $S_m = 20, \forall m \in \mathcal{M}$. The static power consumption is set as $P_{\mathrm{C}} = 20$ W. For the UAV's propulsion power consumption, the setting of corresponding parameters follows the Table I in [17].

In Fig. 5, we set the altitude of UAVs and the azimuth angle of UAV's beam as $H = 120$ m and $\alpha = 60°$, respectively. Hence, the target region can be approximately divided into 100 m $\times$ 100 m small area. In addition, the velocity and the transmit power of UAV are set as $V = 25$ m/s and $P_m^{\mathrm{T}} = 20$ dBm, $\forall m \in \mathcal{M}$, respectively. Fig. 5 portrays the energy efficiency per episode versus the number of episodes by using the centralized algorithm and the proposed decentralized algorithm, respectively. To be specific, in the centralized algorithm, the central controller that may be mounted on a UAV or satellite chooses actions for all UAVs in each time slot and sends the chosen action to each UAV. UAVs execute actions and obtain rewards, where rewards of UAVs are send to the central controller. Then, the central controller update the Q value according to rewards of UAVs. It can be seen that compared to the centralized algorithm, the proposed decentralized algorithm converges to the same energy efficiency with higher rate. The reason is that UAVs in the decentralized algorithm can simultaneously explore multiple state-action pairs and independently learn to update the Q value. Then, each UAV can share information about the learned Q value with others over the communication network, which means that the UAV team simultaneously learns to update the Q value from various state-action pairs. Hence, the decentralized algorithm has a higher convergence rate compared to the centralized algorithm. Moreover, to verify the optimality of the proposed algorithm, we compare the proposed trajectory planning algorithm with the optimal trajectory planning algorithm. In particular, for OOTP, the trajectory of UAVs is obtained under the prior information of environments. The simulation results show that the energy efficiency gap between the optimal algorithm and the proposed algorithm is smaller than 5%.

For the context of Fig. 6 and Fig. 7, the number of UAVs, the altitude of UAVs, and the azimuth angle of UAV's beam are set as $M = 3$, $H = 120$ m, and $\alpha = 60°$, respectively.

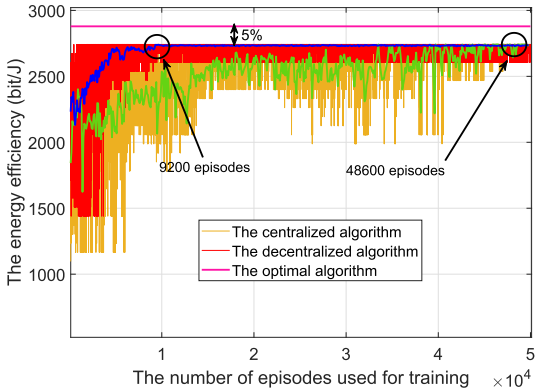


Fig. 5. The energy efficiency of UAVs per episode versus the number of episodes.

Thus, the target region can be approximately divided into 100 m $\times$ 100 m small area. We set $V = 25$ m/s and $P_m^{\mathrm{T}} = 20$ dBm, $\forall m \in \mathcal{M}$. Hence, the hovering time and the duration of each time slot are $T_{\mathrm{H}} = 1$ s and $T_{\mathrm{H}} + l/V = 3$ s. In addition, we set $T_{\mathrm{C}} = 3$ time slots. To provide long-term content coverage, there is at least one UAV not returning to charging station for each time slot in the content coverage process. The service time is defined as the time that all UAV simultaneously return to charging station i.e., content coverage termination. We consider that UAVs need to provide content coverage for GUs with 50 time slots. Fig. 6 portrays the iterative performance of energy learners by using the proposed decentralized algorithm and the centralized algorithm. It can be seen that the proposed decentralized algorithm can converge more quickly than the centralized algorithm. In addition, with the increasing number of iterations, the service time of the UAV team rapidly increases. Moreover, in Fig. 7, we set the total energy of each UAV and the minimum energy unit as $E_m^{\mathrm{T}} = 13500$ J and $E^{\mathrm{U}} = 1350$ J. Hence, the number of energy levels of each UAV equals $\lfloor E_m^{\mathrm{T}}/E^{\mathrm{U}} \rfloor = 10$. In addition, we assume that each UAV returns to charging station need to consume energy 2000J and the remain energy of each UAV is full after it returns to charging station. As we can see in Fig. 7(a), UAV1, UAV2, and UAV3 alternately return to charging station to guarantee the long-term content coverage to GUs. Moreover, in Fig. 7(b), the point at a time slot indicates that the UAV does not return to charging station at this time slot. It is shown that the number of UAVs returning simultaneously is not larger than 1 during the whole content coverage process.

