Deep Reinforcement Learning Based Resource Allocation in Cooperative UAV-Assisted Wireless Networks

Phuong Luong, *Member, IEEE*, François Gagnon, *Senior Member, IEEE*, Le-Nam Tran, *Senior Member, IEEE* and Fabrice Labeau, *Senior Member, IEEE*

***Abstract*—We consider the downlink of an unmanned aerial vehicle (UAV) assisted cellular network consisting of multiple cooperative UAVs, whose operations are coordinated by a central ground controller using wireless fronthaul links, to serve multiple ground user equipments (UEs). A problem of jointly designing UAVs' positions, transmit beamforming, as well as UAV-UE association is formulated in the form of mixed integer nonlinear programming (MINLP) to maximize the sum UEs' achievable rate subject to limited fronthaul capacity constraints. Solving the considered problem is hard owing to its non-convexity and the unavailability of channel state information (CSI) due to the movement of UAVs. To tackle these effects, we propose a novel algorithm comprising of two distinguishing features: (i) exploiting a deep Q-learning approach to tackle the issue of CSI unavailability for determining UAVs' positions, (ii) developing a difference of convex algorithm (DCA) to efficiently solve for the UAV's transmit beamforming and UAV-UE association. The proposed algorithm recursively solves the problem of interest until convergence, where each recursion executes two steps. In the first step, the deep Q-learning (DQL) algorithm allows UAVs to learn the overall network state and account for the joint movement of all UAVs to adapt their locations. In the second step, given the determined UAVs' positions from the DQL algorithm, the DCA iteratively solves a convex approximate subproblem of the original non-convex MINLP problem with the updated parameters, where the problem's variables are transmit beamforming and UAV-UE association. Numerical results show that our design outperforms the existing algorithms in terms of algorithmic convergence and network performance with a gain of up to 70%.**

***Index Terms*—Beamforming, limited fronthaul, optimization, reinforcement learning, UAV placement.**

I. Introduction

The rise of diverse emerging communications types such as Internet-of-Thing (IoT) [1], Machine-to-Machine (M2M) communications [2], Internet-of-Everything (IoE) [3] incites new wireless network infrastructures that lean towards a highly agile network platform. Among others, UAV-assisted networks [4]–[6] are considered the promising solutions. In particular, UAV-assisted networks can flexibly form, destruct, and reform any on-demand access networks by dispatching flying-capable small base stations away from a fixed and grid-connected wireless access infrastructure to communicate with UEs. UAV wireless communications not only provides ubiquitous coverages and received signal strengths, but also embraces beyond Line-of-Sight (LoS) transmissions and allows coordinated UAVs' communications for a better interference management, higher cooperative gains, and lower network latencies [7]. Thus, it is natural to say that UAV communications can inherit many advantages from distributed massive MIMO and ultra-dense heterogeneous networks [**?**], [8].

Harnessing the aforementioned benefits of UAV-assisted networks certainly requires addressing many technical challenges in terms of resource allocation design. Specifically, UAV-assisted networks only perform best when UAVs' positions or trajectories are properly planned, UAV's transmit powers are appropriately allocated, and UAV-UE association is well managed to cope with the dynamic of CSI between UAVs and UEs [9], [10]. Such joint designsoften require an accurate CSI estimation. However, perfect CSI is not always possible because UAVs can flexibly move in space, which causes CSI to quickly change over time and location [11]. Additionally, UAV-assisted networks still suffer similar challenges of coordinated multi-point (CoMP) and multi-cast communication networks [12]–[14]. More practical approaches for efficient development of UAV-assisted networks are of timely importance.

A. Related Work

There have been several studies on a joint design of UAVs' locations, transmit powers and UE association for UAV-assisted networks. For a single-UAV-assisted system, a three dimensional (3D) UAV placement algorithm to maximize the number of covered UEs was proposed in [15]. A 3D UAV positioning based on a Q-learning method considering UE mobility was presented in [16]. The sum energy received by all UEs was maximized in [17] by optimizing the UAV's trajectory in a UAV-enabled wireless power transfer system. The authors in [**?**], [18] proposed a data offloading scheme through the design of UAV's trajectory. For multi-UAV-assisted networks, an efficient iterative algorithm which alternately solves for UEs' scheduling, UAVs' trajectories and transmit powers by applying the block coordinate descent and successive convex optimization techniques was proposed in [19] and [20]. In [21], the quality-of-experience (QoE) of ground UEs was maximized through a joint design of the deployment and movement of multiple UAVs. The downlink sum rate maximization for UEs' association, resource allocation and base

Phuong Luong is with Resilient Machine Learning Institute (ReMi), Montreal, Canada. François Gagnon is with École de Technologie Supérieure (ÉTS), Montreal, QC, Canada. Le-Nam Tran is with University College Dublin, Ireland. Fabrice Labeau is with McGill University, Montreal, Canada (email: {phuong.luongthithu@mcgill.ca}, {francois.gagnon@etsmtl.ca}, {nam.tran@ucd.ie}, {fabrice.labeau@mcgill.ca}).

station placement in multi-UAV-assisted cellular networks was considered in [22].

Although multi-UAV-assisted systems were considered in [19], [21], the UAVs' locations and/or trajectories and power control were mostly designed without considering UAVs' cooperation. Like CoMP transmission in 4G systems, UAVs cooperation can incur significant signaling overhead. To address this issue, some work allows one user to be only served by a subset of the favorable cooperating UAVs, which leads to the problem of UAV-UE association. In CoMP, the design of transmit beamformers and UE data allocation to minimize the backhaul UE data transfer in a multi-cell CoMP network was studied in [23].

Compare to the conventional CoMP networks in which the base stations are static, UAV-assisted networks are highly dynamic. This means that the UAVs' placement and movement as well as UAV's cooperation need to be designed in accordance with the variation of UEs' positions to achieve throughput improvement over time. In this regard, [24] proposed a cooperative beamforming technique for multi-UAV networks (as known as CoMP in the sky) to maximize the uplink network throughput using a proper design of UAVs' placement. [11] aimed at optimizing the decoding order of the NOMA scheme and the cooperative UAVs' positions in space to maximize the sum UEs' achievable rate. The hovering locations of two cooperative UAVs were considered to maximize the signal to noise ratio in [25]. In [26], UAVs' location and transmit beamforming were jointly designed with content placement in a swarm of cooperative UAVs. However, CSI was assumed to be predetermined and used as input to the optimization problem to solve for the UAVs' positions and resource allocation in many previous work such as [11], [19], [24]–[26]. In fact, CSI varies with the UAVs' positions that are also optimization variables. Thus, solving for UAVs' positions assuming the availability of CSI in these aforementioned studies is not practical. In summary, the unavailability of CSI in UAV-assisted networks is the bottleneck which makes its underlying optimization problem very difficult to solve.

Deep reinforcement learning (DRL) has recently been shown as a promising solution for tackling the placement and resource allocation problems in UAV-assisted wireless networks [27]. In [28], a Q-Learning based resource allocation algorithm was proposed for maximizing the expected long-term reward, where each UAV can learn its local observation to independently take an action without cooperation. The work of [29] proposed a DRL algorithm based on echo state network (ESN) cells for optimizing the UAV path, transmit power level and cell association to minimize the intercell-interference level and transmission delay. To cope with the continuous action space, the authors in [30] presented a DRL algorithm which uses an actor-critic method for the UAV coverage and connectivity guarantee. The deployment of UAVs was studied in [31], [32] to minimize the UAVs' transmission powers, where [31] proposed a Gaussian mixture model based machine learning algorithm to predict the cellular traffic. In [32], an ESN framework was used to predict the UEs' content request distribution in a cache-enabled UAV system. Similarly, an ESN algorithm was used to predict the future positions of UEs in [33] where a multi-agent Q-learning based UAVs' trajectories and power control algorithm was then proposed to determine the positions of UAVs. UAV trajectory was also studied in [34] for maximizing the sum UEs' rates by applying a Q-learning based RL algorithm. Despite the proven gains arising from the DRL technique to solve for UAVs' positions/trajectories and resource allocation in the UAV-assisted wireless network, the DRL approaches proposed in previous work [30]–[32], [34] could only tackle single-UAV-assisted or multi-UAV-assisted systems with no UAV cooperation. Moreover, a few discrete levels of power allocation were considered in the action space and no beamforming design was considered in the DRL based UAVs' control policies mentioned above. Hence, the DRL approaches proposed therein can not be readily applied for jointly designing the UAVs' positions, UAV-UE association and transmit beamformers in the cooperative UAV-assisted wireless networks.

B. Motivations and Novel Contributions

In this paper, we study the downlink of a multi-UAV-assisted cellular network whose multiple UAVs can cooperatively serve their UEs using joint processing techniques. A central UAV controller located at the macro base station (MBS) is responsible for processing all baseband signals, coordinating resource allocation/computation, as well as transporting data to the UAVs via wireless fronthaul links [24], [35]. Under this setting, our objective is to maximize the overall system throughput, for which we consider a joint design of UAVs' positions, UAV-UE association, and transmit beamforming at the UAVs. To solve the considered problem we propose a novel deep Q-learning based method in conjunction with a deterministic optimization technique called difference of convex (DC) algorithm. Our novelty lies in the application of a deep Q-learning framework to the UAV-assisted network since it provides a more viable and practical approach to addressing the inherent issue encountered by the optimization techniques for UAVs' positions determination. More specifically, as described in [11], the issue is that an optimization technique requires the CSI to be available without the knowledge of UAVs' positions to find an optimal UAVs' position. Obviously, this requirement is hardly met in practice since CSI itself depends on UAVs' positions which are to be computed. However, since each UAV-to-UE channel varies with UAVs' positions and time, UAVs must necessarily settle at one location for the sake of a highly accurate CSI estimation. Our proposed deep Q-learning approach allows UAVs to learn their environment to jointly adjust their locations in each iteration of the proposed algorithm. In this way, CSI can be obtained based on UAVs' positions so that resource allocation algorithms can be executed. Our contributions are as follows:

- Unlike [15], [16], [19], [29]–[33] where UAVs' locations and power allocation are jointly considered in the system with one or multiple *non-cooperative UAVs*, we propose a UAV cooperation strategy to better coordinate transmit signals and inter-UAV interference to achieve higher system throughput. Our consideration leads to an MINLP optimization problem of jointly designing UAV's positions, UAV-UE association, and transmit beamforming

which incorporates all the network's constraints including the limited fronthaul capacity between the MBS and UAVs.

- The considered problem is hard to solve because of its non-convexity and the CSI unavailability to which the methods in [19], [24]–[26] are no longer applicable. To tackle these issues, we decouple and solve the main problem by two separate phases. In the first phase, a deep Q-learning based RL method (DQL) is adopted to develop an algorithm which allows UAVs to jointly learn the overall current network state and adapt their positions according to the action transmitted from the MBS. Based on the calculated UAVs' positions, we propose a DCA to deal with the resulting non-convex problem of solving for UAV-UE association and transmit beamforming. The output of the proposed DCA algorithm is then used to compute the reward in order to construct the decision policy of the DQL algorithm to update the UAVs' positions. This process is iterated until convergence. To the best of our knowledge, our work is the first to develop an optimization-assisted DQL method for jointly determining UAVs' positions, UAV-UE association and UAVs transmit beamfomers in a cooperative UAV-assisted wireless network.
- We conduct extensive numerical experiments to evaluate the convergence performance of the developed algorithm as well as compare its achieved throughput to other traditional approaches. In particular, to improve the convergence rate of the proposed algorithm, we present an efficient way to choose an initial state of the DQL algorithm rather than an arbitrary one. Numerical results demonstrate that this special way of choosing the initial state of the DQL algorithm significantly improves the performance of the proposed algorithm in terms of convergence rate and network throughput.

The remainder of the paper is as follows. In Section II, we introduce the system model and formulate the problem of interest. Section III presents the proposed solution. We explain the method to choose the initial state of the proposed algorithm in Section III-D. Section IV discusses our numerical results. Finally, conclusion is drawn in Section V.

Notation: We use bold uppercase and lowercase letters to denote matrices and vectors, respectively. $\mathbb{C}$ and $\mathbb{R}$ represent the space of complex and real numbers. $\mathbf{x}^T$ and $\mathbf{x}^H$ stand for the transpose and Hermitian operation of vector $\mathbf{x}$. $|x|$ represents the modulus of $x \in \mathbb{C}$, while $\|\mathbf{x}\|_2$ is the ℓ_2-norm of the vector $\mathbf{x}$. The notation $\mathbb{E}\{\cdot\}$ denotes the expectation operator; x^* represents the complex conjugate of $x \in \mathbb{C}$; $\mathrm{Re}(.)$ and $\mathrm{Im}(.)$ stand for the real and imaginary part of the argument, respectively; $\mathcal{O}$ represents the big O notation.

II. System Model

A. Spatial Model

We consider the downlink of a UAV-assisted wireless network consisting of one MBS and a set $\mathcal{U} = \{1, \ldots, U\}$ of single antenna UAVs operated in a target area as shown in Fig. 1(a). Each UAV has a communication range R and flies at a fixed altitude H to serve a group of single antenna UEs. The UEs are only served by the coordinated UAVs and UAVs connect to the MBS through wireless fronthaul links [11], [33], [34]. We treat the considered model as a multi-UAV relaying system. This scenario is useful for the case in which the signals from MBS to UEs are very weak due to blockages from buildings or mountains or signals being transmitted under expected natural disaster occurrence. In these situations, UEs can only get wireless access through the deployed UAVs.

