Trajectory and Communication Design for Cache-enabled UAVs in Cellular Networks: A Deep Reinforcement Learning Approach

Jiequ Ji, Kun Zhu, *Member, IEEE*, Lin Cai, *Fellow, IEEE*

***Abstract*—In this paper, we investigate the content transmission in a heavy-crowded multiple access cellular network, whose data traffic is offloaded through the combination of edge caching and unmanned aerial vehicle (UAV) communication. In this context, we formulate a novel optimization problem, which minimizes the sum content acquisition delay of users by optimizing the multiuser association and cache placement jointly with UAV trajectory and transmission power over a given flight duration. However, due to the uncertainty of the environment (e.g., random content requests and dynamic UAV positions), it is often difficult and impractical to solve the formulated problem using conventional optimization methods. To this end, we model our problem as a partially observable stochastic game where the macro base station (MBS) and UAVs act as agents to collectively interact with the environment to receive distinctive observations. Moreover, we take advantage of the Proximal Policy Optimization (PPO) learning strategy and propose a novel Dual-Clip PPO-based algorithm to solve the converted problem. To guide agent exploration, a new exploration criterion is proposed in which each UAV agent can obtain an intrinsic reward when it explores beyond the boundary of explored regions (BeBold). Note that the MBS agent has the extrinsic reward given by the environment only. Numerical results reveal that the proposed algorithm outperforms the standard PPO-based deep reinforcement learning algorithm. Moreover, the proposed joint design scheme can achieve a dramatic reduction of content acquisition delay compared with the benchmark schemes without trajectory design or with stochastic cache placement.**

***Index Terms*—Unmanned aerial vehicle, edge caching, trajectory design, cache placement, reinforcement learning.**

I. Introduction

RECENTLY, unmanned aerial vehicles (UAVs) have been widely utilized in various industries due to their high mobility and cost-effectiveness [1]. The inherent characteristics of UAVs enable them to effectively solve problems in conventional terrestrial communications, such as high deployment costs and poor adaptability to exceptional situations. Consequently, UAVs can be employed as aerial base stations (ABSs) to assist traditional infrastructure-based cellular networks [2]. The main application scenarios of UAV-assisted communications include high-speed wireless connection in hotspots, reliable emergency communication, flexible data transmission and so on [3]. With the rapid proliferation of smart mobile devices and emerging mobile applications, data traffic has shown an explosive growth in recent years. The latest Cisco report predicts that global mobile data traffic will reach 77 exabytes per month by 2022, of which about 79 percent is used for content transmission [4]. Edge caching, by which popular contents can be prestored at base stations (BSs) or user equipments (UEs) before being requested by mobile users, has emerged as an efficient technique for alleviating the network traffic load [5]–[7].

To alleviate the traffic pressure of BSs in cellular networks and satisfy the demand of data volume of content transmission, cache-enabled UVAs are deployed to offload part of the traffic from hotspots during peak hours [8], which is a fast and low-cost deployment way for content transmission with low delivery delay and high data rate requirements. It is worth noting that content acquisition delay is an intuitive evaluation of the user experience [9]. In this paper, we study the minimum sum content acquisition delay of all users in UAV-assisted cellular networks for content transmission. One practical application scenario is a stadium that hosts a large-scale concert, in which multiple cache-enabled UAVs are deployed above the theater to provide wireless coverage and help BSs offload data traffic.

A. Related Works

Extensive research works have been recently devoted to the filed of UAV communications. In particular, many researchers have discussed several crucial problems in UAV-enabled communication systems, such as UAV deployment, UAV trajectory design, cache placement, resource allocation, and secure data transmission. The relative position between the UAV and users has a great impact on the performance of content transmission. There have been many studies on UAV deployment in different scenarios [10]–[13]. The work in [10] considered a UAV-relaying system for malfunctioning BSs, where the capacity of the relay network was maximized by optimizing the UAV deployment. The optimal deployment of multiple UAVs was studied in [11] to maximize the downlink coverage. The work in [12] studied a dual-UAV-enabled secure communication system, where the UAV trajectory and user scheduling were jointly optimized to maximize the minimum worst-case secrecy rate over all users. In [13], a joint trajectory design and communication resource allocation algorithm was proposed to maximize the minimum user throughput for multi-UAV enabled networks. In addition, some works have been conducted in UAV scheduling and non-orthogonal multiple access (NOMA) precoding to improve the network performance of UAV-enabled cellular networks [14]–[16]. An efficient UAV scheduling framework was proposed in [14] to provide uninterrupted services for multiple events. The work

J. Ji and K. Zhu are with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China (email: {jiequ, zhukun,wangran}@nuaa.edu.cn).

L. Cai is with the Department of Electrical and Computer Engineering, University of Victoria, Victoria BC V8W 3P6, Canada (e-mail: cai@ece.uvic.ca).

in [15] studied a UAV-assisted NOMA network, where a UAV and a BS cooperatively serve ground users. To maximize the sum rate of all ground users, a joint optimization of UAV trajectory and NOMA precoding was proposed in [15].

Edge caching has always been an interesting research topic in traditional cellular networks. The work in [6] studied a joint design of cache placement and content delivery for achieving secure transmission against eavesdropping attacks. In [7], an optimal content placement strategy was developed to improve the transmission efficiency of the network. In addition, a few recent research has focused on cache-enabled UAV networks [17]–[23]. The main purpose of deploying caches at UAVs is to cache hot popular contents during off-peak periods so that the contents requested by ground users can be directly transmitted without wireless backhaul when they exist in local caches of UAVs. The work in [17] proposed a cache placement strategy based on the prediction of content request distribution and user mobility. The optimal placement of cached contents and UAV locations was studied for maximizing the minimum throughput of all IoT devices in [18]. In [19], a joint content caching and transmission algorithm was proposed to improve the reliability of wireless devices. In [22], cache-enabled secure transmission for UAV-helped scalable videos in hyper-dense networks was studied, where cache-enabled UAVs were used as mobile BSs to transmit video streams to the users together with small base stations (SBSs). The work in [23] studied a joint optimization of UAV trajectory and time scheduling for maximizing the security of UAV-relayed wireless networks with caching.

B. Motivation and Contribution

Note that content acquisition delay, as an important indicator for evaluating network performance, has not been well investigated in the above works on cache-enabled UAV networks. In this paper, we take minimum content acquisition delay as our optimization objective of trajectory design and communication resource allocation in cache-enabled multi-UAV cellular networks. The content acquisition delay is directly related to the transmission distance between the UAV and ground users. Adjusting the trajectories of UAVs can not only establish short-distance transmission links for these desired UAV-user pairs, but also extend the jamming channel distance of all undesirable UAV-user pairs to alleviate the co-channel interference. If the content requested by the user does not exist in local caches of UAVs, the UAV firstly fetches the content from the macro base station (MBS) over a wireless backhaul link and then sends it to the associated user. Obviously, the cache placement plays a vital role in the content acquisition delay. Although trajectory design and cache placement have been widely investigated, to the best of our knowledge, few research works have combined these two aspects to fully exploit their respective strengths. It is worth noting that the joint design of UAV trajectory and cache placement is also affected by user scheduling and association. Motivated by the aforementioned facts, we study the content acquisition delay minimization in a cache-enabled multi-UAV network by jointly optimizing the multiuser association, cache placement, UAV trajectory and transmission power in a given finite period. The formulated problem is a mixed-integer non-convex problem with a highly nonlinear objective function and thousands of binary variables and time-varying UAV trajectory variables. Moreover, since the uncertainty in the environment, it is often impractical to solve applying traditional alternating algorithms based on the block coordinate descent method.

Recently, reinforcement learning (RL) [24] has attracted increasing research interests in solving non-convex optimization problems that standard mathematical methods cannot address, such as trajectory design and multiuser access control for UAV-assisted communications [25]–[27]. Particularly, those hard-to-optimize problems can be simplified as maximizing cumulative rewards through a series of proper reward design and training mechanisms. Motivated by the above considerations, we model our problem as a partially observable stochastic game and then propose a Dual-Clip Proximal Policy Optimization (DC-PPO) algorithm to solve, which is a very promising deep reinforcement learning algorithm based on the actor-critic framework. Intuitively, the ceiling of performance can be broken if there is additional supervised information provided to guide the agent learning. Therefore, we introduce intrinsic rewards to motivate agents to explore before obtaining extrinsic rewards. Since the state transition in our environment is reversible, our simple use of intrinsic rewards to guide exploration may cause each UAV agent to switch back and forth between the new state and the previous state. In order to tackle this issue, we propose a novel criterion for intrinsic rewards that encourages each UAV agent to explore beyond the boundary of explored regions (BeBold).

The main novelty and contributions of this paper are summarized as follows:

- We propose a novel framework of cache-enabled multi-UAV cellular networks for multimedia content dissemination of users in the hotspot area. Then we minimize the sum content acquisition delay of all users by optimizing the UAV flight trajectory and transmission power jointly with user association and cache placement.
- In order to model the uncertainty of the environment (e.g., random content requests and dynamic UAV positions), we formulate our problem as a partially observable stochastic game. The action taken by the MBS agent corresponds to the user association, while the actions taken by each UAV agent correspond to the cache placement, UAV trajectory and power control.
- We propose a DC-PPO algorithm using the BeBold-based exploration criterion to solve the converted problem. To the best of our knowledge, there is no literature that uses the BeBold-based exploration criterion to encourage each UAV agent to gradually expand the area of exploration.
- Simulation results provide several observations. Firstly, the proposed cache-enabled multi-UAV system is superior to the traditional multi-UAV systems without caching in term of content acquisition delay. Secondly, the proposed joint design scheme achieves significantly lower content acquisition delay compared with the benchmark schemes without cache placement and multiuser association optimization. Finally, the high mobility of UAVs is beneficial for achieving better channel conditions and can provide additional flexibility for interference mitigation, and thus greatly reduces the content acquisition delay compared to the traditional cellular networks with static BSs.

The rest of this paper is structured as follows. In Section II, we introduce the system model and formulate the optimization problem for content achieving delay minimization. In Section III, we build a learning system to solve our proposed problem. Section IV describes the algorithm design in details. Numerical results and analysis are presented in Section IV, which confirm the superiority of the proposed algorithm. Finally, we conclude the paper in Section V.

Notations: In this paper, vectors and scalars are respectively denoted by italic and boldface letters. $\mathbb{R}^{N\times 1}$ denotes the space of N-dimensional real vector. For a vector $\mathcal{A}$, $\|\mathcal{A}\|$ denotes its Euclidean norm and $\mathcal{A}^T$ denotes its transpose.

II. System Model

We consider the downlink transmission of a cellular wireless network as shown in Fig. 1, where one MBS and M moving UAVs cooperatively transmit contents for a group of U ground users, denoted by b, $m \in \mathcal{M} = \{1, 2, \dots, M\}$ and $u \in \mathcal{U} = \{1, 2, \dots, U\}$, respectively. It is worth noting that each user is under the overlapping coverage of the MBS and UAVs. We use $i \in \mathcal{M} \cup \{b\}$ to index the access node (i.e., MBS or UAV). The MBS is connected to the core network via a wired fiber link, while each UAV communicates with the MBS via a wireless backhaul link. Suppose that the wireless backhaul link and the radio access link are allocated orthogonal frequency bands. As a result, there is no interference between the radio access link from the MBS (or UAV) to the user and the wireless backhaul link from the MBS to the UAV. In addition, the radio access link of each UAV is orthogonal to that of the MBS in order to avoid co-channel interference between them.

