Path Planning for Cellular-Connected UAV: A DRL Solution With Quantum-Inspired Experience Replay

Yuanjian Li, *Graduate Student Member, IEEE*, A. Hamid Aghvami, *Life Fellow, IEEE*, and Daoyi Dong, *Senior Member, IEEE*

***Abstract*— In cellular-connected unmanned aerial vehicle (UAV) network, a minimization problem on the weighted sum of time cost and expected outage duration is considered. Taking advantage of UAV's adjustable mobility, a UAV navigation approach is formulated to achieve the aforementioned optimization goal. Conventional offline optimization techniques suffer from inefficiency in accomplishing the formulated UAV navigation task due to the practical consideration of local building distribution and directional antenna radiation pattern. Alternatively, after mapping the navigation task into a Markov decision process (MDP), a deep reinforcement learning (DRL)-aided solution is proposed to help the UAV find the optimal flying direction within each time slot, and thus the designed trajectory towards the destination can be generated. To help the DRL agent commit a better trade-off between sampling priority and diversity, a novel quantum-inspired experience replay (QiER) framework is proposed, via relating experienced transition's importance to its associated quantum bit (qubit) and applying Grover iteration based amplitude amplification technique. Compared to several representative DRL-related and non-learning baselines, the effectiveness and supremacy of the proposed DRL-QiER solution are demonstrated and validated in numerical results.**

***Index Terms*— Drone, trajectory design, deep reinforcement learning, quantum-inspired experience replay.**

I. Introduction

With flexible mobility, low cost and on-demand deployment, unmanned aerial vehicles (UAVs) have been widely used in civilian scenarios, e.g., building safety inspections, disaster management, material transport and aerial photography [2]–[4]. In practice, simple point-to-point (P2P) wireless links over unlicensed spectrum are commonly utilized to support the communications between UAVs and ground nodes, leading to constrained communication performance [4]. To further enhance wireless transmission qualities between UAVs and ground transceivers, cellular-connected UAV technique is deemed as a promising solution, via adopting widely-existing terrestrial base stations (BSs) to help establish high-quality ground-to-air (G2A) transmission links [5]–[7]. With the help of today's mature cellular networks and authentication mechanisms, cellular-connected UAV can help achieve better reliability, security and transmission throughput for G2A communications. Besides, cellular-connected UAV solution is significantly cost-effective because no dedicated infrastructures for supporting G2A communications are needed to construct and worldwide cellular BSs can be reused to aid G2A transmissions.

Current cellular networks are genuinely established for serving user equipments on the ground, via downtilting the main lobe of BS's antenna towards the earth. This characteristic can enhance cellular coverage facing the ground but the quality of cellular-aided G2A transmissions cannot be guaranteed in general [5]. To investigate wireless coverage support of current cellular network for UAVs, Lyu and Zhang [8] proposed a novel analytical framework for characterizing G2A uplink/downlink transmissions, where downtilted vertically-directional radiation pattern of BS's antenna is taken into account. Besides, more severe inter-cell interferences (ICIs) introduced by line-of-sight (LoS)-dominated G2A links can further deteriorate the aerial coverage issue, compared to terrestrial communication scenario where non line-of-sight (NLoS) channels are most likely experienced [9]. Fortunately, the controllable mobility feature of UAV makes it possible to tackle the aforementioned aerial coverage obstacles via UAV trajectory planning, either by on-board algorithms or remote pilots. The UAV navigation approach takes advantage of an extra degree of freedom, i.e., UAV's mobility, to realize aerial coverage enhancement and thus poses less or even no requirements on reconstruction of existing cellular infrastructures. Zhang *et al.* [7] studied cellular-connected UAV's mission completion time minimization problem via invoking graph theory and convex optimization to design the optimal flying trajectory from an initial location to a destination, subject to connectivity constraint of the G2A link. Zhan and Zeng [10] maximized data uploading throughput for cellular-connected UAV under constraints of energy cost and minimum transmission rate threshold, via path planning with the help of successive convex approximation (SCA) technique. Bulut and Guevenc [11] proposed a dynamic

Manuscript received 15 September 2021; revised 23 January 2022; accepted 19 March 2022. Date of publication 4 April 2022; date of current version 11 October 2022. The work of Yuanjian Li was supported by the China Scholarship Council and King's College London (joint full-scholarship K-CSC) under Grant CSC201908350102. An earlier version of this paper was presented in part at the IEEE ICC, Seoul, in May 2022 [1] [DOI: 10.48550/arXiv.2108.13184]. The associate editor coordinating the review of this article and approving it for publication was W. Zhang. *(Corresponding author: Yuanjian Li.)*

Yuanjian Li and A. Hamid Aghvami are with the Centre for Telecommunications Research (CTR), King's College London, London WC2R 2LS, U.K. (e-mail: yuanjian.li@kcl.ac.uk; hamid.aghvami@kcl.ac.uk).

Daoyi Dong is with the School of Engineering and Information Technology, University of New South Wales, Canberra, ACT 2600, Australia (e-mail: d.dong@adfa.edu.au).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TWC.2022.3162749.

Digital Object Identifier 10.1109/TWC.2022.3162749

programming solution to help cellular-connected UAV find the best travelling path, subject to a continuous disconnection duration restriction. Other than UAV trajectory design for cellular-connected UAV network, Hu *et al.* investigated joint optimization problems on energy consumption and path planning for scenarios of UAV-aided legitimate monitoring [12] and covert UAV-on-UAV video tracking and surveillance [13], where the specific optimization goals are both solved via convex optimization techniques. However, standard off-line optimization approaches solving trajectory design problem suffer from inefficiency due to non-convex nature of the formulated optimization objective and the corresponding constraints, even under impractical assumptions where perfect knowledge of wireless environment is available, e.g., G2A channel model and BS antenna model. Fortunately, reinforcement learning (RL) serves as a good complement to traditional off-line optimization solutions, which is famous for the favourable ability of learning unknown environment in a trial-and-error manner [14]. Up to date, RL-related techniques have been widely applied to help solve performance optimization problems for UAV-mounted networks, e.g., radio resource allocation, interference mitigation and path planning. Cui *et al.* [15] investigated a real-time design on resource allocation for multiple-UAV network, in which multi-agent reinforcement learning (MARL) framework was proposed to realize optimal user selection, power allocation and sub-channel association. Zeng *et al.* [5] investigated an optimal UAV trajectory planning problem on minimizing the weighted sum of mission completion time and expected transmission outage duration, via deep reinforcement learning (DRL)-aided approaches. Meanwhile, quantum theory has been proven to pose a positive impact on improving learning efficiency for artificial intelligence algorithms in general, and RL-related approaches in particular. Dong *et al.* [16] combined quantum parallelism into conventional RL frameworks (termed as quantum RL (QRL)), in which higher learning efficiency and better trade-off between exploration and exploitation were showcased. Furthermore, Dong *et al.* [17] proposed quantum-inspired reinforcement learning (QiRL) to solve intelligent navigation problem for autonomous mobile robots, where probabilistic action selection method and novel reinforcement approach inspired by quantum phenomenon were integrated into standard RL frameworks. Paparo *et al.* [18] showed that quadratic speed-up is achievable for intelligent agents, with the help of quantum mechanics. Dunjko *et al.* [19] extended traditional agent-environment framework into quantum region, while Saggio *et al.* [20] demonstrated the first experimental result of QRL. In [21], Li *et al.* compared QRL with several RL frameworks in human decision-making scenarios, suggesting that value-based decision-making can be illustrated by QRL at both the behavioural and neural levels. In the field of wireless communications, Li *et al.* [22] investigated an optimal path planning problem for UAV-mounted networks, in which QiRL solution was demonstrated to offer better learning performance than conventional RL methods with ϵ-greedy or Boltzmann action selection policy.

In this paper, we integrate several ideas in quantum mechanics and DRL techniques to solve intelligent trajectory planning problem for cellular-connected UAV networks. The main contributions of this paper are summarized as follows.

- Different from the vast majority of existing literature, more practical G2A pathloss model based on one realization of local building distribution and directional antenna with fixed 3-dimensional (3D) radiation pattern are considered in this paper. Then, a cellular-connected UAV trajectory planning problem is formulated to minimize the weighted sum of flight time cost and the corresponding expected outage duration. Without prior knowledge of the wireless environment, the focused path planning problem is challenging to be tackled via conventional optimization techniques. Alternatively, the proposed optimization problem is mapped into Markov decision process (MDP) and solved by the proposed DRL solution with novel quantum-inspired experience replay (QiER).
- A novel QiER framework is coined to help the learning agent achieve better training performance, via a three-phase quantum-inspired process. Specifically, the quantum initialization phase allocates initial priority for the newly-recorded experiences, the quantum preparation phase generates the updated priority for the sampled transitions with the help of Grover iteration, and the quantum measurement phase outputs distribution of sampling probabilities to help accomplish the mini-batch training procedure.
- To demonstrate advantages offered by the proposed DRL-QiER solution, performance comparison against representative baselines is performed. Compared to DRL approach with standard experience relay (DRL-ER) [23] or prioritized ER (DRL-PER) [24], deep curriculum reinforcement learning (DCRL) method [25] and simultaneous navigation and radio mapping (SNARM) strategy [5], simulation results demonstrate that the proposed DRL-QiER solution can achieve more efficient and steady learning performance. Moreover, the proposed DRL-QiER does not include extra neural networks like SNARM approach, and requires much less hyper-parameter tuning like DCRL or DRL-PER method, which means that it is easier and more robust for implementation.

Although this paper and [5] both focus on designing a DRL-aided solution for intelligently navigating cellular-connected UAV, the main differences are: 1) detailed antenna gain model and pathloss model are provided in this paper, which makes the formulated UAV navigation problem more specific; 2) to overcome the bias issue and relieve the heavy computation burden induced by the extra neural network of SNARM approach [5], i.e., the model-learning component termed as radio map, a light but reliable DRL-QiER solution is proposed, which is model-free and contains only one online training neural network; and 3) quantum mechanism is introduced to aid experience replay efficiency for DRL agent, enabling the proposed DRL-QiER solution have the potential to perform outstandingly than conventional DRL methods. Moreover, with the help of Grover iteration in quantum computation, we extend the QiER method in [26] from 2-dimensional (2D) discrete rotation to its 3D continuous

alternative, which introduces fewer additional hyper-parameters and thus makes the QiER technique more flexible and reliable. Last but not least, compared to our prior work [22], we extend the quantum aid from enhancing action selection quality for RL framework to improving experience replay performance for DRL counterpart, breaking the curse of dimensionality and enabling the agent to practically solve problems with continuous state space.

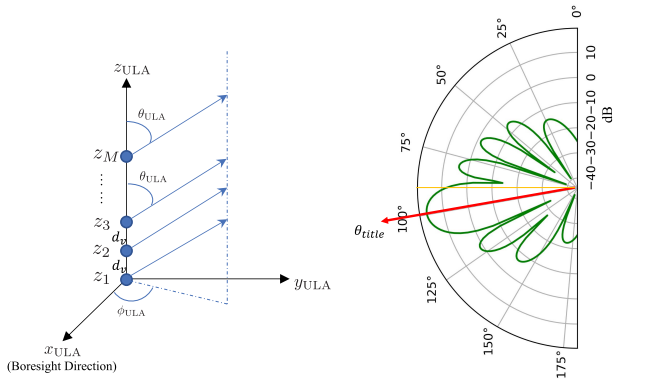
Organization: Section II presents the system model. Section III gives a brief overview on DRL. Section IV briefly introduces quantum state and quantum amplitude amplification. Section V presents the proposed DRL-QiER solution. Simulation results are given in Section VI, while conclusions are drawn in Section VII.

II. System Model

A downlink transmission scenario inside cellular-connected UAV network is considered, where a set $\mathcal{U}$ of U UAVs is served by a set $\mathcal{B}$ of B BSs within cellular coverage. These UAVs are supposed to reach a common destination from their respective initial locations, for accomplishing their own missions.[1] Intuitively, each UAV should be navigated with a feasible trajectory, alongside which the corresponding time consumption should be the shortest and wireless transmission quality provided by the cellular network should be maintained satisfactorily.[2] Without loss of generality, an arbitrary UAV (denoted as u hereafter) out of these U drones are concentrated for investigating the navigation task.[3] For clarity, the UAV's exploration environment is defined as a cubic subregion $\mathbb{A} : [x_{\text{lo}}, x_{\text{up}}] \times [y_{\text{lo}}, y_{\text{up}}] \times [z_{\text{lo}}, z_{\text{up}}]$, where the subscripts "lo" and "up" represent the lower and upper boundaries of this 3D airspace, respectively. Furthermore, the coordinate of the focused UAV at time t should locate in the range of $\vec{q}_{\text{lo}} \preceq \vec{q}_u(t) \preceq \vec{q}_{\text{up}}$, where $\vec{q}_{\text{lo}} = (x_{\text{lo}}, y_{\text{lo}}, z_{\text{lo}})$, $\vec{q}_{\text{up}} = (x_{\text{up}}, y_{\text{up}}, z_{\text{up}})$ and $\preceq$ denotes the element-wise inequality. The initial location and the destination are given by $\vec{q}_u(I) \in \mathbb{R}^{1*3}$ and $\vec{q}_u(D) \in \mathbb{R}^{1*3}$, respectively. Therefore, the overall trajectory of this UAV's flight can be fully traced by $\vec{q}_u(t) = (x_u(t), y_u(t), z_u(t))$, starting from $\vec{q}_u(I)$ and ending at $\vec{q}_u(D)$. Besides, the location of arbitrary BS $b \in \mathcal{B}$ is indicated as $\vec{q}_b = (x_b, y_b, z_b)$, where $\vec{q}_{\text{lo}} \preceq \vec{q}_b \preceq \vec{q}_{\text{up}}$.

