Data Freshness and Energy-Efficient UAV Navigation Optimization: A Deep Reinforcement Learning Approach

Sarder Fakhru1 Abedin1, Member, IEEE, Md. Shirajum Munir1, Graduate Student Member, IEEE, Nguyen H. Tran1, Senior Member, IEEE, Zhu Han2, Fellow, IEEE, and Choong Seon Hong2, Senior Member, IEEE

Abstract—In this paper, we design a navigation policy for multiple unmanned aerial vehicles (UAVs) where mobile base stations (BSs) are deployed to improve the data freshness and connectivity to the Internet of Things (IoT) devices. First, we formulate an energy-efficient trajectory optimization problem in which the objective is to maximize the energy efficiency by optimizing the UAV-BS trajectory policy. We also incorporate different contextual information such as energy and age of information (AoI) constraints to ensure the data freshness at the ground BS. Second, we propose an agile deep reinforcement learning with experience replay model to solve the formulated problem concerning the contextual constraints for the UAV-BS navigation. Moreover, the proposed approach is well-suited for solving the problem, since the state space of the problem is extremely large and finding the best trajectory policy with useful contextual features is too complex for the UAV-BSs. By applying the proposed trained model, an effective real-time trajectory policy for the UAV-BSs captures the observable network states over time. Finally, the simulation results illustrate the proposed approach is 3.6% and 3.13% more energy efficient than those of the greedy and baseline deep Q Network (DQN) approaches.

Index Terms—Unmanned aerial vehicle, age of information, deep reinforcement learning, trajectory optimization.

I. INTRODUCTION

The rapid deployment of the Fifth-generation (5G) wireless network sets an unparalleled criteria for high-quality wireless connectivity and services [1]. As a result, conventional cellular networks face enormous challenges to meet the stringent requirements for different 5G application types, such as enhanced mobile broadband (eMBB), ultra-reliable and low-latency communications (URLLC), and massive machine-type communications (mMTC) applications [2], [3]. One of potential solutions is to deploy unmanned aerial vehicles (UAV) in the 5G network environment where UAVs serve the network applications and users as aerial base stations (UAV-BS) [4]. Unlike the conventional wireless network infrastructure, UAV-BSs are more agile and capable of not only providing better coverage but also significantly strengthening the network capability to meet the stringent demands of high capacity with wide coverage, and low delay constraints. However, the deployment and autonomous navigation of multiple UAV-BSs in 5G networks are still challenging due to the limited energy capacity of UAV-BSs. Moreover, in recent years, the concept of edge computing with 5G [5] has also emerged to complement the need for a remote cloud environment for enabling computation oriented communications (COC) applications such as virtual and augmented reality (VR and AR) [6], real-time monitoring and surveillance [7]. As the demand for the upcoming COC applications [8] becomes more prevalent, the need for receiving fresh information data update from different futuristic applications [9] requires a new metric, age of information (AoI) [10], to measure data freshness apart from the traditional performance metrics for the 5G application types.

In case of the UAV-BS navigation, the existing research [11]–[16] mainly focuses on the path planning and placement of UAV-BSs in the network along with communication and energy constraints. However, to incorporate the computation oriented applications at the edge network requires an agile UAV-BS navigation that not only enhances the energy efficiency [17] but also ensures the up-to-date data delivery on time for seamless operation of COC applications at the edge computing platform.

Under the above circumstances, we focus on optimizing the UAV navigation considering energy efficiency and the AoI context in the 5G enabled edge computing environment. The main contributions of the paper are summarized as follows:

1. First, we formulate an energy-efficient UAV-BS navigation optimization problem in the edge computing network under the contextual constraints such as energy,
navigation, and AoI metric, and then we show that the formulated problem is NP-hard.

- Second, we employ a deep reinforcement learning technique, deep Q-network with experience replay memory, which can achieve energy-efficient UAV navigation under the contextual constraints. With the proposed trained model, an effective real-time trajectory policy for the UAV-BSs can be obtained that captures the observable network states over time. As a result, we design the state, observation, action space and reward explicitly for the proposed deep Q network (DQN) with experience replay which can effectively solve the trajectory optimization problem under the AoI and energy-efficiency constraints. Also, unlike the traditional deep Q network, the proposed model utilizes the benefit of experience replay memory to obtain the optimal trajectory policy while coordinating multiple UAV-BS locations.

- Finally, we perform an extensive experimental analysis to evaluate the performance of the proposed approach. We also conduct an extensive simulation analysis to find the appropriate system parameters to empower the learning model with the proper discount factor and AoI performance metric threshold. The results show that the navigation policy that is obtained by applying the proposed DQN with experience replay achieves significant energy efficiency and data freshness compared to the baseline approaches.

The remainder of the paper is organized as follows. In Section II, we present an extensive literature review based on the current research. In Sections III and IV, we present the system model and problem formulation, respectively. Section V explains in detail how we solve the proposed optimization problem with deep Q-learning with experience replay. In Section VI, we present the simulation analysis to validate the performance and efficiency of our proposed approach for UAV-BS navigation. Finally, in Section VII we conclude the discussion.

II. Literature Review

In this section, we have classified the related works in sub-sections based on UAV navigation, the energy efficiency of UAVs, and the importance of age of information for UAV path planning. Then we provide a summary of challenges that we face to archive the goal of seamless operation of COC applications.

A. UAV Navigation

In [18], the authors proposed an online deep reinforcement learning approach to enable UAV navigation in a large-scale complex environment. The problem formulation for the UAV navigation is based on the partially observable Markov decision process (POMDP) where the authors proposed the actor-critic framework to solve the problem. In [19], the authors focused on capturing the UAV motion while planning the trajectory for UAV navigation through massive multiple-input-multiple-output (MIMO) technique. The UAV agent makes the navigation decision on the basis of the received signal strengths which are used to train the proposed DQN. A trajectory planning method for UAVs in urban environments is proposed in [20], where the authors considered the UAV’s three-dimensional (3-D) environment map to enable navigation to fuse global navigation satellite (GNSS) signals with ambient cellular signals of opportunity. A novel Deep Reinforcement Learning (DRL) algorithm is proposed in [21] for non-holonomic robots with continuous control in an unknown dynamic environment with moving obstacles. A distributed sense-and-send protocol to coordinate the UAVs for sensing and transmission is proposed in [22]. A reinforcement learning technique is therefore applied to solve some of the UAV’s key problems related to trajectory control and wireless resource management. However, most of the works on UAV navigation focused on solving the traditional energy-efficient UAV trajectory optimization problem while mainly considering on wireless resource allocation. Therefore, none of the works on UAV navigation specifically focused on the computation oriented communications (COC) applications where considering both the energy-efficiency and AoI metric are crucial.

