Resource Allocation in UAV-Assisted Wireless Networks Using Reinforcement Learning

Phuong Luong, *Member, IEEE*, François Gagnon, *Senior Member, IEEE*, and Fabrice Labeau, *Senior Member, IEEE*

***Abstract*—In this work, 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 the fronthaul communications, to serve multiple ground users. A problem of jointly designing UAV's locations, transmit beamforming, as well as UAV-user association is formulated in the form of mixed integer nonlinear programming (MINLP) to maximize the sum user achievable rate while considering the constraints of limited fronthaul capacity. Solving the formulated problem is computationally hard owing to the its non-convex nature and the unavailability of channel state information (CSI) due to the undetermined and flexible movement of UAVs. To tackle these effects, we propose a novel algorithm exploiting the deep Q-learning approach to take the hassles of unavailable CSI for determining UAV's location and invoking the difference of convex (DC) based optimization method to efficiently solve for the UAV's transmit beamforming and UAV-user association given the determined UAV's location. The algorithm recursively solves the formulated problem until convergence. Numerical results show that our design outperforms the existing work in terms of algorithmic convergence and network performance and achieve a gain of up to 70% compared to the existing algorithms.**

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

I. Introduction

Wireless communications over the last decade has continuously witnessed tremendous efforts to standardize and implement 5G and its beyond to support many wireless applications and use-cases. The proliferation of Internet-of-Thing (IoT) incites new wireless network infrastructure to lean towards a highly agile network platform such as Unmanned Aerial Vehicle (UAV) assisted network [1]. In fact, this network can flexibly form, destruct, and reform any on-demand access network by dispatching flying-capable small base stations away from the fixed and grid-connected wireless access infrastructure to communicate with end-user, while backhauling data are handled via the backhaul or fronthaul link to the core network. The latest development of UAV wireless communication technology not only provides ubiquitous coverage and received signal strength due to the agility of its 3D movement, but also embraces beyond Line-of-Sight (LoS) transmissions and allows coordinated communication between UAVs in order to better manage interference, achieve cooperative gain, and improve network latency [2]. Thanks to the above prominent capabilities, UAV network (UAN) offers more effective way to adapt the dynamic traffic demands with stringent quality-of-service (QoS) requirement and thus is suitable for a myriad of application such as video streaming, surveillance, etc.

Harnessing the aforementioned benefits of UANs is no simple task since it must encounter many unsolved technical challenges in terms of resource allocation design. There have been several works that study a joint design of UAV location, transmit power and UE association in the UANs. The work in [3] aimed at optimizing the decoding order of the NOMA process and the positions of the UAVs in space to maximize the sum achievable rate of all users. In addition, [4] proposed the CoMP in the sky for the uplink communication of multi-UAV enabled multi-user system to maximize the network throughput via the design of UAV placement. However, they applied the ZF technique to approximate the worst-case achievable rate as well as neglected the fronthaul capacity limitation. However, these work assumed predetermined CSI as input to the optimization problem to solve for the UAVs position and resource allocation, which is not practical.

On the other hand, the deep reinforcement learning (DRL) approach has recently been exploited and applied to the problem of UAV position and resource allocation in the UANs. [5] proposed the 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. The deployment of UAVs was studied to minimize the UAV transmission power in [6]. Again, ESN algorithm using multi-agent Q-learning was used to predict the future position of UEs and determine the position of UAVs in [7]. However, no UAV cooperation and capacity limitation of fronthaul links between UAV and base stations were taken into account in these works.

Unlike [5]–[7] where there is no cooperation among UAVs, in this paper, we study the downlink of an UAN whose multiple UAVs can cooperatively serve their UEs using CoMP techniques. A central UAV controller located at the MBS is responsible for processing all baseband signal, coordinating resource computation, as well as transporting data to the UAVs via wireless fronthaul links [4]. By observing that UAV's cooperation and transmit beamforming correlate with UAV-UE association and each UAV's location, we propose a novel deep Q-learning based design in combination with optimization to jointly determine the UAVs' locations, UAV-UE association, and transmit beamforming at the UAVs. The objective is to maximize the overall system throughput, including the limited fronthaul capacity between the MBS and UAVs. The problem is formulated as a MINLP. The formulated problem is computationally hard to solve because of its non-convexity and the CSI related dilemma in which the methods in [5], [7] are no longer applicable. To tackle these issues, we propose the deep Q-learning based RL method (DQL) to develop an

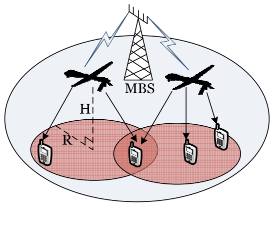


Fig. 1. Cooperative UAVs assisted wireless system

algorithm which allows UAVs to jointly learn the overall network current state and take their movements to adapt their position according to the action transmitted from MBS. Based on the UAV's solution, we employ the DC method based on Lipschitz continuity to handle the remaining non-convex problem of solving for UAV-UE association and beamforming. The outcome of this DC algorithm is then used to construct the decision policy of DQL algorithm to recompute the UAV position and this process is iterated until convergence. To the best of our knowledge, our work is the first to exploit the framework of DQL and DC for jointly determining UAV position, UAV-UE association and beamfomers in the cooperative UAVs-assisted wireless network.

The remaining of the paper is as follows. In Section II, we introduce the system model and formulate the problem of interest. Section III proposes the solution approach. Section IV shows our numerical results. Finally, conclusion are given in Section V.