A. Antenna Model

Terrestrial transmission usually assumes that the distance between transceivers is much greater than the height difference of their antennas. In this regard, antenna modelling for terrestrial communications mainly concerns 2D antenna gain on the horizontal domain. Unfortunately, 2D antenna modelling is not sufficiently feasible for G2A transmissions, where high-altitude UAVs are involved.



(a) Coordinate system of ULA (b) Vertical pattern at boresight

Fig. 1. Demonstration of ULA's coordinate system and vertical radiation pattern.

In compliance with BS's antenna modelling of current cellular networks, directional antenna with fixed 3D radiation pattern is assumed to be equipped at each BS. Following standard sectorization, each BS is portioned to cover three sectors. Therefore, there are $3B$ sectors in total within the interested airspace $\mathbb{A}$. Specifically, it is assumed that three vertically-placed M-element uniform linear arrays (ULAs) are equipped by each BS with boresights directed to their corresponding sectors covered by this BS, subject to the 3GPP specification on cellular BS's antenna model [27]. In individual and independent coordinate system of each ULA (e.g., Fig. 1(a)), antenna element's placing location is denoted as $(0, 0, z_m)$, where $m = \{1, 2, \ldots, M\}$.

Then, the wave factor of ULA can be given by

$$\vec{k} = \frac{2\pi}{\lambda} \left(\sin\theta_{\text{ULA}} \cos\phi_{\text{ULA}}, \sin\theta_{\text{ULA}} \sin\phi_{\text{ULA}}, \cos\theta_{\text{ULA}}\right), \quad (1)$$

where $\lambda = c/f_c$ represents the wavelength, c denotes the light speed and f_c indicates the carrier frequency. Furthermore, the steering vector can be derived as

$$\vec{sv} = \left[\exp(-j\vec{k}(0,0,z_1)^T), \ldots, \exp(-j\vec{k}(0,0,z_M)^T)\right]^T. \quad (2)$$

Suggested by 3GPP, vertical and horizontal radiation patterns in dB of each ULA are given by

$$A_V(\theta_{\text{ULA}}, \phi_{\text{ULA}} = 0°) = -\min\left\{12\left(\frac{\theta_{\text{ULA}} - 90°}{\Theta_{\text{3dB}}}\right)^2, 30\text{dB}\right\}, \quad (3)$$

$$A_H(\theta_{\text{ULA}} = 90°, \phi_{\text{ULA}}) = -\min\left\{12\left(\frac{\phi_{\text{ULA}}}{\Phi_{\text{3dB}}}\right)^2, 30\text{dB}\right\}, \quad (4)$$

respectively. Then, each ULA's 3D element pattern in dB can be calculated as

$$A(\theta_{\text{ULA}}, \phi_{\text{ULA}}) = -\min\left\{-\left[A_V(\theta_{\text{ULA}}, \phi_{\text{ULA}} = 0°) + A_H(\theta_{\text{ULA}} = 90°, \phi_{\text{ULA}})\right], 30\text{dB}\right\}. \quad (5)$$

Note that each antenna element of a ULA is directional, specified by half-power beamwidths Θ_{3dB} and Φ_{3dB} for the vertical and horizontal dimensions, respectively. To suppress ICIs in cellular networks, the main lobe of ULA's radiation pattern

[1]For example, one typical UAV application case is parcel collection. Various UAVs are launched from different costumers' properties carrying parcels to the local distribution centre of delivery firm. Besides, collision avoidance during UAVs' flights needs to be guaranteed, via separating UAV's operation spaces and keeping their flying altitudes higher than the tallest building.

[2]This paper concentrates on UAV navigation task within coverage of cellular networks, while global positioning system (GPS)-supported UAV navigation is beyond the scope of this paper and left as one of future research directions.

[3]These UAVs share the same airspace and common location-dependent database, which means that the trained DRL model can be downloaded by the remaining UAVs, helping them accomplish their navigation tasks.

should be electrically steered by $\theta_{etilt} \in [0°, 180°]$, where $\theta_{etilt} = 90°$ means perpendicular to the ULA. To achieve the steering angle θ_{etilt}, fixed phase shift for each antenna element of ULA can be executed, for which the complex coefficient of the m-th antenna element is given by

$$\omega_m = \frac{1}{M} \exp\left[-j\frac{2\pi}{\lambda}(m-1)d_v \cos\theta_{etilt}\right], \tag{6}$$

where d_v indicates the vertical elements' spacing distance. Furthermore, the array factor can be formulated as

$$AF = \sum_{m=1}^{M} \omega_m \exp(-j\vec{k}(0,0,z_m)^T) = \vec{\omega}\vec{sv}, \tag{7}$$

where $\vec{\omega} = (\omega_1, \ldots, \omega_M)^*$ is the weight vector and the superscript $*$ indicates the complex conjugate. In the end, the 3D antenna gain of each ULA in dB can be calculated as

$$G(\theta_{\text{ULA}}, \phi_{\text{ULA}}) = 10\lg\left(|\sqrt{10^{\frac{A(\theta_{\text{ULA}},\phi_{\text{ULA}})}{10}}}AF|^2\right). \tag{8}$$

Fig. 1(b) illustrates an example for $\theta_{etilt} = 100°$, under parameter setting $\Theta_{3\text{dB}} = \Phi_{3\text{dB}} = 65°$, $d_v = \lambda/2$ and $M = 8$. It is straightforward to observe that the main lobe is downtilted towards the ground for serving terrestrial communications, and the upper side lobes can be utilized to support G2A transmissions. Denote $i \in \{1, \ldots, 3B\}$ as the label of sectors in the considered region. Then, the transmit antenna gain from arbitrary sector to the UAV can be explicitly determined by UAV's location, denoted as $G^i[\vec{q}_u(t)] = G(\theta_{iu}, \phi_{iu})$, where θ_{iu} and ϕ_{iu} can be obtained via taking $\vec{q}_u(t)$, the location of ULA for sector i and the ULA's boresight direction into account.[4]

B. Pathloss Model

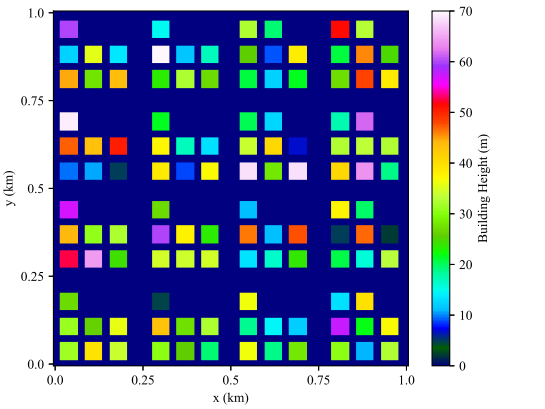
Different from terrestrial transmissions, G2A links are more likely to experience LoS pathloss. In this subsection, the adopted G2A channel model will be interpreted.

According to 3GPP urban-macro (UMa) pathloss model [28], the G2A pathloss in dB from sector i to the UAV at time t is given by

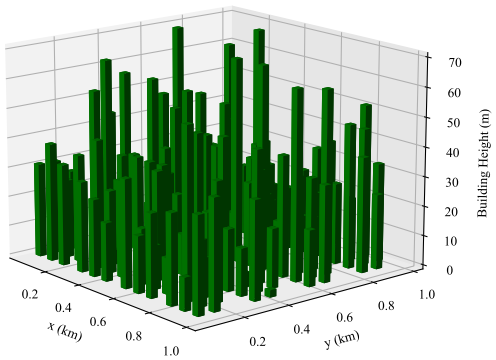
$$\text{PL}^i[\vec{q}_u(t)] = \begin{cases} 28.0 + 22\log_{10}(d_{iu}) + 20\log_{10}(f_c), & \text{LoS} \\ -17.5 + 20\log_{10}\left(\frac{40\pi f_c}{3}\right) \\ + [46 - 7\log_{10}(z_u(t))]\log_{10}(d_{iu}), & \text{NLoS}, \end{cases} \tag{9}$$

where $d_{iu} = ||\vec{q}_u(t) - \vec{q}_i||_2$ outputs the Euclidean distance between the UAV and the location of ULA for sector i.

To practically trace the type of G2A pathlosses, building distribution in the interested airspace $\mathbb{A}$ should be taken into consideration. Fig. 2 illustrates an example of local building distribution, including their horizontal locations on the ground and heights (Fig. 2(a)), as well as the corresponding 3D view



(a) Local building distribution



(b) 3D view of local building distribution

Fig. 2. The building distribution under consideration.

(Fig. 2(b)). With given building distribution, the type of large-scale pathloss of G2A channels for UAV at arbitrary location $\vec{q}_u(t)$, i.e., LoS or NLoS in (9), can be accurately determined via checking the potential blockages between the UAV and sectors.[5]

C. SINR at UAV

With the aforementioned antenna and pathloss models, the received signal of the focused UAV u at arbitrary location $\vec{q}_u$ over time t can be formulated as

$$y_u(t) = \sum_{i=1}^{3B} \sqrt{10^{\frac{G^i[\vec{q}_u(t)] - \text{PL}^i[\vec{q}_u(t)]}{10}}} h_{iu} x_i(t) + n_u(t), \tag{10}$$

where $x_i(t) \sim \mathcal{CN}(0, P_i)$ is the emitted message from sector i to the UAV with average transmit power P_i, h_{iu} represents the corresponding small-scale fading channel[6] and $n_u(t) \sim \mathcal{CN}(0, \sigma^2)$ denotes the received additive complex Gaussian noise (AWGN) at the UAV. Note that the explicit type of pathloss, i.e., LoS or NLoS, can be determined via checking possible blockages according to one realization of local building distribution as mentioned in Section II-B. Assume that the UAV is associated with sector $\hat{i}$ at time t, the instantaneous

[4]Note that the location of ULA for sector i is assumed to be the same as its associated BS, which is a reasonable consideration because the distance among ULAs on the BS is much smaller than that between the UAV and the BS.

[5]Note that our method generating G2A pathloss is more practical than the widely-used probabilistic G2A channel model in current literature because the later can only characterize the average G2A pathloss rather than its real counterpart.

[6]This paper aims to develop a UAV navigation method for arbitrary small-scale channel model. Hence, we do not specify the type of small-scale fading here, e.g., Rayleigh [29]–[33], Rician [34], [35] or Nakagami-m [36].

signal-to-interference-plus-noise ratio (SINR) at the UAV can be derived as

$$\Gamma_u(t) = \frac{P_{\hat{i}} 10^{\frac{G^{\hat{i}}[\vec{q}_u(t)] - \mathrm{PL}^{\hat{i}}[\vec{q}_u(t)]}{10}} |h_{\hat{i}u}|^2}{I_u(t) + \sigma^2}, \tag{11}$$

where $I_u(t) = \sum_{i \neq \hat{i}} P_i 10^{\frac{G^i[\vec{q}_u(t)] - \mathrm{PL}^i[\vec{q}_u(t)]}{10}} |h_{iu}|^2$ means the ICIs from un-associated sectors.[7]

D. Problem Formulation

The received SINR (11) is a random variable because of the randomness introduced by small-scale fadings, with given UAV coordinate $\vec{q}_u(t)$ and cell association $\hat{i}(t)$. Therefore, the corresponding transmission outage probability (TOP) can be formulated as a function of $\vec{q}_u(t)$ and $\hat{i}(t)$, i.e., $TOP_u\{\vec{q}_u(t), \hat{i}(t)\} = \Pr[\Gamma_u(t) < \Gamma_{th}]$, where Pr outputs the probability calculated with respect to (w.r.t.) the aforementioned small-scale fadings. Then, the ergodic outage duration (EOD) of the UAV u travelling with trajectory $\vec{q}_u(t), \forall t \in [0, T_u]$ from $\vec{q}_u(I)$ to $\vec{q}_u(D)$ can be expressed as

$$EOD_u\{\vec{q}_u(t), \hat{i}(t)\} = \int_0^{T_u} TOP_u\{\vec{q}_u(t), \hat{i}(t)\} dt. \tag{12}$$