B. Energy Efficiency

In [23], the authors strive to reduce the total energy consumption of UAVs, including both power propulsion and communication-related energy, while meeting the requirement of multiple ground node (GN) communication throughput. The problem formulation considers the issue of energy minimization by jointly optimizing the allocation of UAV trajectory and contact time between GNs, as well as the overall mission completion time. Using the successive convex approximation (SCA) technique, an effective iterative algorithm is proposed to update the UAV trajectory and contact time allocation at the same time at each iteration, which may converge to a solution that satisfies the Karush-Kuhn-Tucker (KKT) conditions. The goal of the work in [24] is to reduce the propulsion energy consumption of the UAV while meeting the requirement of throughput by optimizing the trajectory of the running track. To transform the problem into a discrete counterpart, a variable discretization approach is used and later, the problem is transformed into a problem of convex optimization where the proposed method can obtain a locally optimal solution. In order to address the critical issue of insufficient on-board UAV energy and CE transmission energy, the authors in [25] focused on maximizing energy efficiency (EE) by jointly optimizing the scheduling of the backscatter devices (BD), the power reflection coefficients of the BDs, the transmission power of the CEs and the trajectory of the UAV. Moreover, the authors considered the BD’s throughput and other realistic constraints for the problem formulation. In [26], the authors considered minimizing UAV and user equipment (UE) weighted sum energy consumption subject to task constraints, information-causality constraints, bandwidth allocation constraints, and trajectory constraints of the UAV. The UAV-BSs energy-efficient repositioning trajectories are designed in [27] using the Kuhn-Munkres based algorithm, where an Echo State Network based algorithm enables user equipment (UEs) to predict future trajectories.
Unlike the existing works, in this paper, we design the trajectory policy of UAV-BSs considering the trade-off between the energy-efficiency and AoI metric along with other trajectory constraints for the computation oriented communications (COC) applications. We further, design a joint action space of considering the multiple UAV-BS navigation that further reduces the overlapping coverage to enhance the energy-efficiency of the UAV-BSs.

C. Age of Information (AoI)

In [28], the authors proposed a dynamic programming approach to study a problem of UAV path planning and data acquisition with the concept of AoI metric. In the proposed approach, the authors jointly considered the selection of data acquisition mode, energy consumption at each node, and age evolution of the information collected by the UAVs. The UAV flight trajectory and status update packet scheduling are jointly configured in [29] to achieve the required weighted sum for the age-of-information (AoI) values of various processes at the UAV, referred to as weighted sum-AoI. A deep reinforcement learning (RL) algorithm is proposed to achieve the optimal policy that minimizes the weighted sum-AoI, called the age-optimal strategy. The authors suggested a combination of ground sensor nodes (SN) and trajectory planning strategy in [30] to strike a balance between the upload time of the SNs and the flight time of the UAVs using dynamic programming in different scenarios. In [31], the authors sought to minimize UAV’s total energy consumption by jointly optimizing the association of the Internet of Things Devices (IoTD), the allocation of computer resources, the UAV hovering time, the wireless power duration and the IoTD service sequence. The authors proposed a UAV trajectory planning model for data collection in [32], where the objective is to minimize expired data packets across the entire sensor network system. The authors also simplified the original problem into a minim-AoI-optimal path scheme due to complex constraints and proposed a reinforcement learning-based strategy for the solution. However, for incorporating the fresh information update by the UAV-BSs for the COC applications, we consider the high frequency Millimeter Wave (mmWave) wireless spectrum for enabling the backhaul communication between the UAV-BSs and ground BS which is captured in the energy efficiency metric design of the UAV-BSs.

III. SYSTEM MODEL

In Fig. 1, we consider a set of given trajectory points, $P = \{1, 2, \cdots, P\}$. The trajectory points in $P$ are covered by a set of battery-powered UAV-BS, $U = \{1, 2, \cdots, U\}$, which act as relay for a set of IoT devices, $I = \{1, 2, \cdots, I\}$. For simplicity, we consider the set of IoT devices are randomly located at different trajectory points. In this paper, we also consider a single ground station (i.e., ground base station) $b$ which is equipped with the multi-access edge computing (MEC) server (BS-MEC) and acts as the information fusion center that receives information updates from the IoT devices through the UAV-BS relays to support the computation oriented communications applications. Therefore, we assume that the BS-MEC is a point in the network where various communication resources are available to achieve a certain computational accuracy where the timely information updates from the different network sources are essential. Moreover, the BS-MEC $b$ is also considered as a trajectory point within the set $P$ where the set of neighboring trajectory points of $b$ (exclude itself) is denoted by $P_b = \{p \in P : (b, p) \in I\}$ where $I \subseteq P \times P$ is the set of trajectory links between the trajectory points. The UAV-BS $u \in U$ traverses within different trajectory points in $P$ over a finite observation time $T$. As UAV-BS $u \in U$ travels with a trajectory $p \in P$, it gathers the information data packets from the active IoT device $i \in I$ located near the trajectory point $p \in P$ using the uplink communication channel. Moreover, UAV-BSs $u \in U$ uses the backhaul communication link when the BS-MEC $b$ is within the transmission range of UAV-BS $u \in U$. As a result, BS-MEC can receive a fresh information update from different trajectory points and calculate the AoI metric. We assume that the BS-MEC is equipped with an array of mmWave directional antennas and provides a dedicated mmWave spectrum for backhaul communication for the UAV-BS. Moreover, the IoT devices and the UAV-BS are also equipped with directional antennas so that the IoT devices can transmit information updates to the UAV-BS using the non-mmWave spectrum. In this paper, we limit the scope by focusing on the deployment, communication, and navigation of the UAV-BSs $u \in U$ for the data relaying from different sources $i \in I$ at trajectory points $p \in P$ to the ground BS $b$. Also, for the user association between the IoT devices and the corresponding UAV-BSs at different trajectory points, we apply the default max-signal-to-interference-plus-noise-ratio (SINR) [33] based approach.

A. IoT-to-UAV-BS Communication Model

At the trajectory point $p \in P$, the air-to-ground path loss probability of the UAV-BS $u \in U$ with IoT devices $i \in I$ is calculated as [34],

$$
\xi_{i,u}^p = \begin{cases} 
\frac{1}{1 + \alpha \exp(-\hat{\alpha}(\frac{180}{\pi} \Theta_u - \alpha))}, & \text{LoS channel,} \\
1 - \frac{1}{1 + \alpha \exp(-\hat{\alpha}(\frac{180}{\pi} \Theta_u - \alpha))}, & \text{NLoS channel.}
\end{cases}
$$

(1)
Here \( \alpha \) and \( \hat{\alpha} \) are the environment dependent constants for the LoS and NLoS channels, respectively, where \( \Theta_u \) is the elevation angle of UAV-BS \( u \in \mathcal{U} \). The intuition of calculating the air-to-ground LoS and NLoS path loss probabilities using (1) is that, in urban/sub-urban environment, the uplink communication link between the UAV-BSs and the IoT devices may be hindered (i.e., multi-path fading) by the surrounding obstacles (e.g., buildings) unlike IoT devices deployed in the rural environment. In addition, the path loss in decibel (dB) is calculated as [35],

\[
P_{i,p}^u = \begin{cases} 
20 \log \left( \frac{4\pi f_c \delta_{i,p}^u}{c} \right) + \epsilon, & \text{LoS channel}, \\
20 \log \left( \frac{4\pi f_c \delta_{i,p}^u}{c} \right) + \bar{\epsilon}, & \text{NLoS channel}.
\end{cases}
\]

(2)