A. UAV Mobility Model

We assume that the locations of the MBS and all the users are fixed on the ground with altitude zero. Accordingly, a two-dimensional cartesian coordinate system is exploited with all the dimensions measured in meters. Let $\mathbf{w}_b = [x_b, y_b] \in \mathbb{R}^{2\times 1}$ and $\mathbf{w}_u = [x_u, y_u]_{u\in\mathcal{U}} \in \mathbb{R}^{2\times 1}$ denote the horizontal coordinates of the MBS and user u, respectively. It is also assumed that all UAVs fly at a fixed height of H meters above the ground and provide content transfer services for U users through a cyclical time-division multiple access (TDMA) protocol with a constant cycle duration T. Let $\mathbf{q}_m(t) = [x_m(t), y_m(t)]^T_{m\in\mathcal{M}} \in \mathbb{R}^{2\times 1}$ denote the horizontal plane coordinate of UAV m at time instant $t \in [0, T]$. For ease of exposition, we further quantize the continuous time T as N time slots with equal duration d_t, i.e., $T = d_t N$ and $n \in \mathcal{N} = \{1, 2, \dots, N\}$. It is worth noting that the value of d_t should be sufficiently small to ensure that the position of each UAV can be considered to be static within each time slot. Thus, the horizontal trajectory of UAV m can be approximately denoted by a sequence of discrete points as $\mathbf{q}_m[n] = [x_m[n], y_m[n]]^T_{n\in\mathcal{N}}$.

At the beginning of time slot n, UAV m flies in a horizontal direction determined by the angle of $\vartheta_m[n] \in (0, 2\pi]$, distance of $d_m[t] \in [0, d_{\max}]$, where $d_{\max}$ denotes the maximum flying distance that each UAV can travel during a time slot. Thus, we use $x_m[n] = x_m[0] + \sum_{n=0}^{n} d_m[n]\cos(\vartheta_m[n])$ and $y_m[n] = y_m[0] + \sum_{n=0}^{n} d_m[n]\sin(\vartheta_m[n])$ to denote the coordinate of UAV m at time slot n. Any two UAVs should keep a minimal safe distance to ensure collision avoidance among them in each time slot. Besides, each UAV cannot fly beyond the boundary of the target area, which is given by $[0, x_{\max}] \times [0, y_{\max}]$. Then the following trajectory constraints should be satisfied

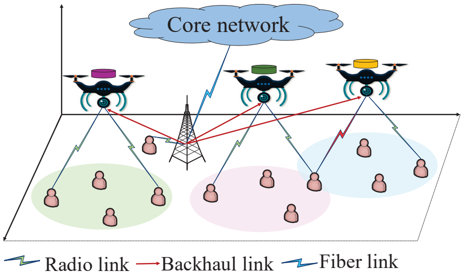


Fig. 1. An illustration of downlink data transmission for a cellular network with multiple cache-enabled UAVs.

$$\begin{aligned}
&0 \le \vartheta_m[n] \le 2\pi,\ \forall m,\\
&0 \le x_m[n] \le x_{\max},\ \forall m,\\
&0 \le y_m[n] \le y_{\max},\ \forall m,\\
&0 \le d_m[n] \le d_{\max},\ \forall m,\\
&\mathbf{q}_m[n] - \mathbf{q}_k[n] \ge d_{\min},\ \forall k \ne m, \forall n,
\end{aligned} \tag{1}$$

where $d_{\min}$ denotes the minimal inter-UAV distance.

B. Cache Placement Model

Note that each UAV can provide content transmission only by connecting to the MBS via wireless backhaul. Due to the limited capacity of wireless backhaul, the transmission rate of each UAV is also limited, which will degrade the quality of user experience at peak-traffic period. To tackle this problem, caching can be used for UAV transmission to alleviate network congestion. Specifically, each UAV is equipped with a cache device with limited capacity, which can pre-store some popular contents at off-peak period. If a content requested by a user exists at the local cache of its associating UAV, the content can be directly delivered to the user over a radio link. Otherwise, the UAV fetches the requested content from the MBS through a wireless backhaul link for the associated user.

In this cache model, we consider a content library containing F contents denoted by $\mathcal{F} = \{1, 2, \dots, F\}$. Moreover, each content is assumed to have the same size of S bits. Actually, this assumption of equal-size contents is reasonable since each content can be divided into blocks of the same size. We define a content request matric $\mathbf{x} \in \{0, 1\}^N_{U\times F}$, where each element $x_{u,f}[n] = 1$ means that user u requests content f at time slot n and otherwise $x_{u,f}[n] = 0$. Note that the request probability of each user for a content is affected by the popularity of this content. The popularity of content f in a period T is assumed to be static and follow a Zipf distribution [28].

We define a cache placement matric $\mathbf{y} \in \{0, 1\}^N_{F\times M}$, where each element $y_{f,m}[n] = 1$ means that UAV m caches content f at time slot n and otherwise $y_{f,m}[n] = 0$. All contents are available in the MBS, while each UAV only stores a subset of the total contents due to its limited cache size. We use $S \times C_m$ to denote the storage capacity of UAV m. To avoid repeated caching, we also assume that content f can only be cached on at most L_f UAVs, where $L_f \leq M$. Then the following cache placement constraints should satisfy

$$\begin{aligned}&\sum\nolimits_{f=1}^{F} y_{f,m}[n] \leq C_m,\ \forall m,\\&\sum\nolimits_{m=1}^{M} y_{f,m}[n] \leq L_f,\ \forall f.\end{aligned} \tag{2}$$

C. Transmission Channel Model

Due to the height characteristics of UAVs and the complexity of environment, the UAV-to-user and MBS-to-UAV wireless channels are more likely to be dominated by the probabilistic line-of-sight (LoS) and non-line-of-sight (NLoS) links [29].

1) UAV-to-user: We use a statistical propagation model for calculating the path loss between a UAV and a user. Similar to [30], the LoS and NLoS path losses of UAV m sending a content to user u at time slot n are given by

$$\begin{aligned}h_{m,u}^{\text{LoS}}[n] &= 20\log\left(\frac{4\pi f_c d_{m,u}[n]}{v_c}\right) + \chi_{\text{LoS}},\\h_{m,u}^{\text{NLoS}}[n] &= 20\log\left(\frac{4\pi f_c d_{m,u}[n]}{v_c}\right) + \chi_{\text{NLoS}},\end{aligned} \tag{3}$$

where f_c denotes the carrier frequency; v_c is the speed of light; $d_{m,u}[n]$ is the distance from UAV m to user u at time slot n; χ_{LoS} and χ_{NLoS} are two different shadowing factors due to the LoS and NLoS links, respectively.

The probability of establishing a LoS connection between UAV m and user u at time slot n is expressed as

$$P_{m,u}^{\text{LoS}}[n] = \frac{1}{1 + c_1\exp(-c_2(\theta_{mu}[n] - c_1))}, \tag{4}$$

where c_1 and c_2 are environment-related constant values (e.g., rural and dense urban); $\theta_{mu}[n] = \frac{180}{\pi}\arcsin(\frac{H}{d_{m,u}[n]})$ is the elevation angle between UAV m and user u at time slot n. Then the NLoS probability is given by $P_{m,u}^{\text{NLoS}}[n] = 1 - P_{m,u}^{\text{LoS}}[n]$. As a result, the average path loss from UAV m to user u at time slot n is expressed as

$$\widetilde{h}_{m,u}[n] = P_{m,u}^{\text{LoS}}[n]h_{m,u}^{\text{LoS}}[n] + P_{m,u}^{\text{NLoS}}[n]h_{m,u}^{\text{NLoS}}[n]. \tag{5}$$

2) MBS-to-UAV: The UAV is connected to the MBS to fetch the required content of the user from the core network but that is not stored in the local cache. According to [31], the average path loss from the MBS to UAV m at time slot n is given by

$$\widetilde{h}_{b,m}[n] = P_{b,m}^{\text{LoS}}[n]h_{b,m}^{\text{LoS}}[n] + P_{b,m}^{\text{NLoS}}[n]h_{b,m}^{\text{NLoS}}[n], \tag{6}$$

where $h_{b,m}^{\text{LoS}}[n] = d_{b,m}^{-\alpha}[n]$ and $h_{b,m}^{\text{NLoS}}[n] = \eta d_{b,m}^{-\alpha}[n]$ are the LoS and NLoS path losses from the MBS to UAV m at time slot n, respectively; η is the additional path loss factor of the NLoS link; $d_{b,m}[n]$ is the distance between the MBS and UAV m at time slot n; α is the path loss exponent; Similar to (4), the LoS and NLoS connection probabilities between the MBS and UAV m at time slot n can be respectively expressed as

$$\begin{aligned}P_{b,m}^{\text{LoS}}[n] &= \frac{1}{1 + c_1\exp(-c_2(\theta_{bm}[n] - c_1))},\\P_{b,m}^{\text{NLoS}}[n] &= 1 - P_{b,m}^{\text{LoS}}[n],\end{aligned} \tag{7}$$

where $\theta_{bm}[n] = \frac{180}{\pi}\arcsin(\frac{H}{d_{b,m}[n]})$.

3) MBS-to-user: Referring to the 3GPP standard in [32], the path loss from the MBS to user u is calculated as

$$h_{b,u} = 128.1 + 37.6\log_{10}(d_{b,u}), \tag{8}$$

where $d_{b,u}$ is the distance from the MBS to user u.

We use $P_m[n]$ to index the transmission power of UAV m at time slot n, which is subject to both peak and average power constraints, denoted by P_{avg} and P_{max}. Then we have

$$\begin{aligned}&P_m[n] \leq P_{\text{max}},\ \forall m, n,\\&\frac{1}{N}\sum\nolimits_{n\in\mathcal{N}} P_m[n] \leq P_{\text{avg}},\ \forall m.\end{aligned} \tag{9}$$

According to the above analysis, the signal-to-interference-plus-noise ratio (SINR) of MBS-to-user link, UAV-to-user link and MBS-to-UAV link at time slot t can be expressed as

$$\begin{aligned}\gamma_{b,u}[n] &= \frac{P_b}{\sigma^2 10^{h_{b,u}/10}},\\\gamma_{m,u}[n] &= \frac{P_m[n]10^{-\widetilde{h}_{m,u}[n]/10}}{\sigma^2 + \sum_{i\in\mathcal{M},i\neq m} P_i[n]10^{-\widetilde{h}_{i,u}[n]/10}},\\\gamma_{b,m}[n] &= \frac{P_b}{\sigma^2 10^{\widetilde{h}_{b,m}[n]/10}},\end{aligned} \tag{10}$$

where σ^2 is the noise power and P_b is the transmission power of the MBS.