According to (12), the UAV has more freedom to adjust its flying trajectory for visiting stronger wireless coverage areas (say, regions with lower TOP) if longer flight time budget T_u is achievable. However, T_u is commonly expected to be as short as possible, for the consideration of energy consumption and time cost for accomplishing the corresponding mission. Therefore, a dilemma of minimizing both T_u and EOD_u exists inevitably. To balance this, we focus on minimizing the weighted sum of T_u and $EOD_u\{\vec{q}_u(t), \hat{i}(t)\}$ via designing $\vec{q}_u(t)$ and $\hat{i}(t)$. Unfortunately, continuous time t implies infinite amount of velocity constraints and location possibilities, leading the UAV path planning task too sophisticated to be handled. Alternatively, the flight period T_u is uniformly divided into N time slots, making the navigation task practically trackable. The duration of each time slot $\Delta_t = T_u/N$ is controlled to be sufficiently small so that the distance, pathloss and antenna gain from each sector towards the UAV can be considered as approximately static within arbitrary time slot.[8] Besides, sector assignment is commonly dependent on pathloss to avoid non-stop handover in practice, and thus the associated sector within each time slot is assumed unchanged. Therefore, (12) can be approximated as $EOD_u\{\vec{q}_u(t), \hat{i}(t)\} \approx \sum_{n=1}^{N} \Delta_t TOP_u\{\vec{q}_u(n), \hat{i}(n)\}$. With given $\vec{q}_u(n)$ and $\hat{i}(n)$ for each time slot, $TOP_u\{\vec{q}_u(n), \hat{i}(n)\}$ can be obtained via numerical signal measurement at the UAV.[9] In this regard, we have

$$TOP_u\{\vec{q}_u(n), \hat{i}(n)\} \simeq \frac{1}{L} \sum_{\iota=1}^{L} ITOP\{\vec{q}_u(n), \hat{i}(n) | h(\iota)\}, \tag{13}$$

where $h(\iota)$ indicates one realization of the involved small-scale fading components, L represents the amount of signal measurements, the TOP indicator $ITOP\{\vec{q}_u(n), \hat{i}(n) | h(\iota)\} = 1$ if $\Gamma_u\{\vec{q}_u(n), \hat{i}(n) | h(\iota)\} < \Gamma_{th}$ and $ITOP\{\vec{q}_u(n), \hat{i}(n) | h(\iota)\} = 0$ otherwise. Note that $L \gg 1$ stands in practice, which means that the approximation (13) is feasible to be treated as an equation. Then, the corresponding optimization problem can be stated as

$$(\mathrm{P1}): \min_{\vec{v}_u(n)} \frac{\tau \Delta_t}{L} \sum_{n=1}^{N} \sum_{\iota=1}^{L} ITOP\{\vec{q}_u(n), \hat{i}(n) | h(\iota)\} + N, \tag{14a}$$

$$\text{s.t. } \hat{i}(n) = \arg\min_{i \in \{1,2,\ldots,3B\}} PL^i[\vec{q}_u(n)], \tag{14b}$$

$$\vec{q}(n+1) = \vec{q}(n) + V_u \Delta_t \vec{v}_u(n),$$
$$\|\vec{v}_u(n)\| = 1, \tag{14c}$$
$$\vec{q}_{\mathrm{lo}} \preceq \vec{q}_u(n) \preceq \vec{q}_{\mathrm{up}}, \quad \vec{q}_u(0) = \vec{q}_u(I),$$
$$\vec{q}_u(N) = \vec{q}_u(D), \tag{14d}$$

where τ is the weight balancing the aforementioned minimization dilemma, V_u represents the UAV's flying velocity and $\vec{v}_u(n)$ specifies the mobility direction. The constraint (14b) holds because the sector association strategy is dependent sorely on pathlosses from all the sectors within each time slot and it is clear that the UAV should always pair with the sector which can offer the least degree of pathloss.

It is straightforward to conclude that antenna gain and LoS/NLoS condition from each sector to the UAV are dependent on the UAV's location with given building and BS distribution, which further impacts the corresponding pathloss and type of small-scale fading. This makes it extremely sophisticated to solve problem (P1) via standard optimization methods, if not impossible. To provide a better alternative solving the proposed optimization problem (P1), a DRL-aided solution with a novel QiER framework is proposed in this paper.

III. Deep Reinforcement Learning

This section is established to give a brief introduction to DRL basics, which is of importance for understanding the proposed DRL-aided solution and the corresponding key notations.

The training of RL agent is based on MDP consisting of five components, listed in a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma)$. A state $s_t \in \mathcal{S}$ denotes RL agent's observation from the environment at trial t. An action $a_t \in \mathcal{A}$ represents the agent's choice

[7]This paper focuses on the worst case where universal frequency reuse is assumed, which means that all the non-associated co-channel sectors will be taken into account as the sources of ICIs.

[8]In the case of $\Delta_t \to 0$, the discrete flight trajectory can accurately approach its continuous counterpart, resulting in extremely heavy computation burden. Therefore, the length of time slot Δ_t should be delicately chosen to achieve satisfactory balance of approximation accuracy and computational complexity.

[9]The closed-form expression of $TOP_u\{\vec{q}_u(n), \hat{i}(n)\}$ cannot be derived because this paper aims to develop a UAV navigation framework for arbitrary small-scale fading environment and the modelling of $h_{iu}, i \in \{1, 2, \ldots, 3B\}$ is not specified. Besides, Δ_t (typically on the magnitude of second) is relatively greater than the length of channel coherence block (on the magnitude of millisecond) caused by the small-scale fading. Therefore, $TOP_u\{\vec{q}_u(n), \hat{i}(n)\}$ can be practically evaluated by numerical measurements on the raw received signals at the UAV.

at trial t following an action selection policy $\pi(s_t, a_t)$. The policy $\pi(s_t, a_t) : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$ claims the probability distribution of picking action a_t for state s_t, constrained by $\sum_{a_t \in \mathcal{A}} \pi(s_t, a_t) = 1$. After executing an action, state transition function $\mathcal{T} = \Pr(s_{t+1}|s_t, a_t) : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ characterizes state transition $s_t \rightarrow s_{t+1}$. An immediate reward $r_t(s_t, a_t)$ acts as performance metric determining how good the selected action a_t is, for state s_t. A scalar factor $\gamma \in [0, 1]$ is invoked to discount future rewards, which can help reduce variance caused by the reward function and achieve the convergence of RL algorithms.

While interacting with the environment, the RL agent chooses an action a_t for observed state s_t at trial t following current action selection policy $\pi(s_t, a_t)$. After executing the selected action, state transition $s_t \rightarrow s_{t+1}$ occurs and a scalar reward $r_t(s_t, a_t)$ will be generated. Then, the experience $exp_t = \{s_t, a_t, r_t, s_{t+1}\}$ can be collected to train the RL agent. The state-action value function $Q_\pi(s_t, a_t)$ (i.e., Q function) derives the accumulated-rewards and reflects the long-term return of acting a_t over s_t following action selection policy π, given by

$$Q_\pi(s_t, a_t) = \mathbb{E}_\pi \left[G_t = \sum_{n_t=0}^{+\infty} \gamma^{n_t} r_{t+n_t} | s_t = s, a_t = a \right], \quad (15)$$

where G_t calculates the discounted accumulated-rewards. The state-action value function $Q_\pi(s_t, a_t)$ satisfies the Bellman equation, shown as

$$Q_\pi(s_t, a_t) = \mathbb{E}_\pi \left[r_t + \gamma \sum_{s_{t+1} \in \mathcal{S}} \mathcal{T}(s_{t+1}|s_t, a_t) \times \sum_{a_{t+1} \in \mathcal{A}} \pi(s_{t+1}, a_{t+1}) Q_\pi(s_{t+1}, a_{t+1}) \right]. \quad (16)$$

An important goal of RL agent is to find the optimal Q function which follows Bellman optimality equation [37], shown as

$$Q^*(s_t, a_t) = r_t + \gamma \sum_{s_{t+1} \in \mathcal{S}} \mathcal{T}(s_{t+1}|s_t, a_t) \max_{a_{t+1} \in \mathcal{A}} Q^*(s_{t+1}, a_{t+1}). \quad (17)$$

Unfortunately, (17) is non-linear and admits no closed-form solution, which can alternatively be solved through iterative algorithms [38]. Specifically, (17) can be deduced recursively to achieve the optimality $Q^*(s_t, a_t)$ via temporal difference (TD) learning, given by.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha_{lr} \left[r_t + \gamma \max_{a_{t+1} \in \mathcal{A}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t) \right], \quad (18)$$

where $\delta_t = r_t + \gamma \max_{a_{t+1} \in \mathcal{A}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)$ represents the TD error and $\alpha_{lr} \in (0, 1]$ denotes the learning rate. It is well-known that the optimum $Q^*(s_t, a_t)$ can be achieved when the state-action pairs are sufficiently experienced and the learning rate is properly chosen [37].

To crack the nut rooted from high-dimensional state and/or action spaces, instead of applying Q-table (e.g., tabular RL) to store $Q(s_t, a_t)$ for state-action pairs, function approximation technique is used to approximate the Q function, e.g., artificial neural networks (ANNs). Adopting DNN [39] (i.e., ANN with deep hidden layers) to approximate Q function, e.g., deep Q network (DQN) [23], is a popular and practical solution, i.e., $Q(s_t, a_t) \approx Q(s_t, a_t|\boldsymbol{\theta})$, where parameter vector $\boldsymbol{\theta}$ corresponds to the weight coefficients and biases of all links in the DNN. The parameter vector $\boldsymbol{\theta}$ can be updated via bootstrapping method to minimize the loss function, defined as

$$\mathcal{L}(\boldsymbol{\theta}) = \left[r_t + \gamma \max_{a_{t+1} \in \mathcal{A}} Q(s_{t+1}, a_{t+1}|\boldsymbol{\theta}^-) - Q(s_t, a_t|\boldsymbol{\theta}) \right]^2, \quad (19)$$

where $Q(s_t, a_t|\boldsymbol{\theta}^-)$ with parameter vector $\boldsymbol{\theta}^-$ indicates the target network, which is applied to enhance the stability of learning process.

Standard DRL algorithms apply one-step information to calculate the loss function (19) and train the online network, which may not be adequate and thus lead to poor predictability. To overcome the aforementioned shortcoming, multi-step learning strategy [37] was proposed via taking N_{ms}-step-forward knowledge into account. Specifically, the N_{ms}-step discounted accumulated-reward from a given state s_t can be rewritten as $r_{t:t+N_{ms}} = \sum_{n_{ms}=0}^{N_{ms}-1} \gamma^{n_{ms}} r_{t+n_{ms}+1}$. Based on (19), the loss function for N_{ms}-step learning can be derived as

$$\mathcal{L}(\boldsymbol{\theta}) = \left[r_{t:t+N_{ms}} + \gamma^{N_{ms}} \max_{a' \in \mathcal{A}} Q(s_{t+N_{ms}}, a'|\boldsymbol{\theta}^-) - Q(s_t, a_t|\boldsymbol{\theta}) \right]^2. \quad (20)$$

IV. Quantum State and Quantum Amplitude Amplification

In this section, several basic concepts in quantum computation are briefly introduced, which is of essence to help understand the proposed QiER framework.

A. Quantum State

In quantum mechanics, a quantum state of a closed quantum system can be described by a unit vector in Hilbert space. Specifically, a quantum state $|\Psi_c\rangle$ (Dirac notation) comprised of $\hat{n}$ quantum bits (qubits[10]) can be expressed as

$$|\Psi_c\rangle = |\Psi_1\rangle \otimes |\Psi_2\rangle \otimes \cdots \otimes |\Psi_{\hat{n}}\rangle = \sum_{p=00\ldots0}^{\overbrace{11\ldots1}^{\hat{n}}} h_p |p\rangle, \quad (21)$$

where $|\Psi_e\rangle, e \in [1, \hat{n}]$ represents the e-th qubit, h_p means the complex coefficient (i.e., probability amplitude) of eigenstate $|p\rangle$ subject to $\sum_{p=00\ldots0}^{11\ldots1} |h_p|^2 = 1$ and $\otimes$ denotes the tensor product. The representation of $\hat{n}$-qubit quantum state $|\Psi_c\rangle$ follows the quantum phenomenon known as *state superposition principle*. That is, the $|\Psi_c\rangle$ can be regarded as

[10] A qubit can be realized by a two-state system, e.g., 1) a two-level atom, in which $|0\rangle$ denotes the ground state and $|1\rangle$ indicates the excited state; 2) a photon, where $|0\rangle$ represents the horizontal polarization state and $|1\rangle$ means the vertical polarization state; or 3) a spin system, in which the states of spin up and spin down are described by $|0\rangle$ and $|1\rangle$, respectively.

the superposition of $2^{\hat{n}}$ eigenstates, ranged from $|00\ldots0\rangle$ to $|11\ldots1\rangle$. As a special case, a two-eigenstate quantum system (say, a single qubit) can be described as an arbitrary superposition state of eigenstates $|0\rangle$ and $|1\rangle$, given by

$$|\Psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \tag{22}$$

where the complex coefficients $\alpha = \langle 0 \mid \Psi\rangle$ and $\beta = \langle 1 \mid \Psi\rangle$ denote the probability amplitudes for eigenstates $|0\rangle$ and $|1\rangle$, respectively. Note that the single-qubit superposition $|\Psi\rangle$ is a unit vector (i.e., $\langle\Psi \mid \Psi\rangle = 1$) in Hilbert space spanned by orthogonal bases $|0\rangle$ and $|1\rangle$, subject to $|\alpha|^2 + |\beta|^2 = 1$. According to *quantum collapse phenomenon*, after measurement or observation of an external experimenter, $|\Psi\rangle$ will collapse from its superposition state onto one of its eigenstates $|0\rangle$ and $|1\rangle$ with probabilities $|\alpha|^2$ and $|\beta|^2$, respectively.

