IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

INNOVATIVE TRENDS IN 6G ECOSYSTEMS

Received 28 January 2024, accepted 15 March 2024, date of publication 19 March 2024, date of current version 27 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3379323

RESEARCH ARTICLE

Cache Sharing in UAV-Enabled Cellular Network: A Deep Reinforcement Learning-Based Approach

HAMIDULLAH MUSLIH[1], S. M. AHSAN KAZMI[2], MANUEL MAZZARA[1], AND GASPARD BAYE[3]

[1]Institute of Software Development and Engineering, Innopolis University, 420500 Innopolis, Russia
[2]Department of Computer Science and Creative Technologies, University of the West of England, BS16 1QY Bristol, U.K.
[3]Department of Computer and Information Science, University of Massachusetts Dartmouth, North Dartmouth, MA 02747, USA

Corresponding author: Hamidullah Muslih (h.muslih@innopolis.university)

ABSTRACT Caching content at base stations has proven effective at reducing transmission delays. This paper investigates the caching problem in a network of highly dynamic cache-enabled Unmanned Aerial Vehicles (UAVs), which serve ground users as aerial base stations. In this scenario, UAVs share their caches to minimize total transmission delays for requested content while simultaneously adjusting their locations. To address this challenge, we formulate a non-convex optimization problem that jointly controls UAV mobility, user association, and content caching to minimize transmission delay time. Considering the highly dynamic environment where traditional optimization approaches fall short, we propose a deep reinforcement learning (RL)-based algorithm. Specifically, we employ the actor-critic-based Deep Deterministic Policy Gradient (DDPG) algorithm to solve the optimization problem effectively. We conducted extensive simulations with respect to different cache sizes and the number of associated users with their home UAVs and compared our proposed algorithm with two baselines. Our proposed solution has demonstrated noteworthy enhancements over the two baseline approaches across various scenarios, including diverse cache sizes and varying numbers of users associated with their respective home UAVs.

INDEX TERMS 5G, 6G, cache sharing in UAVs, wireless networks, UAV-to-UAV communication.

I. INTRODUCTION

Wireless networks face serious difficulties due to the widespread use of the Internet of Things (IoT). As the International Telecommunication Union (ITU) report states, wireless data traffic in 2030 will be 10,000 times higher than in 2010 [1]. Moreover, according to Cisco, the number of smart devices reached 50 billion by 2020, indicating the steady expansion of this sector. These devices will produce more data traffic and burden the communication infrastructure, especially the ground base stations. It will also raise the transmission delay for content downloading. However, designing the infrastructure of future wireless networks requires addressing very critical challenges such as deployment strategies, coverage area, power consumption, link capacity, and transmission delay. Some of the technologies that aim to solve these challenges are Device-to-Device (D2D) communication, ultra-dense small cell network [2], and millimeter-Wave (mmW) communication in 5G networks as well as UAV-carried base station in 5G networks [3].

Deploying more ground base stations is essential to cope with a sudden increase in data traffic and a decrease in the transmission delay in future wireless networks. However, this is not a cost-effective and efficient solution. Therefore, caching is one of the potential techniques for reducing the transmission delay time in cellular networks, especially in small base stations [4], [5]. However, the conventional caching approaches only address the problem of the users who are typically connected to the same base station. Since the content is cached in the fixed base station, it does not offer any significant benefits to mobile users.

There are a number of solutions to overcome this problem. Taking advantage of non-terrestrial networks (NTN). NTN

The associate editor coordinating the review of this manuscript and approving it for publication was Seshadri Mohan.

network is formed by objects in the sky equipped with communication capabilities such as satellites, flying vehicles, and high altitude platform systems (HAPs) [6]. For instance, deploying Unmanned Aerial Vehicles (UAVs) (also referred to as drones) to enhance ground coverage as well as network performance in wireless networks and it is relatively easier to manage and economical to deploy [7]. UAVs are versatile devices that offer various services such as aerial photography, goods delivery and serving as aerial base stations [8]. UAVs are now one of the potential technologies for aerial platform-aided wireless communication to enhance the coverage area and link state for future wireless networks.

The unique flexible flying features of UAVs in 3D space, make them ideal candidates to be used as mobile base stations. Furthermore, UAVs base stations can fly at higher altitudes [9]. Therefore, they can have better Line-of-Sight (LoS) to the ground users in urban areas, compared to the ground base stations [10]. Additionally, UAV-based base stations can offer a 3D coverage for the Unmanned Autonomous Vehicle User Equipments (UAV-UEs) [11]. In the same vein, in [3], the authors argue that UAVs could conserve resources for the deployment of wireless communication services if they are correctly deployed. Especially deploying Low Altitude Platform UAVs (LAP-UAVs) as a small cell for a brief period for events (e.g., sports events and festivals), is simple and cost-effective. On the other hand, the High Altitude Platform UAVs (HAP-UAVs) are suitable for long-term coverage solutions [3].

In the area of UAV-enabled wireless networks, the existing research focused on the trajectory, 2D and 3D deployment, power consumption, front-hauling, etc. It is worth mentioning that many challenges still exist, such as, but not limited to, mobile edge computing, UAV antennas, UAV as base station simulators, and caching in the UAVs [8]. Furthermore, we can leverage the mobility of the cache-enabled UAVs, where UAV base stations can adjust their location dynamically to the specific ground users based on their cached content. In the meantime, UAVs can intelligently cache content based on the number of requests they receive for specific content.

To the best of our knowledge, there is still lack of research and insight on analyzing the performance of wireless communication system where cache-enabled UAVs can be used as base stations and the UAVs share their caches in order to reduce the transmission delay. Therefore, this paper proposes framework for cache-enabled UAV-mounted base stations in the wireless networks where UAVs are able to collaborate and share their caches using Deep Reinforcement Learning (DRL).