D. User Association Model

When a user associates with a UAV, the transmission delay mainly consists of two parts: one is the downlink transmission delay from the UAV to this user and the other is the backhaul transmission delay from the MBS to UAV. Moreover, since all contents are stored in the MBS, we only consider the downlink transmission delay when a user associates with the MBS. We define a user scheduling matric $\mathbf{z} \in \{0,1\}_{U\times M}^{N}$, where each element $z_{u,m}[n] = 1$ means that user u is connected to UAV m at time slot n and otherwise $z_{u,m}[n] = 0$.

With limited endurance, each UAV will be allocated a quota for the number of associated users in each time horizon T to ensure the quality of user experience. We use $Q_m[n]$ to denote the quota of UAV m at time slot n. Since the MBS can supply power continuously on the ground, we consider that its quota at time slot n is equal to the number of users in the cell, i.e., $Q_b[n] = U$. Thus the following constraints need to be satisfied

$$\begin{aligned}&\sum\nolimits_{i\in\mathcal{M}\cup\{b\}} z_{u,i}[n] \leq 1,\ \forall u \in \mathcal{U},\\&\sum\nolimits_{u\in\mathcal{U}} z_{u,i}[n] \leq Q_i[n],\ \forall i \in \mathcal{M}\cup\{b\}.\end{aligned} \tag{11}$$

Suppose that the radio link bandwidth of the MBS and each UAV is equal. Let W and B denote the radio link bandwidth and the backhaul link bandwidth, respectively. Since this paper does not focus on the bandwidth allocation of the radio access link and the backhaul link, W and B are assumed to be equally divided among their associated users and UAVs, respectively. At time slot n, the downlink transmission rate from access node i ($i \in \mathcal{M}\cup\{b\}$) to user u is given by

$$R_{i,u}[n] = \frac{W\sum_{f\in\mathcal{F}} x_{u,f}[n]z_{u,i}[n]}{\sum_{u\in\mathcal{U}}\sum_{f\in\mathcal{F}} x_{u,f}[n]z_{u,i}[n]}\log_2(1 + \gamma_{i,u}[n]). \tag{12}$$

In addition, the backhaul data transmission rate from the MBS to UAV m for user u at time slot n is given by

$$\begin{aligned} R_{b,m,u}[n] &= \log_2(1+\gamma_{b,m}[n]) \\ &\times \frac{B\sum_{f\in\mathcal{F}} x_{u,f}[n] z_{u,m}[n](1-y_{m,f}[n])}{\sum_{u=1}^{U}\sum_{f=1}^{F}\sum_{m=1}^{M} x_{u,f}[n] z_{u,m}[n](1-y_{m,f}[n])}. \end{aligned} \tag{13}$$

The downlink transmission delay from access node i to user u and the backhaul transmission delay from the MBS to UAV m for user u are respectively expressed as

$$T_{i,u}^{\text{down}}[n] = \frac{S}{R_{i,u}[n]}, \quad T_{m,u}^{\text{back}}[n] = \frac{S}{R_{b,m,u}[n]}. \tag{14}$$

Accordingly, the delay for user u associated with the MBS (or UAV m) to obtain all of its required contents in $\mathcal{F}$ within one period T can be expressed as

$$\begin{aligned} D_{b,u} &= \sum_{n\in\mathcal{N}}\sum_{f\in\mathcal{F}} x_{u,f}[n] z_{u,b}[n] T_{b,u}^{\text{down}}[n], \\ D_{m,u} &= \sum_{n\in\mathcal{N}}\sum_{f\in\mathcal{F}} x_{u,f}[n] z_{u,m}[n] \\ &\qquad [T_{m,u}^{\text{down}}[n] + (1-y_{m,f}[n]) T_{m,u}^{\text{back}}[n]]. \end{aligned} \tag{15}$$

E. Problem Formulation

Let $\mathbf{q} = \{\mathbf{q}_m[n], \forall m, n\}$ and $\mathbf{p} = \{P_m[n], \forall m, n\}$. Combing with the aforementioned analysis, our goal is to minimize the sum content acquisition delay for all the users in the cell over the whole period T. To achieve this goal, we formulate an optimization problem by jointly designing the user association $\mathbf{z}$, cache placement $\mathbf{y}$, power allocation $\mathbf{p}$, UAV trajectory $\mathbf{q}$. Mathematically, this problem can be written as

$$\begin{aligned} \text{P1:} \min_{\mathbf{q},\mathbf{p},\mathbf{y},\mathbf{z}} \quad & \sum\nolimits_{u\in\mathcal{U}} \sum\nolimits_{i\in\mathcal{M}\cup\{b\}} D_{i,u} && \text{(16a)} \\ \text{s. t} \quad & \sum\nolimits_{f\in\mathcal{F}} y_{m,f}[n] \le C_m, \forall m && \text{(16b)} \\ & \sum\nolimits_{m\in\mathcal{M}} y_{m,f}[n] \le L_f, \forall f && \text{(16c)} \\ & \sum\nolimits_{i\in\mathcal{M}\cup\{b\}} z_{u,i}[n] \le 1, \forall u, && \text{(16d)} \\ & \sum\nolimits_{u\in\mathcal{U}} z_{u,i}[n] \le Q_i[n], \forall i \in \mathcal{M}\cup\{b\}, && \text{(16e)} \\ & \frac{1}{N}\sum_{n=1}^{N} p_m[n] \le P_{\text{avg}}, P_m[n] \le P_{\max}, \forall m, && \text{(16f)} \\ & 0 \le \vartheta_m[n] \le 2\pi, \ \forall m, && \text{(16g)} \\ & 0 \le x_m[n] \le x_{\max}, \ \forall m, && \text{(16h)} \\ & 0 \le y_m[n] \le y_{\max}, \ \forall m, && \text{(16i)} \\ & 0 \le d_m[n] \le d_{\max}, \ \forall m, && \text{(16j)} \\ & \mathbf{q}_m[n] - \mathbf{q}_i[n] \ge d_{\min}, \ \forall i \ne m, \forall n, && \text{(16k)} \end{aligned}$$

where (16b) and (16c) denote the cache capacity of each UAV and limitation of the number of UAVs for each cached content; (16d) and (16e) indicate that each user is associated with up to one node and each node serves $Q_i[n]$ users in each time slot; (16f) represents the transmission power constraint; (16g)-(16k) describe the movement policy of each UAV.

The presented problem is shown to be a mixed-integer non-convex programming problem, which may be difficult to solve since it includes tremendous binary discrete decision variables and a highly non-convex objective function. To solve such non-convex optimization problems, most works quantize them into several convex subproblems and then solve these subproblems alternately in an iterative manner until the algorithm converges. Such quantization makes the original problem easier to tackle but at the cost of accuracy. Moreover, the optimized results are only applicable to the current environment, while the standard iterative algorithm using the block coordinate descent method will fail when the environment changes. In the following section, we model our problem as a partially observable stochastic game and propose a Dual-Clip Proximal Policy Optimization (DC-PPO)-based algorithm to solve, which falls into the actor-critic framework composed of an interactive pair of policy and value networks. The DC-PPO-based algorithm is effective to tackle the uncertainties in the dynamic environment since it can learn and estimate values through observations.

III. Learning System for Multi-UAV Cooperated Caching and Communication

In this section, we will build a learning system to solve the formulated non-convex optimization problem. In order to cope with the changing UAV positions, we use a stochastic game to model the formulated problem first and then propose a novel DC-PPO-based algorithm to solve it.

A. Game Formulation

Since the optimization objective of the formulated problem is to minimize the sum content acquisition delay of users in a dynamic content transmission system, we model problem (15) as a stochastic game. In general, the stochastic game model is expressed by a tuple $\langle \mathcal{K}, \mathcal{S}, \mathcal{O}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, where $\mathcal{K}$ is the set of agents; $\mathcal{S}$ denotes the state space; $\mathcal{O}$ denotes the observation space; $\mathcal{A}$ denotes the action space; $\mathcal{R}$ denotes the reward space; $\mathcal{P}$ denotes the state transition probability; and $\gamma \in [0,1]$ denotes the reward discount factor.

At each coherence time slot n, the environment state is given by $\boldsymbol{s}[n]$ where $\boldsymbol{s}[n] \in \mathcal{S}$. Accordingly, each agent k receives a local observation $o_k[n]$ of the environment, determined by the observation function $o_k[n] \triangleq b(\boldsymbol{s}[n], k)$, and then chooses an action according to the observation $a_k[n] \triangleq \pi(o_k[n])$, forming a joint action $\mathcal{A}[n]$, where $\pi_k(\cdot)$ represents the policy function of agent k. In order to encourage cooperation between agents, multiple agents operate with the same reward. After choosing $\mathcal{A}[n]$, each agent receives a reward $r_k[n]$ based on the reward function $\mathcal{R}[n]$, where $r_1[n] = r_2[n] = \cdots = r_K[n]$. The environment then turns to the next state $\boldsymbol{s}[n+1]$ based on the transition probability function $P(\boldsymbol{s}[n+1] | \boldsymbol{s}[n], a_1[n], \cdots, a_K[n])$. In our stochastic game model, serval key elements are described below in detail.

B. Agent

In this learning system, we define the controller in both the MBS and each UAV as an agent. Each agent has its own actor network and critic network, which are served as the execution policy and the policy evaluator, respectively. As shown in Fig. 2, each agent receives a local observation and takes an action based on its policy. The environment gives each agent a reward after taking $\mathcal{A}[n]$ and then evolves to its next state.

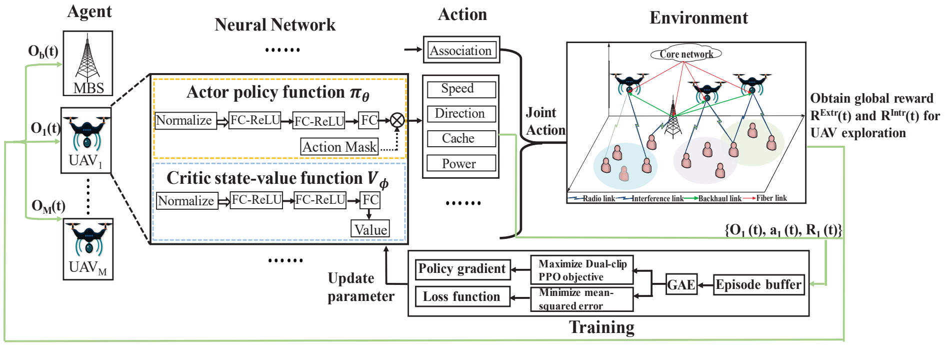


Fig. 2. The framework of Dual-Clip Proximal Policy Optimization (DC-PPO)-based algorithm for multi-UAV cooperative networks.

C. State and Observation Space

Each agent can only obtain the information about the current environment state by running an observation function.