II. System Model

A. Spatial Model

We consider the downlink communications of a UAN consisting of one macro base station (MBS) and a set $\mathcal{U} = \{1, \dots, U\}$ of UAVs operated in an circular area of radius d_0. Each UAV has a ground-projected communication range R and flies at a fixed altitude H to serve a group of ground UEs. We consider a scenario in which U UAVs connect to the ground controller associated with the MBS through wireless fronthaul links. We assume that the fronthaul communications between UAVs and MBS are accommodated on separate spectrum from the access communications between UAVs and UEs. Thus, there is no interference between UAV-MBS and UAV-UE links.

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 $k \in \mathcal{K} = \{1, \dots, K\}$ denotes the set of UE's indices within the given geographical region. In addition, we consider that each ith UAV positioned at $u_i = \{x_i, y_i, H\}$ can cooperate with other UAVs to serve the group of UEs. The UAV's positions are opted to be calculated after some period of time while the UE's positions are assumed fixed during a cycle of UAVs positioning. When the UEs shifts their position, we must recalculate the UAVs position according to the updated positions at any UAV positioning cycle. For simplicity, we only consider for a cycle of UAV positioning in this paper.

B. Channel Model

We assume that the fading channel remains unchanged within a coherence time. Following [3], we consider $h_{ik}(\delta_{ik})$ as the air-to-ground (AtG) channel from the ith UAV to the kth UE, which can be written as $h_{ik}(\delta_{ik}) = \delta_{ik}\tilde{h}_{ik}, \forall i \in \mathcal{U}, k \in \mathcal{K}$ where $\delta_{ik} \triangleq \Big[(x_i - \bar{x}_k)^2 + (y_i - \bar{y}_k)^2 + H^2\Big]^{-1/2}$ represents the path-loss coefficient and and $\tilde{h}_{ik}$ represents the small scale fading between the ith UAV and the kth UE. Since the AtG channel are often governed by LoS propagation, $\tilde{h}_{ik}$ follows the Rician distribution with factor K_b, which consists of a deterministic LoS component $\bar{h}_{ik}$ with $\left|\bar{h}_{ik}\right| = 1$ and a random scattered component $\hat{h}_{ik}$ as $\tilde{h_{ik}} = \sqrt{K_b/(1+K_b)}\bar{h}_{ik} + \sqrt{1/(1+K_b)}\hat{h}_{ik}$ where $\hat{h}_{ik}$ follows a complex Gaussian distribution $\mathcal{CN}(0,1)$. Similarly, we denote by $g_{mi}(\Delta_{mi}) = \Delta_{mi}\tilde{g}_{mi}$ as the ground-to-air (GtA) channel from MBS to the ith UAV, consisting of the pathloss coefficient $\Delta_{mi} \triangleq \Big[(\bar{x}_m - x_i)^2 + (\bar{y}_m - y_i)^2 + (H_m - H)^2\Big]^{-1/2}$ and small scale fading $\tilde{g}_{mi} = \sqrt{K_m/(1+K_m)}\bar{g}_{mi} + \sqrt{1/(1+K_m)}\hat{g}_{mi}$, where $\{\bar{x}_m, \bar{y}_m\}$ and H_m represent the fixed grounded position and height of MBS. Similar to $\tilde{g}_{mi}$ follows the Rician distribution of factor K_m which consists of a deterministic LoS component $\bar{g}_{mi}$ with $|\bar{g}_{mi}| = 1$ and a random scattered component $\hat{g}_{mi} \sim \mathcal{CN}(0,1)$.

C. Transmission Model

1) Transmission from MBS to UAV: Let us assume that the MBS can communicate with each UAV on the orthogonal subchannels. Then, the received signal at the ith UAV is given by $y_i = \sqrt{p_i} g_{mi}(\Delta_{mi}) s_{mi} + \tilde{n}_i$ where $p_i \in \mathbb{R}^+$ is the power from MBS to the ith UAV, s_{mi} is the message for the ith UAV where $\mathbb{E}\{s_i s_i^\star\} = 1$ and $\tilde{n}_i \sim \mathcal{CN}(0, N_0)$ is the AWGN at the ith UAV, where N_0 is the noise power. Thus, the achievable rate in b/s/Hz at the ith UAV is computed as

$$\Upsilon_i\left(p_i, \Delta_{mi}\right) = \log_2\left(1 + \frac{p_i\left|g_{mi}(\Delta_{mi})\right|^2}{N_0}\right) \quad (1)$$

2) Transmission from UAVs to UEs: We consider that all UAVs can serve their UEs simultaneously in the same spectrum by applying the beamforming technique. For notation convenience, we denote the set of beamforming vectors intended for the kth UE as $\mathbf{w}_k \triangleq [w_{1k}, w_{2k}, \dots, w_{Uk}] \in \mathbb{C}^{U \times 1}$, and the vector includ1ing the channels from all UAVs to the kth UE as $\mathbf{h}_k(\delta_k) \triangleq [h_{1k}(\delta_{1k}), h_{2k}(\delta_{2k}), \dots, h_{Uk}(\delta_{Uk})]^T \in \mathbb{C}^{U \times 1}$ where $\delta_k = \{\delta_{ik}\}_{\forall i \in \mathcal{U}}$ represents the location vector of all UAVs to the kth UE. Using these notations, the received signal at the kth UE is given by