The rest of the paper is organized as follows. In Section I-A, we discuss the state of the art regarding caching in the sky. Section II discusses the system model, notations, and problem formulation, and Section III covers the implementation of the proposed solution. In Section V, we discuss the simulation and in Section V-A we discuss the experimental results. Finally, Section VI concludes the paper with future work.

A. RELATED WORK

Cache-enabled networks allow users to experience the expected Quality of Service (QoS) by caching the popular content for future requests [12]. There is a plethora of work already done on caching considering the metrics such as cache hit, maximizing the throughput, and content placement to reduce the delay in accessing the content. For instance, in [13], the authors considered deploying UAVs to deliver content to end users, they proposed timelines solution model that uses DRL-based Proximal Policy Optimization (PPO) method for learning the optimal caching strategy. The simulation results show that the proposed solution outperforms the conventional caching approaches.

In [14], the authors considered deploying UAVs as non-terrestrial relays to improve the quality of experience (QoE) for user equipments (UE) on the ground. UAVs are equipped with storage to proactively store frequently requested contents by the ground users. The authors proposed a novel Deep Federated Echo State Learning (DeepFESL) algorithm that proactively cache content based on UE location, device type and other device behaviors. Due to privacy issue of UE data, the model is trained in a decentralized and federated learning approach, where all the UE devices collaboratively train the model and the popular contents are cached in the UAVs. After extensive simulation, results show the proposed solution outperforms conventional caching schemes in terms of cache hit, system utilization and improving QoE. Likewise, in [15], the authors leverage the user movement and propose a new systematic framework to improve the caching efficiency in Content-Centric Wireless Networks (CCWNs). The authors suggested a mobility-aware caching strategy in the ground base station.

In [16], the authors used the UAV as a relay and a base station in a Cognitive Radio Network (CRN) to improve transmission capability and solve the spectrum scarcity problem while the traffic is huge. On the other hand, in [17], the authors investigated joint optimization of the UAV deployment, caching placement, and user association to improve the Quality of Experience (QoE). Although this paper utilizes the use of UAVs in cellular networks, it concentrates mostly on the end-user side instead of the base station load.

In [18] the authors investigated temporary deployment of UAV base stations in an ultra-dense network while considering the energy efficiency and 3D deployment of the UAVs. The authors proposed a novel price-based proximal policy optimization (3PO) algorithm to maximize the system performance in terms of throughput and energy efficiency. The results shows that the proposed approach outperforms the baselines. However, in this work caching in the UAV base station is not investigated.

The authors in [19], examined cache-enabled UAVs more specifically the cache placement and UAV resource consumption in cellular Non-Orthogonal Multiple Access (NOMA) networks while considering the movement and dynamics of the UAVs and content requests by the users.

Similarly, in [20], the authors optimized energy efficiency in cache-enabled fixed-wing UAV in the IoT domain. Similarly, in [21], the authors examined the deployment of the UAVs as base stations that pre-cache the content for the ground users to improve the quality of experience in a Cloud Radio Access Network (CRAN). In [10], the authors investigated resource allocation and caching of the contents in cache-enabled UAVs-assisted LTE-U (licensed and unlicensed bands) networks. They proposed a distributed algorithm based on the liquid state machine framework to improve the system's performance. In another similar work [22], the authors jointly optimized cache placement, flight trajectory, and transmission power optimization for multi-UAV-assisted wireless networks.

In [23], the authors investigated online UAV-assisted wireless caching by jointly optimizing UAV movement, power consumption, and scheduling of the content caching. The UAVs actively cache the requested content based on popularity. Also, UAVs get recharged from the charging stations.

Khuwaja et al. [24] examined the system performance of a network where the network has both flying base station aka drones and small ground base stations. They used caching to reduce the load from the backhaul. Likewise, in [25], Wang et al. explored content distribution in crowded areas where they used UAV and edge caching to offload the traffic and enhance the performance of the system.

In another work, Kang et al. [26] investigate the technique to lower the load of the traffic on the static ground base station by deploying UAVs as short-period small base stations to assist the cellular network performance. Similarly, in [27], the authors investigated the deployment of the UAVs as base stations aka Aerial Base Station (ABS), considering the backhaul link as constrain and UAVs were backed with storages to cache the content for ground users. The authors focused on delay-tolerant and delay-sensitive users with distinctive rate requirements. The problem was formulated jointly considering the ABS 3D placement, user-to-ABS association, and bandwidth allocation. In [12], the authors examined the deployment of the UAVs in wireless networks, while focusing on constraints such as power consumption and disk capacity. To improve the throughput of the multimedia data in the Internet of Things network, Jiang et al. [28] investigated the flexibility in caching of the UAVs.

In view of the preceding discussion on the existing works, to the best of our knowledge, the carried research papers do not consider cache sharing among the drones aka UAVs in a wireless network. To address this gap, in this paper, we propose a Deep reinforcement learning-based algorithm while considering an effective and efficient cache-sharing scheme to lower the transmission delay and maximize the overall performance of the network system. The main contributions of this paper are as follows:

- We propose a novel framework of cache-enabled UAV wireless network in which the UAVs share their cache capabilities with each other in a highly dynamic and unpredictable environment where the objects such as users and UAVs are constantly moving that effecting the channel gain.
- In order to reduce the total transmission delay incurred in the response to a content request, We formulate a non-convex optimization problem for simultaneously optimizing user to UAV association, UAV mobility, and content caching.
- Since, the problem is non-convex, which means that conventional methods such as optimization, relaxation, or approximation are not effective in solving it. We adopt and extend DRL-based Deep Deterministic Policy Gradient (DDPG) algorithm due to its model-free and online properties where the agent actively interacts with the environment. DDPG can work with continues space action [29], and it is off-policy compared to Proximal Policy Optimization (PPO) [30] and Trust Region Policy Optimization (TRPO) [31]. Thus, we believe DDPG is the best suit in our scenario. We transform the aforementioned non-convex optimization problem to DDPG algorithm to find the optimized power levels, user association, and caching decision to minimize the transmission delay.
- Finally, we evaluate our proposed solution with the two baselines. In baseline 1, we consider no caching is enabled in the UAVs, and the requested content of the users is downloaded via a base station, and baseline 2, cache-enabled UAVs with no cache sharing among neighbors. We conducted the simulation for various cache sizes and different numbers of associated users with their home UAVs. The simulation results show a significant improvement which is summarized as follows:
  - Small cache size: we consider UAVs with 20 MB of cache size and 80 users associated with them, the results show that our proposed solution outperforms baseline 1 with 24.7% margin, and baseline 2 with 10.6% margin in terms of average delay of downloading content. Furthermore, in terms of average system rate usage, our proposed solution outperforms baseline 1 by 31.9% and baseline 2 by 13.7%.
  - Large cache size: we consider UAVs with 40 MB of cache size and 80 users associated with them, the results show that our proposed solution outperforms baseline 1 by 46% and baseline 2 by 16.8% in terms of average delay of content downloading. Furthermore, in terms of average system usage rate, our proposed solution outperforms baseline 1 by 39% and baseline 2 by 12.9%.

– Number of associated users: we also tested system performance with respect to the different number of associated users. The results show that the system performance utilization increases as the number of users increases. Furthermore, there is a significant reduction in terms of cache miss in the home and neighbor UAVs which leads to improvement in total average delay and average system rate usage.

Next, we present our system model and problem formulation.

II. SYSTEM MODEL AND PROBLEM FORMULATION

In our system model, we consider a network illustrated in figure 1 that has a ground base station and a set of UAV base stations with content caching capability that serve users on the ground. We simplify the model by assuming that the UAVs move over the users and proactively cache the content based on the Zipf law at each time step. Which can reduce the transmission delay and backhaul load [27]. Moreover, we assume that a UAV in our network model moves with limited velocity and has limited communication and storage capacities. Finally, each UAV possesses sufficient energy to serve the ground users. We assume each UAV operates with a fixed battery life, determining a consistent service or flight time for our approach.

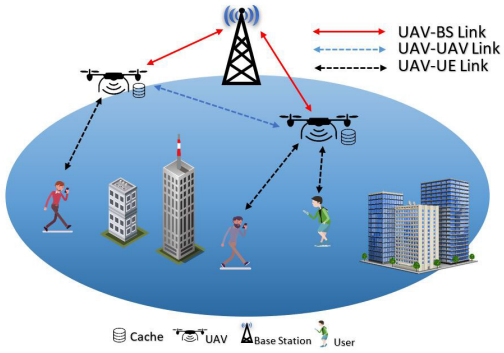


FIGURE 1. The system model for cache-sharing UAVs with three connections.

We use BS notation for the ground station and $u \in \mathcal{U} = \{1, 2, 3, \ldots, U\}$ for the set of UAVs. Furthermore, a set of ground users associated with each UAV u is denoted as H_u and let $\mathcal{H} \triangleq \bigcup_{u=1}^{|U|} H_u$ resemble the total number of ground users in the network. A UAV $u \in \mathcal{U}$ can operate for a limited time T, and T is divided into multiple time steps represented as $t \in [0, T]$.

Table 1 shows the system model parameters. We next describe our models for mobility, transmission, user association and caching.

A. MOBILITY MODEL

Mobility is one of the key features of UAVs as flying base stations.

A ground user $h \in \mathcal{H}$ is considered as randomly distributed at fixed locations at time slot t, which can be expressed in 3D Cartesian coordinates system as $(E_h^T(t), 0(t))$, $\forall h$ such that $E_h(t) = [x_h(t), y_h(t)]^T \in \mathbb{R}^{2*1}$.

TABLE 1. System model parameters.

Notation	Description
BS	The ground base station
$\mathcal{U}$	The set of all UAVs
u	The UAV $u \in \mathcal{U}$
$\mathcal{H}$	The set of all users
H_u	The subset of users served by UAV $u \in \mathcal{U}$
T	The total operational time of UAV $u \in \mathcal{U}$
t	The time slot $t \in [0, T]$

TABLE 2. Mobility model parameters.

Notation	Description
E_h	The 2D coordination of ground user
R_u	The 2D coordination of the UAV $u \in \mathcal{U}$
A_u	The altitude of the UAV $u \in \mathcal{U}$
$\dot{\vec{R}}_u(t)$	Horizontal velocity of UAV $u \in \mathcal{U}$ at time t
$\max(V_u^{horizon})$	Maximum horizontal speed of UAV $u \in \mathcal{U}$
$d_{u,h}(t)$	The distance of the UAV $u \in \mathcal{U}$ and user $h \in \mathcal{H}$ at time t
$d_{u,u'}(t)$	Distance between UAV $u, u' \in \mathcal{U}$ at time t
$Min(d)$	Minimum inter-UAV distance

Moreover, the location of the UAVs at time slot t is expressed in 3D Cartesian coordinate system as $(R_u^T(t), A_u(t))$, $\forall u \in \mathcal{U}$, where $R_u(t) = [x_u(t), y_u(t)]^T$ and $A_u(t)$ is the corresponding altitude.

We may define UAV's horizontal speed constraint as $||\dot{\vec{R}}_u(t)|| \leq \max(V_u^{horizon})$, $t \in [0, T]$ and $\dot{\vec{R}}_u(t)$ is the time-derivative of the horizontal coordinates of the UAV $u \in \mathcal{U}$ such that:

$$\dot{\vec{R}}_u(t) = \frac{\partial \vec{R}_u(t)}{\partial t}, \tag{1}$$

where $\partial \vec{R}_u(t)$ is the partial derivative of the displacement at time t and $\max(V_u^{horizon})$ is the maximum horizontal speed of UAV $u \in \mathcal{U}$. Note that in our work, we assume UAVs' operate at a fixed altitude [32], [33].