1) UAV-agent: As mentioned earlier, the content acquisition delay is directly related to the channel condition between users and UAVs. It is very difficult to acquire accurate channel state information in practice due to the high mobility of UAVs. In contrast, each UAV can perfectly know the locations of ground users by deploying a synthetic aperture radar on it. Hence, the observation of each UAV consists of its own location and the locations of all the users at the current time slot and the request distribution of each user for all required contents at the current time slot. In particular, the observation space $o_m[n]$ of UAV m at time slot t is described as

$$\begin{aligned} o_m[n] =&\{x_m[n], y_m[n], \{x_u[n]\}_{u\in\mathcal{U}}, \\ &\{y_u[n]\}_{u\in\mathcal{U}}, \{x_{u,f}[n]\}_{u\in\mathcal{U}\cup f\in\mathcal{F}}\}. \end{aligned} \tag{17}$$

Similar to [33], we normalize the variables in (17), which are rewritten as follows

$$\begin{aligned} o_m[n] =&\{\overline{x}_m[n], \overline{y}_m[n], \{\overline{x}_u[n]\}_{u\in\mathcal{U}}, \\ &\{\overline{y}_u[n]\}_{u\in\mathcal{U}}, \{x_{u,f}[n]\}_{u\in\mathcal{U}\cup f\in\mathcal{F}}\}. \end{aligned} \tag{18}$$

where $\overline{x}_m[n] = x_m[n]/x_{\max}$, $\overline{y}_m[n] = y_m[n]/y_{\max}$, $\overline{x}_u[n] = x_m[n]/x_{\max}$, and $\overline{y}_u[n] = y_m[n]/y_{\max}$.

2) MBS-agent: The observation of the MBS agent includes the locations of all users and UAVs at the current time slot, and the location of the MBS at the current time slot. Consequently, the observation space of the MBS is given as

$$\begin{aligned} o_b[n] =&\{x_b[n], y_b[n], \{x_u[n]\}_{u\in\mathcal{U}}, \{y_u[n]\}_{u\in\mathcal{U}}, \\ &\{x_m[n]\}_{m\in\mathcal{M}}, \{y_m[n]\}_{m\in\mathcal{M}}\}. \end{aligned} \tag{19}$$

Similarly, we normalize the variables in (18), which are further reexpressed as

$$\begin{aligned} o_b[n] =&\{\overline{x}_b[n], \overline{y}_b[n], \{\overline{x}_u[n]\}_{u\in\mathcal{U}}, \{\overline{y}_u[n]\}_{u\in\mathcal{U}}, \\ &\{\overline{x}_m[n]\}_{m\in\mathcal{M}}, \{\overline{y}_m[n]\}_{m\in\mathcal{M}}\}. \end{aligned} \tag{20}$$

D. Action Space

Existing research shows that if each agent only learns conservative feasible solutions, its exploration becomes daunting. In this learning system, we introduce an Action-Mask module as shown in Fig. 2, which can transform the solution that does not satisfy the constraint into a feasible solution by modifying the actions of each agent for training. In particular, when a UAV agent is flying out of a given boundary, we can modify its actions to keep it within the trajectory constraints.

1) UAV: At each time slot n, each UAV needs to choose its own proper trajectory to provide content transmission services for ground users. Besides, each UAV decides how much power is required for content transmission and which content should be cached. Accordingly, the action space $a_m[n]$ of UAV m at time slot n is given by

$$a_m[n] =\{y_{m,f}[n], d_m[n], \vartheta_m[n], P_m[n]\}. \tag{21}$$

2) MBS: Consider that the MBS intensively controls which node should be selected by each user to establish a connection, the action space $a_b[n]$ of the MBS at time slot n is given by

$$a_b[n] =\{\{z_{u,i}[n]\}_{i\in b\cup\mathcal{M},u\in\mathcal{U}}\}. \tag{22}$$

It is worth noting that the action definitions in (21) and (22) include discrete variables and continuous variables, which can not be directly tackled by our learning algorithm. The reason is that conventional reinforcement learning algorithms can only resolve problems where all action definitions are either discrete or continuous, but can not address the hybrid action space.

In order to address this issue, we convert the discrete action variables $y_{m,f}[n]$ and $z_{u,i}[n]$ to continuous ones. Specifically, we use $y_{m,f}[n]\in[0,1]$ to denote the cache placement indicator and UAV agent m can select the $\lceil y_{m,f}[n]*F\rceil$-th content for caching when it preforms $y_{m,f}[n]$ for cache placement, where $\lceil\cdot\rceil$ denotes the ceiling function. In addition, we use $z_{u,i}[n] \in [0,1]$ to denote the user association indicator and the MBS can control user u to choose the $\lceil z_{u,i}[n]*(1+M)\rceil$-th node as the access node when it takes $z_{u,i}[n]$ for user access selection. We normalize the variables $d_m[n]$, $\vartheta_m[n]$ and $P_m[n]$ in order to facilitate the elimination of the impact of large differences among variables on model performance. As a result, the action space $a_m[n]$ is rewritten as

$$a_m[n] =\{y_{m,f}[n], \overline{d}_m[n], \overline{\vartheta}_m[n], \overline{P}_m[n]\}. \tag{23}$$

where $\overline{d}_m[n] = d_m[n]/d_{\max}$, $\overline{\vartheta}_m[n] = \vartheta_m[n]/\vartheta_{\max}$, and $\overline{P}_m[n] = P_m[n]/P_{\max}$.

Many experiments reveal that each UAV has poor mobility when it moves within the horizontal distance $d_m\in[0,d_{\max}]$ and direction $\vartheta_m[n]\in[0,2\pi]$. To tackle this issue, we introduce an experience setting and change the range of movement distance and direction for each UAV as $[-d_{\max}, d_{\max}]$ and $[-\pi,\pi]$.

E. Reward Design

The definition of the reward is mandatory in order to prompt each agent to take proper action. After executing those selected actions, each agent can obtain an immediate reward in a certain state at each time slot. There have been many works suggesting the use of intrinsic rewards to motivate agents to explore before receiving any extrinsic rewards [34]. In this system, the reward of each UAV agent at each time slot n consists of the extrinsic reward and the intrinsic reward, which is expressed as

$$R_m[n] = R^{\mathrm{Extr}}[n] + \epsilon R_m^{\mathrm{Intr}}[n], \forall m \in \mathcal{M}, \tag{24}$$

where $R^{\mathrm{Extr}}[n]$ denotes the extrinsic reward provided by the environment; ϵ denotes the scaling hyperparameter; and $R_m^{\mathrm{Intr}}[n]$ denotes the intrinsic reward of each UAV from the exploration criterion. Suppose that all UAV agents share the same extrinsic reward to facilitate cooperative behavior between them.

1) Extrinsic Reward: It is clear that the extrinsic reward is generally related to the objective function. According to (16), we find that the optimization objective is to minimize the sum content acquisition delay of users within a given flight period T. Therefore, the extrinsic reward function at each time slot n is defined as

$$R^{\mathrm{Extr}}[n] = L - \sum_{i \in \{b\} \cup \mathcal{M}} \sum_{u \in \mathcal{U}} D_{i,u}[n], \forall i \in \{b\} \cup \mathcal{M}, \tag{25}$$

where L is a large value to ensure that the extrinsic reward is greater than 0. Note that minimizing the cost is equivalent to maximizing the reward. In addition, $D_{b,u}[n]$ and $D_{m,u}[n]$ are formulated as

$$D_{b,u}[n] = \sum_{f \in \mathcal{F}} x_{u,f}[n] z_{u,b}[n] T_{b,u}^{\mathrm{down}}[n], \forall i \in \{b\}, \tag{26}$$

$$\begin{aligned} D_{m,u}[n] = \sum_{f \in \mathcal{F}} & x_{u,f}[n] z_{u,m}[n] [T_{m,u}^{\mathrm{down}}[n] \\ & + (1 - y_{m,f}[n]) T_{m,u}^{\mathrm{back}}[n]], \forall i \in \{\mathcal{M}\}. \end{aligned} \tag{27}$$

2) Intrinsic Reward: In this learning system, we design the intrinsic reward to motivate each agent to constantly learn from the environment. In an environment where the state transition is reversible, simply using intrinsic reward to guide exploration will result in agents going back and forth between novel states $\boldsymbol{s}[n+1]$ and their previous states $\boldsymbol{s}[n]$. To deal with this issue, the BeBold-based exploration criterion [35] uses an aggressive restriction in which each agent is rewarded only when it visits the state $\boldsymbol{s}[n]$ for the first time in an episode. Thus, the intrinsic reward of each UAV agent m at time slot n is defined as

$$\begin{aligned} & R_m^{\mathrm{Intr}}[n] = \mathbb{1}\{N_e(x_m[n+1], y_m[n+1]) = 1\} \\ & * \max\left(\frac{1}{N(x_m[n+1], y_m[n+1])} - \frac{1}{N(x_m[n], y_m[n])}, 0\right), \end{aligned} \tag{28}$$

where $N_e(x_m[n+1], y_m[n+1])$ represents the episodic state count and will be reset every episode.

It should be noted that the reward of the MBS agent at each time slot only contains the extrinsic reward, which is expressed by (25). The aim of this learning system is to find an optimal policy π_*, which maximizes the cumulative discounted reward under the discount factor γ. Hence, the cumulative discounted reward is expressed as

$$R^{\mathrm{cumu}} = \sum\nolimits_n^N \gamma_i R_i[n], \forall i \in \{b\} \cup \mathcal{M}. \tag{29}$$

F. Transition Probability

We use $P(\boldsymbol{s}[n+1] | \boldsymbol{s}[n], a_1[n] \cdots, a_I[n])$ for $i \in \{b\} \cup \mathcal{M}$ to denote the state transition probability, which indicates the probability distribution of the next state after all agents execute their corresponding actions under the current state.

IV. DC-PPO-based Joint Optimization Algorithm

Based on the above-mentioned stochastic game model, it is difficult for the MBS and UAVs to receive enough information to specify their state transition functions. Therefore, we need a model-free algorithm that does not require a priori information of all state transition functions to solve the resulting problem. Since the association control for the MBS and the transmission position and cache placement of each UAV are continuous-valued, the action space of the stochastic game defined in the above section is infinite, which makes tabular-value-based algorithms, such as Q-learning [36], unsuitable for this problem. In addition, policy-based algorithms such as deep deterministic policy gradient (DDPG) [37] may not be able to properly solve this problem, since it may suffer from high variance when the policies of multiple agents are optimized simultaneously. Therefore, we resort to the Dual-Clip Proximal Policy Optimization (DC-PPO) algorithm to solve our formulated stochastic game problem, which follows the actor-critic architecture that composes of an interactive pair of policy and value networks. There are several reasons for choosing the DC-PPO-based algorithm to solve the proposed joint optimization problem. Firstly, the DC-PPO-based algorithm has been recognized as a promising algorithm for the trajectory design of UAVs in [38]. Secondly, the DC-PPO-based algorithm has outstanding performance and lower computational complexity.

In this section, we propose a DC-PPO-based algorithm for cache placement, user association, power allocation, and UAV trajectory design. We first provide a simple introduction for the DC-PPO-based algorithm that is a deep reinforcement learning (DRL) algorithm based on the actor-critic framework [39]. We then describe the proposed DC-PPO-based algorithm in detail.