$$y_k = \mathbf{h}_k^H(\delta_k)\mathbf{w}_k q_k + \sum\nolimits_{j \in \mathcal{K}\backslash k} \mathbf{h}_k^H(\delta_k)\mathbf{w}_j q_j + z_k \quad (2)$$

where q_k is the message for the kth UE where $\mathbb{E}\{q_k q_k^\star\} = 1$, $z_k \sim \mathcal{CN}(0, \sigma_0^2)$ is the additive white Gaussian noise (AWGN) and σ_0^2 is the noise power. In (2), we assumed that the kth UE is connected to all the UAVs, but the ith UAV serves the kth UE only if $\left\|w_{ik}s\right\|_2^2 > 0$. By treating interference as noise, the achievable rate in b/s/Hz at the kth UE is given by

$$\upsilon_k\left(\mathbf{w}, \delta_k\right) = \log_2\left(1 + \frac{|\mathbf{w}_k^H \mathbf{h}_k(\delta_k)|^2}{\sum_{j \in \mathcal{K}\backslash k} |\mathbf{w}_j^H \mathbf{h}_k(\delta_k)|^2 + \sigma_0^2}\right) \quad (3)$$

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

In addition, let us introduce the integer variable $c_{ik} = \{0,1\}_{\forall i \in \mathcal{U}, k \in \mathcal{K}}$ representing the association variable between the ith UAV and the kth UE. Accordingly, $c_{ik} = 1$ implies that the ith UAV serves the kth UE (i.e., $w_{ik} > 0$) and $c_{ik} = 0$ (i.e., $w_{ik} = 0$) otherwise. Importantly, to enable the transmission from a UAV to its associated UEs, the total UE achievable rate served by the ith UAV should be smaller than or equal to the fronthaul capacity provided from MBS to the ith UAV. This fronthaul rate constraint at the ith UAV is expressed as

$$\Upsilon_i \left(p_i, \Delta_{mi}\right) \geq \sum_{\forall k \in \mathcal{K}} c_{ik} \upsilon_k \left(\mathbf{w}, \delta_k\right) \tag{4}$$

D. Problem Formulation

We aim at finding the optimal UAVs location $\mathbf{u} = \{u_i = \{x_i, y_i\}\}, \forall i \in \mathcal{U}$, the UAV-UE association $\mathbf{c} = \{c_{ik}\}, \forall i \in \mathcal{U}, \forall k \in \mathcal{K}$and transmit beamformers $\mathbf{w}$ to maximize the sum achievable rate of all UE in the cooperative UAV-assisted network while guaranteeing the fronthaul rate and power budget constraints. By considering $\boldsymbol{\delta} = \{\delta_k\}, \forall k \in \mathcal{K}$ and $\boldsymbol{\Delta} = \{\Delta_{mi}\}, \forall i \in \mathcal{U}$ and $\Omega = \{\mathbf{c}, \mathbf{w}, \mathbf{u}, \mathbf{p}, \boldsymbol{\lambda}, \boldsymbol{\delta}, \boldsymbol{\Delta}\}$ as the slack variables, the problem can be formulated as

$$(\mathcal{P}) : \max_{\Omega} \sum_{k \in \mathcal{K}} \upsilon_k(\mathbf{w}, \delta_k) \tag{5a}$$

$$\text{s.t.} \delta_{ik}^{-1} \geq \sqrt{\left(x_i - \bar{x}_k\right)^2 + \left(y_i - \bar{y}_k\right)^2 + H^2} \tag{5b}$$

$$\Delta_{mi}^{-1} \geq \sqrt{\left(\bar{x}_m - x_i\right)^2 + \left(\bar{y}_m - y_i\right)^2 + \left(H_m - H\right)^2} \tag{5c}$$

$$\sum_{\forall i \in \mathcal{U}} p_i \leq P_{\max}^{\text{MBS}} \tag{5d}$$

$$\left\|w_{ik}\right\|^2 \leq c_{ik} \lambda_{ik} \tag{5e}$$

$$\sum_{\forall k \in \mathcal{K}} \lambda_{ik} \leq P_{i,\max}^{\text{UAV}} \tag{5f}$$

$$\sqrt{(x_i - \bar{x}_k)^2 + (y_i - \bar{y}_k)^2} \leq R + \eta(1 - c_{ik}) \tag{5g}$$

$$\Upsilon_i \left(p_i, \Delta_{mi}\right) \geq \sum_{\forall k \in \mathcal{K}} c_{ik} \upsilon_k \left(\mathbf{w}, \delta_k\right) \tag{5h}$$

Constraint (5d) is to guarantee the maximum power budget at MBS. The constraint in (5e) guarantees that the transmit power $\left\|w_{ik}\right\|^2$ from the ith UAV to the kth UE is zero if $c_{ik} = 0$ where λ_{ik} represents the soft power level corresponding to the upper bound of power that the ith UAV can transmit to the kth UE. The constraint in (5f) is to guarantee the power budget at each UAV. Constraint (5g) implies that the kth UE may be served by the ith UAV only if it is located in the communication range of the ith UAV where $\eta > 0$ and η is large to make (5g) hold. The constraints in (5h) represent the limited fronthaul rate constraints. Then, we have the observations about the problem $(\mathcal{P})$ as follows. First, problem $(\mathcal{P})$ is generally NP-hard due to the presence of binary variable $\mathbf{c}$. Moreover, even when this binary variable is relaxed to be continuous, the obtained problem is still non-convex because of the nonconvexity of the objective function (5a) and the constraints in (5b), (5c), and (5h). In mathematical programming, $(\mathcal{P})$ is categorized as a MINLP for which methods in previous works is not applicable to find a globally optimal solution. Given the non-convexity and combinatorial nature of $(\mathcal{P})$, a pragmatic goal is to find a sufficiently good feasible solution in a reasonable amount of time.