Then, the distance between a ground user $h \in \mathcal{H}$ to associate it with UAV $u \in \mathcal{U}$ at time $t \in [0, T]$ can be expressed as below:

$$d_{u,h}(t) = \sqrt{||\vec{R}_u(t) - \vec{E}_h(t)||^2 + A_u(t)^2}. \tag{2}$$

Next, we define a constraint in which we define the minimum distance $d_{u,u'}$ required between two UAVs $u, u' \in \mathcal{U}$ to avoid a collision between them:

$$d_{u,u'}(t) = \sqrt{||\vec{R}_u(t) - \vec{R}_{u'}(t)||^2} \geq d_{min}, \forall t \in [0, T], \tag{3}$$

where d_{min} is the minimum inter-UAV distance in meters. The mobility model parameters are listed in the table 2

B. TRANSMISSION MODEL

Our transmission model has air to ground and air to air channels, as explained below.

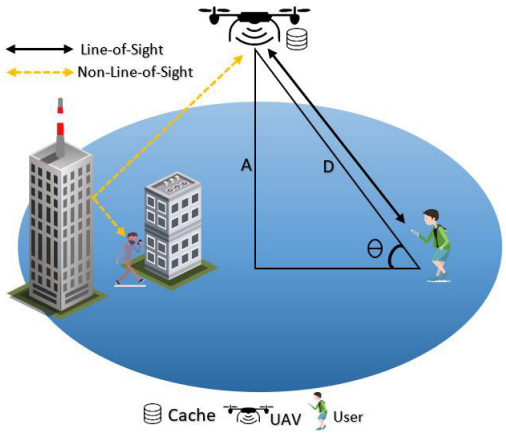


FIGURE 2. UAV to ground users communication via LoS and NLoS links.

1) AIR TO THE GROUND MODEL

In this model UAVs have two possible links: i) from UAV to ground user, ii) from UAV to ground base station. Therefore, we use the ground terminal $g \in \mathcal{H} \cup BS$ for referring both ground user $h \in \mathcal{H}$ and the ground base station BS. The link between the UAV $u \in \mathcal{U}$ and the ground terminal g depends on the carrier frequency, probability of the line-of-sight (LoS), and non-line-of-sight (NLoS), and the attenuation factor of the LoS and NLoS links. In urban environments, there might be obstacles that block the link between the UAV $u \in \mathcal{U}$ and the ground terminal g. Fig. 2 shows the LoS and NLoS links of UAV to two types of users, one with direct LoS with no obstacle and one with NLoS that gets the signal by reflection. Therefore, the LoS probability at time slot t for a ground user is expressed as follows [10].

$$Pr^{los}_{u,g}(t) = (1 + ae^{\tau(A_u, d_{u,g})})^{-1}, \tag{4}$$

such that:

$$\tau(A_u, d_{u,g}) = b(a - sin^{-1}(\frac{A_u}{d_{u,g}})),$$

having A_u and $d_{u,g}$ give us the angle Θ between UAV $u \in \mathcal{U}$ and g. Furthermore, high altitude gives a larger angle with a high probability of LoS. a and b are environmental variables (i.e. rural, urban, dense urban, or others). Probability of NLoS by axioms of probability theory at time t we have the following:

$$Pr^{nlos}_{u,g}(t) = 1 - Pr^{los}_{u,g}. \tag{5}$$

For the path-loss between UAV $u \in \mathcal{U}$ and g, we used the standard log-normal shadowing model for both LoS and NLoS links and it can be expressed as following:

$$PL_{u,g}(t) = fspl(d_0) + 10\mu_{LoS} log(d_{u,g}) + \mathcal{X}_{\sigma LoS}, \tag{6}$$

$$PNL_{u,g}(t) = fspl(d_0) + 10\mu_{NLoS} log(d_{u,g}) + \mathcal{X}_{\sigma NLoS}, \tag{7}$$

where $fspl(d_0)$ is the free space path loss given by the $20log(\frac{4\pi f_c d_0}{c})$, where f_c is the frequency being used, c is the speed of light and d_0 is the reference or close-in distance. μ_{LoS} and μ_{NLoS} are the path loss exponents which are also known as the attenuation factors. $\mathcal{X}_{\sigma LoS}$ and $\mathcal{X}_{\sigma NLoS}$ are the shadowing random variables, which are represented as the Gaussian random variable in dB with zero mean, σLoS and $\sigma NLoS$ the standard deviations in dB.

TABLE 3. Air to ground channel model parameters.

Notation	Description
g	A user $h \in \mathcal{H}$ or the base station BS
$Pr^{los}_{u,g}$	The probability of LoS connection
Θ	The elevation angle between UAV u and g
$d_{u,g}$	The distance between UAV u and ground terminal user h or base station BS
a and b	Environmental variables
$sin^{-1}(\frac{A_u}{d_{u,g}})$	The elevation angle between UAV u and g
$Pr^{nlos}_{u,g}$	The probability of NLoS connection
$PL_{u,g}$	Path loss of LoS channel between UAV u and user h or base station BS
$PNL_{u,g}$	The path loss of NLoS channel between UAV u and user h or base station BS
c	The speed of light
f_c	The frequency band
d_0	The close-in reference distance
μ	The LoS and NLoS attenuation factors.
$\mathcal{X}_{\sigma LoS}$ and $\mathcal{X}_{\sigma NLoS}$	The shadowing random variables
σLoS and $\sigma NLoS$	The standard deviations
μLoS and $\mu NLoS$	Path loss exponents
$\bar{l}_{u,g}$	The average path loss between UAV u and user h or base station BS
$SP_{u,g}$	The transmit power of UAV u to ground terminal g
$SNR_{u,g}$	The signal to noise ratio between UAV u and user h or base station BS
$LC_{u,g}$	The link capacity between UAV u and user h or base station BS
$B_{u,g}$	The bandwidth between UAV u and user h or base station BS
σ^2	The noise power variance