B. Quantum Amplitude Amplification

For a two-eigenstate qubit $|\Psi\rangle$, the probability amplitudes of each eigenstate can be changed via a quantum operation (e.g., Grover iteration [40]), gradually modifying the collapse probability distribution. Two unitary reflections are applied to achieve Grover iteration, given by

$$\boldsymbol{U}_{|0\rangle} = \boldsymbol{I} - (1 - e^{j\phi_1})\,|0\rangle\,\langle 0|\,, \tag{23}$$

$$\boldsymbol{U}_{|\Psi\rangle} = (1 - e^{j\phi_2})\,|\Psi\rangle\,\langle\Psi| - \boldsymbol{I}, \tag{24}$$

where $\{\phi_1, \phi_2\} \in [0, 2\pi]$, $\boldsymbol{I}$ indicates identity matrix, and $\langle 0|$ and $\langle\Psi|$ are Hermitian transposes of $|0\rangle$ and $|\Psi\rangle$, respectively. Then, the Grover iterator can be formulated as $\boldsymbol{G} = \boldsymbol{U}_{|\Psi\rangle}\boldsymbol{U}_{|0\rangle}$, which remains unitary. After m times of acting $\boldsymbol{G}$ on $|\Psi\rangle$, the two-eigenstate qubit with updated probability amplitudes can be given by $|\Psi\rangle \leftarrow \boldsymbol{G}^m\,|\Psi\rangle$. Two updating approaches can be used to accomplish quantum amplitude amplification task: 1) $m = 1$ with dynamic parameters ϕ_1 and ϕ_2; and 2) dynamic m with fixed parameters ϕ_1 and ϕ_2 (e.g., π). The latter updating method can only change the probability amplitudes in a discrete manner, and thus the former solution is chosen in this paper.

Proposition 1: For Grover iteration with flexible parameters, the overall effects of $\boldsymbol{G}$ on the superposition $|\Psi\rangle$ can be derived analytically as $\boldsymbol{G}\,|\Psi\rangle = (\mathcal{Q} - e^{j\phi_1})\alpha\,|0\rangle + (\mathcal{Q} - 1)\beta\,|1\rangle$, where $\mathcal{Q} = (1 - e^{j\phi_2})\left[1 - (1 - e^{j\phi_1})|\alpha|^2\right]$ and $|(\mathcal{Q} - e^{j\phi_1})|^2|\alpha|^2 + |(\mathcal{Q} - 1)|^2|\beta|^2 = 1$.

Proof: The effects of $\boldsymbol{U}_{|0\rangle}$ on $|0\rangle$ and $|1\rangle$ are expressed as

$$\boldsymbol{U}_{|0\rangle}\,|0\rangle = \left[\boldsymbol{I} - (1 - e^{j\phi_1})\,|0\rangle\,\langle 0|\right]|0\rangle = e^{j\phi_1}\,|0\rangle\,, \tag{25}$$

$$\boldsymbol{U}_{|0\rangle}\,|1\rangle = \left[\boldsymbol{I} - (1 - e^{j\phi_1})\,|0\rangle\,\langle 0|\right]|1\rangle = |1\rangle\,, \tag{26}$$

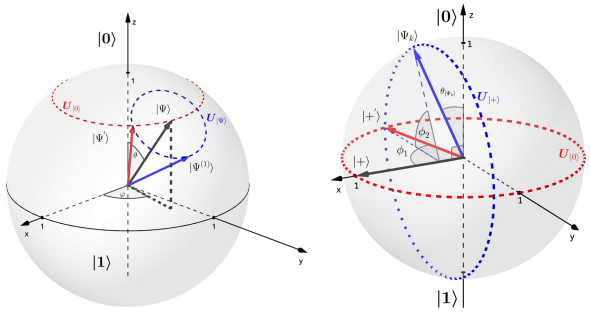
respectively. Then, we obtain

$$\boldsymbol{U}_{|0\rangle}\,|\Psi\rangle = \left[\boldsymbol{I} - (1 - e^{j\phi_1})\,|0\rangle\,\langle 0|\right]|\Psi\rangle = e^{j\phi_1}\alpha\,|0\rangle + \beta\,|1\rangle\,, \tag{27}$$

where $\boldsymbol{U}_{|0\rangle}$ plays the role as a *conditional phase shift operator*.

Furthermore, we get

$$\begin{aligned}\boldsymbol{G}\,|\Psi\rangle &= \boldsymbol{U}_{|\Psi\rangle}\boldsymbol{U}_{|0\rangle}\,|\Psi\rangle = (1 - e^{j\phi_2})\left[\alpha\,|0\rangle + \beta\,|1\rangle\right] \\ &\quad \times \left[\alpha^{\dagger}\,\langle 0| + \beta^{\dagger}\,\langle 1|\right]\boldsymbol{U}_{|0\rangle}\,|\Psi\rangle - \boldsymbol{U}_{|0\rangle}\,|\Psi\rangle \\ &= (\mathcal{Q} - e^{j\phi_1})\alpha\,|0\rangle + (\mathcal{Q} - 1)\beta\,|1\rangle\,,\end{aligned} \tag{28}$$



(a) Grover rotation on $|\Psi\rangle$ (b) Grover rotation on $|+\rangle$

Fig. 3. Geometric explanation of the grover rotation.

where $\mathcal{Q} = (1 - e^{j\phi_2})(e^{j\phi_1}|\alpha|^2 + |\beta|^2) = (1 - e^{j\phi_2})\left[1 - (1 - e^{j\phi_1})|\alpha|^2\right]$.

Because Grover operator $\boldsymbol{G}$ is unitary, the updated superposition $|\Psi\rangle \leftarrow \boldsymbol{G}\,|\Psi\rangle$ still follows the normalization rule of probability amplitudes, i.e., $|(\mathcal{Q} - e^{j\phi_1})|^2|\alpha|^2 + |(\mathcal{Q} - 1)|^2|\beta|^2 = 1$. ■

Corollary 1: The ratio between collapse probabilities of $|\Psi\rangle \to |0\rangle$ before and after being impacted by G can be given by $|\mathcal{R}|^2 = |(1 - e^{j\phi_1} - e^{j\phi_2}) - (1 - e^{j\phi_1})(1 - e^{j\phi_2})|\alpha|^2|^2$, which is symmetric w.r.t. $\phi_1 = \phi_2$ and $\phi_1 = 2\pi - \phi_2$. Then, the updated collapse probabilities onto eigenstates $|0\rangle$ and $|1\rangle$ can be given by $|\mathcal{R}|^2|\alpha|^2$ and $1 - |\mathcal{R}|^2|\alpha|^2$, respectively.

Proof: Based on (22) and (28), the ratio between the probability amplitudes of $|0\rangle$ after being acted by $\boldsymbol{G}$ and before that can be derived as $\mathcal{R} = (1 - e^{j\phi_1} - e^{j\phi_2}) - (1 - e^{j\phi_1})(1 - e^{j\phi_2})|\alpha|^2$, which completes the proof. ■

Remark 1: The process of $|\Psi\rangle \leftarrow \boldsymbol{G}\,|\Psi\rangle$ can be depicted geometrically on the Bloch sphere. In Fig. 3(a), $|\Psi\rangle$ is reconstructed in Polar coordinates, given by

$$\begin{aligned}|\Psi\rangle = e^{j\zeta}\left(\cos\frac{\theta}{2}\,|0\rangle + e^{j\varphi}\sin\frac{\theta}{2}\,|1\rangle\right) \simeq \cos\frac{\theta}{2}\,|0\rangle \\ + e^{j\varphi}\sin\frac{\theta}{2}\,|1\rangle\,,\end{aligned} \tag{29}$$

where $e^{j\zeta}$ poses no observable effects [21]. Then, the unit vector $|\Psi\rangle$ on the Bloch sphere is uniquely specified by angle variables $\theta \in [0, \pi]$ and $\varphi \in [0, 2\pi)$. The effect of $\boldsymbol{U}_{|0\rangle}$ can be regarded as a clockwise rotation around the z-axis by ϕ_1 (the red circle) on the Bloch sphere, leading to the rotation from $|\Psi\rangle$ to $|\Psi'\rangle$. In a similar manner, when the basis is changed from $\{|0\rangle, |1\rangle\}$ to $\{|\Psi\rangle, |\Psi^{\perp}\rangle\}$, $\boldsymbol{U}_{|\Psi\rangle}$ results in a clockwise rotation around the new z-axis $|\Psi\rangle$ by ϕ_2 (the blue circle), rotating $|\Psi'\rangle$ to $|\Psi^{(1)}\rangle$. Hence, the overall impact of $\boldsymbol{G}$ on $|\Psi\rangle$ is a two-step process rotating the polar angle θ, on the perspective of basis $\{|0\rangle, |1\rangle\}$. With flexible ϕ_1 and ϕ_2, it is possible to achieve arbitrary parametric rotation on the Bloch sphere, which serves as the foundation for quantum amplitude amplification task. The smaller θ is, the higher probability $|\Psi\rangle$ will collapse onto $|0\rangle$ when it is observed by an external examiner, and vice versa.

V. DRL-QiER Algorithm

In this section, a DRL-QiER solution is developed to solve optimization problem (P1).

A. The MDP Formulation

To solve the optimal trajectory planning problem (P1) via DRL-aided technique, the first step is to map it into an MDP, which can be described as follows.

- $\mathcal{S}$: The state space consists of possible UAV locations $\vec{q}_u$ under constraint $\vec{q}_{\text{lo}} \preceq \vec{q}_u \preceq \vec{q}_{\text{up}}$, which means that the state space is continuous.
- $\mathcal{A}$: The continuous action space involves all the feasible flying directions $\vec{v}_u$ under constraint $\|\vec{v}_u\| = 1$. To break the curse of dimensionality caused by continuous state and action spaces, the action space is discretized as $\mathcal{A} = \{[1,0,0],[0,1,0],[-1,0,0],[0,-1,0],[\sqrt{2}/2,\sqrt{2}/2,0],[-\sqrt{2}/2,\sqrt{2}/2,0],[\sqrt{2}/2,-\sqrt{2}/2,0],[-\sqrt{2}/2,-\sqrt{2}/2,0]\}$, corresponding to flying directions right, forward, left, backward, right-forward, left-forward, right-backward and left-backward, respectively. Therefore, the action space contains $N_{fd} = 8$ direction options.
- $\mathcal{T}$: The state transition is deterministic and controlled by the mobility constraint (14c).
- r: Our goal is to minimize the weighted sum of time cost and EOD. Thus, we may design the reward function as $r(\vec{q}_u) = -1 - \frac{\tau\Delta_t}{L}\sum_{\iota=1}^{L} ITOP\{\vec{q}_u|h(\iota)\}$. The formulation of $r(\vec{q}_u)$ can be interpreted as follows: 1) for each time of state transition, the agent will receive a movement penality 1, encouraging the UAV to use less steps to generate the trajectory; and 2) on top of the movement penality, the UAV will get a weighted outage duration penality $\frac{\tau\Delta_t}{L}\sum_{\iota=1}^{L} ITOP\{\vec{q}_u|h(\iota)\}$ as well, pushing the UAV to visit locations with stronger wireless coverage quality. Besides, two special cases are considered as follows: 1) once the UAV reaches the predefined destination $\vec{q}_u(D)$, the training episode terminates and a positive value r_D will replace the reward function; and 2) once the UAV crashes onto the boundary of the considered airspace, the training episode terminates and a negative value r_{ob} will replace the reward function instead. In summary, the aforementioned design of reward function aims to encourage the UAV to reach $\vec{q}_u(D)$ with as fewer steps as possible, while avoiding hitting the boundary and visiting areas with weak wireless coverage strength.
- γ: To connect the objective function of (P1) and the discounted accumulated-rewards over each learning episode, the discount factor is chosen as $\gamma = 1$.

B. Quantum-Inspired Representation of Experience's Priority

In the proposed DRL-QiER solution, the priority of experienced transition exp_t is represented by the k-th qubit, where the scalar index k indicates this transition's location index in the QiER buffer. Specifically, the quantum representation of stored transition's priority can be given by

$$|\Psi_k\rangle = \alpha_k |0\rangle + \beta_k |1\rangle, \tag{30}$$

where the complex-valued probability amplitudes α_k and β_k follow the normalization constraint $|\alpha_k|^2 + |\beta_k|^2 = 1$. It is worth noting that the eigenstates $|0\rangle$ and $|1\rangle$ in (30) mean accepting and denying this transition, respectively. After quantum measurement, the superposition $|\Psi_k\rangle$ will collapse onto eigenstate $|0\rangle$ with probability $|\langle 0 \mid \Psi_k\rangle|^2 = |\alpha_k|^2$ or eigenstate $|1\rangle$ with probability $|\langle 1 \mid \Psi_k\rangle|^2 = |\beta_k|^2$. The complex coefficients α_k and β_k are of importance and essence in the QiER system, influencing the occurrence probability of accepting or denying the corresponding transition when $|\Psi_k\rangle$ is observed. The quantum representation $|\Psi_k\rangle$ establishes a bridge between quantum eigenstates and accepting or denying particular transition, which allows us to apply quantum amplitude amplification to realize manipulation of quantum collapse.

C. QiER Framework

The proposed QiER framework consists of the following three phases.

1) Quantum Initialization Phase: When transition exp_t is stored into the QiER buffer with finite capacity C, a label $k \in \{1,\ldots,C\}$ will be assigned to exp_t, which specifies the location of exp_t being recorded within the QiER buffer.[11] Then, experience exp_t and the k-th qubit $|\Psi_k\rangle$ together will be stored into the QiER buffer, which can be regarded as a collection of $(exp_t, |\Psi_k\rangle)$. When a new transition is recorded into the QiER buffer and before being sampled out to feed the training agent, its associated qubit $|\Psi_k\rangle$ should be initialized as eigenstate $|0\rangle$, i.e., $|\Psi_k\rangle \leftarrow |0\rangle$. The reason is that the agent has never been trained with these un-sampled transitions that may have unimaginable potentials to help the agent learn the characteristics of environment with which the agent is interacting. Thus, we set these newly-recorded transitions with the highest priority, encouraging the agent to more likely learn from them.