III. Proposed DQL and DC Algorithm

In this section, we present a novel framework of deep Q-learning based reinforcement learning (DQL) algorithm and difference of convex (DC) algorithm, called DQL-DC algorithm, that enables UAVs to learn the network state to adapt their position jointly with determining the transmit beamforming and UE association.

A. DQL Introduction

In DQL, an agent interacts with a system environment in a sequence of discrete times t. At each time t, the agent including MBS and UAVs observes the overall network state s_t, takes action a_t and receives the reward r_t. Then, the environment of network moves to new state s_{t+1} at time $t+1$. In order to present our proposed DQL algorithm, we first introduce and define the state s_t, action a_t and reward r_t at time step t as follows:

- Action $a_t = \left\{\phi_{i,t}, d_{i,t}\right\}, \forall i \in \mathcal{U}$ decided for all UAVs where $\phi_{i,t} = (0, 2\pi]$ and $d_{i,t} = [0, d_{\max}]$ are the movement direction and distance for each UAV i, respectively.
- The state $s_t = \left\{\left\{h_{ik,t}(\delta_{ik,t})\right\}, g_{mi,t}(\Delta_{mi,t})\right\}, \forall k, \forall i \in \mathcal{U}$ is the CSI obtained from all UAVs corresponding to UAVs location $\left\{u_{i,t}\right\}, \forall i \in \mathcal{U}$ at time t.
- After receiving an selected action a_t sent from MBS via fronthaul link, each UAV i moves to the new position according to movement direction $\phi_{i,t}$ and distance $d_{i,t}$, $\forall i \in \mathcal{U}$. Given the current positions, UAVs update and send to MBS their CSI $\left\{h_{ik,t+1}(\delta_{ik,t+1}), g_{mi,t+1}(\Delta_{mi,t+1})\right\}$ $, \forall k, \forall i \in \mathcal{U}$ which refers to the new state s_{t+1}. The transition from state s_t to s_{t+1} due to action a_t generates a reward $r_t\left(s_t, a_t\right)$ calculated in (6) where $\kappa > 0$ is a constant parameter and $\mathbf{w}_t^\star$ is the optimal beamforming solution obtained by solving the following optimization problem

$$(\mathcal{P}_t) : \underset{\mathbf{w}_t, \mathbf{p}_t, \boldsymbol{\lambda}_t, \mathbf{c}_t}{\text{maximize}} \quad \sum_{k \in \mathcal{K}} \upsilon_k(\mathbf{w}_t, \delta_{k,t}) \tag{7a}$$

$$\text{subject to} \quad \text{(5d); (5e); (5f); (5g); (5h)} \tag{7b}$$

given $\mathbf{u}_t$, δ_t, and Δ_t from the received action a_t and state s_t at the time step t. By doing this way, determining the transmit beamforming $\mathbf{w}_t^\star$ and UE association $\mathbf{c}_t^\star$ solution is associated with the UAVs position $\mathbf{u}_t$ and network environment s_t obtained at time step t of DQL-DC algorithm. However, it is not trivial to calculate the reward since solving $(\mathcal{P}_t)$ is very challenging due to the binary variable $\mathbf{c}$ and non-convexity of UE rate in objective (7a) and constraints (5h). In the following, we present our proposed DC algorithm to solve $(\mathcal{P}_t)$.

B. DC Algorithm for Solving $(\mathcal{P}_t)$

We observe that the problem (7) is difficult to solve mainly because of the non-convex non-concave rate function $\upsilon_k(\mathbf{w}_t, \delta_k)$ and the term $c_{ik,t}\upsilon_k\left(\mathbf{w}_t, \delta_k\right)$ with respect to variable $\mathbf{w}_t$. Based on the concept of DC programming, we will express each of the non-convex non-concave functions as the difference of two convex or concave ones. To illustrate this, we rewrite $\upsilon_k\left(\mathbf{w}_t, \delta_{k,t}\right)$ asoptimization

$$\upsilon_k(\mathbf{w}_t, \delta_{k,t}) = \underbrace{\upsilon_k(\mathbf{w}_t, \delta_{k,t}) + \xi_{k,t}\left\|\mathbf{w}_t\right\|^2}_{f_k(\mathbf{w}_t, \delta_{k,t})} - \xi_{k,t}\left\|\mathbf{w}_t\right\|^2 \tag{8}$$

$$r_t(s_t,a_t)=\begin{cases}\sum_{\forall k\in\mathcal{K}}\upsilon_k(\mathbf{w}_t^\star,\delta_{k,t})+\kappa & \text{if}\sum_{\forall k\in\mathcal{K}}\upsilon_k(\mathbf{w}_\mathbf{t}^\star,\delta_{k,t})>\sum_{\forall k\in\mathcal{K}}\upsilon_k(\mathbf{w}_\mathbf{t-1}^\star,\delta_{k,t-1})\\ \sum_{\forall k\in\mathcal{K}}\upsilon_k(\mathbf{w}_t^\star,\delta_{k,t}) & \text{if }\sum_{\forall k\in\mathcal{K}}\upsilon_k(\mathbf{w}_\mathbf{t}^\star,\delta_{k,t})=\sum_{\forall k\in\mathcal{K}}\upsilon_k(\mathbf{w}_\mathbf{t-1}^\star,\delta_{k,t-1})\\ \sum_{\forall k\in\mathcal{K}}\upsilon_k(\mathbf{w}_t^\star,\delta_{k,t})-\kappa & \text{otherwise}\end{cases}\tag{6}$$