A. Preliminaries

PPO is a model-free, on-policy, policy-gradient, actor-critic reinforcement learning algorithm designed at OpenAI [40]. In addition, PPO-based algorithms can be roughly classified into two categories: Penalty-PPO and Clip-PPO. The former adopts the Kullback-Leibler (KL) divergence to exchange the policy [41], while the latter used in our formulated learning system relies on a specific clipping technique in the objective function. It is clear that the Clip-PPO-based algorithm can approximate the hard constraints applied to the PPO-based learning system by using much more effortless equations. Thus, the Clip-PPO-based algorithm is much simpler algorithm that demonstrates its significant efficiency.

In the learning system, we use π to index the policy network which is optimized with respect to its parameterization θ. The policy network takes the local observation $\boldsymbol{o}$ as its input and then outputs an action $\boldsymbol{a}$, consisting of the user association, cache placement, UAV location, and transmission power. For a continuous action space, the policy network is tasked to output the moment of a probability distribution, where the mean and variance of a multivariate Gaussian can be derived from the set of actions. In the training phase, actions are sampled randomly according to this distribution to increase exploration, while the mean is taken as the action when the training is completed.

Due to each agent directly learns the action policy, the DC-PPO-based algorithm is returned to the class of policy gradient algorithms developed in the past decade. The policy gradient algorithm works by iteratively updating the policy parameters using the stochastic gradient update technique. Similar to [42], the gradient is estimated in a Monte Carlo manner by running the policy in the environment to obtain the sample of the policy loss $J(\theta)$ and its gradient, which are respectively given by

$$\begin{aligned} J(\theta) &= \mathbb{E}_{\boldsymbol{o},\boldsymbol{a}\sim\pi_\theta}\left[\sum_t R(\boldsymbol{s}_t,\boldsymbol{a}_t)\right] = \mathbb{E}_{\boldsymbol{o},\boldsymbol{a}\sim\pi_\theta}[R_t], \\ \nabla_\theta J(\theta) &= \mathbb{E}_{\boldsymbol{o},\boldsymbol{a}\sim\pi_\theta}\left[\left(\sum_{t=1}^{T}\nabla_\theta \log\pi_\theta(\boldsymbol{a}_t|\boldsymbol{s}_t)\right)R_t\right]. \end{aligned} \tag{30}$$

The main challenge of the policy gradient algorithm lies in reducing the variance of policy gradient estimates. We propose a series of policy gradient estimators that substantially reduce variance while maintaining a tolerable level of bias. Moreover, we call this variance reduction scheme parameterized by $\gamma \in [0,1]$ and $\lambda \in [0,1]$ as the generalized advantage estimation. According to [43], we define the generalized advantage estimation as the exponentially-weighted average of l-step estimators, which is expressed as

$$\begin{aligned} \hat{A}_t(\gamma,\lambda) &= \sum_{l=1}^{\infty}(\gamma\lambda)^l\delta_{t+1}^V \\ &= \sum_{l=1}^{\infty}(\gamma\lambda)^l(R_t+\gamma V(s_{t+l+1})-V(s_{t+l})), \end{aligned} \tag{31}$$

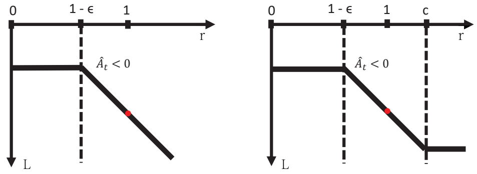
where γ corresponds to the discount factor used in the cumulative discounted reward function (29). It is worth noting that the generalized advantage estimation for $\lambda \in [0,1]$ can make a fundamental bias-variance tradeoff.

B. Learning Algorithm Design

The proposed DC-PPO-based algorithm for the cache placement and multiuser association jointly with the UAV trajectory design is summarized in Algorithm 1. We consider an episodic setting with each episode spanning each UAV battery lifetime constraint T. Each episode begins with a randomly initialized environment state (determined by the initial location and cache placement of each UAV, the initial user association, etc.) and continues until the end of T. The high mobility of UAVs leads to the transition of the environment state and makes each agent adjust its action. The proposed DC-PPO-based algorithm uses the paradigm of centralized training and distributed learning. Each agent receives its own observation $\mathcal{O}_i[n], i \in \mathcal{M} \cup \{b\}$ at time slot n and then selects the action generated from the shared policy π_θ, where the policy is trained with experiences collected simultaneously by all agents. The training process alternates between collecting experiences by running the policy in parallel. The MBS agent and each UAV agent leverages the shared policy, where the MBS agent controls the association and scheduling of users, while each UAV agent stores content within its limited cache capacity and controls its trajectory and transmitted power. All agents generate a joint action $\mathcal{A}[n]$ to interact with the environment and receive a global reward

Algorithm 1 DC-PPO-based algorithm

```
Input:
  Environment $\mathbf{E}$;
  Observation Space $\mathcal{O}$;
  Action Space $\mathcal{A}$;
Process:
  Initialize policy parameter $\theta_i^0$, state-value function parameter $\phi_i^0$ and episode buffer $\mathcal{D}_i$ for each agent $i$;
  for each episode do
    # Collect experiences of all agents
    for time slot $n = 0, \ldots, N$ do
      for each agent $i \in \{b\} \cup \mathcal{M}$ do
        Observe space $o_i[t]$ and choose action $a_i[n]$ by running its policy function $\pi_i[n] = \pi(o_i[t])$.
      end for
      All agents perform a joint action $\mathcal{A}_t$ and interact with the environment to receive a global extrinsic reward $R^{\text{Extr}}[n]$ according to (25).
      for each UAV agent $m$ do
        Compute its intrinsic reward $R_m^{\text{Intr}}[n]$ according to (28) and get its total immediate reward $R_n[t]$ according to (24).
        Store $\tau_m[n] = (o_m[n], a_m[n], R_m[n], o_m[n+1])$ into the episode buffer $\mathcal{D}_m$.
      end for
    end for
    # Perform reinforcement learning algorithm
    for each agent $i$ do
      Compute reward $\hat{R}_i[n]$ in the episode buffer $\mathcal{D}_i$;
      Compute generalized advantage estimates $\hat{A}[n]$ based on the current value function $V_{\phi_n}$;
      Update the policy by maximizing the Dual-Clip PPO objective typically through stochastic gradient ascent with Adam:
        $\theta_i[n+1] = \arg\max_{\theta} \frac{1}{|\mathcal{D}_i|N} \sum_{\tau\in\mathcal{D}_i}\sum_{n=0}^{N} \max\big(\min\big(\hat{A}_{\theta_i^{(n)}}[n] r(\theta), \mathrm{clip}(r(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_{\theta_i^{(n)}}[n]\big), c\hat{A}_{\theta_i^{(n)}}[n]\big).$
      Fit value function by minimizing the mean-squared error through the stochastic gradient algorithm:
        $\phi_i[n+1] = \arg\min_{\phi} \frac{1}{|\mathcal{D}_i|N} \sum_{\tau\in\mathcal{D}_i}\sum_{n=0}^{N} (V_{\phi_i[n]}(o_i[n]) - \hat{R}_i[n]).$
    end for
  end for
Output: the learned policies $\pi$ of all agents.
```



(a) (b)

Fig. 3. (a) Standard PPO-based algorithm (clip with ϵ) and (b) Dual-clip PPO-based algorithm (clip with ϵ and c where $\hat{A}_{\theta_{\text{old}}}(s,a) < 0$).

$\mathcal{R}[n+1]$. Each agent obtains $\mathcal{R}_i[n+1], i \in \mathcal{M} \cup \{b\}$ depending on the content acquisition delay of all users according to (25). The reward is used to verify which actions are beneficial and update the critic network. The actor network outputs the policy and the critic network evaluates the current policy by accessing the observation and action of each agent during the centralized training phase. There are two types of PPO: standard PPO and Dual-Clip PPO, both of which use gradient clipping to ensure that poor actions do not disrupt the training.

1) Standard PPO-based Algorithm: The output of the critic network is a component of the loss function for the actor. We use the sampled experiences to construct the loss that contains two parts: the loss of the critic network $L_{\text{value}}^{\text{PPO}}(\phi)$ and the loss of the actor network $L_{\text{policy}}^{\text{PPO}}(\theta)$. The network structure of the state-value function $V_\phi(s_i[n])$ is the same as that of the policy function π_θ, except that the former has only one unit in its last layer with a linear activation. The state-value function can be used to estimate the advantage $\hat{A}_i[n]$ based on the generalized advantage estimation [43]. We construct the squared-error loss $L_{\text{value}}^{\text{PPO}}(\phi)$ as follow

$$L_{\text{value}}^{\text{PPO}}(\phi) = (V_{\phi_i[n]}(s_i[n]) - R_i[n])^2, \tag{32}$$

which is optimized with the Adam Optimizer. The PPO-based algorithm receives the expectation of samples collected from the old policy $\pi_{\theta_{\text{old}}}$ under the new policy that needs to refine π_θ. The probability ratio between the old policy and the new policy is expressed as

$$r(\theta) = \frac{\pi_\theta(a|s)}{\pi_{\theta_{\text{old}}}(a|s)}, \tag{33}$$

Since the probability ratio $r(\theta)$ may be large, the maximization of the surrogate objective function may result in an excessive policy deviation. To tackle this issue, the standard PPO-based algorithm imposes the constraint by forcing $r(\theta)$ to stay within a small interval around 1, denoted by $[1-\epsilon, 1+\epsilon]$, where ϵ is a hyperparameter that penalizes extreme changes in the policy, which is described as

$$\begin{aligned} L_{\text{policy}}^{\text{PPO}}(\theta) =& \mathbb{E}\big[\min\big(r(\theta)\hat{A}_{\theta_{\text{old}}}(s,a), \\ & \text{clip}(r(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_{\theta_{\text{old}}}(s,a)\big)\big], \end{aligned} \tag{34}$$

where the function $\text{clip}(r(\theta), 1-\epsilon, 1+\epsilon)$ makes the ratio not greater than $1+\epsilon$ and not less than $1-\epsilon$.

2) Dual-Clip PPO-based Algorithm: In the proposed large-scale training environment, we find that the use of actions with negative advantage functions will cause a negative impact on the policy. In particular, when $\pi_\theta(\boldsymbol{a}|\boldsymbol{s}) \gg \pi_{\theta_{\text{old}}}(\boldsymbol{a}|\boldsymbol{s})$, the ratio $r(\theta)$ is a larger value. When $\hat{A}_{\theta_{\text{old}}}(\boldsymbol{s},\boldsymbol{a}) < 0$, such a large ratio will lead to an unbounded variance due to $r(\theta)\hat{A}_{\theta_{\text{old}}}(\boldsymbol{s},\boldsymbol{a}) \ll 0$. It is clear that the old policy and the new policy will diverge, which makes it challenging to ensure the policy convergence. Hence, we propose a dual-clip PPO-based algorithm to support our formulated large-scale training environment, which clips the ratio $r(\theta)$ with a lower bound of the value $r(\theta)\hat{A}_{\theta_{\text{old}}}(\boldsymbol{s},\boldsymbol{a})$. Fig. 3 shows the clipping of the standard PPO-based algorithm and the Dual-Clip PPO-based algorithm. When $\hat{A}_{\theta_{\text{old}}}(\boldsymbol{s},\boldsymbol{a}) < 0$, the new objective function of the proposed dual-clip PPO algorithm is expressed as

$$\begin{aligned} L_{\text{policy}}^{\text{DC-PPO}}(\pi) = \mathbb{E}\big[&\max\Big(\min\big(r(\theta)\hat{A}_{\theta_{\text{old}}}(\boldsymbol{s},\boldsymbol{a}), \\ & \text{clip}(r(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_{\theta_{\text{old}}}(\boldsymbol{s},\boldsymbol{a})\big), c\hat{A}_{\theta_{\text{old}}}\Big)\big], \end{aligned} \tag{35}$$

where $c > 1$ is a constant that indicates the lower bound.