Furthermore, the average path loss between the UAV $u \in \mathcal{U}$ and g can be calculated by equations (4),(5),(6), and (7) which is expressed as follow [34] and [35]:

$$\bar{l}_{u,g}(t) = (Pr^{los}_{u,g}) \times (PL_{u,g}) + (Pr^{nlos}_{u,g}) \times (PNL_{u,g}). \tag{8}$$

Given the average path loss from equation (8), the average signal-to-noise ratio (SNR) of g associated with UAV $u \in \mathcal{U}$ can be expressed as below [35]:

$$SNR_{u,g}(t) = \frac{SP_{u,g}}{10^{\bar{l}_{u,g}/10}/\sigma^2}, \tag{9}$$

where $SP_{u,g}$ is the transmit power density of UAV $u \in \mathcal{U}$ to the ground terminal g. $\sigma^2 = B_{u,g}N_0$ which is the variance of Addictive White Gaussian Noise (AWGN) and $B_{u,g}$ is the bandwidth used and N_0 is denoted as the power spectral density.

Combining all of the above quantities the link capacity between UAV $u \in \mathcal{U}$ and the ground terminal g is expressed in bit/sec as below:

$$LC_{u,g}(t, SP_{(u,g)}) = B_{u,g} \times log_2(1 + SNR_{u,g}). \tag{10}$$

Air to ground model notations listed in table 3.

2) AIR TO AIR MODEL

For UAV to UAV communication we leverage from free space path loss since the UAVs are at a higher altitude and there is a free-space between UAVs. We consider UAV $u \in \mathcal{U}$ transmits signal to neighbor UAV $u' \in \mathcal{U}$. Therefore, the power at which UAV $u \in \mathcal{U}$ sends the signal to UAV $u' \in \mathcal{U}$ is modeled as below [34]:

$$SP_{u,u'}(t) = SP_{u'}(t) - G_u(t) - G_{u'}(t) + FSPL_{u,u'}(t), \quad (11)$$

where $SP_{u,u'}$ is the transmit signal by UAV $u \in \mathcal{U}$ and $SP_{u'}$ is the received signal by UAV $u' \in \mathcal{U}$. G_u and $G_{u'}$ are the gains of UAV $u \in \mathcal{U}$ and UAV $u' \in \mathcal{U}$ respectively. Moreover free-space path loss between UAV $u \in \mathcal{U}$ and UAV $u' \in \mathcal{U}$ is modeled as follow:

$$FSPL_{u,u'}(t) = 10\rho log(\frac{4\pi f}{cd_{u,u'}}) \quad (12)$$

SNR for the UAV to UAV is modeled below:

$$SNR_{u,u'}(t) = \frac{SP_{u,u'}}{\sigma^2} \quad (13)$$

The channel capacity for the UAV to UAV motivated from [10], is modeled as below:

$$LC_{u,u'}(t, SP_{(u,u')}) = B_{u,u'} \times log_2(1 + SNR_{u,u'}) \quad (14)$$

Air to air model notations listed in table 4.

TABLE 4. Air to air channel model parameters.

Notation	Description
$SP_{u,u'}$	The transmit signal by UAV $u \in \mathcal{U}$
$SP_{u'}$	The received signal by UAV $u' \in \mathcal{U}$
G_u	The gain of UAV $u \in \mathcal{U}$
$G_{u'}$	The gain of UAV $u' \in \mathcal{U}$
ρ	The UAV to UAV path loss exponent

C. USER ASSOCIATION

The set of users that is connected to a UAV $u \in \mathcal{U}$ is denoted as H_u and each UAV can serve a limited number of users. We consider dense areas where multiple UAVs are needed to serve the users which lead us to have dynamic user association to UAVs. The total number of the users that a UAV can associated with is expressed below:

$$\sum_{h=1}^{\mathcal{H}} \alpha_u^h \leq H_u \quad \forall u, u \in \mathcal{U}, \quad (15)$$

We introduce a binary variable which controls the association of the users to UAVs.

$$\alpha_u^h = \begin{cases} 1 & \text{Associate user } h \text{ with UAV } u, \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Caching and user association models parameters are listed in table 5

D. CACHING MODEL

In our caching model we assume that each UAV $u \in \mathcal{U}$ is equipped with the cache capacity of M_u and we consider the set of contents as $k \in \mathcal{K} = \{1, 2, 3, \ldots, K\}$, where each UAV $u \in \mathcal{U}$ caches a set of contents $K_u \subset \mathcal{K}$. Note that in our model, we assume each content k is represented with size s^k. Each UAV $u \in \mathcal{U}$ has a limited cache capacity and can only store a limited number of contents, this constraint can be written as below:

$$\sum_{k=1}^{K} y_u^k s^k \leq M_u, \quad \forall u \in \mathcal{U}; \quad (17)$$

Moreover, lets assume that the UAV $u \in \mathcal{U}$ has pre-cached the popular contents using Zipf law and $Pr_{u,k}$ represents that probability of a user request for content k from UAV u, inspired by our previews work [36].

As Fig. 3 illustrates, in our system model there will be three possible scenarios for a user $h \in \mathcal{H}$ associated with its home UAV $u \in \mathcal{U}$ for requesting content $k \in \mathcal{K}$.