Let us denote the position of the kth UE, which is fixed on the ground, by $v_k = (\bar{x}_k, \bar{y}_k, 0)$, where $k \in \mathcal{K} = \{1, \ldots, K\}$ denotes the set of UEs' indices. Without loss of optimality, we set the MBS position to $(0, 0, \bar{H})$ where $\bar{H}$ is the height of the MBS. The ith UAV is positioned at $u_i = (x_i, y_i, H)$.

B. Channel Model

We assume that the fading channel remains unchanged within a coherence time. Following [11], [18], [36], we denote by $\alpha_{ik}^{\text{AtG}}(u_i)$ the air-to-ground (AtG) channel from the ith UAV to the kth UE, which is modeled as

$$\alpha_{ik}^{\text{AtG}}(u_i) = \sqrt{\mu_{ik}^{\text{AtG}}(u_i)}\tilde{h}_{ik}, \forall i \in \mathcal{U}, k \in \mathcal{K} \tag{1}$$

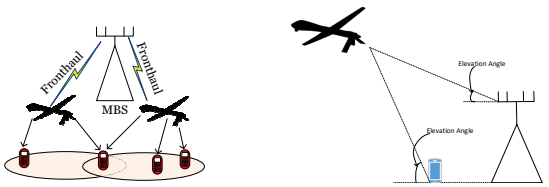
where $\mu_{ik}^{\text{AtG}}(u_i)$ is the large scale fading coefficient accounting for signal attenuation due to both pathloss and shadowing and $\tilde{h}_{ik}$ is the small scale fading coefficient. Let us denote $d_{ik}(u_i) = \sqrt{\left(x_i - \bar{x}_k\right)^2 + \left(y_i - \bar{y}_k\right)^2 + H^2}$ as the distance between the ith UAV and the kth UE. As in [36], [37], the channel between a UAV and a UE can be LoS or non-line-of-sight (NLoS). The probability that the channel between the ith UAV and the kth UE is LoS, denoted by $p_{ik}^{\text{LoS}}(u_i)$, depends on elevation angle steam from this UAV-UE pair (as depicted in Fig. 1(b)) and is approximated as $p_{ik}^{\text{LoS}}(u_i) = \frac{1}{1+C\exp(-D[\frac{180}{\pi}\sin^{-1}\left(\frac{H}{d_{ik}(u_i)}\right)-C])}$ where C and D are parameters that depend on the propagation environment (cf. [36], [37] for the specific values of C and D for some typical environments). Thus, $\mu_{ik}^{\text{AtG}}(u_i)$ can be modeled as

$$\mu_{ik}^{\text{AtG}} = \begin{cases} \mu_0 d_{ik}^{-2}(u_i) & \text{LoS (with probably } p_{ik}^{\text{LoS}}(u_i)) \\ \eta\mu_0 d_{ik}^{-2}(u_i) & \text{NLoS (with probably 1-}p_{ik}^{\text{LoS}}(u_i)) \end{cases} \tag{2}$$

where $\mu_0 = \left(\frac{4\pi f}{c}\right)^{-2}$ and $\eta < 1$ are the average channel power gain at the reference distance of 1 meter and additional attenuation factor due to NLoS, respectively; f is the carrier frequency and c is the speed of light. In addition, we consider the Rician distribution for modeling the small scale fading between the ith UAV and the kth UE as [11], [38]

$$\tilde{h}_{ik} = \sqrt{K_{ik}/(1+K_{ik})}\bar{h}_{ik} + \sqrt{1/(1+K_{ik})}\hat{h}_{ik} \tag{3}$$

where K_{ik} is the Rician factor of channel between the ith UAV at position u_i and the kth UE. $\bar{h}_{ik}$ is the deterministic LoS component with $\left|\bar{h}_{ik}\right| = 1$, and $\hat{h}_{ik} \sim \mathcal{CN}(0, 1)$ is the random scattered component. Note that K_{ik} is related to the elevation angle between the ith UAV and the kth UE and is computed by $K_{ik} = A_1 \exp(A_2 \sin^{-1}(H/d_{ik}(u_i)))$, where A_1 and A_2 are constant coefficients determined by specific environment [38].



(a) (b)

Fig. 1. (a): Cooperative multi-UAV-assisted wireless system;(b): Elevation angles.

Similarly, we denote by $\alpha_i^{\text{GtA}}(u_i) = \sqrt{\mu_i^{\text{GtA}}(u_i)}\tilde{g}_i$ the ground-to-air (GtA) channel from the MBS to the ith UAV, where

$$\mu_i^{\text{GtA}}(u_i) = \begin{cases} \mu_0 d_i^{-2}(u_i) & \text{LoS (with probably } p_i^{\text{LoS}}(u_i)) \\ \eta\mu_0 d_i^{-2}(u_i) & \text{NLoS (with probably 1-}p_i^{\text{LoS}}(u_i)), \end{cases} \tag{4}$$

where $d_i(u_i) = [x_i^2 + y_i^2 + (H - \bar{H})^2]^{1/2}$ is the distance between the ith UAV and the MBS, and $\tilde{g}_i = \sqrt{K_i/(1+K_i)}\bar{g}_i + \sqrt{1/(1+K_i)}\hat{g}_i$ is the small scale fading coefficient between the ith UAV and the MBS. The LoS probability $p_i^{\text{LoS}}(u_i)$ between the MBS and the ith UAV is $p_i^{\text{LoS}}(u_i) = \frac{1}{1+C\exp(-D[\frac{180}{\pi}\sin^{-1}\left(\frac{H-\bar{H}}{d_i(u_i)}\right)-C])}$. In addition, $\tilde{g}_i$ follows the Rician distribution with Rician factor $K_i = A_1\exp(A_2\sin^{-1}((H-\bar{H})/d_i(u_i)))$. $\bar{g}_i$ is the deterministic LoS component with $|\bar{g}_i| = 1$ and $\hat{g}_i \sim \mathcal{CN}(0,1)$ is the random scattered component.

C. Transmission Model

The data transmissions from the MBS to UEs occur in two phases as follows:

1) Phase 1: Transmissions from MBS to UAVs: We assume that MBS communicates with UAVs on orthogonal subchannels by means of frequency devision multiplex access (FDMA). Thus, the received signal at the ith UAV is given by

$$y_i = \sqrt{p_i}\alpha_i^{\text{GtA}}(u_i)s_i + \tilde{n}_i \tag{5}$$

where $p_i \in \mathbb{R}^+$ is the power from the MBS to the ith UAV, s_i is the message for the ith UAV where $\mathbb{E}\{s_i s_i^*\} = 1$ and $\tilde{n}_i \sim \mathcal{CN}(0,\sigma_n^2)$ is the additive white Gaussian noise (AWGN) at the ith UAV, where σ_n^2 is the noise power. Thus, the achievable rate (in b/s/Hz) at the ith UAV is

$$R^{\text{GtA}}{}_i(p_i,u_i) = \log_2(1+\text{SNR}(p_i,u_i)) \tag{6}$$

where

$$\text{SNR}(p_i,u_i) = \frac{p_i\left|\alpha_i^{\text{GtA}}(u_i)\right|^2}{\sigma_n^2} \tag{7}$$

The total transmit powers from the MBS to all UAVs should be less than or equal to the maximum power budget $P_{\max}^{\text{MBS}}$

$$\sum_{\forall i\in\mathcal{U}} p_i \le P_{\max}^{\text{MBS}}. \tag{8}$$

2) Phase 2: Transmissions from UAVs to UEs: Since there are possibly many UEs in the considered system, FDMA is not adopted in this phase. The reason is that serving a large number of UEs with orthogonal transmissions may reduce the spectrum efficiency. Instead, we consider that all UAVs can serve their UEs simultaneously in the same spectrum by applying the linear beamforming technique. Specifically, the UAVs will cooperate to form separate beamforming to different UEs to reduce multiuser interference and improve spectral efficiency. For notational convenience, we denote the set of beamforming vectors intended for the kth UE as $\mathbf{w}_k \triangleq [w_{1k}, w_{2k}, \ldots, w_{Uk}] \in \mathbb{C}^{U\times 1}$ and the vector including the channels from all UAVs to the kth UE as $\mathbf{h}_k(\mathbf{u}) \triangleq [\alpha_{1k}^{\text{AtG}}(u_1), \alpha_{2k}^{\text{AtG}}(u_2), \ldots, \alpha_{Uk}^{\text{AtG}}(u_U)]^T \in \mathbb{C}^{U\times 1}$ where $\mathbf{u} = [u_1, u_2, \ldots, u_U]^T$ represents the location vector of all UAVs. Using these notations, the received signal at the kth UE is

$$y_k = \mathbf{h}_k(\mathbf{u})\mathbf{w}_k q_k + \sum\nolimits_{j\in\mathcal{K}\setminus k}\mathbf{h}_k(\mathbf{u})\mathbf{w}_j q_j + z_k \tag{9}$$

where q_k is the message intended for the kth UE with $\mathbb{E}\{q_k q_k^*\} = 1$, $z_k \sim \mathcal{CN}(0,\sigma_0^2)$ is the additive white Gaussian noise (AWGN) and σ_0^2 is the noise power. Note that in (9), we have assumed that the kth UE is connected to all the UAVs, but the ith UAV effectively serves the kth UE only if $|w_{ik}|^2 > 0$. By treating the interference as noise, the achievable rate in b/s/Hz for a given set of channel realizations at the kth UE is given by

$$R_k^{\text{AtG}}(\mathbf{w},\mathbf{u}) = \log_2(1+\text{SINR}_k(\mathbf{w},\mathbf{u})) \tag{10}$$

where

$$\text{SINR}_k(\mathbf{w},\mathbf{u}) = \frac{|\mathbf{w}_k^H\mathbf{h}_k(\mathbf{u})|^2}{\sum_{j\in\mathcal{K}\setminus k}|\mathbf{w}_j^H\mathbf{h}_k(\mathbf{u})|^2 + \sigma_0^2} \tag{11}$$

and $\mathbf{w} \triangleq [\mathbf{w}_1^T, \mathbf{w}_2^T, \ldots, \mathbf{w}_k^T]^T \in \mathbb{C}^{(KU)\times 1}$ is the vector stacking the beamformers of all UEs.

To formulate the problem of interest we introduce binary variables $c_{ik} = \{0,1,\forall i \in \mathcal{U}, k \in \mathcal{K}\}$ to represent the association between the ith UAV and the kth UE. We refer to these binary variables as the UAV-UE association variables. Specifically, $c_{ik} = 1$ implies that the ith UAV serves the kth UE (i.e., $w_{ik} > 0$) and $c_{ik} = 0$ is otherwise (i.e., $w_{ik} = 0$). The power at the ith UAV should satisfy the following constraints:

$$|w_{ik}|^2 \le c_{ik}\lambda_{ik}, \forall i \in \mathcal{U}, \forall k \in \mathcal{K} \tag{12}$$

$$\sum_{\forall k\in\mathcal{K}} \lambda_{ik} \le P_{i,\max}^{\text{UAV}}, \forall i \in \mathcal{U} \tag{13}$$

$$\lambda_{ik} \le c_{ik}P_{i,\max}^{\text{UAV}}, \forall i \in \mathcal{U}, \forall k \in \mathcal{K} \tag{14}$$

where λ_{ik} represents the soft power level of the ith UAV that can transmit to the kth UE. More specifically, (12) and (14) guarantee that the transmit power $|w_{ik}|^2$ and λ_{ik} are zeros if $c_{ik} = 0$, respectively, and (13) force the actual transmit power to stay within the power budget of the ith UAV. Note that the association variables are also constrained by the ith UAV coverage range. Particularly, the kth UE can be served by the ith UAV if it is within the communication range R, leading

to the following constraint

$$d_{ik}(u_i) \leq R + S(1 - c_{ik}), \forall i \in \mathcal{U}, \forall k \in \mathcal{K} \quad (15)$$

where $S > 0$ is sufficiently large to make (15) hold. The meaning is that when $c_{ik} = 1$, the distance between the kth UE and the ith UAV has to be equal to or smaller than the communication range of the ith UAV. Otherwise, $c_{ik} = 0$ and (15) is always satisfied.

Finally, for the transmissions from a UAV to its associated UEs to be possible, the total UEs' achievable rates served by the ith UAV should be smaller than or equal to the fronthaul capacity provided from MBS to the ith UAV, which is expressed as

$$R_i^{\text{GtA}}(p_i, u_i) \geq \sum_{\forall k \in \mathcal{K}} c_{ik} R_k^{\text{AtG}}(\mathbf{w}, \mathbf{u}), \forall i \in \mathcal{U}. \quad (16)$$

D. Problem Formulation

We aim at finding the UAVs' positions $\mathbf{u} = \{u_{ik}, \forall i \in \mathcal{U}, k \in \mathcal{K}\}$, the UAV-UE association $\mathbf{c} = \{c_{ik}, \forall i \in \mathcal{U}, k \in \mathcal{K}\}$, MBS power allocation $\mathbf{p} = \{p_i, \forall i \in \mathcal{U}\}$ and transmit beamformers $\mathbf{w}$ to maximize the sum UEs' achievable rates in the considered cooperative multi-UAV-assisted network. The problem is formulated as

$$(\mathcal{P}): \underset{\mathbf{c},\mathbf{w},\mathbf{u},\mathbf{p},\boldsymbol{\lambda}}{\text{maximize}} \quad \sum_{k \in \mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}, \mathbf{u}) \quad (17a)$$

$$\text{subject to} \quad R_k^{\text{AtG}}(\mathbf{w}, \mathbf{u}) \geq R_{k,\min}, \forall k \in \mathcal{K} \quad (17b)$$

$$(8); (12); (13); (14); (15); (16) \quad (17c)$$

where $R_{k,\min}$ is the predetermined minimum rate required for the kth UE, and the individual rate constraints in (17b) are to ensure the QoS requirement forthe kth UE, $\forall k \in \mathcal{K}$.