2) Quantum Preparation Phase: After an experience is sampled from the QiER buffer to train the agent, the quantum preparation phase should be performed on its associated qubit, updating the corresponding priority. This is due to two reasons: 1) the TD error of this transition is updated; and 2) the experience becomes older for the agent.

The uniform quantum state is defined as

$$|+\rangle = \frac{\sqrt{2}}{2}\left(|0\rangle + |1\rangle\right), \tag{31}$$

which can be understood as a unit vector on the x-axis of Bloch sphere (Fig. 3(a)) with $\theta = \pi/2$ and $\varphi = 0$. The absolute value of TD error $|\delta_t|$ is chosen to reflect priority of the corresponding transition exp_t. Once a recorded transition is sampled, its associated qubit $|\Psi_k\rangle$ should first be reset to the uniform quantum state, i.e., $|\Psi_k\rangle \leftarrow |+\rangle$. Then, to map the updated priority of exp_t into $|\Psi_k\rangle$, one time of Grover iteration with flexible parameters will be applied on the

[11]The QiER buffer is designed to be with fixed-size capacity in line with standard ER technique of DRL, which means that the first stored experience will be popped out first to create space for recording the new-coming transition when the QiER buffer is fully exploited. Therefore, each recorded experience is supposed to remain in the buffer for a fixed time.

uniform quantum state, shown as

$$|\Psi_k\rangle = \boldsymbol{U}_{|+\rangle}\boldsymbol{U}_{|0\rangle}|+\rangle \overset{(a)}{=} (\mathcal{P} - e^{j\phi_1})\frac{\sqrt{2}}{2}|0\rangle + (\mathcal{P}-1)\frac{\sqrt{2}}{2}|1\rangle, \quad (32)$$

where $\mathcal{P} = (1-e^{j\phi_2})\left[1-0.5(1-e^{j\phi_1})\right]$ and the derivation (a) is based on ***Proposition 1***. According to ***Remark 1***, the transformation from $|+\rangle$ to $|\Psi_k\rangle$ can be depicted on the Bloch sphere as Fig. 3(b). In this example, the phase shift parameters are set as $\phi_1 < \pi/2$ and $\phi_2 < \pi/2$. It is straightforward to observe that the probability of collapsing onto eigenstate $|0\rangle$ enlarges after the quantum preparation phase (i.e., $|+\rangle \overset{\boldsymbol{U}_{|+\rangle}\boldsymbol{U}_{|0\rangle}}{\longrightarrow} |\Psi_k\rangle$), because the polar angle rotates from $\angle 90^\circ$ (of $|+\rangle$) to an acute angle θ_{Ψ_k} (of $|\Psi_k\rangle$). Similarly, the collapse probability onto eigenstate $|0\rangle$ after one time of Grover iteration on $|+\rangle$ can be kept unchanged or shrinked via selecting feasible combination of phase shift parameters $\phi_1 \in [0, 2\pi]$ and $\phi_2 \in [0, 2\pi]$.

In practical applications, some experiences may be sampled for training with undesired high frequency, leading to over-training issue. Besides, the finite size of QiER buffer could further deteriorate this disservice [41], which will cause unfair and biased sampling performance. To circumvent this issue, the replay time of each stored transition should be taken into consideration for the quantum preparation phase, which enables it to enrich sample diversity to improve the learning performance. In the early stage of training the agent, the importance of each experience is ambiguous. However, alongside the learning process, the absolute TD errors of some transitions remain relatively large, despite many times they have been sampled for training. Hence, it is necessary to relate training episode to the quantum preparation phase.

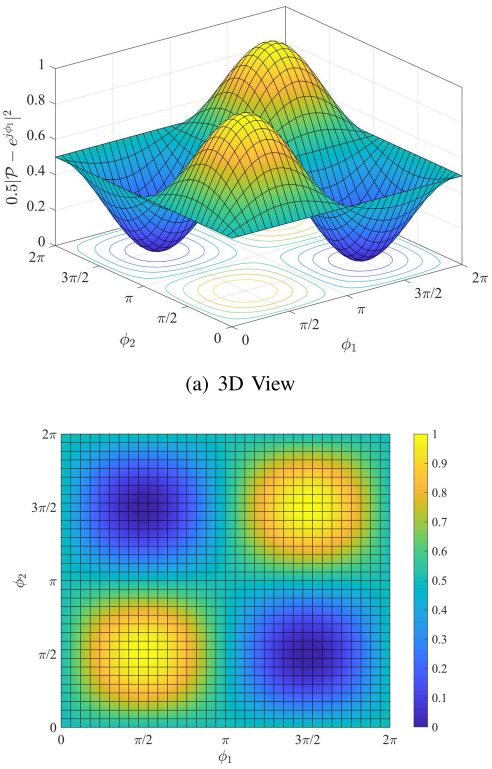
The quantum preparation phase aims to modify the collapse probability onto eigenstate $|0\rangle$, via one time of Grover iteration with free parameters ϕ_1 and ϕ_2. To quantify the amplification step of quantum preparation phase, we let

$$\phi_1 = \frac{e^{\frac{|\delta_t|\pi}{\delta_{\max}}} - e^{-\frac{|\delta_t|\pi}{\delta_{\max}}}}{e^{\frac{|\delta_t|\pi}{\delta_{\max}}} + e^{-\frac{|\delta_t|\pi}{\delta_{\max}}}}\frac{\pi}{2} = \frac{\pi}{2} tanh\left(\frac{|\delta_t|\pi}{\delta_{\max}}\right) \in \left[0, \frac{\pi}{2}\right), \quad (33)$$

$$\phi_2 = \frac{rt_k}{rt_{\max}}\frac{te}{te_{\max}}\pi + \frac{\pi}{2} \in \left(\frac{\pi}{2}, \frac{3\pi}{2}\right]. \quad (34)$$

With (33) and (34), the quantum amplitude amplification is related with the corresponding absolute TD error $|\delta_t|$, maximum TD error $\delta_{\max}$, replay times rt_k, maximum replay time $rt_{\max}$, current training episode te and the total training episode $te_{\max}$, which means that the quantum preparation phase updates the priority of exp_t into its associated k-th qubit $|\Psi_k\rangle$.

Remark 2: The collapse probability of $|\Psi_k\rangle$ onto eigenstate $|0\rangle$ versus $\phi_1 \in [0, 2\pi]$ and $\phi_2 \in [0, 2\pi]$ is depicted in Fig. 4. From this figure, we can find that $|\langle 0 \mid \Psi_k\rangle|^2 = 0.5|\mathcal{P} - e^{j\phi_1}|^2$ is a symmetric function w.r.t. $\phi_1 = \phi_2$ and $\phi_1 = 2\pi - \phi_2$, which is a specific case (i.e., $|\alpha|^2 = 0.5$) of ***Corollary 1***. If we concentrate on surface within $\phi_1 \in [0, \pi/2]$ and $\phi_2 \in [\pi/2, 3\pi/2]$, it is straightforward to conclude that (33) and (34) together can control the quantum amplification step and direction. Specifically, larger ϕ_1 will lead to greater amplitude amplification step, for arbitrary fixed ϕ_2. Besides, ϕ_2 controls the amplification direction, where $\phi_2 \in [\pi/2, \pi)$ means that the probability of collapsing onto $|0\rangle$ will be enlarged, while $\phi_2 \in (\pi, 3\pi/2)$ indicates that the probability of collapsing onto $|0\rangle$ will be reduced.



(a) 3D View

(b) The corresponding top view

Fig. 4. Collapse probability onto $|0\rangle$ versus ϕ_1 and ϕ_2.

Remark 3: In the early stage of training, the radio $rt_k/rt_{\max}$ remains relatively large because $rt_{\max}$ is not sufficiently updated yet. To avoid unreasonably denying all the sampled transitions in the early stage of training, we introduce the factor $te/te_{\max}$ to steer parameter ϕ_2 in (34).

3) Quantum Measurement Phase: After the QiER buffer is fully occupied by recorded transitions, a mini-batch of experiences will be sampled to perform network training for the agent, via standard gradient descent method. To prepare the mini-batch sampling procedure under constraint of priorities, quantum measurement on the associated qubits should be accomplished first. Specifically, the probability of the k-th qubit collapsing onto eigenstate $|0\rangle$ can be calculated as $|\langle 0 \mid \Psi_k\rangle|^2$. Then, the probability of the corresponding experience being picked up during the mini-batch sampling process can be defined as $bp_k = |\langle 0 \mid \Psi_k\rangle|^2 / \sum_{e=1}^{C} |\langle 0 \mid \Psi_e\rangle|^2$, in which the denominator means the sum of collapse probabilities onto eigenstate $|0\rangle$ of qubits that are associated with all stored experiences.

During the mini-batch sampling period, several times of picking recorded experiences from the QiER buffer will be executed, following the generated picking probability vector $\vec{bp} = [bp_1, bp_2, \ldots, bp_C]$ after quantum measurement phase.

Algorithm 1 The Proposed DRL-QiER Solution

Initialization: Initialize the online D3QN network $Q_{D3}(s,a|\boldsymbol{\theta}_{D3})$ and its target network $Q_{D3}(s,a|\boldsymbol{\theta}^{-}_{D3})$, with $\boldsymbol{\theta}^{-}_{D3} \leftarrow \boldsymbol{\theta}_{D3}$. Initialize the QiER buffer R with capacity C. Initialize the vector of replay time as $\vec{rt} = [rt_1, rt_2, \ldots, rt_C] = \vec{0}$. Set the size of mini-batch as N_{mb}. Set the order index of R as $k = 1$. Set the flag indicating whether the QiER buffer is fully occupied or not as $LF = False$. Set the maximum TD error as $\delta_{\max} = 1$;
for $te = [1, te_{\max}]$ **do**
  Set time step $n = 0$. Randomly set the the UAV's initial location as $\vec{q}_u(n) \in \mathcal{S}$. Initialize a sliding buffer $\hat{\text{R}}$ with capacity N_{ms};
  repeat
    Select and execute action a_n, then observe the next state $\vec{q}_u(n+1)$ and the immediate reward $r_n = r[\vec{q}_u(n+1)]$;
    if $LF == True$ **then**
      Perform quantum measurement on all stored experiences' qubits and get the vector of their replaying probabilities $[bp_1, bp_2, \ldots, bp_C]$;
      for $n_{mb} = [1, N_{mb}]$ **do**
        Sample a transition according to $[bp_1, bp_2, \ldots, bp_C]$ and get its location index $d \in \{1, 2, \ldots, C\}$;
        Reset the d-th qubit back to uniform quantum state $|\Psi_d\rangle = |+\rangle$;
        Update the corresponding replay time $rt_d += 1$ and $rt_{\max} = \max(\vec{rt})$;
        Calculate the sampled transition's absolute N_{ms}-step TD error $|\delta_{N_{ms}}|$ and update the maximum TD error $\delta_{\max} = \max(\delta_{\max}, |\delta_{N_{ms}}|)$;
        Perform quantum preparation phase on the d-th qubit;
      end
      Update the online D3QN network $Q_{D3}(s,a|\boldsymbol{\theta}_{D3})$ via gradient descent method using the mini-batch of sampled N_{mb} transitions from R;
    end
    Get and record transition $exp_n = \{\vec{q}_u(n), a_n, r_n, \vec{q}_u(n+1)\}$ into $\hat{\text{R}}$;
    if $n \geq N_{ms}$ **then**
      Generate the N_{ms}-step reward $r_{n-N_{ms}:n}$ from $\hat{\text{R}}$ and record N_{ms}-step experience $exp_{n-N_{ms}:n} = \{\vec{q}_u(n-N_{ms}), a_{n-N_{ms}}, r_{n-N_{ms}:n}, \vec{q}_u(n)\}$ into R with order index k;
      Perform quantum initialization phase on the k-th qubit as $|\Psi_k\rangle = |0\rangle$. Reset $rt_k = 0$ and let $k+ = 1$;
      if $k > C$ **then**
        Set $LF = True$ and reset $k = 1$;
      end
    end
    Let $n+ = 1$;
  until $\vec{q}_u(n) = \vec{q}_u(D)$ || $\vec{q}_u(n) \notin \mathcal{S}$ || $n = N_{\max}$;
  Update $\epsilon \leftarrow \epsilon \times dec_\epsilon$. Update the target D3QN $Q_{D3}(s,a|\boldsymbol{\theta}^{-}_{D3})$ every Υ_{D3} episodes, i.e., $\boldsymbol{\theta}^{-}_{D3} \leftarrow \boldsymbol{\theta}_{D3}$;
end

Note that the total sampling time is equal to the size of mini-batch, which will be specified in the numerical result section later.

Remark 4: Although the QiER buffer involves quantum representations and operations, the corresponding processes, i.e., the quantum initialization phase, the quantum preparation phase and the quantum measurement phase, can be imitated on conventional computing devices without implementing real quantum computations on practical quantum computers.

Remark 5: The associated qubit of sampled experience should be reset to the uniform quantum state, which means that the corresponding quantum preparation phase starts from the uniform quantum state rather than the previous counterpart. This is in line with the quantum phenomenon where a quantum system will collapse onto one of its eigenstates after an observation. Note that the sampled transitions are still remained in the QiER buffer until they are discarded.