1) CACHE HIT AT THE HOME UAV

In the first scenario, the content k requested by the user $h \in \mathcal{H}$ is available in the home UAV $u \in \mathcal{U}$. Thus, we define $y_{u,h}^k$ as cache hit indicator variable at home UAV $u \in \mathcal{U}$ for indicating if content k requested by user h is cached at the home UAV u.

$$y_{u,h}^k = \begin{cases} 1 & \text{if content } k \text{ requested by user } h \text{ is cached at} \\ & \text{home UAV } u \in \mathcal{U}, \\ 0 & \text{otherwise.} \end{cases} \quad (18)$$

In this scenario, since there is a cache hit in the home UAV, the home UAV $u \in \mathcal{U}$ will not ask the content from the neighbor UAV $u' \in \mathcal{U}$, in other words, no cached content will be shared from neighbor with home UAV.

Therefore, the transmission delay for the first scenario is considered for one hop, and it can be calculated as below:

$$t_{u,h}^k = \frac{s^k}{LC_{u,h}} \quad (19)$$

where the $t_{u,h}^k$ is the transmission delay of the content k for the user h from home UAV u and s^k is the size of the requested content and $LC_{u,h}$ is the link capacity between home UAV $u \in \mathcal{U}$ and user $h \in \mathcal{H}$.

2) CACHE HIT AT THE NEIGHBOR UAV

In the second scenario the content k requested by user $h \in \mathcal{H}$ is not cached in the home UAV $u \in \mathcal{U}$ but the content is available in neighbor UAV $u' \in \mathcal{U}$. Thus we define $y_{u',h}^k$ is the cache hit indicator variable at the neighbor UAV $u' \in \mathcal{U}$ for indicating if content k requested by user h is cached at the

TABLE 5. Caching and user association models parameters.

Notation	Description
M_u	The cache storage of each UAV $u \in \mathcal{U}$
$\mathcal{K}$	The set of all contents
k	The content $k \in \mathcal{K}$
K_u	The cached content in the UAV $u \in \mathcal{U}$
s^k	The size of the content $k \in \mathcal{K}$
$Pr_{u,k}$	Probability of requesting content $k \in \mathcal{K}$ from UAV $u \in \mathcal{U}$
α_u^h	The user association decision variable for user $h \in \mathcal{H}$ to UAV $u \in \mathcal{U}$
$y_{u,h}^k$	The cache hit indicator in home UAV
$y_{u',h}^k$	The cache hit indicator in neighbor UAV
$t_{u,h}^k$	The transmission delay for content $k \in \mathcal{K}$ from home UAV
$t_{u',h}^k$	The transmission delay for content $k \in \mathcal{K}$ from neighbor UAV
$t_{BS,h}^k$	The transmission delay for content $k \in \mathcal{K}$ from BS

neighbor UAV u'.

$$y_{u',h}^k = \begin{cases} 1 & \text{if content } k \text{ requested by user } h \text{ is cached at } u' \in \mathcal{U}, \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

Moreover, in this scenario, since there is a cache miss in the home UAV, thus, the home UAV $u \in \mathcal{U}$ requests the content from the neighbor UAV $u' \in \mathcal{U}$ and serves the ground user. In other words, neighbor UAV shares its cached content with the home UAV. Point to be noted, that cache sharing occurs in this scenario.

The transmission delay for the second scenario can be calculated as below:

$$t_{u',h}^k = \frac{s^k}{LC_{u,h} + LC_{u,u'}} \tag{21}$$

where $t_{u',h}^k$ is the transmission delay of the content k for the user h from neighbor UAV u' and $LC_{u,u'}$ is the link capacity between home UAV $u \in \mathcal{U}$ and neighbor UAV $u' \in \mathcal{U}$.

3) CACHE MISS AT THE HOME AND NEIGHBOR UAVS

In the third scenario we assume that the content k requested by user $h \in \mathcal{H}$ is not cached in the home UAV $u \in \mathcal{U}$ as well as at the neighbor UAV $u' \in \mathcal{U}$. Therefore, the requested content is downloaded from the base station BS. The delay for the third scenario can be calculated as below:

$$t_{BS,h}^k = \frac{s^k}{LC_{u,h} + LC_{u,BS}} \tag{22}$$

where $t_{BS,h}^k$ is the transmission delay of the content k for the user h from the base station BS and $LC_{u,BS}$ is the link capacity between home UAV $u \in \mathcal{U}$ and base station BS. It should be noted that once the home UAV receives the missed content from the ground base station, it can cache missed content based on popularity.

E. PROBLEM FORMULATION

We formulate a novel optimization problem to minimize the total transmission delay for the ground users using the cache sharing approach between UAVs. The formulated problem

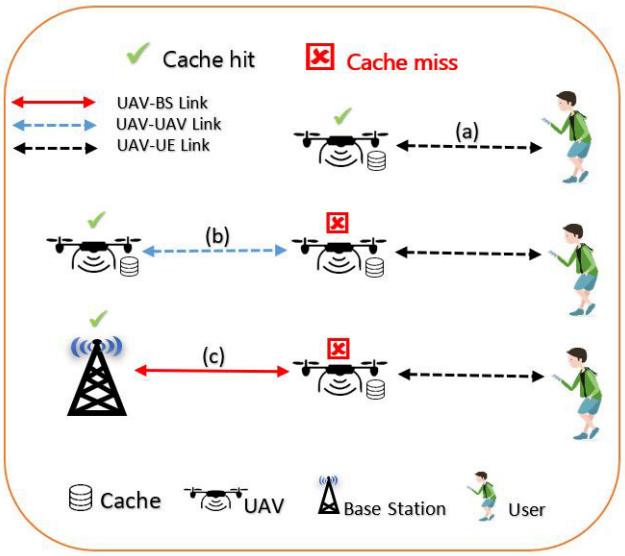


FIGURE 3. Collaboration space for cache-sharing UAVs with three possible scenarios.