The following remarks characterize the difficulties in solving $(\mathcal{P})$. First, due to the presence of binary association variables, non-convex rate functions $R_k^{\text{AtG}}(\mathbf{w}, \mathbf{u})$, and non-convex functions $c_{ik} R_k^{\text{AtG}}(\mathbf{w}, \mathbf{u})$ in (16), problem $(\mathcal{P})$ is a MINLP which is generally NP-hard. Please see Appendix A for a proof of $(\mathcal{P})$'s NP-hardness. Moreover, even when $\mathbf{c}$ is relaxed to be continuous, the relaxed problem is still non-convex. Thus, problem $(\mathcal{P})$ is very difficult to solve and requires highly computational complexity to find a globally optimal solution. Given the inherent non-convexity and combinatorial nature of $(\mathcal{P})$, a pragmatic goal is to find a high-performance solution in a reasonable amount of time.

III. Proposed DCA-assisted DQL Algorithm

In this section, we present a novel framework to solve problem $(\mathcal{P})$, which is called DCA-assisted DQL. This framework is a combination of a deep Q-learning based reinforcement learning (DQL) algorithm and DCA. The proposed DCA-assisted DQL algorithm lets UAVs learn the entire network environment to adjust their positions jointly with determining the transmit beamforming and UAV-UE association to maximize the sum UEs' achievable rates. In particular, the DCA-assisted DQL algorithm comprises two phases: training and testing phase. The former aims at building the optimal decision policy for deep Q-network (DQN) in order to seek the optimal positions of UAVs. The reward of the DQL algorithm in the training phase is computed by solving the non-convex problem of transmit beamforming and UAV-UE association using the proposed DCA. The latter is the actual execution of UAVs to observe the real estimated CSI from the network environment and to fly to their optimal positions guided by the DQN output obtained from the training phase. With the assistance of the DCA, the DQL algorithm only needs to train UAVs with respect to a smaller set of variables (e.g., UAVs' positions), not all the involved variables. As a result, the proposed algorithm greatly reduces the state-action space, especially when all UAVs are coordinated and jointly take actions. Another novel aspect of the proposed DCA-assisted DQL algorithm is that the DCA can produce high-quality transmit beamforming and UAV-UE association, instead of setting a few discrete levels of power as in [29], [33]. The training phase includes the following two steps:

- First, UAVs fly to the positions $\mathbf{u}_t$ according to the action a_t received from the MBS at time step t of the DCA-assisted DQL algorithm. UAVs estimate CSI associated with $\mathbf{u}_t$.
- Second, given the estimated CSI, the remaining variables, e.g., MBS transmit power, UAVs beamforming and UAV-UE association, are found using the DCA. The obtained solution is used to calculate the reward of the DCA-assisted DQL algorithm.
- This process is iterated to build up the decision policy of the DCA-assisted DQL algorithm.

In the next subsections we will present the details of the two steps above.

A. Overview of DCA-assisted DQL Algorithm

In the proposed DCA-assisted DQL algorithm, UAVs are seen as agents which are coordinated and controlled by the MBS. UAVs interact with the system environment in a sequence of discrete times t. At each time t, the UAVs observe the state s_t, take action a_t and receive the reward r_t. The system moves to the new state s_{t+1} at time $t+1$. In the context of the considered problem, we define the state s_t, action a_t and reward r_t at time step t as follows:

- State representation s_t: UAV agents determine state s_t from the positions $\mathbf{u}_t$ and CSI observation associated with $\mathbf{u}_t$ as $s_t = \{\mathbf{u}_t, \alpha_{ik,t}^{\text{AtG}}(u_{ik,t}), \alpha_{i,t}^{\text{GtA}}(u_{i,t}), \forall k \in \mathcal{K}, i \in \mathcal{U}\}$.
- Action $a_t = \{\phi_{i,t}, d_{i,t}, \forall i \in \mathcal{U}\}$ which are decided for all UAVs, where $\phi_{i,t} \in (0, 2\pi]$ and $d_{i,t} \in [0, d_{\max}]$ are the movement direction and distance for each ith UAV, respectively.
- After taking action a_t sent from the MBS, each ith UAV moves to the new position $u_{i,t+1}$ according to movement direction $\phi_{i,t}$ and distance $d_{i,t}$, $\forall i \in \mathcal{U}$. Given the positions $\mathbf{u}_{t+1}$, the UAVs estimate the new CSI, denoted as the state s_{t+1}. The transition from state s_t to s_{t+1} generates a reward $r_t(s_t, a_t)$. The reward is directly related to the sum UEs' achievable rates and defined by the actions. Particularly, if the UAVs carry out action a_t at time step t that improves the sum rates, then the reward gets a high value and vice versa. This reward allows UAVs to learn an action as a good or bad one in the training process to create the optimal policy decision.

Thus the reward is calculated as (18), where $\kappa > 0$, $V > \kappa$ are constant parameters and $\mathbf{w}_t^\star$ is the optimal beamforming solution obtained by solving the following optimization problem

$$(\mathcal{P}_t): \underset{\mathbf{w}_t,\mathbf{p}_t,\boldsymbol{\lambda}_t,\mathbf{c}_t}{\text{maximize}} \quad \sum_{k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t) \tag{19a}$$

$$\text{subject to} \quad (8);(12);(13);(14);(15);(16);(17b). \tag{19b}$$

Note that the positions of UAVs $\mathbf{u}_t$ at time step t are already known and thus not the optimization variables in the above problem.Therefore, unlike [5], [11], [24], [26] where CSI availability is assumed without knowing UAVs' location, our work can obtain the real CSI associated with location $\mathbf{u}_t$. In this way, the reward $r_t(s_t,a_t)$ in (18) is calculated by the transmit beamforming $\mathbf{w}_t^\star$ and UAV-UE association $\mathbf{c}_t^\star$ which depends on the UAVs' positions $\mathbf{u}_t$ obtained at time step t of DCA-assisted DQL algorithm. Remark that problem $(\mathcal{P}_t)$ can be infeasible when individual UE rate requirements in (17b) are not satisfied. Thus, if $(\mathcal{P}_t)$ is infeasible, the reward corresponding to action a_t is assigned a negative value $-V$ in order to avoid this action.

It is now apparent that the challenge is how to calculate the reward efficiently since solving $\big(\mathcal{P}_t\big)$ is intractable. This is due to the binary variable $\mathbf{c}$ and the non-convexity of UE rate functions in (19a) and the constraints in (16). To overcome this issue, we propose a DCA to solve $\big(\mathcal{P}_t\big)$.

B. DCA for Solving $(\mathcal{P}_t)$ to Calculate The Reward $r_t(s_t,a_t)$

1) Difference of Convex Functions Reformulation: We observe that problem (19) is difficult to solve because of the non-convex functions $R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)$ and $c_{ik,t}R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)$. Based on the concept of DC programming, we will express these functions as difference of two convex ones. Specifically, we rewrite $R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)$ equivalently as

$$R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t) = \underbrace{R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t) + \xi_{k,t}\big\|\mathbf{w}_t\big\|^2}_{f_k(\mathbf{w}_t,\mathbf{u}_t)} - \xi_{k,t}\big\|\mathbf{w}_t\big\|^2 \tag{20}$$

for some $\xi_{k,t} > 0$. Intuitively if $\xi_{k,t}$ is sufficiently large, the quadratic term $\xi_{k,t}\big\|\mathbf{w}_t\big\|^2$ will make $f_k(\mathbf{w}_t,\mathbf{u}_t)$ convex with respect to $\mathbf{w}_t$. Particularly, $\nabla R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)$ is shown to be Lipschitz continuous with a constant $\bar{\xi}$ and thus, $R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)$ is $\bar{\xi}$-smooth (cf. [12, Lemma 2] for further details on how to calculate $\bar{\xi}$). Note that for $\xi_{k,t} > \bar{\xi}$, $f_k(\mathbf{w}_t,\mathbf{u}_t)$ is strongly convex [12].

Next a DC decomposition of $c_{ik,t}R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)$ is given by (21), for any $\zeta_{k,t} \geq 0$. Similarly, for $\zeta_{k,t} > \bar{\zeta}$, where $\bar{\zeta}$ is a Lipschitz constant of $\nabla c_{ik,t}R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)$, $y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t})$ is strongly concave.

Furthermore, to deal with the binary variables $\mathbf{c}$, we proceed to equivalently rewrite the binary constraint $c_{ik,t} \in \{0,1\}$ into the continuous DC form constraints as

$$c_{ik,t} - c_{ik,t}^2 \leq 0 \tag{22a}$$

$$0 \leq c_{ik,t} \leq 1. \tag{22b}$$

We note that the equivalent reformulation of a binary constraint into two continuous constraints as shown above is quite popular in mixed integer programming in order to leverage continuous optimization [39]. Although (22a) and (22b) do not satisfy Slater's conditions, we remark that (22a) is further approximated by (23c) and (29c), and Slater's conditions hold for the continuous relaxation. The proof of this is omitted here due to the space limitation.

2) Proposed Relaxation of (35)*:* From the above reformulations we now introduce a new slack variable $\mathbf{z}_t = \{z_{ik,t} \geq 0, \forall i,k\}$ and consider the following problem

$$\underset{\mathbf{w}_t,\mathbf{p}_t,\mathbf{c}_t,\boldsymbol{\lambda}_t,\mathbf{z}_t}{\text{maximize}} \sum_{k\in\mathcal{K}} f_k(\mathbf{w}_t,\mathbf{u}_t) - \xi_{k,t}\left\|\mathbf{w}_t\right\|^2 - \upsilon\sum_i\sum_k z_{ik,t} \tag{23a}$$

$$\text{s.t.} R_i^{\text{GtA}}(p_i,u_i) \geq \sum_{\forall k\in\mathcal{K}} y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t}) + \zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2 + c_{ik,t}^2\right) \tag{23b}$$

$$c_{ik,t} - c_{ik,t}^2 \leq z_{ik,t} \tag{23c}$$

$$(8);(12);(13);(15);(17b);(22b) \tag{23d}$$

where $\upsilon \geq 0$ is a penalty parameter. It is clear that problems (19) and (23) are equivalent when $z_{ik,t} = 0$, $\forall i,k$. A large value of υ will force $z_{ik,t}$ to be zero or close to zero. We observe that the non-convexity of the objective function is due to the maximization over the convex function $f_k(\mathbf{w}_t,\mathbf{u}_t)$. Thus, we can iteratively approximate function $f_k(\mathbf{w}_t,\mathbf{u}_t)$ by its first order Taylor linearization $F_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)$ around the point $\mathbf{w}_t^n$ at the nth iteration as $f_k(\mathbf{w}_t,\mathbf{u}_t) \geq F_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)$ where $F_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)$ is given by (24). In addition, we have denoted $\mathbf{w}_t^{nH} = (\mathbf{w}_t^n)^H$ to lighten the notation and $\breve{f}_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)$ is given by

$$\breve{f}_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n) = \frac{2\,\text{Re}\left(\mathbf{w}_t^{nH}\mathbf{H}_k(\mathbf{u}_t)\mathbf{w}_t - \mathbf{w}_t^{nH}\mathbf{H}_k(\mathbf{u}_t)\mathbf{w}_t^n\right)}{\mathbf{w}_t^{nH}\mathbf{H}_k(\mathbf{u}_t)\mathbf{w}_t^n + \sigma_0^2} - \frac{2\,\text{Re}\left(\mathbf{w}_t^{nH}\tilde{\mathbf{H}}_k(\mathbf{u}_t)\mathbf{w}_t - \mathbf{w}_t^{nH}\tilde{\mathbf{H}}_k(\mathbf{u}_t)\mathbf{w}_t^n\right)}{\mathbf{w}_t^{nH}\tilde{\mathbf{H}}_k(\mathbf{u}_t)\mathbf{w}_t^n + \sigma_0^2}. \tag{25}$$

In (25), we define

$$\mathbf{H}_k(\mathbf{u}_t) = \text{Bdiag}\underbrace{\left(\bar{\mathbf{H}}_k(\mathbf{u}_t),\ldots,\bar{\mathbf{H}}_k(\mathbf{u}_t)\right)}_{K\text{elements}}$$

$$\tilde{\mathbf{H}}_k(\mathbf{u}_t) = \text{Bdiag}(\bar{\mathbf{H}}_k(\mathbf{u}_t),\ldots,\underbrace{\mathbf{0}}_{k\text{th element}},\ldots,\bar{\mathbf{H}}_k(\mathbf{u}_t))$$

where $\bar{\mathbf{H}}_k(\mathbf{u}_t) = \mathbf{h}_k(\mathbf{u}_t)\mathbf{h}_k^H(\mathbf{u}_t)$. Similarly, it can be seen that the non-convexity of (23b) is because of the concave function $y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t})$ on the right side of inequality. In the same way, we can approximate function $y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t})$ by its upper bound $Y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t};\mathbf{w}_t^n,c_{ik,t}^n)$ around the point $\mathbf{w}_t^n,c_{ik,t}^n$ as in (26). As a result, constraint (23b) can be approximated by the following generic convex constraint

$$R_i^{\text{GtA}}(p_i,u_i) \geq \sum_{\forall k\in\mathcal{K}} Y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t};\mathbf{w}_t^n,c_{ik,t}^n) + \zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2 + c_{ik,t}^2\right) \tag{27}$$