D. The Proposed DRL-QiER Solution

The proposed DRL-QiER algorithm is summarized in **Algorithm 1**, and its flow chart is illustrated in Fig. 5. To solve the formulated MDP in Section V-A, double DQN (DDQN) with duelling architecture (D3QN) is adopted to approximate the Q function $Q(\vec{q}_u, \vec{v}_u)$. To further speed up and stabilize

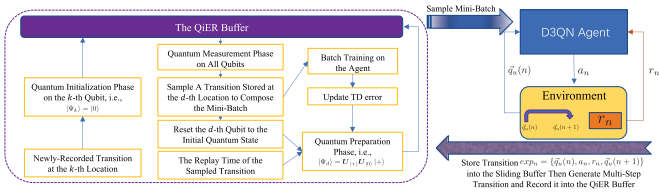


Fig. 5. Flow chart of the proposed DRL-QiER algorithm.

the learning process, N_{ms}-step learning and target network techniques are adopted for updating parameters of the online D3QN. Specifically, the online D3QN aims to minimize the following loss function

$$\mathcal{L}(\boldsymbol{\theta}_{D3}) = \Big[r_{t:t+N_{ms}} + \gamma^{N_{ms}} Q(\vec{q}_{u(t+N_{ms})}, \vec{v}^{*}_{u} | \boldsymbol{\theta}^{-}_{D3}) - Q(\vec{q}_{u(t)}, \vec{v}_{u(t)} | \boldsymbol{\theta}_{D3})\Big]^2, \quad (35)$$

where $\boldsymbol{\theta}_{D3}$ is the parameter vector of the online D3QN, $\boldsymbol{\theta}^{-}_{D3}$ means the parameter vector of the target D3QN. The selected action $\vec{v}^{*}_{u}$ in (35) is chosen from the online D3QN rather than the target D3QN, i.e., $\vec{v}^{*}_{u} = \arg\max_{\vec{v}_u \in \mathcal{A}} Q(\vec{q}_{u(t+N_{ms})}, \vec{v}_u | \boldsymbol{\theta}_{D3})$, which completes the DDQN procedure.

Algorithm 1 starts with network and hyper-parameter initializations, as shown in step 1. At the beginning of each training episode, the UAV's initial location is randomly picked from the state space $\mathcal{S}$ (step 3). Then, the UAV chooses an action following the popular ϵ-greedy action selection policy, which means that the UAV either selects a random action from the action space $\mathcal{A}$ with probability $\epsilon \in [0,1]$ or chooses the optimal action that maximizes the state-action approximation of the online D3QN with probability $1-\epsilon$. After the execution of the selected action, the environment will feed back the next state and the corresponding immediate reward (step 5). The experienced transition exp_n will then be recorded by a sliding buffer, to prepare for the N_{ms}-step learning (step 17). When the sliding buffer is full, the latest N_{ms}-step experience can be generated and then delivered into the QiER buffer (step 18-step 24). Each training episode terminates when one of the following cases are encountered: reaching the destination, hitting the boundary, or exhausting the step threshold (step 26).[12] When one episode is over, the exploration parameter ϵ will be annealed to encourage exploitation from exploration. For every fixed amount of training episodes, the target D3QN will be updated to the online counterpart (step 27). Once the QiER buffer is fully occupied, the mini-batch training for the online D3QN begins (step 6-step 16). With the mini-batch samples, the online D3QN is trained to minimize the mean counterpart of loss function (35), via standard stochastic gradient descent approach (step 15).

Remark 6: The proposed QiER framework for manipulating mini-batch sampling is realized via adopting an unsorted data structure known as binary sum-tree, inspired by the PER approach [24]. The motivation is that for achieving an

[12]It is worth noting that although an explicit energy cost model (commonly for the UAV propulsion power consumption) is not specified in our considered UAV navigation scenario, the global constraint of energy consumption is implied because the step threshold $N_{\max}$ poses a shared budget of propulsion energy cost for all possible trajectories.

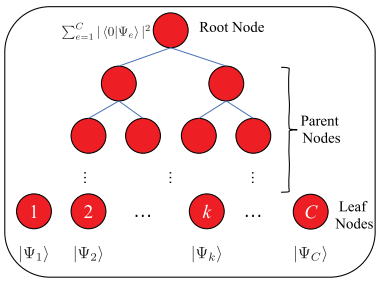


Fig. 6. Sum-tree architecture.

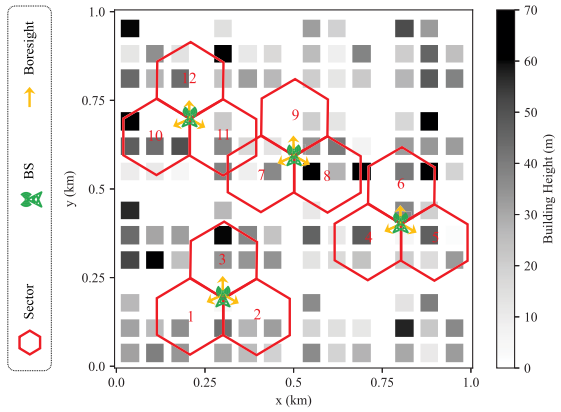
efficient sampling performance based on the current picking distribution $\vec{bp} = [bp_1, bp_2, \ldots, bp_C]$, the complexity should not depend on C which could be unbearably large in practice. An illustration of the used sum-tree architecture can be found in Fig. 6, where either the root node or the parent node contains at most two child nodes as their offspring while their values equal to the sum of their child nodes. Specifically, the k-th leaf node of the sum-tree is pointed to qubit $|\Psi_k\rangle$ and the corresponding stored transition in the QiER buffer, and therefore there are C leaf nodes in total. When performing the quantum measurement phase after the priority updating of quantum initialization phase or quantum preparation phase, the sum of collapse probabilities onto eigenstate $|0\rangle$ of all involved qubits, i.e., $\sum_{e=1}^{C} |\langle 0 \mid \Psi_e \rangle|^2$, can be updated via propagating the measurement of any updated qubit from the corresponding leaf node to the root node, enabling $\mathcal{O}[\log(C)]$ updating and sampling. Besides, the quantum amplitude amplification in quantum preparation phase is based on ***Proposition 1*** and ***Corollary 1***, where the quantum collapse probability updating is steered by closed-form expressions and thus negligible extra computation cost is required. Therefore, the complexity of our proposed QiER framework is comparable to that of propositional PER and DCRL strategies. With the aforementioned efficient implementation, the proposed QiER framework only costs negligible extra computational power and memory, compared to conventional ER approach. Note that SNARM approach adopting ER strategy maintains an extra neural network for radio mapping, which is undoubtedly more computation-expensive than DRL-ER, DRL-PER, DCRL and our proposed DRL-QiER solution. Moreover, our QiER framework does not destruct the convergence of any DRL agent that it is plugged onto, but may result in different convergence curve against DRL agent aided with other experience replay techniques, because it sorely focuses on polishing the picking process of stored transitions, as depicted in Fig. 5.

VI. Numerical Results

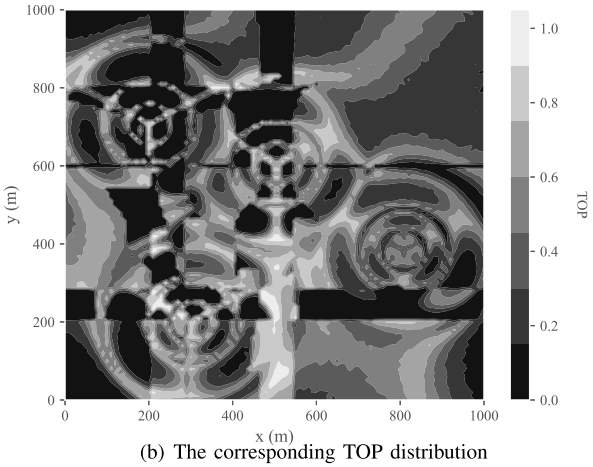
In this section, simulation results for the proposed DRL-QiER solution and the corresponding performance comparison against several baselines are performed.

A. Simulation Environment Setups

For conducting the simulation, the UAV's exploration airspace is set as $\mathbb{A} : [0, 1] \times [0, 1] \times [0, 0.1]$ km. Fig. 7(a) delivers the top view of $\mathbb{A}$, in which the locations of involved BSs and the direction of each ULA's boresight are specified.



(a) The simulation environment



(b) The corresponding TOP distribution

Fig. 7. Simulation environment and the corresponding preview on TOP distribution.

To generate building distribution within $\mathbb{A}$, one realization of statistical model suggested by the International Telecommunication Union (ITU) [42] is invoked, which is subject to the following three parameters: 1) $\hat{\alpha}$ indicates the ratio of region covered by buildings to the whole land; 2) $\hat{\beta}$ represents average amount of buildings; and 3) $\hat{\gamma}$ determines building heights' distribution (say, Rayleigh distribution with mean $\hat{\gamma} > 0$). Besides, the small-scale fading component of G2A link is assumed to follow block Nakagami-m channel model. The common destination's location is fixed at $\vec{q}_u = (0.8, 0.8, 0.1)$ km, without loss of generality.

Unless otherwise mentioned, the parameter setups regarding simulation environment are in line with Table I. With the generated local building distribution, antenna model and small-scale fading model, the corresponding TOP distribution over arbitrary UAV location within $\mathbb{A}$ can be previewed as Fig. 7(b).

B. Structure of DNNs and Hyper-Parameter Settings for Learning Process

The proposed DRL-QiER algorithm is implemented on Python 3.8 with TensorFlow 2.3.1 and Keras. Specifically, the DNNs of online D3QN agent are constructed with fully-connected feedforward ANNs. The shapes of the online D3QN's input and output layers are subject to the UAV's horizontal locations and the amount of possible flying directions,

TABLE I
PARAMETER SETTINGS FOR SIMULATION ENVIRONMENT

Parameters	Values	Parameters	Values
Amount of BSs B	4	Amount of sectors $3B$	12
Horizontal side-length of $\mathbb{A}$ D	1 km	Amount of each ULA's array elements M	8
Half-power beamwidth $\Theta_{3\text{dB}}/\Phi_{3\text{dB}}$	65°/65°	Speed of light c	3×10^8 m/s
Carrier frequency f_c	2 GHz	Wave length λ	15 cm
ULA's element spacing distance d_v	7.5 cm	ULA's electrically titled angle θ_{etilt}	100°
Antenna height of BS	25 m	Flying altitude of UAV	100 m
ITU building distribution parameter $\hat{\alpha}$	0.3	ITU building distribution parameter $\hat{\beta}$	118
ITU building distribution parameter $\hat{\gamma}$	25	Total amount of buildings $\hat{\beta}D^2$	118
Expected size of each building $\hat{\alpha}/\hat{\beta}$	0.0025 km^2	Maximum height of buildings	70 m
Transmit power of each sector P_i	20 dBm	Nakagami shape factor m for LoS/NLoS	3/1
Transmission outage threshold Γ_{th}	0 dB	Average power of AWGN σ^2	-90 dBm
Duration of time slot Δ_t	0.5 s	Velocity of the UAV V_u	30 m/s
Amount of signal Measurements L	1000	Weight balancing the minimization τ	50

TABLE II
HYPER-PARAMETER SETTINGS FOR LEARNING PROCESS

Parameters	Values	Parameters	Values
Capacity of QiER buffer C	20000	Size of mini-batch N_{mb}	128
Initial ϵ-greedy factor ϵ	0.5	Annealing speed dec_ϵ	0.994/episode
Target D3QN update frequency Υ_{D3}	5	Length of sliding buffer N_{ms}	30
Positive special reward r_D	400	Negative special reward r_{ob}	-10000
Learning rate α_{lr}	Adam's default	Discount factor γ	1
Maximum training episodes $te_{\max}$	2000	Step threshold $N_{\max}$	400

respectively. Between the input and output layers, there are 4 hidden layers, where the first 3 hidden layers contain 512, 256, 128 neurons, respectively. The last hidden layer plays the role as duelling layer consisting of $N_{fd}+1$ neurons, where one neuron indicates the estimation of state-value and the other N_{fd} neurons reflect action advantages. Then, the outputs of the duelling layer will be aggregated to generate the estimation of the N_{fd} actions at the output layer. Besides, the optimizer minimizing the mean square error (MSE) for the DRL-QiER agent is Adam with fixed learning rate. The activation functions for each hidden layer and the output layer are Relu and Linear, respectively. Note that the target D3QN shares the same structure as its online counterpart.

The settings of hyper-parameter for learning process are stated in Table II.

C. Training of the DRL-QiER Algorithm

Fig. 8(a) and Fig. 8(b) depict the return history and designed trajectories of the proposed DRL-QiER solution, respectively. Note that the moving average return for each training episode is calculated via a moving window with length of 200 episodes, while the corresponding designed trajectories are picked with spatially-separated initial locations in the late training stage (in the range of episodes 1900-2000), for the sake of neat and sufficient demonstration. From Fig. 8(a), it is straightforward to conclude that the moving average returns steadily converge to the maximum alongside the training process, although some fluctuations are experienced, which is a typical phenomenon in DRL field. Besides, from Fig. 8(b), it is observed that the proposed DRL-QiER solution can direct the UAV from various initial locations to the common destination, with designed trajectories adaptive to the TOP distribution. Regions with higher TOP are avoided while keeping the UAV being directed to reach the common destination with possibly fewer moving steps (equivalently, as short flying time cost as possible). For instance, even the near-to-zero but extremely narrow TOP slots around $(0.4, 0.76, 0.1)$ km and $(0.6, 0.79, 0.1)$ km can be recognized. On the contrary, higher TOP regions in the range of $(0.4-0.6, 0-0.5, 0.1)$ km are bypassed as much as possible. Another good example is the trajectory starting from location around $(0.95, 0.09, 0.1)$ km, where the "V" shape around $(0.95, 0.2, 0.1)$ km perfectly demonstrates the effectiveness of the proposed DRL-QiER solution, in which the higher TOP fields are avoided. Note that larger weight factor τ will generally lead the designed path to experience lower TOP regions, but inevitably enlarging the time cost (say, longer and more tortuous trajectory) reaching the common destination. This is the reason why weight factor τ is invoked to balance the proposed minimization problem (P1).