Here \( f_c \) is the uplink channel frequency, and \( \epsilon \) and \( \bar{\epsilon} \) are the attenuation factors for the LoS and NLoS channels, respectively. Using (2), the received signal power from the IoT device \( i \in \mathcal{I} \) at trajectory point \( p \in \mathcal{P} \) to UAV-BS \( u \in \mathcal{U} \) is calculated as,

\[
\hat{P}_{i,p}^u = \frac{\hat{P}_{i,p}^u}{P_{i,p}^u}.
\]

(3)

where \( \hat{P}_{i,p}^u \) is the transmit power of IoT device \( i \in \mathcal{I} \) for offloading the data to UAV-BS \( u \in \mathcal{U} \). At time \( t \), the received SINR for UAV-BS \( u \in \mathcal{U} \) with IoT device \( i \in \mathcal{I} \) at trajectory point \( p \in \mathcal{P} \) is calculated as,

\[
\gamma_{i,p}^u(t) = \frac{\hat{P}_{i,p}^u(t) (10^{\eta/10})^{-1}}{I_{i,p}^u + \sigma^2}.
\]

(4)

Here \( I_{i,p}^u = \sum_{p' \in \mathcal{P}} \sum_{i' \in \mathcal{I}, i \neq i'} \hat{P}_{i',p'}^u (10^{\eta/10})^{-1} \) is the received interference of UAV-BS \( u \in \mathcal{U} \) from the other UAV-BSs \( u' \in \mathcal{U}, u \neq u' \) which is serving IoT \( i' \in \mathcal{I}, i \neq i' \) that is located in different neighboring and overlapping trajectory points \( p' \in \mathcal{P}, p \neq p' \) and \( \sigma^2 \) is the noise power. Using (4), the channel capacity at time \( t \) is defined as,

\[
r_{i,p}^u(t) = \begin{cases} 
\beta_u \cdot \log \left( 1 + \gamma_{i,p}^u(t) \right), & \text{if } \gamma_{i,p}^u(t) > \gamma_{th}, \\
0, & \text{otherwise}.
\end{cases}
\]

(5)

Here \( \beta_u \) is the fixed non-mmWave uplink channel bandwidth that is equally distributed to the IoT devices \( \mathcal{I} \) at the trajectory point \( p \in \mathcal{P}, \gamma_{th} \) is the SINR threshold for ensuring successful uplink transmission between IoT devices and UAV-BSs.

### B. UAV-BS-to-BS Communication Model

The received power of the ground BS \( b \) from UAV-BS \( u \in \mathcal{U} \) is calculated as [36],

\[
P_{b,u} = P_{b,u}^{tx} \cdot G_{u}^{tx} \cdot G_{b}^{rx} \cdot \frac{c}{4\pi \delta_{b,u} f_{\text{mmWave}}}. 
\]

(6)

Here \( P_{b,u}^{tx} \) is the transmit power of UAV-BS \( u \in \mathcal{U} \) to BS \( b \), \( \delta_{b,u} \) is the distance between the UAV-BS \( u \) and ground BS \( b \), \( c \) is the speed of light, \( f_{\text{mmWave}} \) is the carrier frequency of the mmWave back-haul link, \( G_{u}^{tx} \) and \( G_{b}^{rx} \) are the antenna gains of the transmitter UAV \( u \in \mathcal{U} \) and receiver ground BS \( b \), receptively. At time \( t \), the back-haul capacity of the channel between UAV-BS \( u \in \mathcal{U} \) and ground BS \( b \) at time slot \( t \) is calculated as,

\[
r_{b,u}^{\text{mmWave}}(t) = \begin{cases} 
\beta_{b,u}^{\text{mmWave}} \cdot \log \left( 1 + \frac{\hat{P}_{b,u}^{\text{mmWave}}}{\beta_{b,u}^{\text{mmWave}} \sigma^2} \right), & \text{if } \delta_{u,b} \leq \hat{\alpha}, \\
0, & \text{otherwise}.
\end{cases}
\]

(7)

Here \( \beta_{b,u}^{\text{mmWave}} \) is the mmWave back-haul bandwidth and \( \sigma^2 \) is the additive noise. If the distance \( \delta_{u,b} = \sqrt{h_u^2 + (x_u - x_b)^2 + (y_u - y_b)^2} \) between the UAV-BS and the ground BS \( b \) at UAV-BS height \( h_u \) is less than a threshold distance \( \hat{\alpha} \), the UAV-BS transmits the information update to the ground base station using \( \beta_{b,u}^{\text{mmWave}} \). Using (7), the transmission energy of UAV-BS \( u \in \mathcal{U} \) while using back-haul link at time \( t \) is calculated as,

\[
E_{u}^{\text{mmWave}}(t) = P_{b,u}^{tx} \times r_{b,u}^{\text{mmWave}}(t).
\]

(8)

### C. UAV-BS Relay Network Energy Efficiency Metric Design

UAV-BS \( u \in \mathcal{U} \) covers the observation area horizontally at a constance altitude \( h_u \) where different UAV-BSs may maintain different altitudes. The assumption is practical for UAV-BSs according to the Federal Aviation Administration (FAA) regulations for small unmanned aircraft (UAS) operations. Moreover, the UAV-BS trajectory at time \( t \) is defined as, \( t_u(t) = (x_u(t), y_u(t))^T \in \mathbb{R}^{2 \times 1} \). Therefore, the time varying distance covered by UAV-BS \( u \in \mathcal{U} \) horizontally at constant altitude \( h_u \) from the current trajectory position to the next position (i.e., \( \hat{x}_u(t) \) and \( \hat{y}_u(t) \)) is defined as [37],

\[
\delta_t(t) = \sqrt{h_u^2 + (x_u(t) - \hat{x}_u(t))^2 + (y_u(t) - \hat{y}_u(t))^2}, \forall u \in \mathcal{U}, 0 \leq t \leq T.
\]

(9)

The total mobility energy cost of UAV-BS \( u \in \mathcal{U} \) for covering distance \( \delta_t(t) \) at time \( t \) is calculated as,

\[
E_u(t) = \delta_t(t) \times E_{\text{prop}}.
\]

(10)

Here \( E_{\text{prop}} = k_1 |v|^3 + k_2 |v|^2 (1 + |a|^2) \) is the upper bound of the propulsion power consumption where \( k_1 \) and \( k_2 \) depends on the UAV-BS design and \( g = 9.8 \text{ m/s}^2 \) is the gravitational acceleration [38].