Fig. 8 shows the impact of UAV velocity on the energy efficiency. Similarly, we divide the target region into 100 m $\times$ 100 m small area. We compare DMUCRL algorithm with three benchmarks, i.e., the optimal offline trajectory planning (OOTP), the throughput maximization trajectory planning (TMTP), and straight flight trajectory planning (SFTP). In particular, for OOTP, the trajectory of UAVs is obtained under the prior information of environments. Moreover, for TMTP, the UAV trajectory and power control are optimized by maximizing the overall throughput. For SFTP, UAVs moves sequentially to each hotspot and returns to charging station when their energy is exhausted. As we can see, the energy

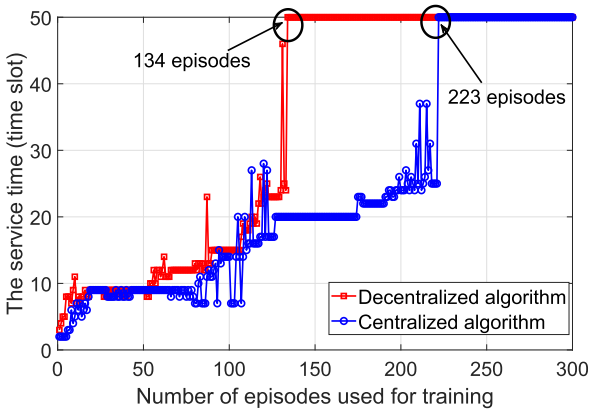
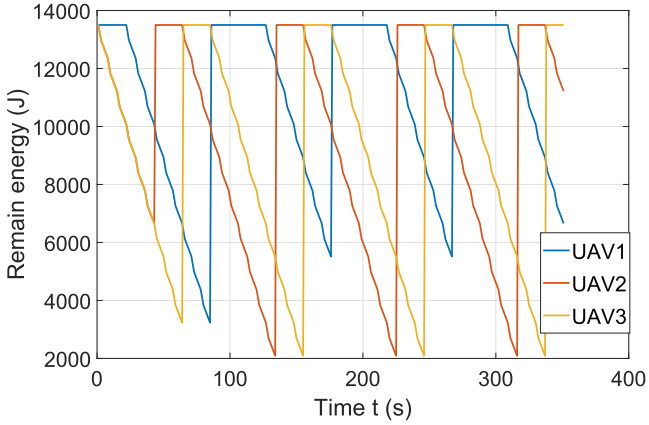
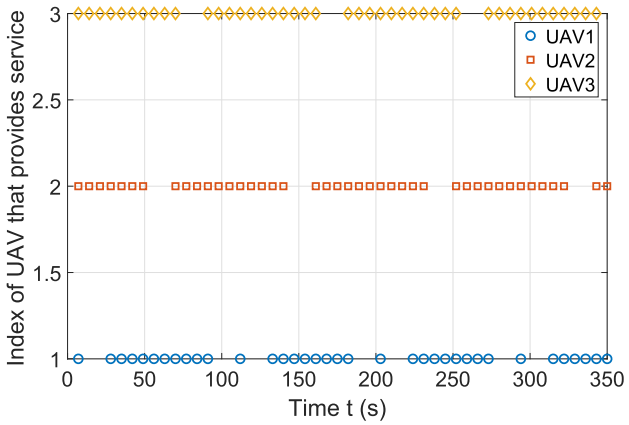


Fig. 6. The service time of the UAV team per episode versus the number of episodes.



(a) The remain energy of each UAV.



(b) The schedule of the UAV team.

Fig. 7. The remain energy of each UAV and the schedule of the UAV team with setting $E_m^{\mathrm{T}} = 13500\mathrm{J}$ and $E^{\mathrm{U}} = 1350\mathrm{J}$.

efficiency gap between the optimal algorithm and the proposed algorithm is smaller than 5%. In addition, the proposed algorithm always outperforms other two compared algorithms. Moreover, the energy efficiency of UAVs increases at first and then decreases as the increase of the UAV velocity. Intuitively, higher UAV velocity can reduce the total time of completing task, which decreases the UAV static energy cost, e.g., computation energy cost, thus increasing energy efficiency. However, the propulsion power consumption of UAVs may increase with

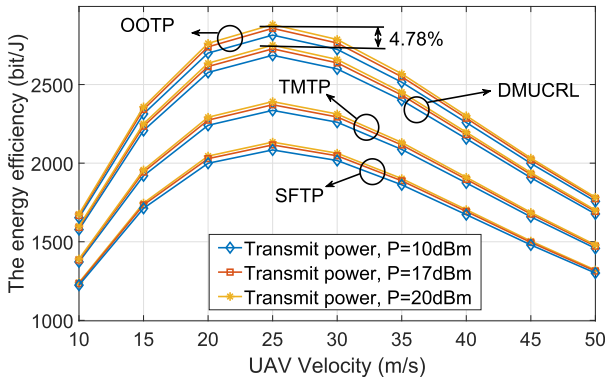


Fig. 8. The energy efficiency of UAVs versus the diameter of UAV velocity.