will reduce the transmission delay time, which will lead to maximizing the sum achievable throughput. Hereafter, the total delay for user $h \in \mathcal{H}$ to download the requested content $k \in \mathcal{K}$ from UAV $u \in \mathcal{U}$ calculated as below:

$$\begin{aligned} &D_{u,h}^k(\alpha, y, SP) \\ &= \alpha_u^h \Big((y_{u,h}^k)(t_{u,h}^k) + (1 - y_{u,h}^k) \\ &\times \Big[\big[y_{u',h}^k (t_{u,h}^k + t_{u',h}^k) \big] + (1 - y_{u',h}^k)(t_{u,h}^k + t_{BS,h}^k) \Big] \Big). \end{aligned} \tag{23}$$

where α_u^h is the association control variable, $y_{u,h}^k$ is cache hit indicator variable at home UAV $u \in \mathcal{U}$, $t_{u,h}^k$ is the transmission delay time between user $h \in \mathcal{H}$ and the home UAV $u \in \mathcal{U}$. $y_{u',h}^k$ is the cache hit indicator variable of the neighbor UAV $u' \in \mathcal{U}$ and $t_{u',h}^k$ is the transmission delay between the neighbor UAV $u' \in \mathcal{U}$ and user $h \in \mathcal{H}$. $t_{BS,h}^k$ is the transmission delay time between user $h \in \mathcal{H}$ and base station BS. Furthermore, we denote SP for both transmit power of UAV $u \in \mathcal{U}$ to UAV $u' \in \mathcal{U}$ or ground terminal g.

Then, the total transmission delay for downloading all the content $k \in \mathcal{K}$ of a user $h \in \mathcal{H}$ from UAV $u \in \mathcal{U}$ can be defined as below:

$$\zeta_{u,h} = \sum_{k=1}^{K} D_{u,h}^k(\alpha, y, SP) \tag{24}$$

Therefore, above optimization problem can be expressed as follow:

$$\min_{\boldsymbol{\alpha}, \boldsymbol{y}, \boldsymbol{sp}} \quad \sum_{h=1}^{H} \sum_{u=1}^{U} \zeta_{u,h} \tag{25}$$

$$\text{Subject to:} \quad \sum_{k=1}^{K} y_u^k s^k \leq M_u, \quad \forall u \in \mathcal{U}; \tag{26a}$$

$$D^k_{u,h}(\alpha, y, SP) \leq \mathcal{T}_{u,h}, \quad \forall u, h; \quad h \in \mathcal{H}, u \in \mathcal{U} \tag{26b}$$

$$\sum_{u=1}^{U} ||\dot{\vec{R}}_u(t)|| \leq \max(V_u^{horizon}) \quad \forall t \in [0, T] \tag{26c}$$

$$\sum_{h=1}^{\mathcal{H}} \alpha_u^h \leq H_u \quad \forall u \in \mathcal{U}, \tag{26d}$$

$$\sum_{u=1}^{U} 0 \leq R_u(t) \leq R_u^{max}, \quad \forall t \in [0, T] \tag{26e}$$

$$\sum_{u=11}^{U} \sum_{u'=1}^{U} d_{u,u'}(t) = \sqrt{||\vec{R}_u(t) - \vec{R}_{u'}(t)||^2} \geq Min(d), \forall t \in [0, T] \tag{26f}$$

$$y = \{0, 1\}, \forall u, u', h, k; \quad h \in \mathcal{H}, u \in \mathcal{U}, k \in \mathcal{K} \tag{26g}$$

$$\alpha_u^h = \{0, 1\}, \forall u, h; \quad h \in \mathcal{H}, u \in \mathcal{U} \tag{26h}$$

Where the constraint 26a states that each UAV $u \in \mathcal{U}$ has finite amount of capacity M_u which can cache limited number of content $k \in \mathcal{K}$ with size s^k. The constraint 26b states that the total transmission delay time should be under a certain threshold to ensure that the user does not starve. Moreover, 26c is the constraint for the max horizontal speed of the UAV. Constraint 26d states the number of users that can associate with a UAV. Constraint 26e is ensuring the UAV is in serving area. Constraint 26f is for avoiding the UAVs collision. Finally, the constraints 26g and 26h represent binary decision variables for cache hit ensuring content availability either at home or neighbor UAVs and user association to UAV, respectively.

III. PROBLEM TRANSFORMATION TO DRL

In this section, we present our solution and transform our problem into a deep reinforcement learning algorithm. Minimizing the transmission delay of a content requested by a user $h \in \mathcal{H}$, depends on the user association (α), rate between UAV $u \in \mathcal{U}$ and user $h \in \mathcal{H}$, rate between UAV $u \in \mathcal{U}$ and ground base station BS, rate between UAV $u \in \mathcal{U}$ and UAV $u' \in \mathcal{U}$, caching availability in home UAV (y_u), and caching availability in neighbor UAV ($y_{u'}$). In RL, the agent tries to interact with the environment to find the optimal policy and maximize the cumulative reward. In our approach, the agent is our UAV $u \in \mathcal{U}$ that is trying to find the best policy by learning about the environment at each interaction.

The environment in our RL problem is captured by representing the state of mobility of the UAV $u \in \mathcal{U}$, the distance of the UAV $u \in \mathcal{U}$ with the associated user $h \in \mathcal{H}$ which will affect the channel rates to calculate the delay, available cache in the home UAV $u \in \mathcal{U}$, and available cache in the neighbor UAV $u' \in \mathcal{U}$. Based on the environment observation at each time slot $t \in [0, T]$ the agent chooses an optimal action represented by the tuple (α^*, y^*, and SP^*). Note that these actions are the control variables represented in our problem. Since the number of states in our approach becomes very large, we employ deep neural network (DNN) with RL to overcome this challenge to capture huge state space [37]. This is a common approach used to tackle such problems [38], [39].