To proceed further, we derive the upper bound convex function in the left side of the constraint (22a) by deriving the lower bound of function $c_{ik,t}^2$ around the point $c_{ik,t}^n$ as

$$c_{ik,t} - 2c_{ik,t}^n c_{ik,t} + \left(c_{ik,t}^n\right)^2 \leq z_{ik,t} \tag{28}$$

Finally, by applying the above approximations, we can ap-

$$r_t(s_t,a_t)=\begin{cases}\sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_t^\star,\mathbf{u}_t)+\kappa & \text{if}\sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_{t+1}^\star,\mathbf{u}_{t+1})>\sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_t^\star,\mathbf{u}_t)\\ \sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_t^\star,\mathbf{u}_t) & \text{if } \sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_{t+1}^\star,\mathbf{u}_{t+1})=\sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_t^\star,\mathbf{u}_t)\\ \sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_t^\star,\mathbf{u}_t)-\kappa & \text{if } \sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_{t+1}^\star,\mathbf{u}_{t+1})<\sum_{\forall k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w}_t^\star,\mathbf{u}_t)\\ -V & \text{if infeasible solution } \mathbf{w}_{t+1}^\star\end{cases} \tag{18}$$

$$c_{ik,t}R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)=\underbrace{\left(c_{ik,t}R_k^{\text{AtG}}(\mathbf{w}_t,\mathbf{u}_t)-\zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2+c_{ik,t}^2\right)\right)}_{y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t})}+\zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2+c_{ik,t}^2\right) \tag{21}$$

$$F_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)=f_k(\mathbf{w}_t^n,\mathbf{u}_t)+\breve{f}_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)+2\xi_{k,t}\,\text{Re}\left(\mathbf{w}_t^{nH}\mathbf{w}_t-\left\|\mathbf{w}_t^n\right\|^2\right) \tag{24}$$

$$\begin{aligned}Y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t};\mathbf{w}_t^n,c_{ik,t}^n)=y_k(\mathbf{w}_t^n,\mathbf{u}_t,c_{ik,t}^n)+c_{ik,t}^n\breve{f}_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)+\left(c_{ik,t}-c_{ik,t}^n\right)R_k^{\text{AtG}}\left(\mathbf{w}_t^n,\mathbf{u}_t\right)\\ -2\zeta_{k,t}\left(\mathbf{w}_t^{nH}\mathbf{w}_t-\left\|\mathbf{w}_t^n\right\|^2+c_{ik,t}^nc_{ik,t}-\left(c_{ik,t}^n\right)^2\right)\end{aligned} \tag{26}$$

proximate (23) at the nth iteration as

$$\underset{\mathbf{w},\mathbf{p},\mathbf{c},\lambda,\mathbf{z}}{\text{maximize}} \sum_{k\in\mathcal{K}} F_k(\mathbf{w}_t,\mathbf{u}_t;\mathbf{w}_t^n)-\xi_{k,t}\left\|\mathbf{w}_t\right\|^2-\upsilon^n\sum_i\sum_k z_{ik,t} \tag{29a}$$

$$\text{s.t.} R_i^{\text{GtA}}(p_i,u_i)\geq\sum_{\forall k\in\mathcal{K}} Y_k(\mathbf{w}_t,\mathbf{u}_t,c_{ik,t};\mathbf{w}_t^n,c_{ik,t}^n)+\zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2+c_{ik,t}^2\right) \tag{29b}$$

$$c_{ik,t}-2c_{ik,t}^nc_{ik,t}+\left(c_{ik,t}^n\right)^2\leq z_{ik,t} \tag{29c}$$

$$(8);(12);(13);(15);(17b);(22b) \tag{29d}$$

where $\mathbf{w}^n,\mathbf{c}^n$, υ^n are not the optimization variables but parameters obtained from the previous iteration.

We remark that the value υ^0 is initiated with some small positive value and increased by multiplying with $\varrho>1$ after each iteration to ensure that $\sum_i\sum_k z_{ik,t}$ approaches 0 when υ^n approaches some large values. Note that that constraint (17b) is equivalent to the two SOC constraints: $\sqrt{e^{R_{k,\min}}/(e^{R_{k,\min}}-1)}\,\text{Re}((\mathbf{h}_k(\mathbf{u}))^H\mathbf{w}_k)\geq\left\|[(\mathbf{h}_k(\mathbf{u}))^H\mathbf{w}_1,\ldots,(\mathbf{h}_k(\mathbf{u}))^H\mathbf{w}_K,\sigma_0]^T\right\|_2$ and $\text{Im}((\mathbf{h}_k(\mathbf{u}))^H\mathbf{w}_k)=0$ [13]. Thus, problem (29) at the nth iteration of the DCA-based algorithm is a convex optimization problem which can be handled by off-the-shelf convex solvers. As a result, a solution to $(\mathcal{P}_t)$ can be obtained at the convergence of the DCA-based algorithm, which is outlined in Algorithm 1. The proof of the convergence of Algorithm 1 is similar to that in [12], and thus is omitted for the sake of brevity.

Algorithm 1 Proposed DCA-based algorithm.

1: Set $n:=0$, $\upsilon^n:=0$, and initialize starting points of $\mathbf{w}^n,\mathbf{c}^n$;
2: **repeat**
3: Solve the approximated problem (29) to achieve the optimal solution $\mathbf{c}^\star,\mathbf{w}^\star,\mathbf{p}^\star,\lambda^\star,\mathbf{z}^\star$;
4: Update $\mathbf{w}^{n+1}=\mathbf{w}^\star,\mathbf{c}^{n+1}=\mathbf{c}^\star$, and $\upsilon^{n+1}:=\varrho\upsilon^n$;
5: Set $n:=n+1$;
6: **until** Convergence of the objective (29a)

C. The DQL

In this section, we present the learning part of the proposed DCA-assisted DQL algorithm whose decision policy is built based on the reward $r_t(s_t,a_t)$ in (18) to update the UAVs' position until convergence. We recall that DQL is a popular method of reinforcement learning (RL) that incorporates a deep neural network (DNN) as an approximator of $Q(.)$ function to seek the optimal actions from the current state. The action-value function $Q^\pi(s_t,a_t)$ is the expected accumulated reward when an action a_t is taken in the environmental state s_t under decision policy π

$$Q^\pi(s_t,a_t)=\mathbb{E}\left[R_t|s_t,a_t,\pi(s_t)\right] \tag{30}$$

where the cumulative discounted reward is defined as $R_t=\sum_{j=0}^{\infty}\gamma^j r_{t+j+1}(s_{t+j+1},a_{t+j+1})$ and $\gamma\in(0,1]$ is a discount factor for weighting future rewards. The optimal action-value function $Q^\star(s_t,a_t)=\max_\pi Q^\pi(s_t,a_t)$ can be iteratively estimated as

$$Q^\star(s_t,a_t)=\mathbb{E}_{s_{t+1}}\left[r_{t+1}+\gamma\max_a Q^\star(s_{t+1},a)\right] \tag{31}$$

In DQL, $Q(s_t,a_t;\theta)\approx Q^\star(s_t,a_t)$ is the estimated action-value function during the iterative process, which is approximated by the DNN where θ is the weights of the edges in the DNN. $Q(s_t,a_t;\theta)$ is updated by adjusting weights θ in DQL through a training process. More specifically, weights θ in DQL are trained and optimized by minimizing prediction errors of $Q(s_t,a_t;\theta)$. This can be done as follows. At time step t, state s_t is input into DNN which has weights θ, action a_t is chosen as $a_t=\arg\max_a Q(s_t,a;\theta)$ where $Q(s_t,a;\theta)$ is the output of the DNN corresponding to all different possible actions a. When action a_t is taken, DQL generates reward $r_t(s_t,a_t)$ calculated in (18) and the system moves to the next state s_{t+1}. Let us define an experience sample (s_t,a_t,r_t,s_{t+1}) at time step t. Then, DQL is trained by minimizing prediction error of $Q(s_t,a_t;\theta)$ through the loss function $\mathcal{L}_t(\theta)$ defined as

$$\mathcal{L}_t(\theta)=\mathbb{E}\left[Z_t(r_t,s_{t+1})-Q(s_t,a_t;\theta)\right] \tag{32}$$

where $Z_t(r_t, s_{t+1})$ is the target value which can be estimated as

$$Z_t(r_t, s_{t+1}) = r_t(s_t, a_t) + \gamma \max_a Q(s_{t+1}, a; \theta) \quad (33)$$

Here, the target value is computed based on the current reward as well as predicted discounted reward $\gamma \max_a Q(s_{t+1}, a; \theta)$ given by the DNN. Thus, weights θ of DNN can now be updated by minimizing the loss function $\mathcal{L}_t(\theta)$ using a semi-gradient algorithm [40].

To improve the learning stability, we employ the experience replay technique [41]. The agent stores the collected samples into the replay buffer with capacity $\mathbf{B}$ and pick a mini batch of them from the buffer to calculate the loss function rather than using a single sample as in (32). Note that the buffer is always updated by removing the oldest samples and adding the newest samples whenever the buffer is full. Consequently, by sampling N experience samples from the buffer $\mathbf{B}$, the loss function can be computed as

$$\bar{\mathcal{L}}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left(Z_i(r_i, s_{i+1}) - Q(s_i, a_i; \theta) \right)^2 \quad (34)$$

In addition, we also employ the target DNN with parameter θ^{target} for training purpose. Particularly, after every E time steps, the target DNN is replaced by the latest DNN by assigning θ^{target} to the latest θ of the DNN and the target values are computed based on this target DNN as $Z_t(r_t, s_{t+1}) = r_t(s_t, a_t) + \gamma \max_a Q(s_{t+1}, a; \theta^{\text{target}})$. The overall training and testing phases of DCA-assisted DQL algorithm are presented in Algorithm 2 and Algorithm 3, respectively.

1) The complexity and signal overhead analysis: The computational complexity of our proposed DCA-assisted DQL algorithm mainly comes from Step 6 of Algorithm 2, i.e., to run Algorithm1 for solving the problem $(\mathcal{P}_t)$ and Steps 12 and 14 of Algorithm 2, i.e., to update target values $Z_i(r_i, s_{i+1}), \forall i \in \{1, N\}$ for N experience samples in replay buffer $\mathbf{B}$ and weights θ, respectively. Particularly, Algorithm 1 iteratively solves the convex problem (29) until convergence. Since (29) is in fact a second order cone program (SOCP), whose total number of variables is $4UK + U$ and total number of constraints is $4UK + 2U + 1$. Thus, the worst-case per-iteration computational complexity of Algorithm 1 is $\mathcal{O}(\log \frac{1}{\epsilon} U^{3.5} K^{3.5})$, where ϵ is the required precision [42, Chapter 6]. Thus, the overall complexity of Algorithm 1 is $\mathcal{O}(\log \frac{1}{\epsilon} L U^{3.5} K^{3.5})$ where L is the number of iterations required for Algorithm 1 to converge. Note that the true complexity is much reduced in practice. The computational complexity of Algorithm 2 in each time step t is thus $\mathcal{O}(\log \frac{1}{\epsilon} L U^{3.5} K^{3.5} + N + |\theta|)$ where $|\theta|$ denotes the cardinal of weights θ and N is the number of experience samples in the buffer.

As can be seen, the amount of exchanged information in Algorithm 2 is mainly for sending action information a_t from the MBS to UAVs, i.e., Step 6, and CSI from UAVs to the MBS, i.e., Step 7, via fronthaul links. Specifically, there are UK channel coefficients $\mathbf{h}_k(\mathbf{u}) = [\alpha_{1k}^{\text{AtG}}(u_1), \ldots, \alpha_{Uk}^{\text{AtG}}(u_U)]^T \in \mathbb{C}^U$, $\forall k \in \mathcal{K}$, required to be transported from all the UAVs to the MBS. The overhead for sending relevant information in Steps 6 and 7 of Algorithm 2 is U and UK, respectively. Thus the total cost of overhead should be $\rho(UK + U)$, where ρ is the constant representing the cost to transport each information unit.

Algorithm 2 The training phase of DCA-assisted DQL Algorithm

1: MBS initializes weights θ of DNN, weights θ^{target} of target DNN, replay buffer with capacity $\mathbf{B}$, ϵ , β, γ, ρ, N, E and global step $l \coloneqq 0$
2: **for** Episode: $0 : L$ **do**
3: UAVs are randomly positioned $\mathbf{u}_0$, estimate CSI state s_0 which is then sent to MBS
4: **for** Time step: $t = 0 : T$ **do**
5: Decide action $a_t = \begin{cases} \arg\max_a Q(s_t, a; \theta) & \text{with probability 1-}\epsilon \\ \text{random action} & \text{with probability } \epsilon \end{cases}$
6: Apply Algorithm 1 to solve (19) to obtain solution $\{\mathbf{w}_t^\star, \mathbf{c}_t^\star\}$ to calculate $r_t(s_t, a_t)$ in (18); then MBS sends the action a_t to UAVs via fronthaul links.
7: UAVs move to new positions $\mathbf{u}_{t+1}$ according to the action a_t and new estimated CSI is sent back to MBS.
8: Store sample (s_t, a_t, r_t, s_{t+1}) into replay buffer $\mathbf{B}$
9: **if** Remainder($\frac{l}{T_{\text{train}}}$)== 0 and $l > T_{\text{start}}$ **then**
10: Randomly sampling N experience samples from the replay buffer $\mathbf{B}$
11: **for** $i = 1 : N$ **do**
12: Compute target value $Z_i(r_i, s_{i+1}) = r_i(s_i, a_i) + \gamma \max_a Q(s_{i+1}, a; \theta^{\text{target}})$
13: **end for**
14: Update weights θ by minimizing the loss: $\bar{\mathcal{L}}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left(Z_i(r_i, s_{i+1}) - Q(s_i, a_i; \theta) \right)^2$
15: **end if**
16: **if** Remainder($\frac{l}{E}$)==0 and $l > T_{\text{start}}$ **then**
17: Update $\theta^{\text{target}} \coloneqq \theta$
18: **end if**
19: $t \coloneqq t + 1$; $l \coloneqq l + 1$
20: **end for**
21: **end for**

Algorithm 3 Testing phase of the proposed DCA-assisted DQL algorithm.