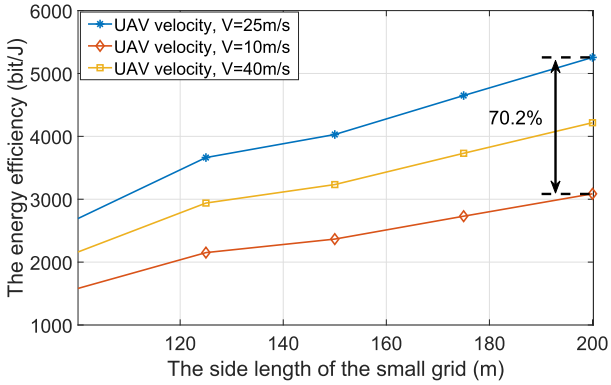


Fig. 9. The energy efficiency of UAVs versus the side length of the small grid.

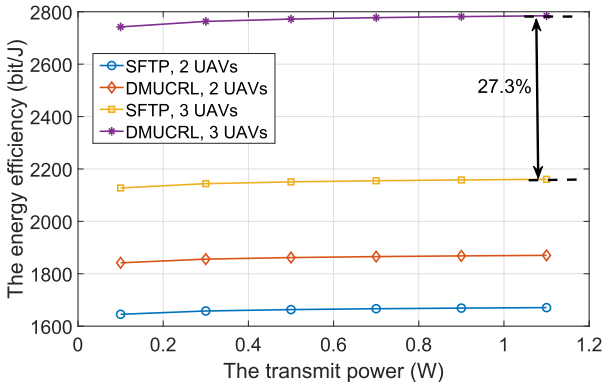


Fig. 10. The energy efficiency of UAVs versus the transmit power of UAV.

the increase of the UAV velocity. Hence, the energy efficiency will decrease if the UAV velocity is higher than the most energy-efficient velocity. The most energy-efficient velocity of UAVs is mainly impacted by the static power consumption and the propulsion power consumption.

Fig. 9 shows the impact of the side length of the small grid on the energy efficiency, where the side length of the small grid is determined by the coverage range of UAVs. We set the transmit power of UAVs as $P_m^{\mathrm{T}} = 20$ dBm, $\forall m \in \mathcal{M}$. UAVs move horizontally at a certain altitude $H = 120$ m and change the azimuth angle of beam to resize the coverage range. The larger coverage range of UAV means the longer side length of the small grid. Fig. 9 shows that the energy efficiency is improved as the coverage range increases. The reason is that a

larger coverage range can increase the number of GUs served in a time slot. To be specific, since UAVs deliver contents to GUs in multicast mode, a large coverage range can accomplish in successfully delivering contents to more GUs with the same energy cost. However, the large coverage range will increase the probability of UAVs hovering over adjacent grids, which may result in failing to deliver contents to GUs located the in interference region (See Fig. 3), thereby decreasing the energy efficiency.

Fig. 10 portrays the energy efficiency versus the transmit power of UAV. We set the UAV velocity and side length of the small grid as $V = 25$ m/s and $l = 100$ m, respectively. As we can see, the increase of the transmit power of UAV can improve the energy efficiency. The reason is that the increase of transmit power of UAV can decrease the hovering time of UAV, which can decrease the total static energy cost and further improve the energy efficiency. In addition, the proposed DMUCRL algorithm can greatly improve the energy efficiency compared to SFTP. Fig. 10 also shows that more UAVs can provide more energy-efficient content coverage. More UAVs can decrease the total length of the trajectory of UAVs. Furthermore, more UAVs can increase the probability of UAV hovering over the grids having GUs, which results in more contents delivered to GUs with the same energy consumption.

VI. Conclusion

In this paper, we investigate multi-UAV trajectory planning to provide a long-term energy-efficient content coverage. We formulate the multi-UAV trajectory planning problem as two related multi-agent cooperative stochastic games. To obtain equilibrium of the games, we propose a Q-learning based decentralized multi-UAV cooperative RL (DMUCRL) algorithm. The proposed algorithm enables UAVs to independently choose their cruise policy and recharging scheduling. Besides, UAVs share their learning results with other UAVs over a time-varying communication network. We prove that the proposed algorithm can converge to the optimal solution of Bellman equation with a higher convergence rate compared to the centralized algorithm. Simulation results show that the proposed algorithm can achieve higher energy efficiency compared to SFTP and ensure a long-term run by recharging regularly.