The total energy efficiency for UAV-BS \( u \in \mathcal{U} \) that covers trajectory points \( \mathcal{P} \) to serve IoT devices in \( \mathcal{I} \) over time \( T \) is defined as,

\[
\eta(\mathcal{P}, u) = \frac{\sum_{t=1}^{T} \sum_{p=1}^{m} r_{b,u}^{\text{mmWave}}(t) + \sum_{i=1}^{m} r_{i,p}(t)}{E_{u}^{\text{mmWave}}(t) + E_u(t)}.
\]

(11)

### D. Age of Information Model for Ground Station

The AoI metric is used to measure the freshness of information collected by the UAV-BSs from the trajectory points in \( \mathcal{P} \) where the UAV-BSs act as relay node for the IoT devices.
Therefore, at the BS $b$, the AoI of the trajectory $p \in \mathcal{P}$ at time $t$ is calculated as,

$$\Delta_u(p, t) = t - \Delta'_u(p, t), \quad \forall p \in \mathcal{P}.$$  \hspace{1cm} (12)

Here $\Delta'_u(p, t)$ is denoted as the time-stamp of the most recent received data packet from the trajectory point $p \in \mathcal{P}$ at the base station $b$ by the UAV-BS $u$. Assume that, at $t$, the information update from the trajectory way-point $p \in \mathcal{P}$ is the most recent. Then, the Age of Information (AoI) associated with the trajectory way-point $p \in \mathcal{P}$ at time $t$ is given by (12).

While the base station $b$ does not receive new information update from the trajectory way-point $p \in \mathcal{P}$ at time $t$ which in turn indicates the fact that information getting older. As soon as the base station $b$ receives new information update from the trajectory way-point $p \in \mathcal{P}$, the corresponding time-stamp is instantaneously updated from $\Delta_u(p, t)$ to $\Delta'_u(p, t + 1)$, reducing the value of $\Delta_u(p, t), \forall p \in \mathcal{P}$ by $\Delta'_u(p, t + 1) - \Delta'_u(p, t)$. In other words, at the moment $t + 1$ the information update from $p \in \mathcal{P}$ occurs at the base station $b$ by the UAV $u \in \mathcal{U}$, the value of $\Delta_u(p, t), \forall p \in \mathcal{P}$ matches the delay of the information update. At $t = 0$, we assume $\Delta_u(p, 0) = 0$ where we adopt the just-in-time transmission policy [39]. The average AoI is calculated at the base station $b$ for trajectory points $\mathcal{P}$ over time slot $T$ as,

$$\hat{\Delta}_b(\mathcal{P}) = \frac{1}{T|\mathcal{P}|} \sum_{t=1}^{T} \sum_{p \in \mathcal{P}} \Delta_u(p, t).$$ \hspace{1cm} (13)

IV. PROBLEM FORMULATION

To formulate the UAV-BS navigation optimization problem under contextual constraints (i.e., trajectory, AoI, energy efficiency constraints), first, we consider each UAV-BS $u \in \mathcal{U}$ can cover only a sub-set of trajectory points in a given time window $T$. Moreover, we consider sub-sets for each $u \in \mathcal{U}$ comprised of trajectory points which are denoted as, $\mathcal{P}_u \subset \mathcal{P}, \mathcal{P}_u \cap \mathcal{P}_{u'} = \emptyset$ where $u \neq u'$. As a result, the objective of energy efficient UAV-BS navigation optimization problem is to find the cooperative trajectory path configuration of the UAV-BSs that maximizes the total energy efficiency of the UAV-BSs relay network subject to the energy and AoI metric. Therefore, the optimization problem formulation is represented as follows,

$$\begin{align*}
\max_{\{\mathcal{P}_u\}_{u \in \mathcal{U}}} & \quad \sum_{u \in \mathcal{U}} \eta(\mathcal{P}_u, u), \quad (14) \\
\text{subject to} & \quad \bigcap_{u \in \mathcal{U}} \mathcal{P}_u = \{b\}, \quad \forall u \in \mathcal{U}, \quad (15) \\
& \quad \bigcup_{u \in \mathcal{U}} \mathcal{P}_u = \mathcal{P}, \quad \forall u \in \mathcal{U}, \quad (16) \\
& \quad \eta(\mathcal{P}_u, u) \geq \eta_{bh}, \quad \forall u \in \mathcal{U}, \quad (17) \\
& \quad \hat{\Delta}_b(\mathcal{P}_u) \leq \hat{\Delta}^b_{bh}, \quad \forall p \in \mathcal{P}_u \backslash \{b\}. \quad (18)
\end{align*}$$

In the above formulated problem, the constraints (15)-(18) are the trajectory, energy efficiency, and AoI constraints, respectively. Constraint (15) indicates non-overlapping trajectories of the UAV-BSs except the ground BS trajectory point where the information update occurs. Constraint (16) indicates the joint trajectory configuration of the UAV-BSs where all the trajectory points are covered interdependently. Constraints (17) and (18) are coupled with the decision variable $\mathcal{P}_u$ where both the energy efficiency and AoI metric are the functions of $\mathcal{P}_u$. Constraint (17) ensures the total energy efficiency of the UAV-BSs where the communication and mobility energy should be greater than a minimum energy efficiency threshold $\eta_{bh}$. Finally, constraint (18) ensures the average freshness of information updates by configuration $\mathcal{P}_u$ should be less than an AoI threshold $\hat{\Delta}^b_{bh}$. Due to constraint (18) in problem (14), the UAV-BSs jointly navigate different trajectory points under not only the energy efficiency constraint but also the AoI constraint where the performance of the computation oriented communication applications depend on the up-to-date information update from different trajectory points. Therefore, problem (14) is different from the traditional energy efficiency maximization problem for UAV navigation.

The decision problem in (14) can be reduced to a base problem of vertex cover problem (i.e., Maximum Clique Problem) [40] with the corresponding constraints (15)-(18), which is NP-Complete. Similar to the maximum clique problem, problem (14) is combinatorial in nature. Moreover, there is no known polynomial algorithm that can tell, given a solution of (14), whether it is optimal. As a result, we can infer that the decision problem in (14) belongs to the same category of the problem of the vertex cover problem, which is proven to be NP-hard. In the next section, we solve problem (14) with the corresponding constraints (15)-(18) by using a deep Q learning technique.

V. PROPOSED TRAJECTORY POLICY ALGORITHM BASED ON DEEP Q-LEARNING

To solve problem (14), we apply a deep reinforcement-learning model which is the combination of a deep neural network and a reinforcement learning algorithm. Specifically, the proposed DQN approach is comprised of three components: (i) a deep neural network to reduce the dimension of the state space that is used to extract the contextual features (e.g., AoI, energy consumption) necessary for UAV-BS navigation, (ii) An experience replay memory to store the state transitions that the UAV-BS agents observe, and (iii) An reinforcement learning (RL) framework to find the best trajectory policy that achieves the objective of problem (14) with the corresponding constraints (15)-(18). Unlike the state-of-the-art method such as control methods, the DQN does not need a network dynamic model as it is model free. Moreover, in the proposed approach, the use of experience replay ensures stability by breaking the temporal dependency among the observations used in the training of the deep neural network.