1: Initialize $t = 0$ and UAVs are randomly positioned $\mathbf{u}_t$ to obtain initial state s_t
2: **repeat**
3: Given state s_t, MBS sends an action $a_t = \arg\max_a Q(s_t, a; \theta)$ to UAVs and solves (19) using Algorithm 1 to obtain $\{\mathbf{w}_t^\star, \mathbf{p}_t^\star, \lambda_t^\star, \mathbf{c}_t^\star\}$ to calculate reward $r_t(s_t, a_t)$
4: UAVs move to their new positions $\mathbf{u}_{t+1}$ according to received a_t, estimate new CSIs which are sent to MBS
5: $t \coloneqq t + 1$
6: **until** Convergence of reward value
7: Given the UAVs' positions $\mathbf{u}^\star$ obtained at the convergence, $\{\mathbf{w}^\star, \mathbf{p}^\star, \lambda^\star, \mathbf{c}^\star\}$ solution is obtained by solving (19) using Algorithm 1.

D. Initial State of DCA-assisted DQL Algorithm

The initial state s_0 is the estimated CSI associated with UAVs' positions $\mathbf{u}_0$ which is determined randomly as in Step 1 of Algorithm 3. Note that the convergence rate of the proposed DCA-assisted DQL algorithm in the testing phase varies according to the initial position of the UAVs. It can

be seen that if the initial positions $\mathbf{u}_0$ are chosen closer to the optimal UAVs' positions, the convergence speed can be faster. In this section, we present a method using the past CSI to calculate the initial positions of the UAVs so that a better performance of proposed algorithm can be achieved. Obviously, the past CSI can reflect to some extent the current CSI at any location within the region of service. Roughly speaking, using the past CSI to maximize the average UEs' rates can be viewed as the optimization of the long-term network performance, which is reasonable for predicting the initial UAV's locations. We exploit this idea to determine an efficient initial state, named s_0^{eff} for the DCA-assisted DQL algorithm. Specifically, we propose that the MBS can collect and store the historical CSI forwarded from UAVs in order to calculate the initial locations. We denote the historical initial network state by $s_h = [s_{-M}, \ldots, s_{-1}]$ which contains all the M previous CSI. Hence, we harness these past CSI in the historical state s_h to efficiently determine the initial locations of UAVs s_0^{eff}. In what follows, we denote the past CSI used to calculate the initial locations of UAVs as $\tilde{h}_{ik,t}$, $\tilde{g}_{i,t}\ \forall t = -M, \ldots, -1$. Using these past CSI, we solve for the initial location of UAVs $\mathbf{u}$ jointly with UE association $\tilde{\mathbf{c}} = \{\tilde{\mathbf{c}}_t\}, \forall t = -M, \ldots, -1$ and transmit beamforming vector $\tilde{\mathbf{w}} = \{\tilde{\mathbf{w}}_t\}, \forall t = -M, \ldots, -1$, $\tilde{\mathbf{p}} = \{\tilde{\mathbf{p}}_t\}, \forall t = -M, \ldots, -1$ and $\tilde{\lambda} = \left\{\tilde{\lambda}_t\right\}, \forall t = -M, \ldots, -1$ in the following problem

$$(\mathcal{P}_0): \underset{\tilde{\mathbf{c}}, \tilde{\mathbf{w}}, \mathbf{u}, \tilde{\mathbf{p}}, \tilde{\boldsymbol{\lambda}}}{\text{maximize}} \quad \frac{1}{M} \sum_{t=-M}^{-1} \sum_{k \in \mathcal{K}} R_k^{\text{AtG}}(\tilde{\mathbf{w}}_t, \mathbf{u}) \tag{35a}$$

$$\text{s.t.} \quad R_k^{\text{AtG}}(\tilde{\mathbf{w}}_t, \mathbf{u}) \geq R_{k,\min}, \forall k \in \mathcal{K}, \forall t = -M, \ldots, -1 \tag{35b}$$

$$|\tilde{w}_{ik,t}|^2 \leq \tilde{c}_{ik,t} \tilde{\lambda}_{ik,t}, \forall i \in \mathcal{U}, k \in \mathcal{K}, \forall t = -M, \ldots, -1 \tag{35c}$$

$$\sum_{\forall k \in \mathcal{K}} \tilde{\lambda}_{ik,t} \leq P_{i,\max}, \forall i \in \mathcal{U}, \forall t = -M, \ldots, -1 \tag{35d}$$

$$d_{ik}(u_i) \leq R + \eta(1 - \tilde{c}_{ik,t}), \forall i \in \mathcal{U}, k \in \mathcal{K}, \forall t = -M, \ldots, -1 \tag{35e}$$

$$R_i^{\text{GtA}}(\tilde{p}_{i,t}, u_i) \geq \sum_{\forall k \in \mathcal{K}} \tilde{c}_{ik,t} R_k^{\text{AtG}}(\tilde{\mathbf{w}}_t, \mathbf{u}), \forall i \in \mathcal{U}, \forall t = -M, \ldots, -1 \tag{35f}$$

$$\sum_{\forall i \in \mathcal{U}} \tilde{p}_{i,t} \leq P_{\max}^{\text{MBS}} \tag{35g}$$

$$\tilde{c}_{ik,t} \in \{0, 1\}, \forall i, k \tag{35h}$$

Note that $\tilde{\mathbf{c}}$, $\tilde{\mathbf{w}}$, $\tilde{\mathbf{p}}$ are different from $\mathbf{c}$, $\mathbf{w}$, $\mathbf{p}$ in problem (17) while $\mathbf{u}$ in problem (17) and (35) is similar. The objective in (35a) is in fact an estimate of the average sum rate evaluated from the past CSI $\tilde{h}_{ik,t}, \tilde{g}_{i,t}, \forall t = -M, \ldots, -1$ and thus represents the long-term historical overall network performance. The reason for the choice of this objective function is that collecting CSI can be done offline in any previous time before running our DCA-assisted DQL algorithm. Due to the structural similarities between $(\mathcal{P}_0)$ and $(\mathcal{P}_t)$, we slightly modify Algorithm 1 to solve $(\mathcal{P}_0)$. We briefly explain some main modifications here. First, a lower bound function $F_k(\tilde{\mathbf{w}}_t, \mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n)$ of $f_k(\tilde{\mathbf{w}}_t, \mathbf{u})$ around the point $\tilde{\mathbf{w}}_t^n$ and $\mathbf{u}^n$ is given by

$$\begin{aligned} &F_k(\tilde{\mathbf{w}}_t, \mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) \\ &= f_k(\tilde{\mathbf{w}}_t^n, \mathbf{u}^n) + \breve{f}_k(\tilde{\mathbf{w}}_t; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) + \hat{f}_k(\mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) \\ &+ 2\xi_k \operatorname{Re}\left(\tilde{\mathbf{w}}_t^{nH} \tilde{\mathbf{w}}_t - \|\tilde{\mathbf{w}}_t^n\|^2 + \mathbf{u}^{nH} \mathbf{u}_k - \|\mathbf{u}^n\|^2\right) \end{aligned} \tag{36}$$

where $\breve{f}_k(\tilde{\mathbf{w}}_t; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n)$ and $\hat{f}_k(\mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n)$ are respectively given by

$$\begin{aligned} &\breve{f}_k(\tilde{\mathbf{w}}_t; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) = \\ &\frac{2\operatorname{Re}\left(\tilde{\mathbf{w}}_t^{nH} \mathbf{H}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t - \tilde{\mathbf{w}}_t^{nH} \mathbf{H}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t^n\right)}{\tilde{\mathbf{w}}_t^{nH} \mathbf{H}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t^n + \sigma_0^2} \\ &- \frac{2\operatorname{Re}\left(\tilde{\mathbf{w}}_t^{nH} \tilde{\mathbf{H}}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t - \tilde{\mathbf{w}}_t^{nH} \tilde{\mathbf{H}}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t^n\right)}{\tilde{\mathbf{w}}_t^{nH} \tilde{\mathbf{H}}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t^n + \sigma_0^2} \end{aligned} \tag{37}$$

$$\begin{aligned} \hat{f}_k(\mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) &= \frac{2\operatorname{Re}\left(\mathbf{u}^{nH} \Phi_k(\tilde{\mathbf{w}}_t^n) \mathbf{u} - \mathbf{u}^{nH} \Phi_k(\tilde{\mathbf{w}}_t^n) \mathbf{u}^n\right)}{\tilde{\mathbf{w}}_t^{nH} \mathbf{H}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t^n + \sigma_0^2} \\ &- \frac{2\operatorname{Re}\left(\mathbf{u}^{nH} \tilde{\Phi}_k(\tilde{\mathbf{w}}_t^n) \mathbf{u} - \mathbf{u}^{nH} \tilde{\Phi}_k(\tilde{\mathbf{w}}_t^n) \mathbf{u}^n\right)}{\tilde{\mathbf{w}}_t^{nH} \tilde{\mathbf{H}}_k(\mathbf{u}^n) \tilde{\mathbf{w}}_t^n + \sigma_0^2} \end{aligned} \tag{38}$$

where $\Phi_k(\tilde{\mathbf{w}}_t) = \sum_{\forall j \in \mathcal{K}} \omega_{jk}(\tilde{\mathbf{w}}_t)$,$\tilde{\Phi}_k(\tilde{\mathbf{w}}_t) = \sum_{\forall j \in \mathcal{K} \setminus k} \omega_{jk}(\tilde{\mathbf{w}}_t)$, $\phi_{jk}(\tilde{\mathbf{w}}_t) = \tilde{\mathbf{w}}_{j,t} \circ \tilde{\mathbf{h}}_{k,t}$, and $\omega_{jk}(\tilde{\mathbf{w}}_t) = \phi_{jk}(\tilde{\mathbf{w}}_t) (\phi_{jk}(\tilde{\mathbf{w}}_t))^H$. In the same way, we also can approximate function $y_k(\tilde{\mathbf{w}}_t, \mathbf{u}, \tilde{c}_{ik,t})$ around the point $\tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n$ as

$$\begin{aligned} &Y_k(\tilde{\mathbf{w}}_t, \mathbf{u}, \tilde{c}_{ik,t}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) = y_k(\tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) \\ &+ \breve{y}_k(\tilde{\mathbf{w}}_t; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) + \hat{y}_k(\mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) \\ &-2\zeta_{k,t}\left(\tilde{\mathbf{w}}_t^{nH} \tilde{\mathbf{w}}_t - \|\tilde{\mathbf{w}}_t^n\|^2 + \mathbf{u}^{nH} \mathbf{u} - \|\mathbf{u}^n\|^2 + \tilde{c}_{ik,t}^n \tilde{c}_{ik,t} - \left(\tilde{c}_{ik,t}^n\right)^2\right) \\ &+ \mathring{y}_k(\tilde{c}_{ik,t}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) \end{aligned} \tag{39}$$

where

$$\breve{y}_k(\tilde{\mathbf{w}}_t; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) = \tilde{c}_{ik,t}^n \breve{f}_k(\tilde{\mathbf{w}}_t; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) \tag{40}$$

$$\hat{y}_k(\mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) = \tilde{c}_{ik,t}^n \hat{f}_k(\mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) \tag{41}$$

$$\mathring{y}_k(\tilde{c}_{ik,t}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) = \left(\tilde{c}_{ik,t} - \tilde{c}_{ik,t}^n\right) R_k^{\text{AtG}}(\tilde{\mathbf{w}}_t^n, \mathbf{u}^n) \tag{42}$$

and the function $R_i^{\text{GtA}}(\tilde{p}_{i,t}, u_i)$ around the point $\tilde{p}_{i,t}^n, u_i^n$ as

$$\begin{aligned} &R_i^{\text{GtA}}(\tilde{p}_{i,t}, u_i; \tilde{p}_{i,t}^n, u_i^n) = R_i^{\text{GtA}}(\tilde{p}_{i,t}^n, u_i^n) \\ &+ \frac{|u_i^n \tilde{g}_{mi}|^2}{\tilde{p}_{i,t}^n |u_i^n \tilde{g}_{mi}|^2 + N_0}(\tilde{p}_{i,t} - \tilde{p}_{i,t}^n) + \frac{2\tilde{p}_{i,t}^n u_i^n \tilde{g}_{mi}^2 (u_i - u_i^n)}{\tilde{p}_{i,t}^n |u_i^n \tilde{g}_{mi}|^2 + N_0} \\ &+ 2\psi_{i,t}\left(p_{i,t}^n p_{i,t} - \left(\tilde{p}_{i,t}^n\right)^2 + u_i^n u_i - \left(u_i^n\right)^2\right). \end{aligned} \tag{43}$$