Appendix

A. Proof of Theorem 1

This proof is inspired by [27] and [28]. Denoting $\left\{\bar{Q}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right)\right\}$ as the network-average iterate process for any state-action pair $\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$ at time slot t, we have

$$\bar{Q}^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right) = \frac{1}{M}\sum_{m=1}^{M} Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right), \quad \forall t \geq 0. \tag{35}$$

For notational simplicity, we put the time slot index t in the superscript for notation compactness to replace the E and we omit the action and state in the bracket in this proof, e.g., $Q_m^t = Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right)$, $\bar{Q}^t = \bar{Q}^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, t\right)$, $\beta^t = \beta^t\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$, $\alpha^t = \alpha^t\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$, $r_m^t = r_m\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}, s^{\mathrm{E}\prime}\right)$ and $Q_m^{t\prime} = Q_m^{\mathrm{E}}\left(s^{\mathrm{E}\prime}, \mathbf{a}^{\mathrm{E}\prime}, t\right)$. Thus, (28) can be rewritten as

$$Q_m^{t+1} = Q_m^t - \beta^t \sum_{n\in\mathcal{N}_m^t}\left(Q_m^t - Q_n^t\right) + \alpha^t\left(r_m + \gamma \max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_m^{t\prime} - Q_m^t\right). \tag{36}$$

Since the Q value of state-action pair $\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$ is updated if and only if the joint action $\mathbf{a}^{\mathrm{E}}$ occurs at state s^{E}, we denote $\{k\}, \forall k \geq 0$ as the sequence of update moment of state-action pair $\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)$ for energy learner. Hence, we have

$$\mathbf{Q}^{k+1} = \left(\mathbf{I}_M - \beta^k L^k - \alpha^k \mathbf{I}_M\right)\mathbf{Q}^k + \alpha^t\left(\mathbf{R}^k + \mathbf{U}^k\right), \tag{37}$$

where $\mathbf{Q}^{k+1} = \left(Q_1^{k+1}, \ldots, Q_M^{k+1}\right)^{\mathrm{T}}$ and $\mathbf{I}_M$ is the $M \times M$ identity matrix. In (37), we have $\mathbf{R}^k = \left(r_1^k, \ldots, r_M^k\right)^{\mathrm{T}}$ and $\mathbf{U}^k = \left(\gamma \max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_1^{k\prime}, \ldots, \gamma \max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_M^{k\prime}\right)^{\mathrm{T}}$. Moreover, we can obtain

$$\begin{aligned}\mathbf{Q}^{k+1} - \bar{\mathbf{Q}}^{k+1} &= \left(\mathbf{I}_M - \beta^k L^k - \alpha^k \mathbf{I}_M\right)\mathbf{Q}^k - \bar{\mathbf{Q}}^{k+1} + \alpha^k\left(\mathbf{R}^k + \mathbf{U}^k\right)\\ &= \left(\mathbf{I}_M - \beta^k L^k - \alpha^k \mathbf{I}_M\right)\left(\mathbf{Q}^k - \bar{\mathbf{Q}}^k\right) + \alpha^k\left(\hat{\mathbf{R}}^k + \hat{\mathbf{U}}^k\right)\end{aligned} \tag{38}$$

where $\bar{\mathbf{Q}}^k = \bar{Q}^k \mathbf{1}_M$ with $\mathbf{1}_M$ denoting the M-dimensional column vectors of ones. In addition, we have $\hat{\mathbf{R}}^k = \left(\mathbf{I}_M - (1/M)\,\mathbf{1}_M\left(\mathbf{1}_M\right)^{\mathrm{T}}\right)\mathbf{R}^k$ and $\hat{\mathbf{U}}^k = \left(\mathbf{I}_M - (1/M)\,\mathbf{1}_M\left(\mathbf{1}_M\right)^{\mathrm{T}}\right)\mathbf{U}^k$. Hence, we can obtain

$$\begin{aligned}\left\|\mathbf{Q}^{k+1} - \bar{\mathbf{Q}}^{k+1}\right\| &= \left\|\left(\mathbf{I}_M - \beta^k L^k - \alpha^k \mathbf{I}_M\right)\mathbf{Q}^k - \bar{\mathbf{Q}}^{k+1}\right\| + \left\|\alpha^k\left(\mathbf{R}^k + \mathbf{U}^k\right)\right\|\\ &\leq \left\|\left(\mathbf{I}_M - \beta^k L^k\right)\left(\mathbf{Q}^k - \bar{\mathbf{Q}}^k\right)\right\| + \alpha^k\left\|\left(\mathbf{Q}^k - \bar{\mathbf{Q}}^k\right)\right\| + \alpha^k\left\|\left(\hat{\mathbf{R}}^k + \hat{\mathbf{U}}^k\right)\right\|\\ &\overset{(a)}{\leq} \left(1 - C_k + \alpha^k\right)\left\|\left(\mathbf{Q}^k - \bar{\mathbf{Q}}^k\right)\right\| + \alpha^k\left(\left\|\hat{\mathbf{R}}^k\right\| + \left\|\hat{\mathbf{U}}^k\right\|\right)\end{aligned} \tag{39}$$