In Section V(A), first, we model the state and action space of problem (14). After that, in Section V(B), we model the reward and control policy based on problem (14) with the corresponding constraints. Finally, in Section V(C), we provide the proposed training and testing model for UAV-BS trajectory policy.
A. State and Action Space

The state space for trajectory policy of the UA-VBS is a four-dimensional state space. At each time step $t = \{1, 2, \cdots, T\}$, the state or joint observation space of the virtual learning agent (i.e., deployed at the ground BS) is denoted by, $S = \{s_t = (p_{\text{current}}^u, p_{\text{end}}^u, \eta, \Delta)|\eta \in [0, \eta_{\text{th}}], \Delta \in [1, \Delta_{\text{th}}]\}$ where $p_{\text{current}}^u$ is the current location of UA-VBS $u \in U$ at individual heights $h_u$, $p_{\text{end}}^u$ is the target position, $\eta$ is the average energy efficiency of UA-VBSs, and $\Delta$ is the average age for the navigation optimization. Moreover, the trajectory position for navigation of the UA-VBS $u \in U$ is comprised of $x_u \in [0, X_u]$ and $y_u \in [0, Y_u]$, where $X_u$ and $Y_u$ are the maximum coordinate of a particular geographic location. Furthermore, the initial position of each of the UA-VBSs is randomly assigned for each trial along with the number of IoT devices. The lower and upper bounds of continuous state variables $\eta$ and $\Delta$ in the state space are calibrated from the real-world trajectory data.

The action space of the UA-VBSs is the trajectory planning each of the UA-VBS’s navigation from one feasible state (i.e., position) to the next state while satisfying the trajectory and communication constraints (i.e., constraints (15)-(18) of problem (14)). The learning agent selects an action $a_t$ from the set of available actions upon state $s_t$ where $a_t \in A_{s_t} \subset A$, and $A = \{a_1, \cdots, a_T\} = \{P_{\text{valid}}\}_{u \in U}$ is the configurations of the UA-VBS navigation. An example of the action space design is provided in Appendix A.

B. Reward and Control Policy

When a learning agent implements action $a_t$, the environment moves to a new state $s_{t+1}$ and the immediate reward $R_{t+1}$ with the transition $(s_t, a_t, s_{t+1})$ is associated and the learning agent receives the reward through feedbacking. In other words, at each state transition, the agent receives the immediate reward which is used to form the trajectory control policy for navigation. For future usage, the control policy is used by the learning agent that maps the current state to optimal control action. The immediate reward is formulated by the instantaneous energy efficiency metric of the UA-VBS’s and defined as follows,

$$R_t(a_t) = \begin{cases} a_1 \eta(a_t), & \text{if constraints (15)-(18) of (14) are true}, \\ -a_1, & \text{if constraints (15)-(17) of (14) are violated}, \\ 0, & \text{otherwise}. \end{cases}$$

(19)

Here $a_t$ is a coefficient multiplied to the energy efficiency function and also used to penalize the agent when the trajectory constraints are violated. However, when constraint (18) of (14) is violated, the system receives zero reward as the information update from the trajectory waypoints becomes outdated for the COC applications.

The objective of the learning agent over $T$ time slots is, therefore, to maximize the future reward which is defined as,

$$\hat{R}(s, a; t) = \sum_{t_0=0}^{T} \gamma \times R_{t-t_0}(a_t),$$

(20)

Here $\gamma \in [0, 1]$ reflects the trade-off between the importance of immediate and future rewards. The reward $R_t(a_t)$ changes by the energy efficiency metric of the UA-VBSs over the duration of $T$ due to the joint movement (i.e., action) from one trajectory way-point to the next way-point in different time slots. Therefore, $R_{t-t_0}(a_t)$ is the difference of rewards between two time slots. Moreover, the reward function (20) is obtained at time $t$ after learning the current state of the UA-VBSs over the last $T$ time steps duration. Therefore, we define a control policy as $\pi$ for the agent where the $Q$-function or the action-value function is defined as,

$$Q^\pi(s, a) = \hat{R}(s, a) + \gamma \sum_{s' \in S} P_{s,s'} \sum_{a' \in A} \pi(a' | s') Q^\pi(s', a').$$

(21)

Here $P_{s,s'}$ is the transition probability of the states in the environment where $s' = s_{t+1}$, $\pi$ is the control policy, and action $a$ is enforced through the environment simulator. Here, the state and reward update is based on the information received by the ground BS $b$. The detailed description of $Q$-function and the theoretical background on obtaining the best control policy $\pi^{\text{opt}}$ using (21) for UA-VBSs trajectory optimization are discussed in Appendix B.

C. Training With Experience Replay

The proposed DQN approach learns how to optimally control the trajectory configurations of the UA-VBSs for navigation during the simulation. Therefore, it is vital for the simulation process to train a Q-network where the target value for each trajectory observation environment state is given as,

$$y_k = \hat{R}(s, a) + \gamma \max_{a'} Q(s', a' ; \theta_k).$$

(22)

Here we introduce $\theta_k$ which is the network weight obtained by the training during the $k^{th}$ iteration. Hence, using (22), the loss function of the training network is designed as,

$$L_k(\theta_k) = \mathbb{E}_{(s,a) \sim \rho_{(s,a)}} \left[ y_k - Q(s, a; \theta_k) \right]^2.$$  

(23)

Here $\rho(s, a)$ is the probability distribution over the sequences $s$ and actions $a$, $y_k$ is the target value of the training network which is derived from (22), and the optimal network weights $\theta^{\text{opt}}$ are obtained by training. Furthermore, to enhance and stabilize the training of the DQN, we apply the mini-batch method to randomly collect examples from all the training episode steps $e_t = (s_t, a_t, R_{t}(a_t), s_{t+1})$ in a fixed size replay memory $M_t = \{e_1, \cdots, E\}$. As a result, one sample is used multiple times in the training that improves the data efficiency significantly. Therefore, using (22), the loss function in (23) is represented with a uniform distribution over $M$ as,

$$L(\theta) = \mathbb{E}_{(s,a,s') \sim U(M)} \left[ (R + \gamma \max_{a'} Q^{\text{opt}}(s', a' ; \theta^-) - Q(s, a; \theta))^2 \right].$$

(24)
Here $U(\mathcal{M})$ is the uniform distribution over the experience replay memory $\mathcal{M}$ and $\theta^-$ is the stored weight parameters of the target DQN network. The loss function (24) is then updated using the gradient descent method as,

$$\nabla_{\theta_k} \mathcal{L}_k(\theta_k) = \mathbb{E}_{(s, a, r, s') \sim U(\mathcal{M})} \left[ (R + \gamma \max_{a'} Q^\text{opt}(s', a'; \theta^-) - Q(s, a; \theta_k)) \nabla_{\theta_k} Q(s, a; \theta_k) \right]. \quad (25)$$

In step 1 of Alg. 1, we first initialize the network parameters randomly and introduce the target DQN with the same network structure as the original DQN network (lines 1-2 in Alg. 1). Then, at each training episode steps $e_i$, the energy efficiency metric is calculated using (11) considering the navigation and communication parameters in ((11)-(11)) (line 7 in Alg. 1). As a part of the $\epsilon$-greedy policy framework, in the exploration stage, the energy efficiency metric is used to calculate the reward function where the action is derived from the current DQN with the exploration probability $\epsilon$ (line 8 in Alg. 1). The reward function is observed considering the AoI context along with other constraints in problem (14) and transit to the next state $s'_t$ where $t' = t + 1$ (line 9 in Alg. 1). Moreover, the exploration stage enables the UAV-BSs to explore all the joint actions for achieving the better reward values that lead toward choosing the appropriate action with the highest energy efficiency. Subsequently, we adopt the mini-batch approach that shuffles the experience from the replay memory buffer at random to remove the correlation in the observation sequence, and thus, smoothing the changes in the observation data distribution (lines 10-11 in Alg. 1). To train the DQN, we adopt the stochastic gradient descent (SGD) algorithm using the training loss function and update the network parameter $\theta$, and network bias (lines 12-13, in Alg. 1). The training process stops when the UAV-BSs arrive at the terminal trajectory and the DQN network is finally stored for testing (line 14 in Alg. 1).