V. SIMULATION RESULTS

In this section, we provide simulation results to demonstrate the content distribution performance of our proposed DC-PPO-based algorithm with the joint design of the user association, cache placement, UAV trajectory and transmission power.

A. Simulation Setup

We consider a cache-enabled UAV-assisted cellular network where $U = 10$ users are arbitrarily and fixedly distributed in a square area with the size of 1.8 km $\times$ 1.8 km. For comparison, the total number of users will increase from 10 to 50. There are a static MBS and $M = 4$ UAVs deployed to jointly provide content transfer services for ground users within this area. The MBS is located at $[1000, 800]$. The initial horizontal locations of the four UAVs are $[800, 1200]$, $[1200, 1200]$, $[800, 600]$ and $[1200, 600]$, respectively. The departure direction of each UAV is randomly generated in an angle interval of $[0, 2\pi]$. All UAVs fly at a given altitude $H = 200$ m within a finite flight period $T = 100$ s. For convenience, the flight period T is partitioned into multiple time slots with the duration of $d_t = 0.5$ s. The initial transmission power of each UAV is set as the maximum transmission power $P_{\text{max}} = 1$ W. The average power budget of each UAV is fixed to $P_{\text{avg}} = 0.25$ W. To avoid collisions between UAVs, the minimum safe distance is set as $D_{\text{min}} = 1$ m. In addition, the maximum distance that each UAV can travel during one time slot is $d_{\text{max}} = 25$ m. At each time slot n, each user randomly requests one content f from the content library consisting of $F = 30$ contents with probability $x_{u,f}[n]$. The size of each content is equal and is set as $S = 10$ Mbits. We assume that each user makes a request for a content as per the Zipf distribution, which has been widely used to model the content popularity [44]–[46]. Thus, the probability that user u requests content f at time slot n can be expressed as

$$x_{u,f}[n] = \frac{\frac{1}{f^\kappa}}{\sum_{f=1}^{F} \frac{1}{f^\kappa}}, \tag{36}$$

where the Zipf parameter κ is fixed to $\kappa = 0.8$, which reflects the skewness of each user's preference for each content. Each content f can only be stored on at most $L_f = 2$ UAVs. Each UAV m is required to have a limited cache memory with a

TABLE I
NUMERICAL CALCULATION PARAMETER SETTINGS

Description	Symbol	Value
MBS transmission power	P_b	2 W
Radio link bandwidth	W	10 MHz
Backhaul link bandwidth	B	20 MHz
Speed of light	v_c	$3 * 10^8$
Carrier frequency	f_c	2 GHz
Shadowing factor	$\chi_{\text{LoS}}, \chi_{\text{NLoS}}$	6dB, 20dB
Environmental factor	c_1, c_2	11.9, 0.13
Additional path loss factor	η	20 dB
Path loss exponent	α	2
Noise variance	σ^2	-100 dB/Hz
Quota of each UAV	Q_m	4
Quota of MBS	Q_b	20

capacity of C_m contents, which will increase from 3 to 7. Note that the setting of communication-related parameters follows the 3GPP specification [47], which are shown in Table I.

In order to illustrate the effectiveness of the formulated joint design scheme, we consider the following several benchmark schemes with stochastic cache placement and user association or without power and trajectory optimization:

- Stochastic cache placement scheme: UAV trajectory and transmission power as well as user association are jointly optimized with stochastic cache placement;
- Stochastic user association scheme: UAV trajectory and transmission power as well as cache placement are jointly optimized with stochastic user association;
- Fixed UAV trajectory: cache placement and user association as well as transmission power are jointly optimized with given UAV trajectory where each UAV flies along a circle trajectory defined in [48].

B. Network Architecture

We conduct the experimental simulations using a server with an NVIDIA GTX 2080 Ti GPU. The software platform of the experiment is Python 3.6 with PyTorch [49]. It is clear that the proposed DC-PPO-based algorithm consists of the actor-network and critic-network, each of which has one input layer, two hidden layers and one output layer. Moreover, each hidden layer is assumed to have the same number of neurons and is set as $e = 64$. We use the rectified linear unit (ReLU) function $f_{\text{ReLU}}(x) = \max\{0, 1\}$ to describe the activation function in each hidden layer. The Adam optimizer can be used to update the actor network and critic network. The learning rate for both neural networks is set as 0.0001. The clip parameter and the discount factor in our proposed algorithm are set as $\epsilon = 0.2$ and $\gamma = 0.999$, respectively. In addition, the training process of our proposed algorithm has $N_{\text{ept}} = 20000$ episodes, each of which contains $N = 100$ or $N = 200$ time slots.

C. Result Analysis

In Fig. 4, we investigate two UAVs case where one MBS and $M = 2$ UAVs cooperatively provide content transfer services for $U = 20$ ground users. Fig. 4 shows the trajectories of UAV$_1$ and UAV$_2$ projected onto the horizontal plane for $T = 100$ s. All users are assumed to be uniformly distributed in this figure and the MBS is located at $[1000, 800]$. We use the black circle to denote the positions of ground users. As can be seen, UAV$_1$ and UAV$_2$ begin from $[600, 800]$ and $[1400, 800]$, respectively, which are marked with blue stars. Although all the users are randomly distributed in this square area, it can be seen from Fig. 4 that the optimized trajectories of the two UAVs can well match the distribution of all users. During the flight period T, UAV$_1$ and UAV$_2$ constantly learn and update their movement policies to provide better channel conditions and lower delay content transfer services for ground users. As expected, UAV$_1$ and UAV$_2$ fly to the ground users and stay near their serving users for a certain amount of time. In addition, the two UAVs try to stay as far away from each other as possible to alleviate the co-channel interference.

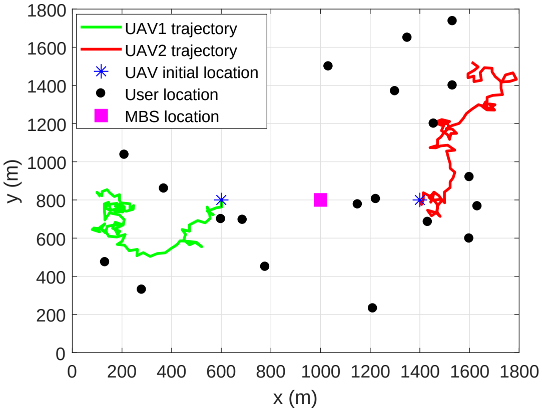


Fig. 4. Optimized UAV trajectory for a two-UAV assisted cellular network under the period $T = 100$ s.

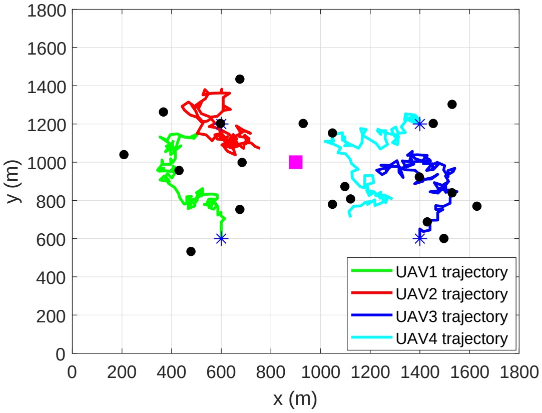
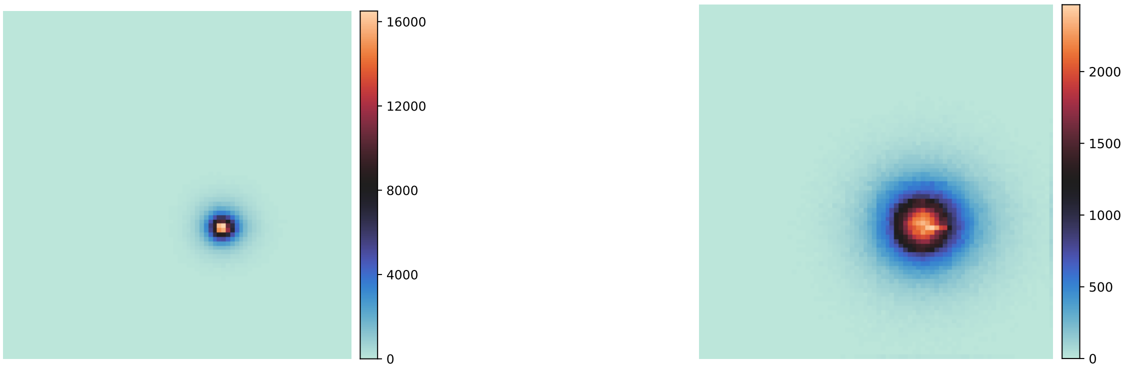


Fig. 5. Optimized UAV trajectory for a four-UAV assisted cellular network under the period $T = 100$ s.

In Fig. 5 and Fig. 6, we investigate four UAVs case in which one MBS and $M = 4$ UAVs are employed to transmit contents to $U = 20$ users. Fig. 5 and Fig. 6 depict the trajectories of the four UAVs obtained by the proposed DC-PPO-based algorithm under $T = 100$ s and $T = 200$ s, respectively. Similar to Fig. 4, the black circle and magenta square index the positions of the



(a) PPO

(b) DC-PPO+BeBold

Fig. 7. Heatmaps for the location of UAV$_3$ agent learnt with different algorithms in the four-UAV environment at 500K training steps. The color depth reflects the number of visitation counts for the location of UAV$_3$ agent.

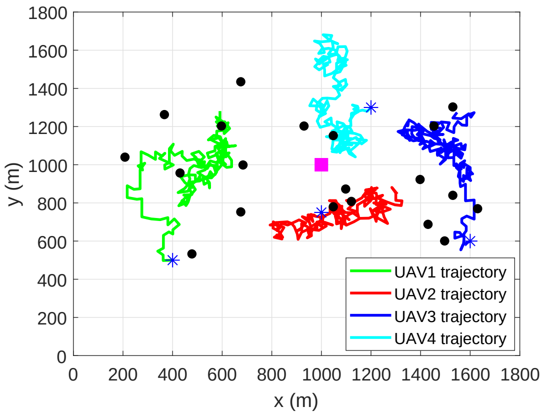


Fig. 6. Optimized UAV trajectory for a four-UAV assisted cellular network under the period $T = 200$ s.