where (a) follows the Lemma 4.4 in [29] and $C_k \to 0$ as $k \to \infty$ with $C_k \in [0, 1]$. Since $\alpha^k \to 0$ as $k \to \infty$, it can be obtained that $(1 - C_k + \alpha^t) \to 0$ as $k \to \infty$. Hence, we can obtain that $\mathbb{P}\left(\lim_{t\to\infty}\left\|\mathbf{Q}^k - \bar{\mathbf{Q}}^k\right\| = 0\right) = 1$. Namely,

$$\mathbb{P}\left(\lim_{t\to\infty} Q_m^{\mathrm{E}}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) = \bar{Q}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)\right) = 1, \quad \forall m \in \mathcal{M}, \;\; s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, \;\; \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{\mathrm{E}}. \tag{40}$$

Then, we give the proof of $\mathbb{P}\left(\lim_{t\to\infty}\bar{Q}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right) = Q^{\mathrm{E}*}\left(s^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}}\right)\right) = 1, \forall m \in \mathcal{M}, s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{\mathrm{E}}$. The proof is inspired by [27]. A result of stochastic approximation given in [27] is first introduced.

Lemma 1: A random iterative process $\triangle^{t+1}(x) = (1 - \alpha^t(x))\triangle^t(x) + \beta^t(x)\Psi^t(x)$ *converges to zeros w.p.1 under the following assumptions*:

1) *The state space is finite.*
2) $\sum_t \alpha^t(x) = \infty$, $\sum_t \beta^t(x) = \infty$, $\sum_t (\alpha^t(x))^2 = \infty$, $\sum_t (\beta^t(x))^2 = \infty$, *and* $E\{\beta^t(x)\,|\Lambda^t\} \leq E\{\alpha^t(x)\,|\Lambda^t\}$ *uniformly w.p.1.*

3) $\|E\{\Psi^t(x)|\Lambda^t\}\|_W \le \zeta \|\triangle^t\|_W$, *where* $\zeta \in (0,1)$.
4) $\mathrm{Var}\{\Psi^t(x)|\Lambda^t\} \le C(1+\|\triangle^t\|_W)^2$, *where* C *is some constant.*

Here $\Lambda^t = \{\triangle^t, \triangle^{t-1}, \ldots, \Psi^{t-1}, \ldots, \alpha^{t-1}, \ldots, \beta^{t-1}, \ldots\}$ *stands for the past at time slot* t. *The notation* $\|\cdot\|_W$ *refers to some weighted maximum norm.*

According to (35), we can obtain

$$\bar{Q}^{t+1} = (1-\alpha^t)\bar{Q}^t + \alpha^t \frac{1}{M}\sum_{m=1}^{M}\left(r_m + \gamma \max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_m^{t\prime}\right). \tag{41}$$

By subtracting Q^* from both side of (41), we have

$$\bar{Q}^{t+1} - Q^* = (1-\alpha^t)(\bar{Q}^t - Q^*) + \alpha^t\left(\frac{1}{M}\sum_{m=1}^{M}\left(r_m + \gamma \max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_m^{t\prime}\right) - Q^*\right). \tag{42}$$

Note that the temporal difference algorithm in (42) can be seen as a random process mentioned in Lemma 1 with $\triangle^t = \bar{Q}^t - Q^*$, $\Psi^t = \frac{1}{M}\sum_{m=1}^{M}\left(r_m + \gamma \max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_m^{t\prime}\right) - Q^*$ and $\beta^t = \alpha^t$. Hence, the condition 1) and 2) in Lemma 1 are satisfied. Then, we give the proof of the temporal difference algorithm in (42) satisfying the condition 3) and 4) in Lemma 1.

According to Proposition 5.1 in [28], we can obtain that $\mathcal{G}(\cdot)$ is a contraction mapping and Q^* is the unique fixed point of operator $\mathcal{G}(\cdot)$, where $\mathcal{G}(\cdot)$ is given by