### VI. PERFORMANCE EVALUATION

In this section, we first address the performance analysis experiment environment through various key metrics. We then describe the outcomes obtained from the experiment and finally provide an in-depth discussion and key observations from the results of the simulation. In order to train the deep neural network (DNN), we consider a neural network architecture with two fully connected (FC) hidden layer with 100 hidden nodes. We also set the experience replay memory size, $M = 200$. The simulation results are obtained by averaging and normalizing the values over 100 episodes.

#### A. Experiment Setting

For the performance evaluation of the proposed approach, we consider the simulation settings which is summarized in Table I. In addition, we compare the proposed approach with two baseline approaches which are,

- **Baseline DQN**: The structure of the baseline DQN is different from the proposed DQN with experience replay approach in terms of not having the experience replay memory.

- **Greedy**: In case of the greedy approach, at each timeslot $t$ in an episode, each UAV-BSs $u \in \mathcal{U}$ co-operatively finds the trajectory paths for navigation that may provide the maximum immediate reward. In addition, the approach applies penalty for violating the system constraints in order to make fair comparison between the proposed and the baseline DQN approaches.

- **Multiple Travelling Salesman Problem (mTSP)**: In mTSP [41], the UAV-BSs cooperatively choose a set of trajectory way-points that is the minimum of total
TABLE I SIMULATION SETTINGS

<table>
<thead>
<tr>
<th>Simulation Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of UAV-BS</td>
<td>3</td>
</tr>
<tr>
<td>No. of MEC-BS</td>
<td>1</td>
</tr>
<tr>
<td>No. of IoT devices</td>
<td>100</td>
</tr>
<tr>
<td>No. of Trajectory points</td>
<td>[6, 14]</td>
</tr>
<tr>
<td>Max UAV-BS height</td>
<td>[140, 250] (m)</td>
</tr>
<tr>
<td>Maximum UAV-BS velocity and acceleration</td>
<td>100 m/s and 5 m/s²</td>
</tr>
<tr>
<td>Radius of UAV-BS</td>
<td>300 (m)</td>
</tr>
<tr>
<td>f_momwave</td>
<td>28 GHz [36]</td>
</tr>
<tr>
<td>β_momwave</td>
<td>20 × 100 MHz [36]</td>
</tr>
<tr>
<td>f_c, β_a</td>
<td>2 GHz, 20 MHz [36]</td>
</tr>
<tr>
<td>P_f, σ, γ, t</td>
<td>20 dBm [36], −100 dBm, 5 dB</td>
</tr>
<tr>
<td>For urban scenario, α, δ, ε, τ, k, λ</td>
<td>9.61, 0.16, 1.20, 9.26 × 10⁻⁴, 2250 [36]</td>
</tr>
<tr>
<td>Normalized AoI threshold Δₖ, α₂</td>
<td>[0.3, 0.9, 1]</td>
</tr>
</tbody>
</table>

B. Convergence Analysis of the Neural Network

In Fig. 2, we observe several statistical characteristics of the reward function. The asymptotic slope depicts the effectiveness of the trajectory policy of the UAVs after the proposed approach with the reward set up in (19) has stabilized. Moreover, the minimum point of the curve (i.e., timeslot = 9) shows how much reward must be sacrificed before the proposed approach starts to improve. This reason behind this phenomenon is, when the UAVs start from a fixed point, all the UAVs navigate randomly and may easily choose less energy-efficient or overlapping trajectory way-points that result in an increased amount of penalty. As the accumulated reward becomes positive after the timeslot 10, the proposed reward setup shows increased cumulative reward. This indicates how long the UAV-BSs jointly take until the proposed approach has recouped the cost of learning the navigation environment. It is noteworthy to mention, the minimum and the zero-crossing points in Fig. 2 indicate the balanced and reasonable behavior of the proposed algorithm with the reward design in (19) where we consider both positive and negative rewards (i.e., penalty). Since the cumulative reward is measured by considering the total reward, the proposed algorithms optimize the discounted reward at each step until the last timestep and thus, converge to a feasible solution (i.e., feasible trajectory policy).

C. Finding Appropriate Discount Factor and AoI Threshold

We present the experimental results for finding the appropriate discount factor (i.e., γ) and AoI threshold (i.e., Δₖ) for the proposed DQN with replay memory approach. Here, we first find the discount factor while considering the energy efficiency metric, and then we fix the discount factor to find the appropriate AoI threshold concerning the trade-off between the average AoI and average rewards metrics.

In Table II and III, we evaluate the appropriate discount factor and AoI threshold considering the average energy efficiency metric and we set the number of trajectory points, |P| = 14. In Table II, we observe how the discount factor affects energy efficiency. More specifically, when the discount factor increases in the case of the proposed DQN with replay memory, the energy efficiency metric goes up at 1.73% at the discount factor is γ = 0.5. On the other hand, the energy efficiency metric for the baseline DQN slightly drops by 0.23% at discount factor γ = 0.5 and the trend continues until discount factor γ = 0.6. Similarly, the energy efficiency metric for the proposed approach also slightly fluctuates at discount factor γ = 0.6. However, up to this point, the proposed DQN with replay memory still outperforms the baseline DQN approach. At γ = 0.7, the energy efficiency metric is at the peak for the proposed DQN with replay memory and the performance of the baseline DQN also improves significantly. However, with a higher value of γ, the performance of both the approach decreases drastically. The explanation behind such phenomenon is that with a lower discount factor value (i.e.,
levels of AoI threshold which are between the two performance metrics by setting four different levels of AoI threshold that are $\Delta_{th}^{\hat{1}} = [0.3, 0.5, 0.7, 0.9]$. At $\Delta_{th}^{\hat{1}} = 0.3$, we observe that with strict AoI threshold, the normalized average reward is at the minimum whereas the AoI is at the peak point (disregarding the normalized average AoI metric value at $\Delta_{th}^{\hat{1}} = 0.9$). This is normal because the strict AoI threshold leads to an increasing penalty, and therefore, the performance of the system in terms of average cumulative reward decreases. As we relax the AoI threshold (i.e., $\Delta_{th}^{\hat{1}} > 0.5$), the average AoI decreases significantly and the average reward increases. However, at $\Delta_{th}^{\hat{1}} = 0.9$, the reward value is the maximum but at this point the effect of the AoI threshold is trivial. More specifically, the system at this point the UAV-BSs operate disregarding the freshness of information updates at the ground BS which is not desirable. Therefore, we set the acceptable AoI threshold at $\Delta_{th}^{\hat{1}} = 0.7$ to balance between the average reward and the AoI metric.