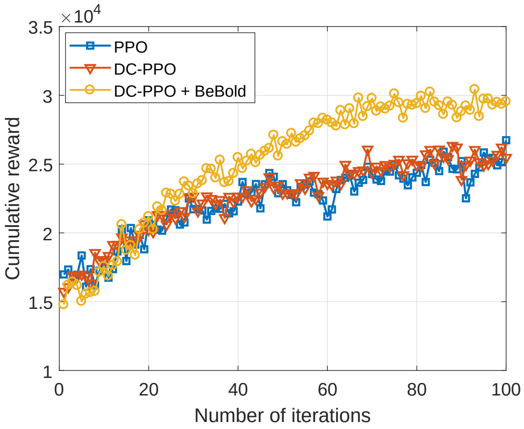


Fig. 8. Cumulative reward versus the number of iterations for a four-UAV system with different algorithms.

MBS and users, respectively. Then the four UAVs begin from their initial positions, which are provided in Section V-A. By optimizing the trajectories of the four UAVs, we cannot only establish high-quality wireless links for the desired UAV-user pairs, but also enlarge the interfering channel distance between those undesired UAV-user pairs to reduce the co-interference. Therefore, the proposed algorithm attempts to make a tradeoff between the co-channel interference and the good channel. It can be observed from Fig. 5 and Fig. 6 that the four UAVs fly near to their respective serving users and hover above each of them for a period of time. As expressed, all UAVs continuously adjust their trajectories to achieve a better reward as the reward is directly related to the air-to-ground channel conditions. It is clear that with the increase of T, each UAV has more freedom to move closer to the users to obtain better channel conditions, which results in the decrease in terms of the content acquisition delay.

To study how different algorithms affect the exploration of each UAV agent, we analyze the visualized results of visitation counts for the location of UAV$_3$ agent in the four-UAV system environment. The target for each UAV agent is to constantly learn and update its policy to provide content delivery services with minimum total content acquisition delay for 20 ground users randomly distributed in a square area $1.8\ \text{km} \times 1.8\ \text{km}$. We define the number of visitation counts $N(s)$ at every state as the metric to evaluate the effectiveness of the BeBold-based exploration criterion. Fig. 7 depicts the heatmap of visitation counts for the location of UAV$_3$ agent at 500K training steps with different algorithms. It can be observed from Fig. 7 that the color depth area of the standard PPO-based algorithm is obviously smaller than that of the DC-PPO-based algorithm + the BeBold-based exploration criterion. The reason is that the standard PPO-based algorithm only uses the extrinsic reward to guide exploration, which thus leads to the UAV$_3$ agent going back and forth between the new state $s_3[n+1]$ and its previous state $s_3[n]$. Moreover, the BeBold-based exploration criterion is capable of providing an intrinsic reward to the UAV$_3$ agent when it explores beyond the boundary of explored areas, which motivates the UAV$_3$ agent for exploration before its extrinsic reward is received.

In Fig. 8, we show the cumulative reward versus the number of iterations under the following algorithms: (i) the DC-PPO-based algorithm; (ii) the standard PPO-based algorithm; and (iii) the DC-PPO-based algorithm + the BeBold-based exploration criterion. The received reward of each agent contains the intrinsic reward from the exploration criterion and the extrinsic

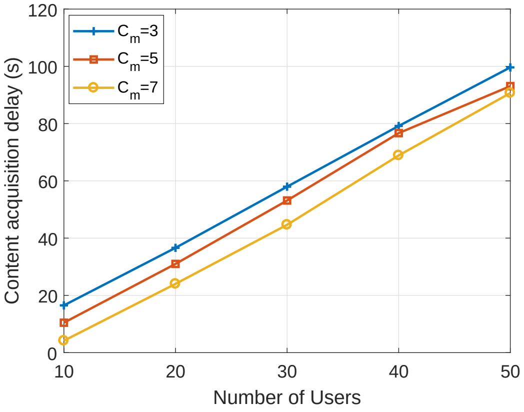


Fig. 9. The content acquisition delay versus the number of users for a four-UAV system with different cache capacities C_m.

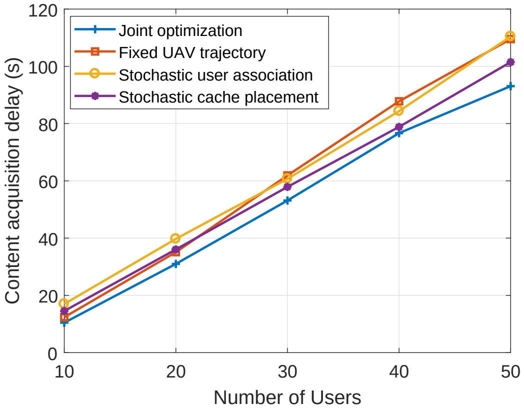


Fig. 11. Computation of the content acquisition delays achieved by different design schemes for a four-UAV system, where $F = 40$.

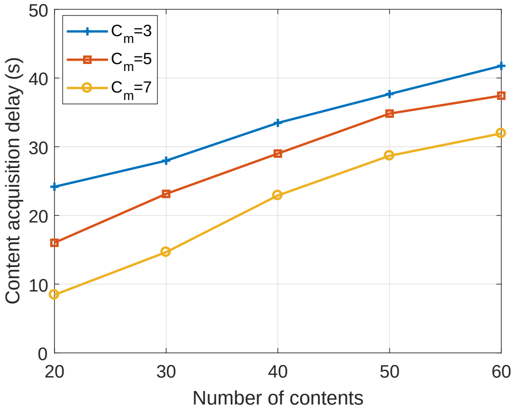


Fig. 10. The content acquisition delay versus the number of contents for a four-UAV system with different cache capacities C_m.

reward given by the environment. Moreover, the BeBold-based exploration criterion can help each agent explore Beyond the Boundary of explored regions. With the increase of the number of iterations, we find that the cumulative rewards of the three algorithm have an obvious tendency to increase and converge. It is notable that the proposed DC-PPO+BeBold algorithm is guaranteed to nearly converge at 65 iterations, while the DC-PPO-based and the standard PPO-based algorithm converge after 85 and 75 iterations, respectively. It can also be seen that the proposed DC-PPO+BeBold algorithm achieves a significant higher cumulative reward than the DC-PPO-based algorithm and the standard PPO-based algorithm, which further verifies the superiority of the DC-PPO+BeBold algorithm.

In Fig. 9, we show the relationship between the total content acquisition delay and the number of users under varying cache capacity, where $M = 4$ and $F = 40$. It can be seen from Fig. 9 that the total content acquisition delay for the three different cache capacities increases monotonically with the number of users. It should be noted that the state space and action space increases largely as the number of users increases, which thus decreases the probability that the optimal legal action is taken. It can also be seen from Fig. 9 that the total content acquisition delay in the case of $C_m = 7$ is significantly smaller than that in the case of $C_m = 3$. The reason is that as the cache capacity becomes large, each UAV m can prestore more contents in its cache memory and users associated with this UAV have more possibilities to directly access their requested contents.

In Fig. 10, we plot the total content acquisition delay versus the number of contents with different cache capacities, where $M = 4$ and $U = 20$. It can be seen from Fig. 10 that the total content acquisition delay increases monotonically for the three different cache capacities as the number of contents increases. The reason is that increasing the number of contents largely increases the state space and legal action space, which results in the decrease of the probability that the optimal legal action is taken. It is clear that the case of $C_m = 7$ contents achieves significantly smaller delay as compared with the two cases of $C_m = 3$ and $C_m = 5$ contents. This is because when $C_m = 7$, most of the multimedia contents requested by the ground users can be stored at different UAVs instead of being fetched from the MBS using the backhaul links of the UAVs.

Fig. 11 compares the content acquisition delay achieved by the following schemes: (i) the proposed joint design scheme; (ii) the stochastic cache placement scheme; (iii) the stochastic user association scheme; and (iv) the fixed trajectory scheme. It can be seen from Fig. 11 that the content acquisition delays of all the four scheme increases largely as the number of users increases. In addition, we note that the curves of the stochastic cache placement scheme and the fixed UAV trajectory scheme are very close to each other in the regime of $U \leq 23$, while the former scheme obtains lower content acquisition delay than the latter scheme when $U \geq 23$. Such results suggest that the cache placement optimization is more important for reducing the content acquisition delay when the number of ground users is smaller, while the trajectory optimization is more prominent with increasing U. Note that the proposed joint design scheme always achieves great gain compared with the three benchmark schemes in term of the content acquisition delay, which further verifies the effectiveness of the proposed joint design scheme. It is clear that the performance gaps between the proposed joint design scheme and the three benchmark schemes increase with the number of users. This means that the proposed joint design

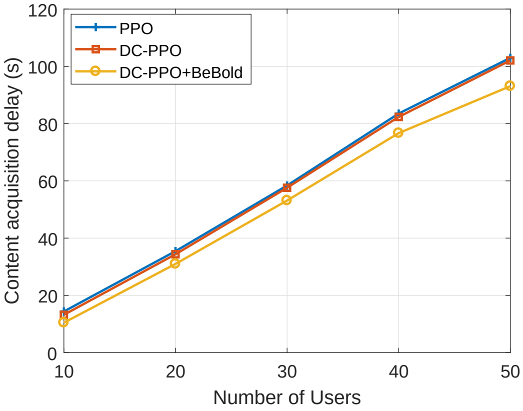


Fig. 12. Computation of the content acquisition delays achieved by different optimization algorithms for a four-UAV system, where $F = 40$.

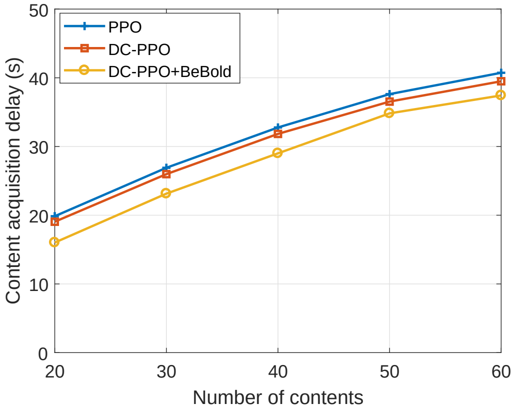


Fig. 13. The content acquisition delays for different number of contents and different optimization algorithms under a four-UAV system, where $U = 20$.

scheme is more capable of providing efficient and low-delay content transfer services for users. All those results in Fig. 11 show that our proposed joint design scheme on UAV trajectory and power control, user association, and fully using the storage capacity per UAV, has a prominent impact on minimizing the content acquisition delay.

In Fig. 12, we compare the content acquisition delay brought by the following algorithms: (i) the DC-PPO-based algorithm; (ii) the standard PPO-based algorithm; and (iii) the proposed DC-PPO+BeBold algorithm. It is observed that the proposed algorithm always achieves the lowest content acquisition delay as compared with the standard PPO-based algorithm and the DC-PPO-based algorithm, which verifies the superiority of the proposed algorithm. It is notable that the content acquisition delay gaps between the proposed algorithm and the other two algorithms increases with the increase of the user number. The reason is that the increase of the number of users results in a lager state action and action space, which hence increases the difficulty in finding an optimal policy for the MBS and each UAV. For different values of U, the content acquisition delay achieved by the standard PPO-based algorithm is observed to be much less than that achieved by the proposed algorithm.