$$\begin{aligned}\mathcal{G}(Q) &= \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}\left(r_m\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right)\right) + \gamma\sum_{s^{\mathrm{E}\prime}\in\mathcal{S}^{\mathrm{E}}} P^{\mathrm{E}}_{s^{\mathrm{E}}s^{\mathrm{E}\prime}}\left(\mathbf{a}^{\mathrm{E}}\right)\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q\left(s^{\mathrm{E}\prime},\mathbf{a}^{\mathrm{E}\prime}\right)\\ &= \frac{1}{M}\sum_{m=1}^{M}\sum_{s^{\mathrm{E}\prime}\in\mathcal{S}^{\mathrm{E}}} P^{\mathrm{E}}_{s^{\mathrm{E}}s^{\mathrm{E}\prime}}\left(\mathbf{a}^{\mathrm{E}}\right) r_m\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}},s^{\mathrm{E}\prime}\right) + \gamma\sum_{s^{\mathrm{E}\prime}\in\mathcal{S}^{\mathrm{E}}} P^{\mathrm{E}}_{s^{\mathrm{E}}s^{\mathrm{E}\prime}}\left(\mathbf{a}^{\mathrm{E}}\right)\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q\left(s^{\mathrm{E}\prime},\mathbf{a}^{\mathrm{E}\prime}\right)\\ &= \sum_{s^{\mathrm{E}\prime}\in\mathcal{S}^{\mathrm{E}}} P^{\mathrm{E}}_{s^{\mathrm{E}}s^{\mathrm{E}\prime}}\left(\mathbf{a}^{\mathrm{E}}\right)\left(\frac{1}{M}\sum_{m=1}^{M} r_m\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}},s^{\mathrm{E}\prime}\right) + \gamma\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q\left(s^{\mathrm{E}\prime},\mathbf{a}^{\mathrm{E}\prime}\right)\right).\end{aligned} \tag{43}$$

Thus, we have $\mathcal{G}(Q^*) = Q^*$ and

$$\left\|\mathcal{G}\left(Q_1\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right)\right) - \mathcal{G}\left(Q_2\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right)\right)\right\|_\infty = \left\|Q_1\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right) - Q_2\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right)\right\|_\infty. \tag{44}$$

It can be obtained

$$\begin{aligned}\mathrm{E}\left\{\Psi^t\right\} &= \sum_{s^{\mathrm{E}\prime}\in\mathcal{S}^{\mathrm{E}}} P^{\mathrm{E}}_{s^{\mathrm{E}}s^{\mathrm{E}\prime}}\left(\mathbf{a}^{\mathrm{E}}\right) \times\left(\frac{1}{M}\sum_{m=1}^{M}\left(r_m + \gamma\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_m^{t\prime}\right) - Q^*\right) \qquad (45)\\ &= \sum_{s^{\mathrm{E}\prime}\in\mathcal{S}^{\mathrm{E}}} P^{\mathrm{E}}_{s^{\mathrm{E}}s^{\mathrm{E}\prime}}\left(\mathbf{a}^{\mathrm{E}}\right) \times\left(\frac{1}{M}\sum_{m=1}^{M} r_m + \gamma\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} \bar{Q}^{t\prime} - Q^*\right) \qquad (46)\\ &= \mathcal{G}\left(\bar{Q}^t\right) - Q^* \qquad (47)\end{aligned}$$

As a result, we have $\|\mathrm{E}\{\Psi^t(x)\}\|_\infty = \left\|\mathcal{G}\left(\bar{Q}^t\right) - \mathcal{G}\left(Q^*\right)\right\|_\infty \le \gamma\left\|\bar{Q}^t - Q^*\right\|_\infty$. Let $\|\cdot\|_\infty$ replace $\|\cdot\|_W$ in Lemma 1, the condition 3) in Lemma 1 is satisfied.

For the condition 4) in Lemma 1, we have

$$\begin{aligned}&\mathbf{var}\left\{\Psi^t\right\}\\ &= \mathbb{E}\left\{\left(\frac{1}{M}\sum_{m=1}^{M}\left(r_m + \gamma\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} Q_m^{t\prime}\right) - Q^*\right\} - \left(\mathcal{G}\left(\bar{Q}^t\right) - \mathcal{G}\left(Q^*\right)\right)\right)^2\right\} \qquad (48)\\ &= \mathbb{E}\left\{\left(\frac{1}{M}\sum_{m=1}^{M} r_m + \gamma\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} \bar{Q}^{t\prime} - \mathcal{G}\left(\bar{Q}^t\right)\right)^2\right\}\\ &= \mathbf{var}\left\{\frac{1}{M}\sum_{m=1}^{M} r_m + \gamma\max_{\mathbf{a}^{\mathrm{E}\prime}\in\mathcal{A}^{\mathrm{E}}} \bar{Q}^{t\prime}\right\}, \qquad (49)\end{aligned}$$