Now $(\mathcal{P}_0)$ is approximated at the nth iteration as

$$\begin{aligned} &\underset{\tilde{\mathbf{c}}, \tilde{\mathbf{w}}, \mathbf{u}, \tilde{\mathbf{p}}, \tilde{\boldsymbol{\lambda}}, \mathbf{z}}{\text{maximize}} \frac{1}{M} \sum_{t=-M}^{-1} \sum_{k \in \mathcal{K}} F_k(\tilde{\mathbf{w}}_t, \mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) \\ &- \xi_{k,t}\left(\|\tilde{\mathbf{w}}_t\|^2 + \|\mathbf{u}\|^2\right) - \upsilon^n \sum_{t=-M}^{-1} \sum_i \sum_k z_{ik,t} \end{aligned} \tag{44a}$$

$$\text{s.t.}\ F_k(\tilde{\mathbf{w}}_t, \mathbf{u}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n) - \xi_{k,t}\left(\|\tilde{\mathbf{w}}_t\|^2 + \|\mathbf{u}\|^2\right) \geq R_{k,\min} \tag{44b}$$

$$\begin{aligned} &R_i^{\text{GtA}}(\tilde{p}_{i,t}, u_i; \tilde{p}_{i,t}^n, u_i^n) - \psi_{i,t}\left(\tilde{p}_{i,t}^2 + u_i^2\right) \geq \\ &\sum_{\forall k \in \mathcal{K}} Y_k(\tilde{\mathbf{w}}_t, \mathbf{u}, \tilde{c}_{ik,t}; \tilde{\mathbf{w}}_t^n, \mathbf{u}^n, \tilde{c}_{ik,t}^n) \\ &+ \zeta_{k,t}\left(\|\tilde{\mathbf{w}}_t\|^2 + \|\mathbf{u}\|^2 + \tilde{c}_{ik,t}^2\right) \end{aligned} \tag{44c}$$

$$\tilde{c}_{ik,t} - 2\tilde{c}_{ik,t}^n \tilde{c}_{ik,t} + \left(\tilde{c}_{ik,t}^n\right)^2 \leq z_{ik,t} \tag{44d}$$

$$(35\text{c}) - (35\text{e}); (35\text{g}); (22\text{b}) \tag{44e}$$

where $\tilde{\mathbf{w}}^n, \tilde{\mathbf{c}}^n, \tilde{\mathbf{p}}^n$, υ^n are not the optimization variables but parameters obtained from the previous iteration. In the rest of the paper, Algorithm 2 with the initial state s_0^{eff} obtained from

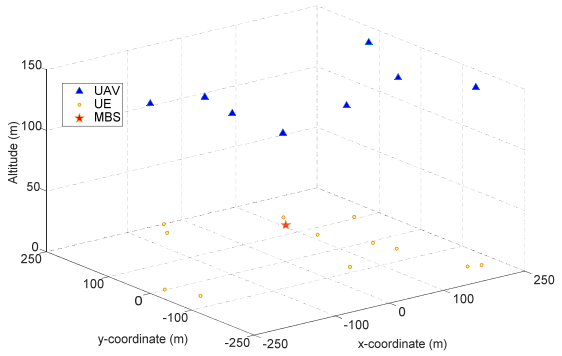


Fig. 2. 3D deployment of UAVs at the initial position $\mathbf{u}_0$, the MBS and ground UEs.

(44) is called Enhanced DCA-assisted DQL (EDCA-assisted DQL) Algorithm.

IV. Performance Evaluation

In this section, we evaluate the performance of our proposed scheme. The parameters and simulation settings used to produce our numerical results are listed in Table I. In our simulation as (c.f. Fig. 2), the ground UEs are randomly distributed in a square area of 500×500 m^2. Unless otherwise stated, we set $R_{k,\min} = 2.5$ b/s/Hz, $C = 10$, $D = 0.6$, $A_1 = 1$, $A_2 = 4.39$, $f = 6$ GHz, $\eta = 0.2$, and the communication range $R = 300$ meters [29], [36], [38]. Here, the proposed DQL algorithm was trained using Tensorflow 1.15 and Python 3.6 on Windows 10 for $L = 1000$ episodes, each of which has $T = 1000$ time steps. In addition, the maximum distance is set to $d_{\max} = 10$ meters. In Fig. 2, we plot the 3D deployment of the MBS at the position $(0, 0, \bar{H})$ and UAVs' positions $\mathbf{u}_0$ at the initial time.

A. Convergence Behavior of Our Proposed Algorithms

It is important to mention that at each iteration of Algorithm 2, we need to perform Algorithm 1. Clearly, the convergence speed of Algorithm 1 directly impacts the performance of Algorithm 2. To illustrate this point, we numerically show the convergence of Algorithm 1 in Fig. 3(a) with three different initial points and two different network settings $U = 8, K = 12$ and $U = 10, K = 15$. From Fig. 3(a), despite choosing different initial points and relatively larger numbers of UAVs and UEs, Algorithm 1 only requires a few iterations, e.g., approximately 20 iterations, to converge. This demonstrates a fast and stable convergence speed of our proposed DCA-based algorithm. Therefore, Algorithm 1 can be integrated into Algorithm 2 without significant performance loss due to the channel variations.

In Fig. 3(b), we show the convergence of the total reward obtained by our proposed DCA-assisted DQL algorithm (Algorithm 2) and the classical QL algorithm with different settings of $U = 2, 4$, and $K = 4$. In this experiment, we consider a small setting with 2 UAVs for the classical QL algorithm due to the exponential increase of the possible number of states and actions with the number of UAVs resulting in the increased computational complexity in maintaining the Q-table in the classical QL algorithm. Here, the total reward corresponding to each episode is the sum of reward $\sum_{t=0}^{T} r_t(s_t, a_t)$ over $T = 1000$ time steps. As seen in Fig. 3(b), our proposed DCA-assisted DQL algorithm converges to a total reward much higher and faster than that obtained by the classical QL algorithm for the same number of UAVs. Particularly, the proposed DCA-assisted DQL algorithm requires only around 300 episodes while classical-QL algorithm needs more than 500 episodes to converge. It can be observed that when U increases, the convergence speed of proposed DCA-assisted DQL algorithm varies slightly. For example, the total reward converges at episode 300 and 400 with $U = 2$ and $U = 4$, respectively. This shows the stable operation of the proposed DCA-assisted DQL algorithm.

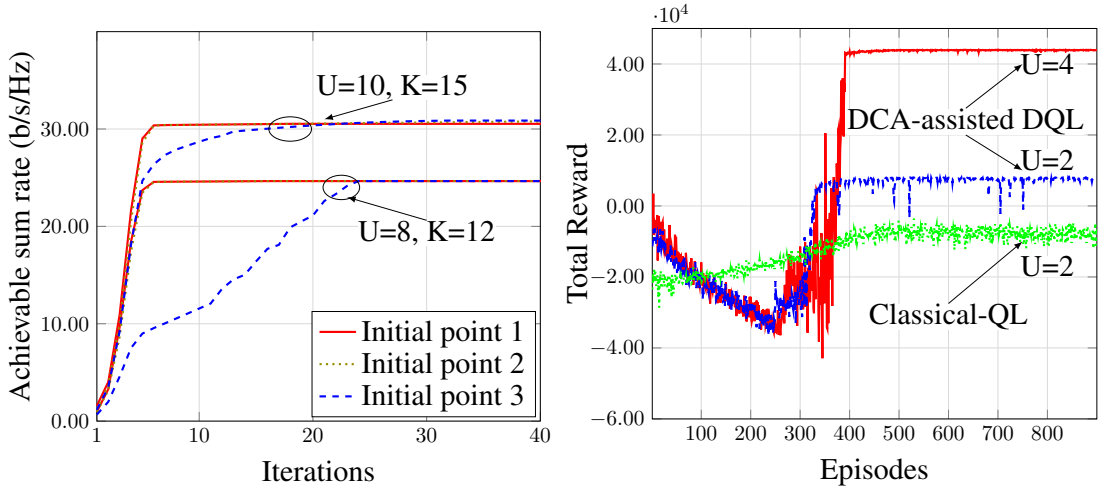
In Fig. 4(a), we illustrate the effect of infeasible cases during the training phase of DCA-DQL algorithm by varying $R_{k,\min}$. We assume that $R_{k,\min} = R_{\min}$, $\forall k \in \mathcal{K}$. It is shown that the convergence speed of DCA-assisted DQL algorithm at high $R_{\min} = 4$ b/s/Hz is less than that at low $R_{\min} = 2$ b/s/Hz. However, both scenarios saturate at the same value of total rewards at the convergence. This is explained as follows. When $R_{\min}$ increases, the infeasible cases occur more frequently. This slightly decreases the total rewards in the first episodes and needs a few more episodes to learn the task compare to the scenario of lower rate requirement. In addition, the deep Q-network is completely trained at the convergence for both scenarios. Thus, it can find the routes from the initial UAVs' positions to the optimal UAVs' positions which yield the same objective of maximal sum user rate. Fig. 4(b) illustrates the training loss calculated from (34). The loss first increases, then decreases, and saturates at the minimum value. This means that our proposed DCA-assisted DQL algorithm is quickly and successfully trained to approximate the $Q(s, a; \theta)$ values of states. This illustrates the effectiveness of our proposed DCA-assisted DQL algorithm in judging the relative consequences of different actions given a state so that higher reward decisions can be made.

Fig. 5(a) compares the convergence speed and gains between EDCA-assisted DQL algorithm with the efficient initial state s_0^{eff} at two different initial points (IPs) and Algorithm 3 with a random initial state s_0^{rand}. We observe that the curves of the EDCA-assisted DQL algorithm with IP 1 and IP 2 have almost the same convergence rate and achieved sum rate which are much better than that of the DCA-assisted DQL algorithm with the random initial state. Particularly, the EDCA-assisted DQL algorithm with IP 1 and IP 2 require only around 10 time steps to converge, but Algorithm 3 with s_o^{rand} requires up to 100 time steps to converge to a lower achievable sum rate. This can be explained as the EDCA-assisted DQL algorithm starts from the UAVs' positions which are very close to the optimal locations. Thus, it needs a small number of steps to reach the optimal solution. This again illustrates the effectiveness of our proposed EDCA-assisted DQL algorithm since the performance of the DRL method is very sensitive to the choice of the initial state.

The DRL based algorithm can adapt the resource allocation with time-varying channels since the agents (i.e. the UAVs) interact with the network environment in each iteration to get the states updated. Thus, the channel evolution is taken

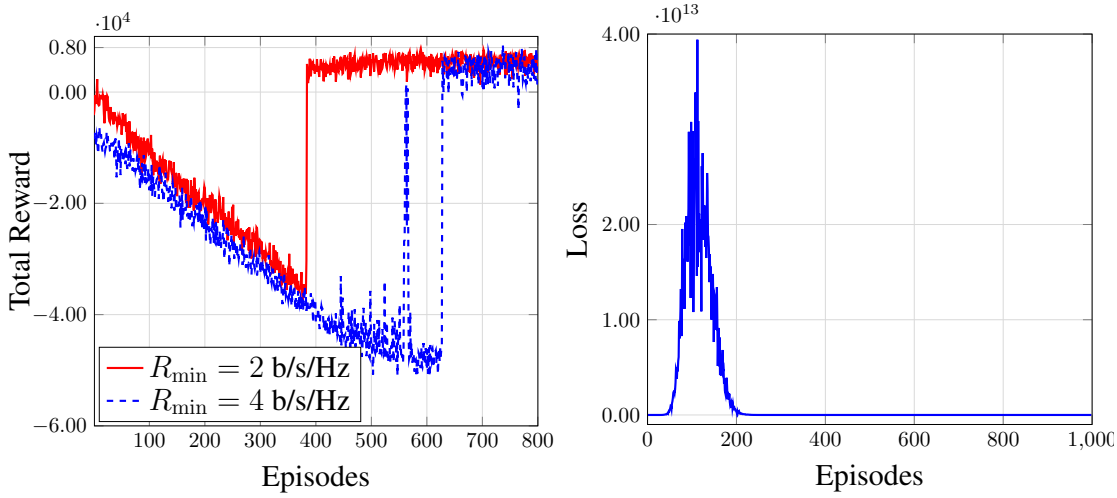
TABLE I
SIMULATION PARAMETERS

Parameters	Notation	Value	Parameters	Notation	Value
Height of UAV	H	150 meters	Replay buffer capacity	B	2000
Height of MBS	H	30 meters	Started training time	T_{start}	2000
Communication range of UAV	R	300 meters	Training time	T_{train}	4
Discount factor	γ	0.97	Batch size	N	48
Learning rate	β	0.001	Maximum transmit power of MBS	$P_{\text{max}}^{\text{MBS}}$	45 dBm
Random action probability	ϵ	1.0 to 0.05	Maximum transmit power of UAV	$P_{\text{max}}^{\text{UAV}}$	30 dBm
Target network update steps	E	100	Noise power	σ_0^2, σ_n^2	-120 dB



(a) (b)

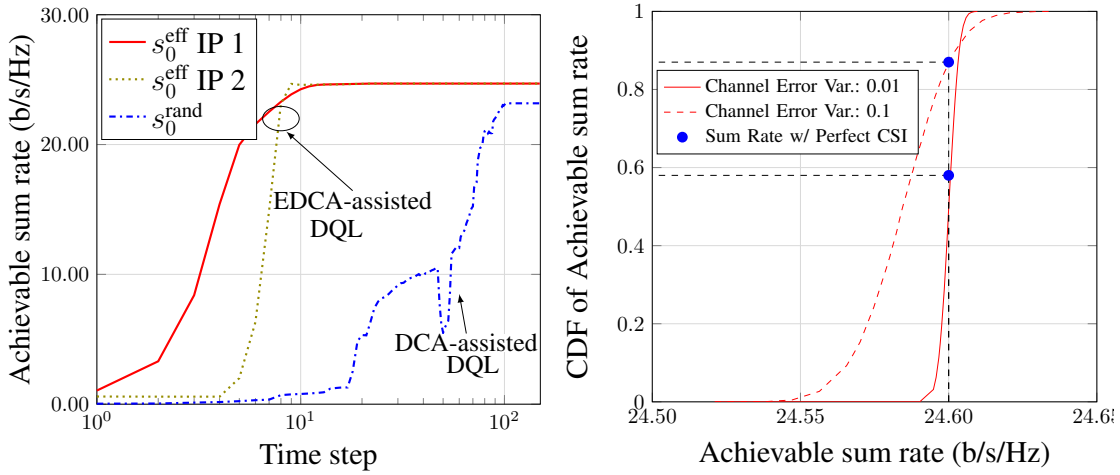
Fig. 3. (a): The convergence of Algorithm 1 with three different initial points and two network settings;(b): The convergence of training phase for DCA-assisted DQL algorithm and classical-QL algorithm with $U = 2$, 4 and $K = 4$.