for any $\xi_{k,t} \geq 0$. Intuitively if $\xi_{k,t}$ is chosen sufficiently large, the function $f_k(\mathbf{w}_t, \delta_{k,t})$ will become convex with respect to the variable $\mathbf{w}_t$ due to the dominance of the strongly convex quadratic term $\xi_{k,t}\left\|\mathbf{w}_t\right\|^2$. To find a proper value of $\xi_{k,t}$ to make (8) DC, we have for $\xi_{k,t} > \bar{\xi}_k$ where $\bar{\xi}_k = ||\mathbf{H}_k(\delta_{k,t})||_F + (2P_{\max}^{\text{UAV}}||\mathbf{H}_k(\delta_{k,t})||_F)^2 + ||\tilde{\mathbf{H}}_k(\delta_{k,t})||_F + (2P_{\max}^{\text{UAV}}||\tilde{\mathbf{H}}_k(\delta_{k,t})||_F)^2$, $f_k(\mathbf{w}_t, \delta_{k,t})$ is a $\bar{\xi}_k$-smooth function and strongly convex [8]. Here, we denote $\bar{\mathbf{H}}_k(\delta_{k,t}) = \mathbf{h}_k(\delta_{k,t})\mathbf{h}_k^H(\delta_{k,t})$ to support the notation of$\mathbf{H}_k(\delta_k) = \text{Bdiag}\underbrace{\left(\bar{\mathbf{H}}_k(\delta_{k,t}),\ldots,\bar{\mathbf{H}}_k(\delta_{k,t})\right)}_{K\text{elements}}$ and $\tilde{\mathbf{H}}_k(\delta_{k,t}) = \text{Bdiag}(\bar{\mathbf{H}}_k(\delta_{k,t}),\ldots,\underbrace{\mathbf{0}}_{k\text{th element}},\ldots,\bar{\mathbf{H}}_k(\delta_{k,t}))$ Similarly, we consider the following DC decomposition $c_{ik,t}\upsilon_k\left(\mathbf{w}_t,\delta_{k,t}\right) = \underbrace{\left(c_{ik,t}\upsilon_k\left(\mathbf{w}_t,\delta_{k,t}\right) - \zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2 + c_{ik,t}^2\right)\right)}_{y_k(\mathbf{w}_t,\delta_{k,t},c_{ik,t})} + \zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2 + c_{ik,t}^2\right)$ for any $\zeta_{k,t} \geq 0$. Similarly, for $\zeta_{k,t} > \bar{\zeta}_k$ where $\bar{\zeta}_k = \sqrt{2\bar{\xi}_k^2 + 8\left(||\mathbf{H}_k(\delta_{k,t})||_F^2 + ||\tilde{\mathbf{H}}_k(\delta_{k,t})||_F^2\right)P_{\max}^{\text{UAV}2}}$, $y_k(\mathbf{w}_t, \delta_k, c_{ik,t})$ is strongly concave [8]. 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

$$c_{ik,t} - c_{ik,t}^2 \leq 0 \tag{9}$$

$$0 \leq c_{ik,t} \leq 1 \tag{10}$$

Now, we observe that the non-convexity of objective function is due to the maximization over the convex function $f_k(\mathbf{w}_t, \delta_{k,t})$. Thus, we can iteratively approximate function $f_k(\mathbf{w}_t, \delta_{k,t})$ by its first order Taylor linearization $F_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]})$ around the point $\mathbf{w}_t^{[n]}$ as $F_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]}) = f_k(\mathbf{w}_t^{[n]}, \delta_{k,t}) + \breve{f}_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]}) + 2\xi_k \operatorname{Re}(\mathbf{w}_t^{[n]H}\mathbf{w}_t - ||\mathbf{w}_t^{[n]}||^2)$where $\breve{f}_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]})$ is in (11). Similarly, it can be seen that the non-convexity of (5h) is because of the concave function $y_k(\mathbf{w}_t, \delta_{t,k}, c_{ik,t})$ on the lesser side of inequality. In the same way, we can approximate function $y_k(\mathbf{w}_t, \delta_{k,t}, c_{ik,t})$ by its upper bound $Y_k(\mathbf{w}_t, \delta_{k,t}, c_{ik,t}; \mathbf{w}_t^{[n]}, c_{ik,t}^{[n]})$ around the point $\mathbf{w}_t^{[n]}, c_{ik,t}^{[n]}$ as $Y_k(\mathbf{w}_t, \delta_{k,t}, c_{ik,t}; \mathbf{w}_t^{[n]}, c_{ik,t}^{[n]}) = y_k(\mathbf{w}_t^{[n]}, \delta_{k,t}, c_{ik,t}^{[n]}) + \breve{y}_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]}, c_{ik,t}^{[n]}) + \mathring{y}_k(c_{ik,t}, \delta_{k,t}; \mathbf{w}_t^{[n]}, c_{ik,t}^{[n]}) - 2\zeta_{k,t}(\mathbf{w}_t^{[n]H}\mathbf{w}_t - ||\mathbf{w}_t^{[n]}||^2 + c_{ik,t}^{[n]}c_{ik,t} - (c_{ik,t}^{[n]})^2)$ where $\breve{y}_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]}, c_{ik,t}^{[n]}) = c_{ik,t}^{[n]}\breve{f}_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]})$ and $\mathring{y}_k(c_{ik,t}; \mathbf{w}_t^{[n]}, c_{ik,t}^{[n]}) = (c_{ik,t} - c_{ik,t}^{[n]})\upsilon_k(\mathbf{w}_t^{[n]}, \delta_{k,t})$. Similarly, we also approximated $c_{ik,t}^2$ in the left side of (9). Finally, by applying above approximations and introducing a new slack variable $\mathbf{z}_t = \{z_{ik,t} \geq 0a, \forall i, k\}$, we can formulate the convex approximation of (7) at the nth iteration as