which clearly verifies $\mathbf{var}\{\Psi^t\} \le C\left(1 + \left\|\bar{Q}^t - Q^*\right\|_W^2\right)$ for some constant C due to the fact that $\frac{1}{M}\sum_{m=1}^{M} r_m$ is bounded [30]. Hence, the condition 4) in Lemma 1 is satisfied. Then, according to Lemma 1, we can obtain $\mathbb{P}\left(\lim_{t\to\infty}\bar{Q}\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right) = Q^{\mathrm{E}*}\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right)\right) = 1, \forall m \in \mathcal{M}, s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{\mathrm{E}}$. Additionally, we have verified $\mathbb{P}\left(\lim_{t\to\infty} Q_m^{\mathrm{E}}\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right) = \bar{Q}\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right)\right) = 1, \forall m \in \mathcal{M}, s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{\mathrm{E}}$. Thus, we have $\mathbb{P}\left(\lim_{t\to\infty} Q_m^{\mathrm{E}}\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right) = Q^{\mathrm{E}*}\left(s^{\mathrm{E}},\mathbf{a}^{\mathrm{E}}\right)\right) = 1, \forall m \in \mathcal{M}, s^{\mathrm{E}} \in \mathcal{S}^{\mathrm{E}}, \mathbf{a}^{\mathrm{E}} \in \mathcal{A}^{\mathrm{E}}$. We complete the proof of Theorem 1.

REFERENCES

[1] C. Liu, W. Feng, Y. Chen, C.-X. Wang, and N. Ge, "Cell-free satellite-UAV networks for 6G wide-area Internet of Things," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 4, pp. 1116–1131, Apr. 2021.

[2] B. Li, Z. Fei, and Y. Zhang, "UAV communications for 5G and beyond: Recent advances and future trends," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2241–2263, Apr. 2019.

[3] Z. Jia, M. Sheng, J. Li, D. Niyato, and Z. Han, "LEO-satellite-assisted UAV: Joint trajectory and data collection for Internet of remote things in 6G aerial access networks," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9814–9826, Jun. 2021.

[4] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Drone small cells in the clouds: Design, deployment and performance analysis," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 1–6.

[5] X. Liu, Y. Liu, and Y. Chen, "Reinforcement learning in multiple-UAV networks: Deployment and movement design," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 8036–8049, Aug. 2019.

[6] 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.

[7] J. Xu, Y. Zeng, and R. Zhang, "UAV-enabled wireless power transfer: Trajectory design and energy optimization," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5092–5106, Aug. 2018.

[8] J. Hu, H. Zhang, and L. Song, "Reinforcement learning for decentralized trajectory design in cellular UAV networks with sense-and-send protocol," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6177–6189, Aug. 2019.

[9] V. Saxena, J. Jalden, and H. Klessig, "Optimal UAV base station trajectories using flow-level models for reinforcement learning," *IEEE Trans. Cognit. Commun. Netw.*, vol. 5, no. 4, pp. 1101–1112, Dec. 2019.

[10] C. H. Liu, Z. Chen, J. Tang, J. Xu, and C. Piao, "Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2059–2070, Sep. 2018.

[11] X. Li, H. Yao, J. Wang, S. Wu, C. Jiang, and Y. Qian, "Rechargeable multi-UAV aided seamless coverage for QoS-guaranteed IoT networks," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10902–10914, Dec. 2019.

[12] S. Chai and V. K. N. Lau, "Online trajectory and radio resource optimization of cache-enabled UAV wireless networks with content and energy recharging," *IEEE Trans. Signal Process.*, vol. 68, pp. 1286–1299, 2020.
[13] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 2109–2121, Mar. 2018.
[14] H. Wu, X. Tao, N. Zhang, and X. Shen, "Cooperative UAV cluster-assisted terrestrial cellular networks for ubiquitous coverage," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2045–2058, Sep. 2018.
[15] H. Zhao, H. Wang, W. Wu, and J. Wei, "Deployment algorithms for UAV airborne networks toward on-demand coverage," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2015–2031, Sep. 2018.
[16] Y. Zeng, R. Zhang, and T. J. Lim, "Throughput maximization for UAV-enabled mobile relaying systems," *IEEE Trans. Commun.*, vol. 64, no. 12, pp. 4983–4996, Dec. 2016.
[17] Y. Zeng, J. Xu, and R. Zhang, "Energy minimization for wireless communication with rotary-wing UAV," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329–2345, Apr. 2019.
[18] G. Arslan and S. Yuksel, "Decentralized Q-learning for stochastic teams and games," *IEEE Trans. Autom. Control*, vol. 62, no. 4, pp. 1545–1558, Apr. 2017.
[19] X. Wang and T. Sandholm, "Reinforcement learning to play an optimal Nash equilibrium in team Markov games," *Proc. Adv. Neural Inf. Process. Syst.*, vol. 15, 2002, pp. 1603–1610.
[20] M. Lauer and M. Riedmiller, "An algorithm for distributed reinforcement learning in cooperative multi-agent systems," in *Proc. 17th Int. Conf. Mach. Learn. (ICML)*, pp. 535–542.
[21] J. Foerster *et al.*, "Stabilising experience replay for deep multi-agent reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1146–1155.
[22] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian, "Deep decentralized multi-task multi-agent reinforcement learning under partial observability," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2681–2690.
[23] J. Cui, Y. Liu, and A. Nallanathan, "Multi-agent reinforcement learning-based resource allocation for UAV networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 729–743, Feb. 2020.
[24] K. Zhang, Z. Yang, H. Liu, T. Zhang, and T. Başar, "Fully decentralized multi-agent reinforcement learning with networked agents," 2018, *arXiv:1802.08757*. [Online]. Available: http://arxiv.org/abs/1802.08757
[25] R. Elliott, "A measure of fairness of service for scheduling algorithms in multiuser systems," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. Conf. (CCECE)*, vol. 3, May 2002, pp. 1583–1588.
[26] B. Jiang, J. Yang, H. Xu, H. Song, and G. Zheng, "Multimedia data throughput maximization in Internet-of-Things system based on optimization of cache-enabled UAV," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3525–3532, Apr. 2019.
[27] T. Jaakkola, M. I. Jordan, and S. P. Singh, "Convergence of stochastic iterative dynamic programming algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 703–710.
[28] S. Kar, J. M. F. Moura, and H. V. Poor, "$\mathcal{QD}$-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus + innovations," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1848–1862, Apr. 2013.
[29] S. Kar, J. M. F. Moura, and H. V. Poor, "Distributed linear parameter estimation: Asymptotically efficient adaptive strategies," *SIAM J. Control Optim.*, vol. 51, no. 3, pp. 2200–2229, Jan. 2013.
[30] F. S. Melo, "Convergence of Q-learning: A simple proof," Inst. Syst. Robot., Tech. Rep.