D. Performance Comparison

Four DRL-aided baselines are considered for performance comparison, listed as follows.

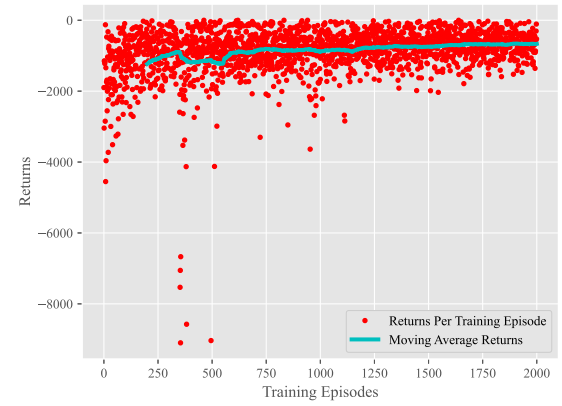
- *DRL-ER*: The D3QN is trained via mini-batch sampling from standard ER buffer with uniform sample-picking strategy.
- *DRL-PER*: The D3QN is trained via stochastic mini-batch sampling from the PER buffer with proportional prioritization method, in line with [24]. In this approach, the priority of each recorded transition x is measured by its corresponding absolute TD error $|\delta(x)|$. Then, the probability of picking a transition from the PER buffer follows $p(x) = (|\delta(x)| + \xi)^{\alpha_{\text{PER}}} / \sum_{x'} (|\delta(x')| + \xi)^{\alpha_{\text{PER}}}$, where a small positive constant ξ is used to avoid zero-priority case and α_{PER} determines how much

prioritization is applied, with $\alpha_{\text{PER}} = 0$ corresponding to the special case that is equivalent to DRL-ER baseline. To correct the bias caused by priority-based sampling, normalized importance-sampling (IS) weight $W(x) = (C \times p(x))^{-\beta_{\text{PER}}} / \max_{x'} W(x')$ is calculated to scale the updating of DNNs, where C is the capacity of the PER buffer and β_{PER} reflects the amount of IS correction. The parameter β_{PER} should be incremented from a relatively small positive constant to 1 over the training process because a full-step update is more important when the algorithm begins to converge.

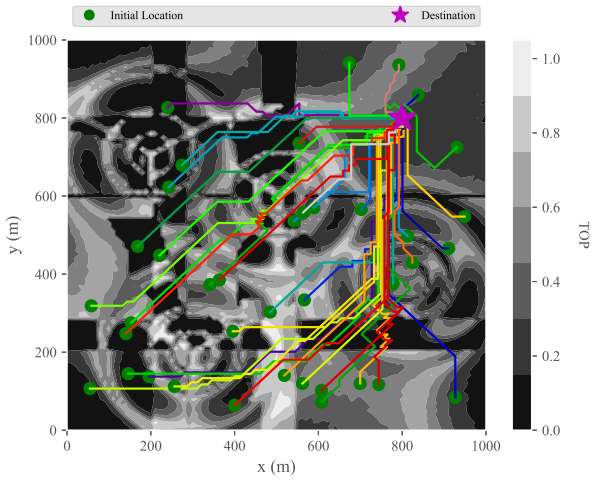
- *DCRL*: The DCRL training paradigm aims to offer better mini-batch sampling efficiency, according to the complexities of recorded experiences. Specifically, the complexity of each transition is determined by self-paced priority and coverage penalty, where self-paced priority maps TD error into the difficulty of current curriculum and coverage penalty uses replay times of transitions to enhance sampling diversity. For detailed implementation of DCRL, please refer to [25].
- *SNARM*: The framework SNARM invokes an extra DNN termed as radio map to help improve the overall learning efficiency. The signal measurements alongside the UAV's trajectory are utilized to train not only the online D3QN but also the radio map. The radio map enables it to generate simulated trajectories and thus reduces actual trials. Based on standard Dyna architecture, one D3QN update with the actual experiences follows several extra updates with the simulated transitions. Therefore, the SNARM approach is promised to help achieve better learning performance while reducing the cost of data acquisition from actual experiences. For more details of SNARM, please refer to [5].

For fair comparison, the structures of online and target D3QNs for all baselines are the same as those of the proposed DRL-QiER solution, while the hyper-parameter settings of these baselines are in line with Table II. Besides, the construction of radio map's DNN and the corresponding hyper-parameter settings of baseline SNARM are in accordance to [5], while the complexity index function, the curriculum evaluation function, the self-paced prioritized function, the coverage penalty function and the corresponding DCRL hyper-parameter settings are in line with [25]. Furthermore, the additional hyper-parameters regarding PER in DRL-PER baseline are set as $\alpha_{\text{PER}} = 1$, $\xi = 0.01$ and $\beta_{\text{PER}} = 0.4$. All the baselines are altered to involve multi-step learning and start training after their replay buffers are fully exploited. Moreover, all the baselines share the same randomly-generated initial UAV locations with the proposed DRL-QiER solution, for each training episode.

Fig. 9(a) delivers the performance comparison on moving average returns of the proposed DRL-QiER solution and considered baselines, versus training episodes. From this subfigure, it is easy to find that SNARM approach can offer satisfactory learning performance, thanks to the simulated trajectories enabled by the extra DNN (i.e., the radio map). Especially, in the range of training episode from 400 to 1000,



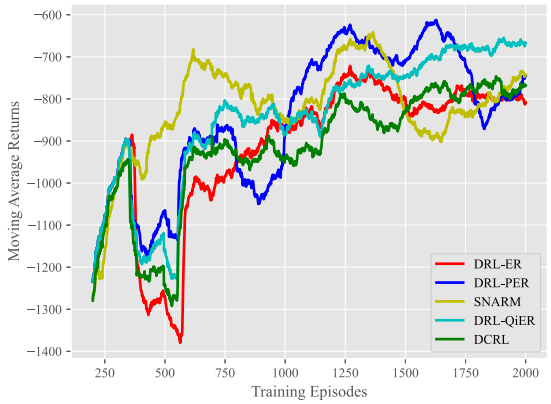
(a) Training return history



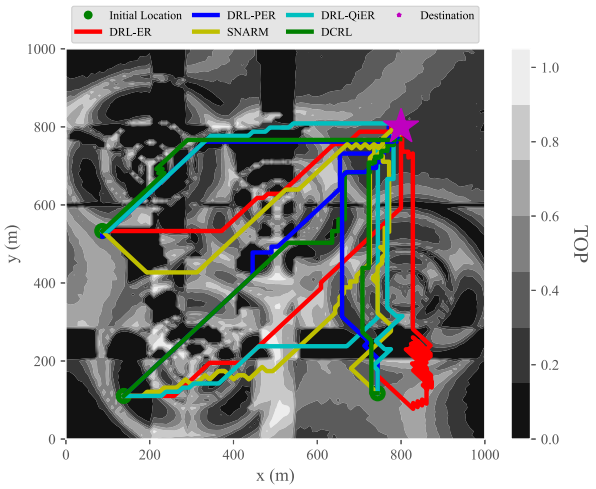
(b) The corresponding designed trajectories

Fig. 8. Training results of the proposed DRL-QiER solution.

despite that the radio map is getting well trained as the training process going. Besides, DRL-PER, DRL-QiER and DCRL approaches can achieve better moving average returns than DRL-ER method, in the early training stage (e.g., episodes 500-750). The reason is that DRL-ER solution samples transitions uniformly without considering their priorities, which leads transitions with higher importance to have less opportunities for training the online D3QN. However, DRL-PER method experiences server fluctuations than DRL-QiER and DCRL (e.g., episodes 1250-2000), which is because DRL-PER does not take transitions' replay time into account and thus some transitions are sampled with undesired high frequency while their absolute TD errors remain relatively large. The proposed DRL-QiER solution showcases more steady learning ability, with less amplification of fluctuation and overall raising trend, thanks to the QiER technique which balances sampling priority and diversity in a better manner. Although SNARM and DCRL approaches can offer satisfactory learning performances, their respective shortcomings are: 1) SNARM framework needs to train an extra model-learning DNN, which thus introduces extra biases and heavy training burden, and 2) it is difficult to set up feasible complexity index function, curriculum evaluation function, self-paced prioritized function, coverage penalty function and the corresponding DCRL hyper-parameters, which limits the robustness of DCRL solution. The

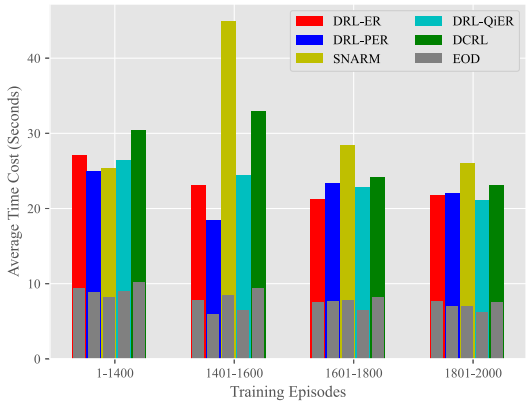


(a) Comparison on moving average returns

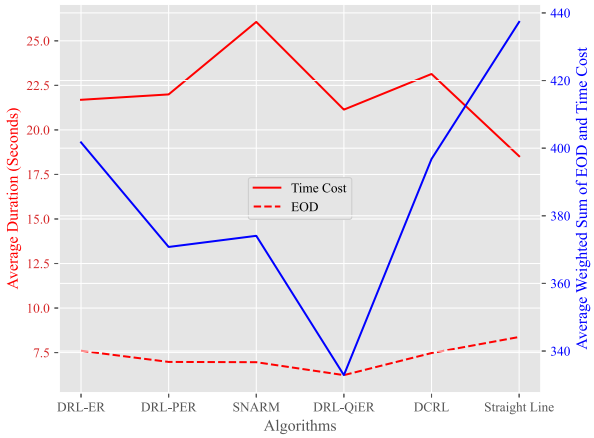


(b) Designed trajectories of trained agents

Fig. 9. Performance comparison on moving average returns and designed trajectories.



(a) Comparison on average time cost



(b) Comparison on average duration

Fig. 10. Performance comparison on average time costs and EODs.

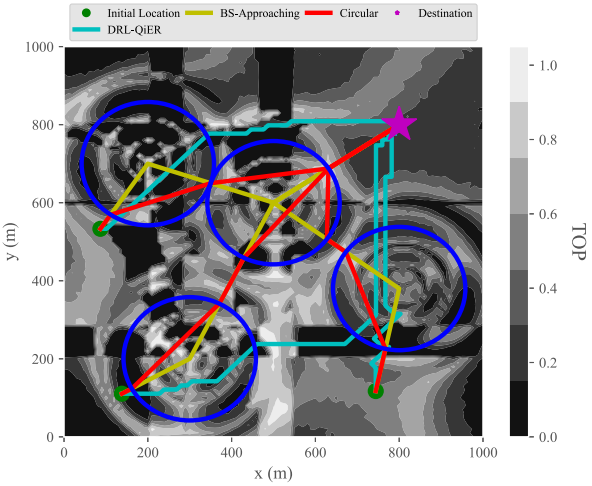


Fig. 11. Comparison on designed trajectories of circular, BS-approaching and DRL-QiER solutions.

proposed DRL-QiER method requires less hyper-parameters tuning and contains no extra DNN, and therefore is easier and more robust for implementation. To deliver more insights, Fig. 9(b) depicts the comparison on designed trajectories of the implemented algorithms, over three representative starting locations chosen from episodes 1910-2000. It is straightforward to observe that the proposed DRL-QiER and the considered baselines direct the UAV to hit the common destination with different trajectories.

Fig. 10(a) demonstrates comparison on average time cost of designed trajectories and the corresponding EOD for the considered algorithms, over four episode slots 1-1400, 1401-1600, 1601-1800 and 1801-2000. From this figure, one can find that the proposed DRL-QiER solution can help achieve both lower average EOD and average time cost, within each episode slot. Especially, in the late training state (e.g., episode slot 1800-2000), the proposed DRL-QiER method outperforms other baselines, in terms of both average EOD and average time cost. Furthermore, Fig. 10(b) illustrates comparison on average duration and average weighted sum of EOD and time cost over the last 200 training episodes, for all the DRL-aided approaches and non-learning-based strategy termed as straight line. From this figure, it is easy to find that while the straight line solution offers the cheapest average time cost, it leads the UAV to suffer the highest average EOD, which is extremely non-preferable and thus unveils the benefits provided by DRL-aided approaches. On the contrary, the proposed DRL-QiER solution can not only help the UAV experience the lowest average EOD, compared to both other DRL-aided approaches and the straight line strategy, but also direct the UAV to reach the common destination with the cheapest average time cost, against other DRL-aided solutions.