$$\max_{\mathbf{w},\mathbf{p},\mathbf{c},\lambda,\mathbf{z}} \sum_{k\in\mathcal{K}} F_k(\mathbf{w}_t, \delta_k; \mathbf{w}_t^{[n]}) - \xi_{k,t}\left\|\mathbf{w}_t\right\|^2 - V\sum_i\sum_k z_{ik,t} \tag{12a}$$

$$\text{s.t.}\Upsilon_i\left(p_i, \Delta_{mi}\right) \geq \sum_{\forall k\in\mathcal{K}} Y_k(\mathbf{w}_t, \delta_k, c_{ik,t}; \mathbf{w}_t^{[n]}, c_{ik,t}^{[n]}) + \zeta_{k,t}\left(\left\|\mathbf{w}_t\right\|^2 + c_{ik,t}^2\right) \tag{12b}$$

$$c_{ik,t} - 2c_{ik,t}^{[n]}c_{ik,t} + \left(c_{ik,t}^{[n]}\right)^2 \leq z_{ik,t} \tag{12c}$$

$$(5d); (5e); (5g); (10) \tag{12d}$$

where $\mathbf{w}^{[n]}, \mathbf{c}^{[n]}$ are not the optimization variables but parameters obtained from the previous iteration and $V \geq 0$ is a penalty parameter. Thus, the problem (7) can be solved by the DCA based algorithm, which is outlined in Algorithm 1. The proof that Algorithm 1 converges after a finite number of iterations is similar to [8], which is omitted here.

Algorithm 1 DCA-based algorithm.

```
Set $n := 0$ and initialize starting points of $\mathbf{w}^{[n]}, \mathbf{c}^{[n]}$;
repeat
    Solve the approximated problem (12) to achieve the optimal solution $\mathbf{c}^\star, \mathbf{w}^\star, \mathbf{p}^\star, \lambda^\star, \mathbf{z}^\star$;
    Update $\mathbf{w}^{[n+1]} = \mathbf{w}^\star, \mathbf{c}^{[n+1]} = \mathbf{c}^\star$;
    Set $n := n + 1$;
until Convergence of the objective (12a)
```

C. DQL-DC Algorithm

In this section, we present the DQL-DC algorithm whose the decision policy is built based on the calculated reward $r_t(s_t, a_t)$ in (6) to recompute the UAV position until convergence. We recall that DQL is a popular method of RL algorithm that incorporates the deep Q-neural networks (DQNN) as the approximator of $Q(s_t, a_t, \theta)$ function to seek for the optimal actions from current states, where θ is the weights of the edges in DQNN. Action-value function $Q^\pi(s, a, \theta)$ is the expected accumulated reward when an action a is taken in the environmental state s under decision policy π

$$Q^\pi(s_t, a_t, \theta) = \mathbb{E}\left[R_t|s_t, a_t, \pi(s_t)\right] \tag{13}$$

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. In DQL, $Q(s_t, a_t; \theta)$ is updated by adjusting the weights θ in DQL through a training process. Specifically, the weights θ in DQL is trained and optimized by minimizing prediction errors of $Q(s_t, a_t; \theta)$ as follows. At time step t, given the state s_t input into DQNN which currently has the weights θ, the action a_t is chosen as $a_t = \arg\max_a Q(s_t, a; \theta)$ where $Q(s_t, a; \theta)$ are ouputs of DQNN corresponding to all different possible actions a. Given the action a_t is taken, DQL generates the reward $r_t(s_t, a_t)$ calculated as in (6) and the overall environment 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 able to be trained by minimizing prediction error of $Q(s_t, a_t; \theta)$ through the loss function $L_t(\theta)$ defined as

$$L_t(\theta) = \mathbb{E}\left[y_t(r_t, s_{t+1}) - Q(s_t, a_t; \theta)\right] \tag{14}$$

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

$$y_t(r_t, s_{t+1}) = r_t(s_t, a_t) + \gamma\max_a Q(s_{t+1}, a; \theta) \tag{15}$$

$$\breve{f}_k(\mathbf{w}_t, \delta_{k,t}; \mathbf{w}_t^{[n]}) \;=\; \frac{2\,\mathrm{Re}\left(\mathbf{w}_t^{[n]H}\mathbf{H}_k(\delta_{k,t})\mathbf{w}_t - \mathbf{w}_t^{[n]H}\mathbf{H}_k(\delta_{k,t})\mathbf{w}_t^{[n]}\right)}{\mathbf{w}_t^{[n]H}\mathbf{H}_k(\delta_{k,t}^{[n]})\mathbf{w}_t^{[n]} + \sigma_0^2} - \frac{2\,\mathrm{Re}\left(\mathbf{w}_t^{[n]H}\tilde{\mathbf{H}}_k(\delta_{k,t})\mathbf{w}_t - \mathbf{w}_t^{[n]H}\tilde{\mathbf{H}}_k(\delta_{k,t})\mathbf{w}_t^{[n]}\right)}{\mathbf{w}_t^{[n]H}\tilde{\mathbf{H}}_k(\delta_{k,t})\mathbf{w}_t^{[n]} + \sigma_0^2} \qquad (11)$$

TABLE I
SIMULATION PARAMETERS

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

Thus, the weights θ of DQNN now can be updated by minimizing loss function $L_t(\theta)$.