In Fig. 13, we show the impact of the number of contents on the content acquisition delay under different algorithms. In addition, we compare the proposed DC-PPO+BeBold algorithm with the standard PPO-based algorithm and the DC-PPO-based algorithm. It is clear that the proposed DC-PPO+BeBold algorithm can achieve a great performance improvement compared with the other two algorithms in terms of content acquisition delay. In the presence of $F \leq 40$ contents, the standard PPO-based algorithm can achieve a content acquisition delay close to the DC-PPO-based algorithm, while the former achieves a little lower delay. In addition, we can observe that the content acquisition delay increases gradually for the three algorithms as the total number of contents grows. It can also be seen that the performance gaps between the proposed algorithm and the other two algorithms become larger with increasing F. All the numerical results in Fig. 12 and Fig. 13 confirm the superiority of the proposed DC-PPO+BeBold algorithm.

VI. Conclusions

In this paper, we investigated multimedia content delivery in UAV-assisted cellular networks where one MBS and multiple cache-enabled UAVs were deployed to provide content transfer services to ground users. We formulated a delay minimization problem by optimizing the cache placement and multiuser association jointly with UAV trajectory and transmission power. To cope with the uncertainty in the high mobility environment, we transformed the delay minimization problem as a partially observable stochastic game model in which the MBS and each UAV acted as an agent and the total content acquisition delay of users was defined as the extrinsic reward. In addition, the actions taken by the MBS were related to the user association, while the actions taken by each UAV corresponded to the UAV trajectory, cache placement and power control. We proposed a DC-PPO+BeBold algorithm to solve our joint optimization problem. Extensive simulation results shown that the proposed joint design scheme can achieve significant performance gains compared to the benchmark schemes. Our results also indicated that the proposed DC-PPO+BeBold algorithm is capable of outperforming the DC-PPO-based algorithm and the standard PPO-based algorithm in terms of content acquisition delay.

References

[1] Y. Zeng, R. Zhang, and T. J. Lim, “Wireless communications with unmanned aerial vehicles: opportunities and challenges,” *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May. 2016.

[2] M. Mozaffari, W. Saad, Y. Nam, and M. Debbah, “A tutorial on UAVs for wireless networks: Applications, challenges, and open problems,” *IEEE Commun. Surv. Tut.*, vol. 21, no. 3, pp. 2334–2360, Sep. 2019.

[3] L. Gupta, R. Jain, and G. Vaszkun, “Survey of important issues in UAV communication networks,” *IEEE Commun. Surv. Tut.*, vol. 18, no. 2, pp. 1123–1152, May. 2016.

[4] “Cosic visual networking index: Global mobile data traffic forecast update, 2017-2022 white paper.” [Online]. https://www.cisco.com/c/en/us/solutions/collateral/service-provider/white-paper-c11-738429.html, accessed on Mar. 2020.

[5] E. Bastug, M. Bennis, and M. Debbah, “Living on the edge: The role of proactive caching in 5G wireless networks,” *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, Aug. 2014.

[6] T. Zheng, H. Wang, and J. Yuan, “Secure and energy-efficient transmissions in cache-enabled heterogeneous cellular networks: Performance analysis and optimization,” *IEEE Trans. Commun.*, vol. 66, no. 11, pp. 5554–5567, Nov. 2018.

[7] Z. Zhao, M. Peng, Z. Ding, W. Wang, and H. V. Poor, "Cluster content caching: An energy-efficient approach to improve quality of service in cloud radio access networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1207–1221, May. 2016.

[8] 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.

[9] P. Brooks and B. Hestnes, "User measures of quality of experience: why being objective and quantitative is important," *IEEE Netw.*, vol. 24, no. 2, pp. 8–13, May. 2010.

[10] X. Zhong, Y. Guo, N. Li, Y. Chen, and S. Li, "Deployment optimization of UAV relay for malfunctioning base station: Model-free approaches," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 11 971–11 984, Dec. 2019.

[11] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient deployment of multiple unmanned aerial vehicles for optimal wireless coverage," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1647–1650, Aug.

[12] Y. Cai, F. Cui, Q. Shi, M. Zhao, and G. Y. Li, "Dual-UAV-enabled secure communications: Joint trajectory design and user scheduling," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1972–1985, Sep. 2018.

[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. Ghazzai, A. Kadri, M. Ben Ghorbel, and H. Menouar, "Optimal sequential and parallel UAV scheduling for multi-event applications," in *Proc IEEE Veh. Technol. Conf. (VTC)*, Jun. 2018, pp. 1–6.

[15] N. Zhao, X. Pang, Z. Li, Y. Chen, F. Li, Z. Ding, and M. Alouini, "Joint trajectory and precoding optimization for UAV-assisted NOMA networks," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3723–3735, May. 2019.

[16] X. Pang, G. Gui, N. Zhao, W. Zhang, Y. Chen, Z. Ding, and F. Adachi, "Uplink precoding optimization for NOMA cellular-connected UAV networks," *IEEE Trans. Commun.*, vol. 68, no. 2, pp. 1271–1283, Feb. 2020.

[17] M. Chen, W. Saad, and C. Yin, "Liquid state machine learning for resource and cache management in LTE-U unmanned aerial vehicle (UAV) networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1504–1517, Mar. 2019.

[18] 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.

[19] M. Chen, W. Saad, and C. Yin, "Echo-liquid state deep learning for 360 content transmission and caching in wireless VR networks with cellular-connected UAVs," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 6386–6400, Sep. 2019.

[20] N. Zhao, F. R. Yu, L. Fan, Y. Chen, J. Tang, A. Nallanathan, and V. C. M. Leung, "Caching unmanned aerial vehicle-enabled small-cell networks: Employing energy-efficient methods that store and retrieve popular content," *IEEE Veh. Technol. Mag.*, vol. 14, no. 1, pp. 71–79, Mar. 2019.

[21] T. Zhang, Y. Wang, Y. Liu, W. Xu, and A. Nallanathan, "Cache-enabling UAV communications: Network deployment and resource allocation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7470–7483, Nov. 2020.

[22] N. Zhao, F. Cheng, F. R. Yu, J. Tang, Y. Chen, G. Gui, and H. Sari, "Caching UAV assisted secure transmission in hyper-dense networks based on interference alignment," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2281–2294, May. 2018.

[23] 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.

[24] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. Cambridge, MA, USA: MIT Press,, 1998.

[25] S. Yin, S. Zhao, Y. Zhao, and F. R. Yu, "Intelligent trajectory design in UAV-aided communications with reinforcement learning," *IEEE Trans. Veh Technol.*, vol. 68, no. 8, pp. 8227–8231, Aug. 2019.

[26] R. Ding, Y. Xu, F. Gao, and X. Shen, "Trajectory design and access control for air-ground coordinated communications system with multi-agent deep reinforcement learning," *IEEE Internet Things J.*, 2021. Early Access.

[27] X. Liu, Y. Liu, Y. Chen, and L. Hanzo, "Trajectory design and power control for multi-UAV assisted wireless networks: A machine learning approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7957–7969, Aug. 2019.

[28] L. Breslau, Pei Cao, Li Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc IEEE INFOCOM*, 1999, pp. 126–134.

[29] Q. Feng, J. McGeehan, E. K. Tameh, and A. R. Nix, "Path loss models for air-to-ground radio channels in urban environments," in *proc IEEE Veh. Technol. Conf. (VTC)*, May. 2006, pp. 2901–2905.

[30] A. Hourani, S. Kandeepan, and A. Jamalipour, "Modeling air-to-ground path loss for low altitude platforms in urban environments," in *Proc IEEE Global Commun. Conf. (GLOCOM)*, Dec. 2014, pp. 2898–2904.

[31] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Unmanned aerial vehicle with underlaid device-to-device communications: Performance and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 3949–3963, Jun. 2016.

[32] 3rd Generation Part-nership Project (3GPP), *Requirements for further advancements for Evolved Universal Terrestrial Radio Access (E-UTRA)*. TR 36.814-920, Mar. 2017.

[33] L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and L. Hanzo, "Multi-agent deep reinforcement learning-based trajectory planning for multi-UAV assisted mobile edge computing," *IEEE Trans. Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 73–84, Mar. 2021.

[34] N. D. J. B. Jingwei Zhang, Niklas Wetzel and W. Burgard, "Scheduled intrinsic drive: A hierarchical take on intrinsically motivated exploration," Accessed on Mar. 2019. [Online]. Available: http://arxiv.org/abs/2019.07400.

[35] T. Zhang, H. Xu, X. Wang, Y. Wu, K. Keutzer, J. E. Gonzalez, and Y. Tian, "BeBold: Exploration beyond the boundary of explored regions," Accessed on Dec. 2020. [Online]. Available: http://arxiv.org/abs/2012.08621.

[36] C. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 6, pp. 279–292, Dec. 1992.

[37] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, and D. Wieratra, "Continuous control with deep reinforcement learning," in *Proc Int. Conf. Learning Represent. (ICLR)*, May. 2016, pp. 1–6.

[38] A. Al-Hilo, M. Samir, C. Assi, S. Sharafeddine, and D. Ebrahimi, "UAV-assisted content delivery in intelligent transportation systems-joint trajectory planning and cache management," *IEEE Trans. Intell. Transp. Syst.*, 2020. Early Access.

[39] V. R. Konda and J. N. Tsitsiklis, "Actor-critic algorithms," in *Proc Neural Information Processing Systems(NIPS)*, Jun. 2000, pp. 1008–1014.

[40] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," Accessed on Jul. 2017. [Online]. Available: http://arxiv.org/abs/1707.06347.

[41] E. Bohn, E. M. Coates, S. Moe, and T. A. Johansen, "Deep reinforcement learning attitude control of fixed-wing UAVs using proximal policy optimization," in *Proc IEEE Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Apr. 2019, pp. 523–533.

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

[43] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," Accessed on Jul. 2015. [Online]. Available: http://arxiv.org/abs/1506.02438.

[44] J. Ji, K. Zhu, D. Niyato, and R. Wang, "Probabilistic cache placement in UAV-assisted networks with D2D connections: Performance analysis and trajectory optimization," *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6331–6345, Oct. 2020.

[45] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176–189, Jan. 2016.

[46] S. H. Chae and W. Choi, "Caching placement in stochastic wireless caching helper networks: Channel selection diversity via caching," *IEEE Trans. Wireless Commun.*, vol. 15, no. 10, pp. 6626–6637, Oct. 2016.

[47] G. T. 36.777, *3GPP TR 36.777, Enhanced LTE Support for Aerial Vehicle*. Release 15, Dec. 2017.

[48] N. Zhao, X. Pang, Z. Li, Y. Chen, F. Li, Z. Ding, and M.-S. Alouini, "Joint trajectory and precoding optimization for UAV-assisted NOMA networks," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3723–3735, May. 2019.

[49] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proc Advances in Neural Information Processing Systems*, 2019, pp. 8026–8037.