Chenxi Zhao received the B.E. degree in telecommunications engineering from Xidian University, Xian, China, in 2016, where he is currently pursuing the Ph.D. degree with the State Key Laboratory of Integrated Service Networks. His research interests include proactive caching, reinforcement learning, and UAV-enabled wireless networks.

Junyu Liu (Member, IEEE) received the B.S. and Ph.D. degrees in communication and information systems from Xidian University, Shaanxi, China, in 2011 and 2016, respectively. He is currently an Associate Professor with the State Key Laboratory of Integrated Service Networks, Institute of Information and Science, Xidian University. His research interests include interference management and performance evaluation of wireless heterogeneous networks and ultra-dense wireless networks.

Min Sheng (Senior Member, IEEE) received the M.S. and Ph.D. degrees in communication and information systems from Xidian University, Shaanxi, China, in 2000 and 2004, respectively. She is currently a Full Professor and the Director with the State Key Laboratory of Integrated Service Networks, Xidian University. Her general research interests include mobile *ad hoc* networks, 5G mobile communication systems, and satellite communications networks. She was awarded as a Distinguished Young Researcher from NSFC and a Changjiang Scholar from the Ministry of Education, China. She is a fellow of the China Institute of Electronics (CIE).

Wei Teng received the B.E. degree in telecommunications engineering from Xidian University, Xi'an, China, in 2014, where he is currently pursuing the Ph.D. degree in communication and information systems. From 2018 to 2019, he was a Visiting Ph.D. Student with the Bradley Department of Electrical and Computer Engineering, Virginia Tech. His research interests focus on resource allocation and optimization applications in small cell networks.

Yang Zheng received the B.E. and Ph.D. degrees in communications engineering from Xidian University, Xi'an, China, in 2014 and 2020, respectively. He is currently an Assistant Professor with the State Key Laboratory of Integrated Service Networks, Xidian University. His research interests include indoor localization and signal processing.

Jiandong Li (Fellow, IEEE) received the M.S. and Ph.D. degrees from Xidian University in 1985 and 1991, respectively. Since 1985, he has been a Faculty Member with the School of Telecommunications Engineering, Xidian University, where he is currently a Professor. From 2002 to 2003, he was a Visiting Professor with the Department of Electrical and Computer Engineering, Cornell University. His major research interests include wireless communication theory, cognitive radio, and signal processing. He was a member of Personal Communications Networks (PCN), specialist group for China 863 Communication High Technology Program, from January 1993 to October 1994 and from 1999 to 2000. He is a member of specialist group of the new generation of broadband wireless mobile communication networks for the Ministry of Industry and Information Technology, and the Chair of Broadband Wireless IP Standard Work Group, China. He is a fellow of the China Institute of Electronics (CIE) and the China Institute of Communication (CIC). He was awarded as a Distinguished Young Researcher from NSFC and a Changjiang Scholar from the Ministry of Education, China. He served as the General Vice Chair for ChinaCom 2009 and the TPC Chair for IEEE ICCC 2013.