(a) (b)

Fig. 4. (a): The convergence of DCA-assisted DQL algorithm with two values $R_{k,\min} = R_{\min}, \forall k \in \mathcal{K}$, $U = 2$, $K = 4$; (b): The loss behavior of DCA-assisted DQL algorithm with $U = 4$ and $K = 4$.

into account in each iteration so that the impact of channel time-variation can be reduced. In our proposed DCA-assisted DQL algorithm, at each time step t we need to run Algorithm 1 to compute the reward. Therefore, it is best if the proposed solution for the problem $(\mathcal{P}_t)$ can be solved quickly so that the CSI error has minimal impact on the eventual resource allocation solution. To assess the time sensitivity of our proposed solution, in Fig. 5(b), we plot the cumulative distribution function (CDF) of the sum UEs' achievable rates for Algorithm 1 when the channels are time-varying. The CDF



(a) (b)

Fig. 5. (a): The convergence of testing phases for EDCA-assisted DQL algorithm with two different initial points (IPs), e.g., s_0^{eff} IP 1 and s_0^{eff} IP 2, and DCA-assisted DQL algorithm with random initial state s_0^{rand}; (b): Cumulative distribution function (CDF) of achievable sum rate according to the channel error variance of 0.01 and 0.1, respectively. In these results, we consider $U = 8$, $K = 12$.

is obtained as follows. First, problem $(\mathcal{P}_t)$ is solved for given CSI using Algorithm 1. The resulting sum rate is denoted as expected sum achievable rate. Then, the obtained solution of UAVs' beamforming, UAV-UE association and MBS power allocation is used to calculate the sum UEs' rates for the new set of CSI, which is deviated from the given CSI. Particularly, the new channel realizations are generated by adding errors to the given channel realizations. The channel errors are drawn from a zero-mean Gaussian distribution with a variance of 0.01. As shown from Fig. 5(b), even if the CSI is outdated, the achieved performance is not far from the expected sum achievable rate. These results are positive in terms of the solution's time sensitivity.

B. Performance Comparison Between Our Proposed Algorithms and Other Existing Algorithms

To compare our proposed DCA-assisted DQL algorithm to other traditional convex approximation based solutions, we consider the following three optimization approaches: 1) the block coordinate descent and successive convex approximation based optimization methods in [19], [20] are applied to solve for UAVs' positions, UAV-UE association and transmit power allocation. Note that CSIs obtained from our DCA-assisted DQL algorithm are used prior to the knowledge of UAVs' positions for solving the problem in [19], [20]. The performance obtained from these two methods serves as the upper bounds for our proposed schema's performance; 2) UAVs are randomly placed and a beamforming technique, where cooperative UAVs transmit signals to their served UEs simultaneously in the same spectrum, is employed. Then, the beamforming and UAV-UE association variables are optimized by applying Algorithm 1. This scheme is named as random placement-DCA; 3) To further reduce the computational complexity, we employ the FDMA scheme where UAVs serve their UEs on their equally allocated spectrum fraction, e.g., $1/K$. Then,

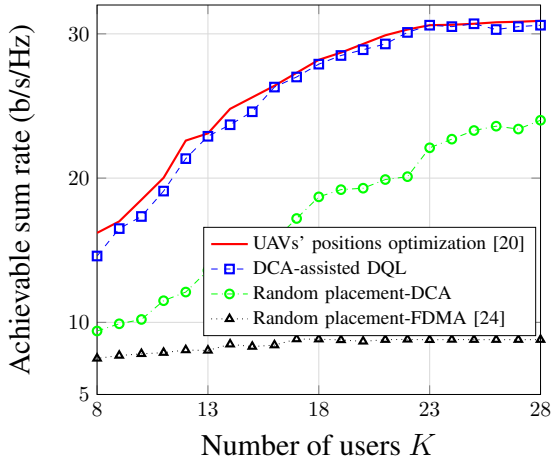


Fig. 6. Performance comparison between DCA-assisted DQL algorithm and three optimization approaches, namely UAVs' positions optimization, random placement-DCA and random placement-FDMA, respectively vs number of UEs K.

power allocation and UAV-UE association solution can be obtained by applying the successive convex approximation (SCA) method in [19] and [24]. This scheme is called random placement-FDMA. In Fig. 6, when the number of users K increases, the sum of UEs' achievable rates increases and saturates after some large K. As shown in Fig. 6, the performance of our proposed algorithm is very close to that of UAVs' positions optimization solution and outperforms other random placement based algorithms. Moreover, although the random placement-FDMA has lower computational complexity, its performance is far worse than that of other algorithms using beamforming technique.

In the next experiments, we further evaluate the performance of our proposed algorithm. In particular, the following schemes are compared:

- Proposed scheme: Joint UAVs' positions, transmit beam-

forming and UAV-UE association in the UAV cooperation based system is considered, where our proposed Enhanced DCA-assisted DQL algorithm is employed.

- Scheme A: Joint UAVs' positions, transmit beamforming and UAV-UE association design without UAV cooperation in [19] is considered, where our proposed Enhanced DCA-assisted DQL algorithm is applied. To express the "no cooperation between UAVs", we additionally impose the constraint $\sum_{\forall i \in \mathcal{U}} c_{ik} = 1, \forall k \in \mathcal{K}$ into the optimization problem. This scheme is included to understand the value of UAV cooperation in our considered system model.
- Scheme B: Joint UAVs' positions, power control and UE association without UAV cooperation is considered, where the ESN algorithm in [19], [29], [33] is applied. For power control design, we consider the transmit power for each UAV i: $\hat{p}_{ik,t} = [\hat{p_1}, \hat{p_2}, \hat{p_3}, \hat{p_4}]$ corresponding to 4 different maximum transmit power levels such that $\hat{p_1} = 1/4P_{i,\max}^{\text{UAV}}, \hat{p_2} = 1/2P_{i,\max}^{\text{UAV}}, \hat{p_3} = 3/4P_{i,\max}^{\text{UAV}}$ and $\hat{p_4} = P_{i,\max}^{\text{UAV}}$.
- Scheme C: UAVs' positions and UAV-UE association without UAV cooperation and fixed transmit power is considered, where the algorithm in [34] is involved.

In Fig. 7(a), we show the performance of different schemes versus the number of UAVs. In this setting, K is set to 12. It can be seen that the sum UEs' achievable rates of all schemes increase when the number of UAVs increases. The reason is that when more UAVs are deployed, more UEs can be served. In addition, we observe that our proposed scheme outperforms schemes A, B and C, which verifies the benefits of adopting the UAVs' cooperation in the considered system model. Particularly, our proposed scheme outperforms scheme B and C up to 70 % and 67% for high number of UAVs, e.g., $U = 10$, respectively. This is because when there are more UAVs, the interference issue becomes worse in scheme B and C than our scheme due to no UAVs' cooperation and interference management.. In addition, scheme A outperforms schemes B and C, which illustrates the effectiveness of the proposed DCA-assisted DQL algorithm compared to the ESN algorithm in [19], [29], [33]. This also demonstrates that choosing a fixed number of transmit power levels in the ESN algorithm yields lower performance compared to our proposed DCA-assisted DQL algorithm for solving the transmit beamforming optimization.

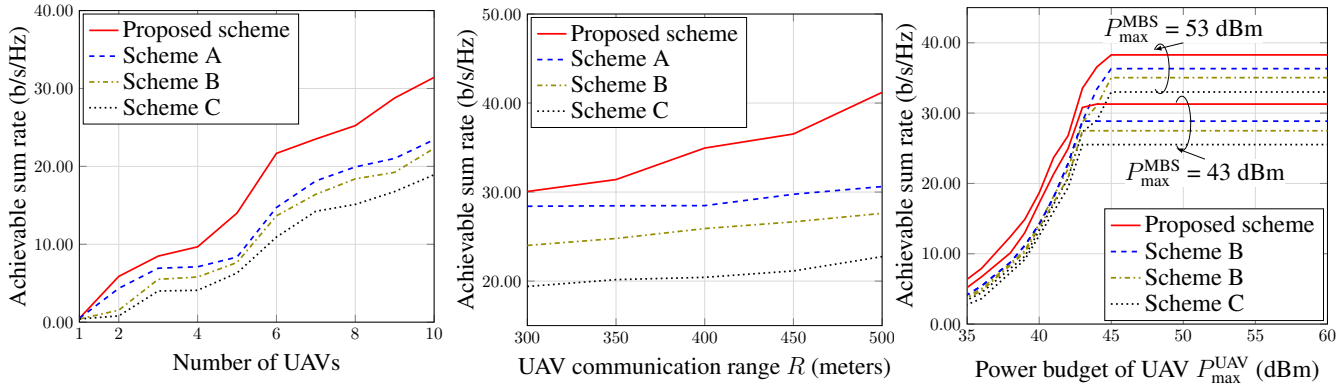
Fig. 7(b) demonstrates the impact of UAV communication range R on the sum UEs' achievable rates for our proposed scheme and schemes A, B and C. In this figure, we set $U = 8$ and $K = 12$. It is clear that the sum UEs' achievable rates of all schemes increase with R. This is because a larger communication range allows more UEs to be served and subsequently increases the sum UEs' achievable rates. It is worth mentioning that when R becomes sufficiently large, the performance gaps between our proposed scheme and schemes A, B and C are more profound. This is because more UEs can be served by coordinated UAVs as each UAV's coverage area overlaps with its neighbors. This leads to larger cooperative gains and increases the sum UEs' achievable rates in the proposed scheme. This again proves the advantages of our proposed method over the others.

In Fig. 7(c), we illustrate the impact of fronthaul rate constraints in the considered UAVs-assisted wireless networks. Particularly, we show the sum UEs' achievable rates with respect to $P_{\max}^{\text{UAV}}$ for $P_{\max}^{\text{MBS}} = 43$ and 53 dBm. We again observe that when $P_{\max}^{\text{UAV}}$ increases, the sum UEs' achievable rates of all schemes increase and saturate at the high regime of $P_{\max}^{\text{UAV}}$, where the proposed scheme always outperforms schemes A, B and C. It is clear that, when $P_{\max}^{\text{UAV}}$ is high, the UAVs can allocate more power to increase the UEs' rate. Moreover, as explained earlier, with UAVs' cooperation, the system can achieve higher cooperative gains, as shown by the remarkable gaps between the proposed scheme and the "no UAV cooperation" schemes A, B and C. In addition, limiting the transmit power to a few discrete levels in scheme B or assigning a fixed transmit power to UAVs as done in scheme C will reduce the flexibility of power allocation and cannot attain the optimal transmit power solution. This decreases the sum UEs' achievable rates performance compared to our proposed DCA-assisted DQL algorithm. Another observation is that when $P_{\max}^{\text{UAV}}$ is significantly high, UAVs do not allocate all of their available powers to the served UEs due to the bottleneck of the fronthaul capacity in (16). This results in a saturation of sum UEs' achievable rates in all schemes. Besides, the higher $P_{\max}^{\text{MBS}}$ is, the higher sum UEs' achievable rates can be obtained. This can be explained as when $P_{\max}^{\text{MBS}}$ increases, the MBS can allocate more power to increase the fronthaul rate, which in turn allows UAVs to allocate more power to their served UEs. Again, this corroborates the impact of the fronthaul rate capacity on the performance of UAV-assisted wireless networks as well as the improvement of our proposed scheme compared to the others.

Table II shows the computational complexity and signal overhead comparisons between our proposed algorithm and the others. It can be seen that our proposed algorithm requires lower computational complexity and signal overhead than that in [19]. Moreover, the computational complexity of other deep RL algorithms in [29], [33] is smaller than that of our proposed DCA-assisted DQL algorithm since the proposed DCA-assisted DQL algorithm needs to perform the optimization algorithm (e.g., Algorithm 1) inside. However, overhead signaling of our proposed algorithm is much less than that in [29] and [33] because UAVs do not have to exchange their information to each other. It is important to emphasize that our proposed DCA-assisted DQL algorithm outperforms other deep RL algorithms in [19], [29], [33] in terms of the sum achievable rates as shown in Fig. 7(a), 7(b), and 7(c).