**D. Experiment Results**

In Fig. 3, we analyze the performance of the proposed DQN with a replay memory with two baseline approaches in terms of average cumulative reward with the increasing number of trajectory way-points. With a small number of trajectory points (i.e., up to $|P| = 8$), the performance of the approaches is not distinct since the density of the way-points in the geographic environment is less. However, as the number of trajectory points increases, the performance gaps between the approaches increase significantly. More specifically, the proposed DQN with the replay memory approach outperforms the greedy and baseline DQN correspondingly up to 2.84% and 1.91%. The performance gaps between the approaches further increase with dense trajectory way-points (i.e., $|P| = 14$) where the proposed DQN with replay buffer outperforms the greedy and the baseline DQN by 5.08% and 4.02%, respectively.

One of the key factors in the proposed model is to consider minimizing the AoI metric while finding a policy for UAV-BSs navigation. Therefore, in Fig. 4, we evaluate the performance of the proposed DQN with replay memory with the greedy and baseline DQN approaches. As we can see from Fig. 4 that the proposed approach outperforms the greedy and baseline DQN by reducing the average AoI correspondingly up to 1.21% and 1.17%. The performance gaps between different approaches increase gradually up to $|P| = 8$. At $|P| = 10$, the performance of the approaches slightly fluctuates since in the experiment we independently run the simulations with the different numbers of trajectory points. Therefore, this has some impacts on the experiment results. However, the trend of the proposed approach to reduce the AoI metric than the baseline approaches continues to carry on as the number of trajectory way-points increases. At $|P| = 14$, the performance gaps among the proposed DQN with replay memory, greedy, and baseline DQN approaches are increased by 0.72% and 1.29%, respectively.

In Fig. 5, we compare the performance of the proposed DQN with replay memory with the baseline approaches concerning the average energy efficiency. We observe that, the proposed DQN with replay memory gradually performs better than that of the baseline approaches. When the number of trajectory way-points is relatively high (i.e., $|P| = 10$), the proposed DQN with replay memory is proven to be slightly energy efficient by 0.32% and 1.97% than the greedy and baseline DQN, respectively. On the other hand, the mTSP performs worst compare to all the approaches. The reason
is, with the higher number of trajectory way-points, the average energy consumption of the UAV-BSs in case of mTSP increases significantly as the trajectory space of the UAV-BSs increases. At each time-slot $t$, each UAV-BS must navigate from one trajectory way-point to the neighboring way-point to serve IoT devices which incurs energy cost for navigation and the battery drains much quickly compare to all the other approaches. However, the proposed approach is more energy efficient with the higher number of trajectory way-points (i.e., $|P| = 14$) where the proposed approach outperforms the greedy, baseline DQN, and mTSP correspondingly up to 3.6%, 3.13%, and 7.87%.

The front-haul capacity is limited with an increasing number of IoT devices at the trajectory points and therefore, the bandwidth should be utilized efficiently. In Fig. 6, we compare the efficacy of the proposed approach with the greedy and baseline DQN with the varying number of trajectory points and different IoT device density. The proposed DQN with replay buffer efficiently utilizes the front-haul and back-haul bandwidth while traversing across different trajectory waypoints. The average bandwidth efficiency is quite similar when the number of trajectory points is less dense in the environment and the distance between the points is large. Therefore, the received interference level at the IoT devices which are served by different UAV-BS is significantly less in all the approaches. However, with a lightly dense trajectory way-point network with the increasing number of IoT devices (i.e., $|P| = 10$), all the approaches face interference and we observe a slight decrease in bandwidth efficiency. Nevertheless, the proposed DQN with replay buffer still outperforms the greedy and baseline DQN by 2.41% and 2.87%, respectively in terms of ensuring bandwidth efficiency.

Fig. 7 illustrates the utilization of the UAV-BSs or network resources under the proposed approach and the other two baseline approaches. As the number of trajectory way-points increases, the number of IoT devices using the network resources of the UAV-BSs also increases due to the increased number of associations per UAV-BSs. For a fixed number of UAV-BSs (i.e., $|\mathcal{U}| = 3$), the IoT devices at different trajectory way-points tend to utilize the maximum network resource provided by the UAV-BSs. However, since the proposed DQN with replay buffer covers the trajectory way-points more efficiently than that of the baseline approaches, the network resource provided by the UAV-BSs are utilized 9.26% and 4.71% more efficiently compared to the greedy and baseline DQN.

Fig. 8 depicts the average energy efficiency of the UAV-BSs that operate under different AoI threshold values.
that with relatively relaxed AoI threshold $\Delta_{th}^1 = 0.7$ the energy efficiency of the UAV-BSs using the proposed approach is 1.63% and 0.95% more energy efficient than that of the greedy and baseline DQN, respectively. The performance gap between the proposed approach and the baseline DQN is relatively close because we use the same discount factor $\gamma = 0.7$. However, the performance gap between the proposed approach and the greedy approach is significant over all the threshold values.

E. Discussion
From the experiment results described above, we find some important observations to prove the efficacy of the proposed approach than that of the baseline approaches. The in-depth discussion on the experiment results can be summarized as below,

- The proposed approach can significantly enhance the UAV-BS trajectory decision where unlike the baseline DQN, the proposed approach can effectively store the transition (i.e., experience) of different environment states to reuse the transition data by random sampling. This stabilizes and improves the DQN training which eventually leads to better trajectory policy that considers the energy consumption of the UAV-BSs, data freshness and bandwidth utilization.
- We can observe from Table II and III that the objective of the UAV-BSs is largely dependent on the setting of the appropriate discount factor and the AoI threshold. Especially, the appropriate discount factor value can effectively enhance the training of the DQN network by not only providing better convergence but also giving a chance of improving the training of the DQN network which has a replay memory.
- The greedy approach sometimes performs better than that of the proposed DQN with replay memory and baseline DQN. However, the performance gain is limited to a small network and in the case of a large network, the proposed DQN with replay memory significantly outperforms the greedy approach.

VII. Conclusion
In this paper, we focused on developing the UAV-BS navigation policy to improve data freshness and accessibility to the IoT network. As a result, we have introduced an agile deep learning reinforcement with an experience replay model that is well-suited to solving the energy-efficient UAV-BS navigation problem under trajectory and AoI constraints. We also performed a comprehensive simulation study to determine appropriate system parameters with the applicable discount factor and AoI efficiency metric threshold to empower the learning model. The simulation results show a strong correlation between energy efficiency and AoI thresholds whereby setting the proper threshold values can effectively enhance the energy efficiency and data freshness for the COC applications. The simulation findings also confirmed the effectiveness of the proposed DQN with experience replay memory under different network conditions.

APPENDIX A
EXAMPLE OF THE ACTION SPACE DESIGN
For example, suppose there are six trajectory way-points in $P = \{1, 2, 3, 4, 5, 6\}$ and two UAV-BSs which are, $u_1$ and $u_2$. The starting points are random for UAV-BSs and therefore, assume that the starting points for $u_1$ and $u_2$ are 1 and 2, respectively. In this example, also assume that the ground base station is located in 6 and all the UAV-BSs must reach that location to perform the information update. Therefore, the goal of the UAV-BSs are fixed at a location 6. Now, the action space $A$ is defined as the combination of the trajectory points in $P$.