TABLE III
COMPARISON ON AVERAGE DURATIONS OF CIRCULAR, BS-APPROACHING AND DRL-QiER SOLUTIONS

	Circular	BS-Approaching	DRL-QiER
Time Cost	28.440 s	31.916 s	31.234 s
EOD	10.469 s	12.128 s	8.136 s
Weighted Sum of Time Cost and EOD	551.890	638.316	438.034

To further highlight superiority of the proposed DRL-QiER solution against conventional path planning approach, performance comparisons between DRL-QiER and two other non-learning baselines are demonstrated in Fig. 11 and Table III, for three different initial locations. Specifically, the BS-approaching baseline aims to direct UAV to travel across the nearest BS alongside the flight because intuitively locations nearby BS can provide stronger coverage quality. The other non-learning baseline is based on the assumption of BS's circular coverage, within which arbitrary location is simply treated as that can provide satisfactory coverage strength. The circles in Fig. 11 are taken as examples to evaluate the designed trajectories under the circular coverage assumption. Note that unlike the aforementioned DRL-related approaches, both of these two considered baselines are not dependent on the actual TOP distribution, and thus naive and inferior trajectories could be generated. To validate this, Table III delivers comparison on average durations of circular, BS-approaching and DRL-QiER solutions, over trajectories started from the considered three initial locations. From this table, it is straightforward to observe that the proposed DRL-QiER solution can direct UAV to achieve the minimum amount of average weighted sum of time cost and EOD where the corresponding average EOD is the cheapest, while the other two non-learning baselines suffer from greater average EOD. The corresponding reason can be interpreted as that the proposed DRL-QiER solution (more generally, DRL-aided approaches) is trained via interacting with the actual TOP distribution, which validates the advantages provided by DRL-related solutions against non-learning alternatives.

VII. CONCLUSION

In this work, an intelligent navigation task for cellular-connected UAV networks was investigated, aiming at minimizing the weighted sum of time cost and expected outage duration alongside UAVs' flying trajectories towards the common destination with randomly-generated initial UAV locations. To overcome the inefficiency of offline optimization techniques in navigating the UAV, a DRL-aided algorithm, i.e., DRL-QiER solution, was proposed, in which the innovative QiER framework is coined to help the D3QN agent hit a better learning efficiency. Simulation results validated the effectiveness of the proposed DRL-QiER solution, while performance comparison against both several DRL-aided baselines and straight line strategy showcased DRL-QiER method's superiority. Moreover, the proposed QiER framework can be potentially extended into other existing DRL frameworks that are dependent on ER technique, e.g., deep deterministic policy gradient (DDPG), soft actor-critic (SAC) and Rainbow.

REFERENCES

[1] Y. Li and A. H. Aghvami, "Intelligent UAV navigation: A DRL-QiER solution," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Seoul, South Korea, May 2022, pp. 1–6.

[2] Z. Xiao, H. Dong, L. Bai, D. O. Wu, and X.-G. Xia, "Unmanned aerial vehicle base station (UAV-BS) deployment with millimeter-wave beamforming," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1336–1349, Feb. 2020.

[3] W. Mei, Q. Wu, and R. Zhang, "Cellular-connected UAV: Uplink association, power control and interference coordination," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5380–5393, Nov. 2019.

[4] L. Liu, S. Zhang, and R. Zhang, "Multi-beam UAV communication in cellular uplink: Cooperative interference cancellation and sum-rate maximization," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4679–4691, Oct. 2019.

[5] Y. Zeng, X. Xu, S. Jin, and R. Zhang, "Simultaneous navigation and radio mapping for cellular-connected UAV with deep reinforcement learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4205–4220, Jul. 2021.

[6] Y. Zeng and X. Xu, "Path design for cellular-connected UAV with reinforcement learning," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)* Waikoloa, HI, USA, Dec. 2019, pp. 1–6.

[7] S. Zhang, Y. Zeng, and R. Zhang, "Cellular-enabled UAV communication: A connectivity-constrained trajectory optimization perspective," *IEEE Trans. Commun.*, vol. 67, no. 3, pp. 2580–2604, Mar. 2018.

[8] J. Lyu and R. Zhang, "Network-connected UAV: 3-D system modeling and coverage performance analysis," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 7048–7060, Aug. 2019.

[9] W. Mei and R. Zhang, "Cooperative downlink interference transmission and cancellation for cellular-connected UAV: A divide-and-conquer approach," *IEEE Trans. Commun.*, vol. 68, no. 2, pp. 1297–1311, Feb. 2020.

[10] C. Zhan and Y. Zeng, "Energy-efficient data uploading for cellular-connected UAV systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7279–7292, Nov. 2020.

[11] E. Bulut and I. Guevenc, "Trajectory optimization for cellular-connected UAVs with disconnectivity constraint," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Kansas City, MO, USA, May 2018, pp. 1–6.

[12] S. Hu, Q. Wu, and X. Wang, "Energy management and trajectory optimization for UAV-enabled legitimate monitoring systems," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 142–155, Jan. 2021.

[13] S. Hu, W. Ni, X. Wang, A. Jamalipour, and D. Ta, "Joint optimization of trajectory, propulsion, and thrust powers for covert UAV-on-UAV video tracking and surveillance," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 1959–1972, 2021.

[14] S. Hu, X. Chen, W. Ni, E. Hossain, and X. Wang, "Distributed machine learning for wireless communication networks: Techniques, architectures, and applications," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1458–1493, 3rd Quart., 2021.

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

[16] D. Dong, C. Chen, H. Li, and T. J. Tarn, "Quantum reinforcement learning," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 38, no. 5, pp. 1207–1220, Oct. 2008.

[17] D. Dong, C. Chen, J. Chu, and T.-J. Tarn, "Robust quantum-inspired reinforcement learning for robot navigation," *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 1, pp. 86–97, Feb. 2012.

[18] G. D. Paparo, V. Dunjko, A. Makmal, M. A. Martin-Delgado, and H. J. Briegel, "Quantum speedup for active learning agents," *Phys. Rev. X*, vol. 4, Jul. 2014, Art. no. 031002.

[19] V. Dunjko, J. M. Taylor, and H. J. Briegel, "Quantum-enhanced machine learning," *Phys. Rev. Lett.*, vol. 117, no. 13, 2016, Art. no. 130501.

[20] V. Saggio *et al.*, "Experimental quantum speed-up in reinforcement learning agents," *Nature*, vol. 591, no. 7849, pp. 229–233, Mar. 2021.

[21] J.-A. Li *et al.*, "Quantum reinforcement learning during human decision-making," *Nature Hum. Behav.*, vol. 4, no. 3, pp. 294–307, Mar. 2020.

[22] Y. Li, A. H. Aghvami, and D. Dong, "Intelligent trajectory planning in UAV-mounted wireless networks: A quantum-inspired reinforcement learning perspective," *IEEE Wireless Commun. Lett.*, vol. 10, no. 9, pp. 1994–1998, Sep. 2021.

[23] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
[24] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in *Proc. IEEE Int. Conf. Learn. Represent.*, Oct. 2016, pp. 1–21.
[25] Z. Ren, D. Dong, H. Li, and C. Chen, "Self-paced prioritized curriculum learning with coverage penalty in deep reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2216–2226, Jun. 2018.
[26] Q. Wei, H. Ma, C. Chen, and D. Dong, "Deep reinforcement learning with quantum-inspired experience replay," *IEEE Trans. Cybern.*, early access, Feb. 18, 2021, doi: 10.1109/TCYB.2021.3053414.
[27] *Study on 3D Channel Model for LTE*, 3GPP, document TR 36.873, Dec. 2017.
[28] *Enhanced LTE Support for Aerial Vehicles*, 3GPP, document TR 36.777, Dec. 2017.
[29] W. Zhang and K. B. Letaief, "Cooperative spectrum sensing with transmit and relay diversity in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4761–4766, Dec. 2008.
[30] M. Dong, L. Tong, and B. M. Sadler, "Optimal insertion of pilot symbols for transmissions over time-varying flat fading channels," *IEEE Trans. Signal Process.*, vol. 52, no. 5, pp. 1403–1418, May 2004.
[31] S. Ma, X. Pan, G.-H. Yang, and T.-S. Ng, "Blind symbol synchronization based on cyclic prefix for OFDM systems," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 1746–1751, May 2009.
[32] M. Sheng, Y. Li, X. Wang, J. Li, and Y. Shi, "Energy efficiency and delay tradeoff in device-to-device communications underlaying cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 92–106, Jan. 2016.
[33] J. Tang, D. K. C. So, E. Alsusa, and K. A. Hamdi, "Resource efficiency: A new paradigm on energy efficiency and spectral efficiency tradeoff," *IEEE Trans. Wireless Commun.*, vol. 13, no. 8, pp. 4656–4669, Aug. 2014.
[34] X. Cheng, Q. Yao, M. Wen, C.-X. Wang, L.-Y. Song, and B.-L. Jiao, "Wideband channel modeling and intercarrier interference cancellation for vehicle-to-vehicle communication systems," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 9, pp. 434–448, Sep. 2013.
[35] C.-K. Wen, S. Jin, and K.-K. Wong, "On the sum-rate of multiuser MIMO uplink channels with jointly-correlated Rician fading," *IEEE Trans. Commun.*, vol. 59, no. 10, pp. 2883–2895, Oct. 2011.
[36] P. K. Upadhyay and S. Prakriya, "Performance of two-way opportunistic relaying with analog network coding over Nakagami-m fading," *IEEE Trans. Veh. Technol.*, vol. 60, no. 4, pp. 1965–1971, May 2011.
[37] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.
[38] J. Liu, B. Krishnamachari, S. Zhou, and Z. Niu, "DeepNap: Data-driven base station sleeping operations through deep reinforcement learning," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4273–4282, Dec. 2018.
[39] X. Tao, Y. Duan, M. Xu, Z. Meng, and J. Lu, "Learning QoE of mobile video transmission with deep neural network: A data-driven approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1337–1348, Jun. 2019.
[40] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
[41] T. De Bruin, J. Kober, K. Tuyls, and R. Babuška, "The importance of experience replay database composition in deep reinforcement learning," in *Proc. Deep Reinforcement Learn. Workshop, NIPS*, Dec. 2015, pp. 1–9.
[42] P. Series, *Propagation Data and Prediction Methods Required for the Design of Terrestrial Broadband Radio Access Systems Operating in a Frequency Range From 3 to 60 GHz*, document ITU-R, Rec. 1410-5, Feb. 2012.

Yuanjian Li (Graduate Student Member, IEEE) received the B.Eng. degree from the Faculty of Computer Science and Technology, Nanjing Tech University, Nanjing, China, and the M.Eng. degree from the Faculty of Information Science and Engineering, Huaqiao University, Xiamen, China, in 2015 and 2019, respectively, both in information and communication engineering. He is currently pursuing the Ph.D. degree with the Department of Engineering, Faculty of Natural, Mathematical and Engineering Sciences, King's College London, London, U.K. His current research interests include UAV-aided wireless networks, machine learning, and radio resource management.

A. Hamid Aghvami (Life Fellow, IEEE) joined as the Academic Staff at King's College London in 1984. In 1989, he was promoted to a Reader and in 1993, was promoted to a Professor in telecommunications engineering. He was the Director of the Centre from 1994 to 2014. He was a Visiting Professor at NTT Radio Communication Systems Laboratories in 1990, a Senior Research Fellow at BT Laboratories from 1998 to 1999, and an Executive Advisor at Wireless Facilities Inc., USA, from 1996 to 2002. He is the Founder of the Centre for Telecommunications Research, at King's College London. He is the Chairperson of Advanced Wireless Technology Group Ltd. He is also the Managing Director of Wireless Multimedia Communications Ltd., his own consultancy company. He is a Visiting Professor at Imperial College. He carries out consulting work on digital radio communications systems for British and international companies. He has published over 560 technical journal and conference papers, filed 30 patents, and given invited talks and courses the world over on various aspects of mobile radio communications. He leads an active research team working on numerous mobile and personal communications projects for fourth and fifth generation networks, these projects are supported both by government and industry.

He was a member of the Board of Governors of the IEEE Communications Society (2001–2003), a Distinguished Lecturer of the IEEE Communications Society (2004–2007), and has been a member, the Chairperson, and the Vice-Chairperson of the technical program and organizing committees of a large number of international conferences. He is a fellow of the Royal Academy of Engineering and IET. He is also the Founder of the International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), a major yearly conference attracting around 1,000 attendees. He was awarded the IEEE Technical Committee on Personal Communications (TCPC) Recognition Award in 2005 for his outstanding technical contributions to the communications field and for his service to the scientific and engineering communities. In 2009, he was awarded a Fellowship of the Wireless World Research Forum in recognition of his personal contributions to the wireless world and for his research achievements as the Director at the Centre for Telecommunications Research, at King's College London.

Daoyi Dong (Senior Member, IEEE) received the B.E. degree in automatic control and the Ph.D. degree in engineering from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively.

He was with the Institute of Systems Science, Chinese Academy of Sciences, and Zhejiang University. He had visiting positions at Princeton University, NJ, USA, RIKEN, Wako-shi, Japan, the University of Duisburg-Essen, Germany, and The University of Hong Kong, Hong Kong. He is currently a Scientia Associate Professor at the University of New South Wales, Canberra, Australia. His research interests include quantum control and machine learning. He was awarded an ACA Temasek Young Educator Award by the Asian Control Association and was a recipient of the International Collaboration Award and the Australian Post-Doctoral Fellowship from the Australian Research Council, and the Humboldt Research Fellowship from the Alexander von Humboldt Foundation of Germany. He is the Member-at-Large and the Board of Governors of the IEEE Systems, Man and Cybernetics Society. He served as an Associate Editor for IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS (2015–2021). He is currently an Associate Editor of IEEE TRANSACTIONS ON CYBERNETICS and IEEE/CAA JOURNAL OF AUTOMATICA SINICA and a Technical Editor of IEEE/ASME TRANSACTIONS ON MECHATRONICS.