V. Conclusion

We have investigated a joint design of UAVs' positions and resource allocation in the downlink of a multi-UAV-assisted wireless network where cooperative UAVs are considered. Specifically, we have jointly optimized the radio resource allocation at UAVs and the MBS along with UAVs' positions to maximize the sum UEs' achievable rates. To overcome



(a) (b) (c)

Fig. 7. (a): Performance comparison of different schemes vs number of UAVs; (b): Performance comparison of different schemes when varying the UAV communication range R; (c): Performance comparison of different schemes vs $P_{\max}^{\text{UAV}}$ with $P_{\max}^{\text{MBS}} = 43$, 53 dBm. In these results, we consider U= 8 and $K = 12$.

TABLE II
COMPLEXITY AND OVERHEAD COMPARISON

Schemes	UAVs cooperation	Computational Complexity	Signal overhead
Our scheme	Yes	$\mathcal{O}(\log \frac{1}{\epsilon} LU^{3.5}K^{3.5} + N + \lvert\theta\rvert)$	$\rho(UK + U)$
[19]	No	$\mathcal{O}(2^{KUN}K^2U^2N^2)$	$\rho K(U-1)UN$
[29]	No	$\mathcal{O}(nU + U + U)$	$\rho(3U + OU + SU)(U-1)$
[33]	No	$\mathcal{O}(U\lvert S\rvert\lvert A\rvert)$	$\rho\lvert S\rvert\lvert A\rvert(U-1)U$

the difficulties of solving the problem pertaining to the non-convexity and the CSI unavailability, we have proposed a novel method based on the deep reinforcement Q-learning framework in combination with a DCA based optimization technique to jointly solve for the UAV's positions and radio resource solution. By exploiting the historical CSI to calculate the long-term sum achievable rate of the network, we have also derived an efficient initial state which can improve the convergence speed of the proposed DCA-assisted DQL algorithm. Numerical results have showed that our proposed algorithm outperforms other known designs which aim at optimizing the network performance without using cooperative UAVs.

VI. ACKNOWLEDGEMENT

This publication has emanated from research supported in part by a Grant from Science Foundation Ireland under Grant number 17/CDA/4786.

APPENDIX

Problem (17) is an MINLP, which is generally NP-hard in the sense that we normally have to search over all possibilities of the integer variables to find an optimal solution. Here, we prove that (17) is NP-hard even when binary variable $\mathbf{c}$, UAVs' positions $\mathbf{u}$ are held fixed. For given binary variable $\mathbf{c}$ and UAVs' positions $\mathbf{u}$, (17) is reduced to

$$\underset{\mathbf{w},\mathbf{p},\boldsymbol{\lambda}}{\text{maximize}} \quad \sum_{k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w},\mathbf{u}) \tag{45a}$$

$$\text{subject to} \quad (8);(12);(13);(14);(16);(17b). \tag{45b}$$

We further fix the MBS power allocation $\mathbf{p}$ to simplify (45) as

$$\underset{\mathbf{w},\boldsymbol{\lambda}}{\text{maximize}} \quad \sum_{k\in\mathcal{K}} R_k^{\text{AtG}}(\mathbf{w},\mathbf{u}) \tag{46a}$$

$$\text{subject to} \quad (12);(13);(14);(16);(17b) \tag{46b}$$

where the left hand side of constraint (16) is now a constant. We remark that (46) is nothing but the sum rate maximization problem for spectrum management which was proved to be NP-hard in [43] based on a polynomial time reduction from the maximum independent set problem. This completes our proof.

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[2] M. Hasan, E. Hossain, and D. Niyato, "Random access for machine-to-machine communication in LTE-advanced networks: Issues and approaches," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 86–93, May 2013.

[3] P. Yang, Y. Xiao, M. Xiao, and S. Li, "6G wireless communications: Vision and potential techniques," *IEEE Network*, vol. 33, no. 4, pp. 70–75, Jul. 2019.

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

[5] M. Samir, S. Sharafeddine, C. M. Assi, T. M. Nguyen, and A. Ghrayeb, "UAV trajectory planning for data collection from time-constrained IoT devices," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 34–36, Jan. 2020.

[6] M. Mozaffari, W. Saad, M.Bennis, Y.-H. Nam, and M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2334–2360, Third Quater 2019.

[7] M. Gapeyenko *et al*, "Flexible and reliable UAV-Assisted backhaul operation in 5G mmWave cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2486–2496, Nov. 2018.

[8] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive mimo versus small cells," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1834–1850, Mar. 2017.

[9] G. Zhang, Q. Wu, M. Cui, and R. Zhang, "Securing UAV communications via joint trajectory and power control," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 1376–1389, Feb. 2019.

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

[11] T. M. Nguyen, W. Ajib, and C. Assi, "A novel cooperative NOMA for designing UAV-assisted wireless backhaul networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2497–2507, Nov. 2018.

[12] P. Luong, F. Gagnon, C. Despins, and L.-N. Tran, "Joint virtual computing and radio resource allocation in limited fronthaul green C-RANs," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2602–2617, Apr. 2018.

[13] ——, "Optimal joint remote radio head selection and beamforming design for limited fronthaul C-RAN," *IEEE Trans. Signal Process.*, vol. 65, no. 21, pp. 5605–5620, Nov. 2017.

[14] ——, "Optimal energy-efficient beamforming designs for Cloud-RANs with rate-dependent fronthaul power," *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 5099–5113, Jul. 2019.

[15] C.-C. Lai, C.-T. Chen, and L.-C. Wang, "On-Demand Density-Aware UAV base station 3D placement for arbitrarily distributed users with guaranteed data rates," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 913–916, 2019.

[16] R. Ghanavi, E. Kalantariy, M. Sabbaghian, H. Yanikomerogluz, and A. Yongacoglu, "Efficient 3D aerial base station placement considering users mobility by reinforcement learning," in *Proc. IEEE Wireless Commun. and Net. Conf. (WCNC)*, Barcelona, Spain, Apr. 2018, pp. 1–6.

[17] 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, 2018.

[18] J. Lyu, Y. Zeng, and R. Zhang, "UAV-aided offloading for cellular hotspot," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 3988–4001, 2018.

[19] 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.

[20] C. Shen, T.-H. Chang, J. Gong, Y. Zeng, and R. Zhang, "Multi-UAV interference coordination via joint trajectory and power control," *IEEE Trans. Signal Process.*, vol. 68, pp. 843–858, 2020.

[21] 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.

[22] S. Yin, L. Li, and F. R. Yu, "Resource allocation and basestation placement in downlink cellular networks assisted by multiple wireless powered UAVs," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 2171–2184, feb. 2020.

[23] J. Zhao, T. Q. S. Quek, and Z. Lei, "Coordinated multipoint transmission with limited backhaul data transfer," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2762–2775, 2013.

[24] L. Liu, S. Zhang, and R. Zhang, "CoMP in the sky: UAV placement and movement optimization for multi-user communications," *IEEE Trans. Commun.*, pp. 1–1, Mar. 2019.

[25] N. Rupasinghe, A. S. Ibrahim, and I. Guvenc, "Optimum hovering locations with angular domain user separation for cooperative UAV networks," in *2016 IEEE Global Commun. Conf. (GLOBECOM)*, Washington, DC, Dec. 2016, pp. 1–6.

[26] P. Dinh, T. M. Nguyen, S. Sharafeddine, and C. Assi, "Joint location and beamforming design for cooperative UAVs with limited storage capacity," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 8112–8123, Nov. 2019.

[27] P. S. Bithas, E. T. Michailidis, N. Nomikos, D. Vouyioukas, and A. G. Kanatas, "A survey on machine-learning techniques for UAV-based communications," *Sensor*, vol. 19, no. 23, pp. 1–39, Nov. 2019.

[28] J. Cui, Y. Liu, and A. Nallanathan, "Multi-agent reinforcement learning-based resource allocation for UAV networks," *IEEE Trans. Veh. Technol.*, vol. 19, no. 2, pp. 729–743, Feb. 2020.

[29] U. Challita, W. Saad, and C. Bettstetter, "Interference management for cellular-connected UAVs: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2125–2140, Apr. 2019.

[30] C. H. Liu *et al*, "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.

[31] Q. Zhang *et al*, "Machine learning for predictive on-demand deployment of UAVs for wireless communications," in *2018 IEEE Global Commun. Conf. (GLOBECOM)*, Abu Dhabi, Dec. 2018, pp. 1–6.

[32] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.

[33] 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.*, pp. 1–1, May 2019.

[34] H. Bayerlein, P. de Kerret, and D. Gesbert, "Trajectory optimization for autonomous flying base station via reinforcement learning," in *Proc. the 19th IEEE Int. Workshop on Signal Process. Advances in Wireless Commun. (SPAWC)*, Kalamata, Greece, June 2018, pp. 1–6.

[35] Q. Wu, L. Liu, and R. Zhang, "Fundamental trade-offs in communication and trajectory design for UAV-enabled wireless network," *IEEE Wireless Commun. Mag.*, pp. 36–44, Feb. 2019.

[36] 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. 2329–23 451, 2019.

[37] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal lap altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, 2014.

[38] C. You and R. Zhang, "3D trajectory optimization in Rician fading for UAV-enabled data harvesting," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3192–3207, 2019.

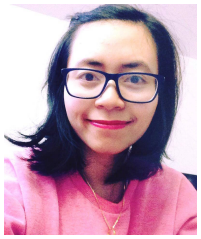
[39] S. Drewes and S. Ulbrich, *Mixed Integer Nonlinear Programming*. New York, NY: Springer, 2012.

[40] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[41] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.

[42] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization*. Philadelphia, PA, USA: SIAM, 2001.

[43] Z.-Q. T. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE J. Sel. Areas Commun.*, vol. 2, no. 1, pp. 57–73, Feb. 2008.



Phuong Luong (S'15–M'19) received the B.Eng. degree in telecommunications and electrical engineering from the Hanoi University of Science and Technology, Vietnam, in 2009, the M.E. degree from the Department of Computer Science, Kyung Hee University, Yongin, South Korea, in 2012 and the Ph.D. degree with the Ecole de Technologie Superieure (ETS), Montreal, QC, Canada. She had held the postdoc position at Mcgill University from 2019 to 2020. Her research interests include optimization and machine learning techniques in radio resource management in the cloud radio access networks (C-RAN) and the Unmanned Aerial Vehicles (UAVs) networks. She is currently working on machine learning technologies in wireless communications systems at Resilient Machine Learning Institute, Montreal, Canada.

François Gagnon (Senior Member, IEEE) received the B.Eng. and Ph.D. degrees in electrical engineering from École Polytechnique de Montréal, Montréal, QC, Canada. Since 1991 he has been a Professor with the Department of Electrical Engineering, École de Technologie Supérieure (ÉTS), Université du Québec, Montréal, Canada. He chaired the department from 1999 to 2001 and was the Director of the Institut pour la resilience et lapprentissage automatis (Resilient Machine Learning Institute). In addition to holding the Richard J. Marceau Industrial Research Chair for Wireless Internet in developing countries with Media5, he also holds the NSERC-Ultra Electronics Chair in Wireless Emergency and Tactical Communication. Most recently, he was appointed as the Director General of TS in June 2019. His research interests cover wireless high-speed communications, modulation, coding, high-speed DSP implementations, and military point-to-point communications.

Le-Nam Tran (M'10–SM'17) received the B.S. degree in electrical engineering from Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam, in 2003, and the M.S. and Ph.D. degrees in radio engineering from Kyung Hee University, Seoul, Korea, in 2006 and 2009, respectively. He is currently an Assistant Professor at the School of Electrical and Electronic Engineering, University College Dublin, Ireland. Prior to this, he was a Lecturer at the Department of Electronic Engineering, Maynooth University, Ireland. From 2010 to 2014, he had held postdoc positions at the Signal Processing Laboratory, ACCESS Linnaeus Centre, KTH Royal Institute of Technology, Stockholm, Sweden (2010-2011), and at Centre for Wireless Communications, University of Oulu, Finland (2011-2014). His research interests are mainly on applications of optimization techniques on wireless communications design. Some recent topics include energy-efficient communications, physical layer security, cloud radio access networks, cell-free massive MIMO, and full-duplex transmission. He has authored or co-authored in more than 100 papers published in international journals and conference proceedings.

Dr. Tran is an Associate Editor of EURASIP Journal on Wireless Communications and Networking. He was Symposium Co-Chair of Cognitive Computing and Networking Symposium of International Conference on Computing, Networking and Communication (ICNC 2016) and was a CoChair of the workshop on Scalable Massive MIMO Technologies for Beyond 5G of IEEE International Conference on Communications 2020.

Fabrice Labeau is the Deputy Provost (Student Life and Learning) at McGill University, where he also holds the NSERC/Hydro-Québec Industrial Research Chair in Interactive Information Infrastructure for the Power Grid. His research interests are in applications of signal processing. He has (co-)authored more than 200 papers in refereed journals and conference proceedings in these areas.

He is the Director of Operations of STARaCom, an inter-university research center grouping 50 professors and 500 researchers from 10 universities in the province of Quebec, Canada. He is President of the Institute of Electrical and Electronics Engineers (IEEE) Sensors Council, Senior Past President of the IEEE Vehicular Technology Society, and the past chair of the Montreal IEEE Section. He was a recipient in 2015 and 2017 of the McGill University Equity and Community Building Award (team category), of the 2008 and 2016 Outstanding Service Award from the IEEE Vehicular Technology Society and of the 2017 W.S. Read Outstanding Service Award form IEEE Canada. He was recognized in 2018 "Ambassadeur Accrédité" for the Montreal Convention Center. He is a "champion" for Engineers Canada?s 30 by 30 initiative and a member of Engineers Canada?s Indigenous Participation in Engineering working group.