Especially, the appropriate discount factor value can sharply enhance the training of the DQN network by not only providing better convergence but also giving a chance of improving the training of the DQN network which has a replay memory.

APPENDIX B
OBTAINING OPTIMAL CONTROL POLICY $\pi^{opt}$

The Bellman Equation is used to find optimal policies and value function which gives maximum value compared to all other value functions. In other words, the value of state $s$ is the reward the agent got upon leaving that state, plus the discounted value of the state the agent landed upon, $s'$, multiplied by the transition probability that the agent will move into. The Bellman Equation for value function is defined as,

$$V(s) = \mathbb{E}[R_{t'} + \gamma V(s_{t'} | s_t = s)], \quad (26)$$

where $t' = t + 1$. However, using (1), in the Bellman Expectation Equation for value function (i.e., state-value function) the agent finds the value of a particular state subjected to some policy $\pi$ which is defined as,

$$V^\pi(s) = \mathbb{E}^\pi[R_{t'} + \gamma V^\pi(s_{t'} | s_t = s)] \quad (27)$$

Similarly, the state-action value function (Q-function) is defined as,

$$Q^\pi(s, a) = \mathbb{E}^\pi[R_{t'} + \gamma V^\pi(s_{t'}, a_{t'}) | s_t = s, a_t = a] \quad (28)$$

So, when the agent averages the Q-values, the value of being in a state is defined as,

$$V^\pi(s) = \sum_{a \in A} \pi(a | s) Q^\pi(s, a). \quad (29)$$

The equation (27) provides a connection between the state-value function and the state-action function. However, the agent needs to evaluate the goodness of taking an
action \(a\). Therefore, we calculate the goodness of an action \(a\) in a state \(s\) as,
\[
Q^\pi(s, a) = \hat{R}(s, a) + \gamma \sum_{s' \in S} P(s, s') V^\pi(s'),
\]
where \(Q^\pi(s, a)\) is the value function.

Here, \(P\) is the transition probability. Finally, the effectiveness of a particular action \(a\) following a particular policy \(\pi\) is defined as (21). Using (21), the goal of our model is to obtain the best control policy \(\pi_{opt}\). Therefore, the maximum \(Q\)-function is defined as,
\[
Q^\pi_{opt}(s, a) = \mathbb{E}[\hat{R}(s, a) + \gamma \max_{a'} Q^\pi_{opt}(s', a') | s, a],
\]
where the discounted cumulative state function is,
\[
V^\pi_{opt}(s) = \max_{a'} \{Q^\pi_{opt}(s, a')\}.
\]

To derive the optimal control policy \(\pi_{opt}\), the \(Q\)-function is updated as,
\[
Q_{t+1}(s, a) = Q_t(s, a) + \psi(\hat{R}(s, a) + \gamma \max_{a'} Q_t(s', a')) - Q_t(s, a),
\]
where \(t' = t + 1\) and \(a = a_{t+1}\) where the \(Q\)-function is updated using the recursive mechanism and \(\psi\) is the learning rate.

**References**


Sarder Fakhrul Abedin (Member, IEEE) received the B.S. degree in computer science from Kristianstad University, Kristianstad, Sweden, in 2013, and the Ph.D. degree in computer engineering from Kyung Hee University, South Korea, in 2020. He served as a Post-Doctoral Researcher with the Department of Computer Science and Engineering, Kyung Hee University. He is currently serving as a Post-Doctoral Researcher with Mid Sweden University, Sweden. His research interests include the Internet of Things (IoT) network management, fog computing, machine learning, Industrial 5G, and wireless networking.

Md. Shirajum Munir (Graduate Student Member, IEEE) received the B.S. degree in computer science and engineering from Khulna University, Bangladesh, in 2010. He is currently pursuing the Ph.D. degree in computer science and engineering with Kyung Hee University, South Korea. He served as a Lead Engineer with the Solution Laboratory, Samsung Research and Development Institute, Bangladesh, from 2010 to 2016. His research interests include the Internet of Things (IoT) network management, fog computing, mobile edge computing, software-defined networking, smart grid, and machine learning.

Nguyen H. Tran (Senior Member, IEEE) received the B.S. degree in electrical and computer engineering from Ho Chi Minh City University of Technology in 2005, and the Ph.D. degree in electrical and computer engineering from Kyung Hee University, South Korea, in 2011. Since 2018, he has been with the School of Computer Science, The University of Sydney, where he is currently a Senior Lecturer. He was an Assistant Professor with the Department of Computer Science and Engineering, Kyung Hee University, from 2012 to 2017. His research interests include applying analytic techniques of optimization, game theory, and stochastic modeling to cutting-edge applications, such as cloud and mobile edge computing, data centers, heterogeneous wireless networks, and big data for networks. He received the Best KUH Thesis Award in Engineering in 2011 and the Best Paper Award at IEEE ICC 2016. He has been an Editor of IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING since 2016, and served as an Editor for the 2017 Newsletter of Technical Committee on Cognitive Networks on Internet of Things.

Zhu Han (Fellow, IEEE) received the B.S. degree in electronic engineering from Tsinghua University in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, in 1999 and 2003, respectively.

From 2000 to 2002, he was a Research and Development Engineer with JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor at Boise State University, Boise. He is currently a John and Rebecca Moores Professor with the Electrical and Computer Engineering Department and the Computer Science Department, University of Houston, TX, USA. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. He received the NSF CAREER Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the Best Paper Award for the EURASIP Journal on Advances in Signal Processing in 2015, the IEEE Leonard G. Abraham Prize in the field of Communications Systems (Best Paper Award in IEEE JSAC) in 2016, and several best paper awards in IEEE conferences. He was an IEEE Communications Society Distinguished Lecturer from 2015–2018, a fellow of AAAS since 2019, and a Distinguished Member of ACM since 2019. Since 2017, he has been 1% Highly Cited Researcher according to Web of Science. He is also the winner of 2021 IEEE Kiyo Tomiyasu Award, for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation: "for contributions to game theory and distributed management of autonomous communication networks."

Choong Seon Hong (Senior Member, IEEE) received the B.S. and M.S. degrees in electronic engineering from Kyung Hee University, Seoul, South Korea, in 1983 and 1985, respectively, and the Ph.D. degree from Keio University in 1997. In 1988, he joined KT, where he was involved in broadband networks, as a Member of Technical Staff. In 1993, he joined Keio University, Japan. He was with the Telecommunications Network Laboratory, KT, as a Senior Member of Technical Staff and as the Director of the Networking Research Team until 1999. Since 1999, he has been a Professor with the Department of Computer Engineering, Kyung Hee University. His research interests include future internet, ad hoc networks, network management, and network security. He is a member of the ACM, IEICE, IPSJ, KIISE, KICS, KIPS, and OSIA. He has served as the general chair, TPC chair/member, or an organizing committee member for international conferences, such as NOMS, IM, APNOMS, E2EMON, CCNC, ADMSN, ICPP, DIM, WISA, BeCN, TINA, SAINT, and ICOIN. He is currently an Associate Editor of the IEEE Transactions on Network and Service Management, International Journal of Network Management, and the JOURNAL OF COMMUNICATIONS AND NETWORKS, and an Associate Technical Editor of IEEE Communications Magazine.