To improve learning stability, we employ the experience replay technique where 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 (14). 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

$$\bar{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(y_i(r_i, s_{i+1}) - Q(s_i, a_i; \theta)\right)^2 \qquad (16)$$

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

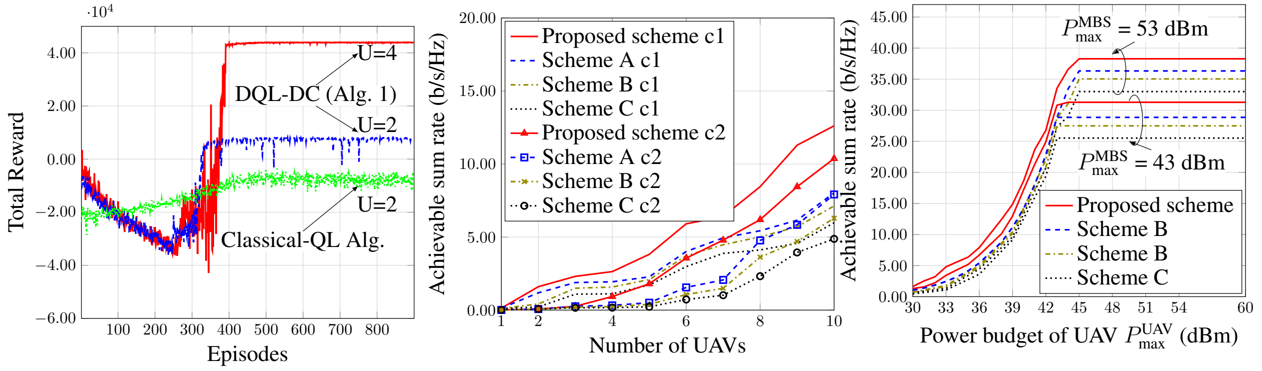
Algorithm 2 The DQL-DC Algorithm

```
MBS initializes weights $\theta$ of DQNN, weights $\theta^{\text{target}}$ of target DQNN, replay buffer with capacity $\mathbf{B}$, $\epsilon$ , $\beta$, $\gamma$, $\rho$, $N$, $E$ and global step $l := 0$
for Episode: $0 : L$ do
    UAVs are randomly positioned $\mathbf{u}_0$, estimates CSI state $s_0^{\text{rand}}$ and send to MBS
    for Time step: $t = 0 : T$ do
        Decide action $a_t = \begin{cases} \arg\max_a Q(s_t, a; \theta) & \text{with probability 1-}\epsilon \\ \text{random action} & \text{with probability } \epsilon \end{cases}$
        Applying Alg. 1 to solve (7) and achieve $\mathbf{w}_t^\star, \mathbf{c}_t^\star$ to calculate $r_t(s_t, a_t)$ in (6); then MBS sends the actions $a_t$, $\mathbf{w}_t^\star$ and $\mathbf{c}_t^\star$ to UAVs.
        UAVs move to their new positions $\mathbf{u}_{t+1}$ according to the received action $a_t$ and estimate the new CSI $s_{t+1}$ and send back to MBS.
        for $i$=0...U do
            if $\Delta_{mi,t} > d_0$ then
                Punish action $a_t$ by deducting the reward: $r_t(s_t, a_t) := r_t(s_t, a_t) - p$ with a penalty $p$ and put UAV $i$ back to previous position: $s_{t+1} \leftarrow s_t$
            end if
        end for
        Store sample $(s_t, a_t, r_t, s_{t+1})$ into replay buffer $\mathbf{B}$
        if Remainder($\frac{l}{T_{\text{train}}}$)==0 and $l > T_{\text{start}}$ then
            Randomly sampling $N$ experience samples from the replay buffer $\mathbf{B}$
            for $i = 1 : N$ do
                Compute target value $y_i(r_i, s_{i+1}) = r_i(s_i, a_i) + \gamma \max_a Q(s_{i+1}, a; \theta^{\text{target}})$
            end for
            Update weights $\theta$ by minimizing the loss: $\bar{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(y_i(r_i, s_{i+1}) - Q(s_i, a_i; \theta)\right)^2$
        end if
        if Remainder($\frac{l}{E}$)==0 and $l > T_{\text{start}}$ then
            Update $\theta^{\text{target}} := \theta$
        end if
        $t := t + 1$; $l := l + 1$
    end for
end for
```

IV. PERFORMANCE EVALUATION

In this section, we evaluate the performance of our proposed scheme with DQL algorithm. We also introduce the "no UAVs cooperation" scheme in the reference [5] and classical Q-learning algorithm as the baseline schemes. The parameters and simulation settings used to produce our results are listed in Table I. In our simulation, we consider the circular coverage of radius d_0 = 10 meters centered at the MBS with coordinate $v_m = \{0, 0\}$. The ground UEs are randomly distributed within the circle. Unless otherwise stated, we set K_b = 3dB, $K_m = 4$ dB, $R = 4$ meters. Here, the proposed DQL algorithm was trained using Tensorflow 1.15 and Python 3.6 on Window 10 for $L = 1000$ episodes, each of which has $T = 1000$ time steps. In Fig. 2(a), we show the convergence of total reward obtained by our proposed DQL-DC algorithm (Alg. 2) and classical-QL algorithm with different setting of $U = 2$ and 4 UAVs and $K = 4$ UEs. In this experiment, we consider the small setting with 2 UAVs for classical-QL algorithm due to the exponential increase of the possible number of states and actions with number of UAVs resulting in the increased computational complexity in maintaining the Q-table in 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, our proposed DQL-DC algorithm converges a total reward much higher and faster than that obtained by the classical-QL algorithm for the same number of UAVs. It can be observed that when U increases, the convergence speed of proposed DQL-DC algorithm varies slightly. This shows the stable operation of proposed DQL-DC algorithm which can be scalable when UAV number increases.

In the next experiments, we compare the performance of our proposed scheme and algorithm to other related schemes. Scheme A considers no UAV cooperation [9], where our proposed DQL-DC algorithm is applied. Scheme B also considers no UAV cooperation, however the ESN algorithm



(a) (b) (c)

Fig. 2. (a) The convergence of total rewards between proposed DQL-DC algorithm (Alg. 2) and classical-QL algorithm vs number of episodes; (b) Performance comparison of our proposed cooperative UAVs with no cooperation scheme vs number of UAVs; (c) Performance comparison of different schemes vs $P_{\max}^{\text{UAV}}$.

in the reference [5], [7], [9] is applied. For power control design, 4 different transmit power level for each UAV i: $\hat{p}_{ik,t} = [1/4P_{i,\max}^{\text{UAV}}, 1/2P_{i,\max}^{\text{UAV}}, 3/4P_{i,\max}^{\text{UAV}}, P_{i,\max}^{\text{UAV}}]$ is considered. Scheme C considers no UAV cooperation and fixed transmit power, where the algorithm in [10] is used. In Fig. 2(b), we consider two network configurations for the comparison, which we set the circular coverage of radius $d_0 = 10$ and $d_0 = 20$ meters centered at the MBS, respectively while the same number of UEs $K = 12$ are randomly placed in configuration 1 (c1) and configuration 2 (c2). In Fig. 2(b), it can be seen that the achievable sum rate of all schemes in the c1 is higher than the corresponding schemes in the c2 and the achievable sum rate increases when number of UAVs increases for all schemes and both configurations. It is simply explained that in the smaller considered coverage area, more UEs can be covered by UAVs than in the larger area. In addition, we observe that our proposed scheme outperforms scheme A, B and C, which verifying the benefit of considering the cooperation between UAVs as well as the effectiveness of proposed DQL-DC algorithm compared to ESN algorithm in [5], [7], [9]. Particularly, our proposed scheme in c1 outperforms the scheme B and C up to 70 % and 67% at very high number of UAVs, e.g., U=10, respectively.

In Fig. 2(c), we show the achievable sum rate with respect to the UAV's maximum power budget $P_{\max}^{\text{UAV}}$ at two different MBS's maximum power budget $P_{\max}^{\text{MBS}} = 43$ and 53 dBm, where U= 6 UAVs and $K = 12$ UEs. We again observe that when $P_{\max}^{\text{UAV}}$ increases, the achievable sum rate of all schemes increase and saturate at high regime of $P_{\max}^{\text{UAV}}$, where the proposed scheme always outperforms scheme A, B and C. When the UAVs have higher $P_{\max}^{\text{UAV}}$, the UAVs can allocate more power to increase the users' rate. An other observation is that when $P_{\max}^{\text{UAV}}$ is significant high, UAVs do not allocate all of their available power to served UEs due to the bottleneck on the fronthaul rate between UAV and MBS as shown in the limited fronthaul rate constraints (4). This results in a saturation of rate of all users in all schemes. Besides, the higher $P_{\max}^{\text{MBS}}$, the higher achievable sum rate 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 UAV allocating more available its power to the served UEs. This corroborates the impact of fronthaul rate capacity on the performance of UANs.

V. Conclusion

In this paper, we investigated the design of UAVs position and resource allocation in the downlink of an UANwhere cooperative UAVs scheme is considered to enhance the system performance. We jointly optimized the radio resource allocation at UAVs and MBS along with UAVs position to maximize the users' sum rate. We proposed a novel framework based on the deep reinforcement Q-learning method in combination with DC program based optimization to jointly solve for the UAV's positions and radio resource solution. Numerical results showed that our achieved solution, under the proposed model and developed algorithm, can outperform the other designs which aim at optimizing without using cooperative UAVs.

References

[1] M. Samir *et al.*, "UAV trajectory planning for data collection from time-constrained IoT devices," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 34–36, 2020.

[2] M. Gapeyenko, V. Petrov, D. Moltchanov, S. Andreev, N. Himayat, and Y. Koucheryavy, "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.

[3] 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.

[4] 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.

[5] 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, 2019.

[6] 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.

[7] 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.

[8] 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, 2018.

[9] 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.

[10] H. Bayerlein, P. de Kerret, and D. Gesbert, "Trajectory optimization for autonomous flying base station via reinforcement learning," in *Proc. SPAWC*, Kalamata, Greece, June 2018, pp. 1–6.