A. SYSTEM OBSERVATION

In our model, the environment collects four variable types of information and transmits them to the agent during each time slot, $t \in [0, T]$. These variables include $t^k_{u,h}$, $t^k_{u',h}$, and $t^k_{BS,h}$ which represent the delays between the user $h \in \mathcal{H}$ to home UAV, neighbor UAV and ground base station. These delays are further elaborated in section II-D and illustrated in Fig.3. The final variable pertains to the horizontal mobility of the agent UAV, denoted by $h_u^{speed} = ||\dot{\vec{R}}_u||$, where $u \in \mathcal{U}$. The state observation is represented by a vector in the following manner.

$$\boldsymbol{n}(t) \in \mathcal{N} = \{t^k_{u,h}(t), t^k_{u',h}(t), t^k_{BS,h}(t), h_u^{speed}\}.$$

B. ACTION

In our reinforcement learning (RL) approach, the agent UAV $u \in \mathcal{U}$ adheres to a stationary control policy and takes actions based on specific conditions. At every time slot $t \in [0, T]$, the agent selects actions such as user association (α), caching decision (y), and power (SP).

$$\boldsymbol{i}(t) \in \mathcal{I} = \{\alpha_u^h, y_u^k, SP\}.$$

C. STATIONARY CONTROL POLICY

In RL, the stationary control policy $\boldsymbol{\Psi}$ is used to map the given state to an action, more specifically $\boldsymbol{\Psi}$ function maps from state $\boldsymbol{n}(t) \in \mathcal{N}$ to the probability of selecting an action from the action space $\boldsymbol{i}(t) \in \mathcal{I}$. It could be written as follow: $\boldsymbol{\Psi} : \mathcal{N} \rightarrow \mathcal{X}$.

D. REWARDS

In addition to the states and actions, the environment also provides the agent with rewards, denoted as $r(t)$, based on its performed actions. The reward function in our case is defined as inversely proportional to the total transmission delay required to download content requested by user $h \in \mathcal{H}$. By utilizing these rewards, the agent learns and strives to determine the optimal policy $\boldsymbol{\Psi}^*$, which maximizes the cumulative reward over a given time horizon.

IV. DRL-BASED ACTOR-CRITIC ALGORITHM

The actor-critic algorithm is widely utilized for solving problems that involve continuous domains. Since we designed our system model considering a realistic and highly dynamic and unpredictable environment where the UAVs and the users are mobile, which causes the communication channels gains to be changed at each moment of the time, resulting in continuous state and action spaces. Therefore, in our work, we employ the deep deterministic policy gradient (DDPG) approach,

which is an actor-critic-based algorithm specifically designed for continuous domains.

DDPG is an off-policy algorithm, in which the agent is continuously learning and exploring the environment regardless of a specific policy. Also, DDPG is an online algorithm, in which the agent is directly interacting with the environment and generating samples of experience and takes the best action in the given state [29].

The actor-critic algorithm employs two deep neural networks, namely the actor-network and the critic-network. The actor-network, denoted as $\pi(\boldsymbol{n};\boldsymbol{\theta^{\pi}})$, maps the state to an action, yielding the policy. The critic-network, denoted as $Q(\boldsymbol{n}, i;\boldsymbol{\theta^{Q}})$, evaluates the quality of the policy by criticizing the actions produced by the actor-network. The parameters of the actor and critic networks, $\boldsymbol{\theta^{\pi}}$ and $\boldsymbol{\theta^{Q}}$, are updated at every time slot, $t \in [0, T]$.

Algorithm 1 : DRL-Based Cache-Sharing Algorithm

1: **Initialize Phase:** $\mathcal{O}^{(t)}$ of size Z and mini-batch $\tilde{\mathcal{O}}^{(t)}$ of size W, $Q(\boldsymbol{n}, i|\boldsymbol{\theta^{Q}})$ as critic network and $Q'(.)$ as target critic network with weights $\boldsymbol{\theta^{Q}}$ and $\boldsymbol{\theta^{Q'}} \leftarrow \boldsymbol{\theta^{Q}}$, respectively, actor network $\pi(\boldsymbol{n}|\boldsymbol{\theta^{\pi}})$ and target actor network $\pi'()$ with weights $\boldsymbol{\theta^{\pi}}$ and $\boldsymbol{\theta^{\pi'}} \leftarrow \boldsymbol{\theta^{\pi}}$,$\epsilon$-greedy probability ϵ^{Max}, ϵ^{Min} and ϵ^{d} decay factor.;
2: **Learning Phase:**
3: **for** each episode **do**
4: Environment reset;
5: $t \leftarrow 0$;
6: **repeat**
7: Observe $\boldsymbol{n}(t)$ & take action $\boldsymbol{i}(t) = \pi(\boldsymbol{n}(t)|\boldsymbol{\theta^{\pi}}) + \Gamma$;
8: Apply $\boldsymbol{i}(t)$, obtain $\boldsymbol{n}(t+1)$, and calculate reward $r(t)$;
9: Save $(\boldsymbol{n}(t), \boldsymbol{i}(t), r(t), \boldsymbol{n}(t+1))$ in $\mathcal{O}$;
10: Randomly sample mini batch $\tilde{\mathcal{O}}(t) \subseteq \mathcal{O}$;
11: Update $\boldsymbol{\theta^{Q}}$ and $\boldsymbol{\theta^{\pi}}$
12: Update $\boldsymbol{\theta^{Q'}}$ and $\boldsymbol{\theta^{\pi'}}$ as follows:
13: $\boldsymbol{\theta^{Q'}} \leftarrow \kappa\boldsymbol{\theta^{Q}} + (1-\kappa)\boldsymbol{\theta^{Q'}}$;
14: $\boldsymbol{\theta^{\pi'}} \leftarrow \kappa\boldsymbol{\theta^{\pi}} + (1-\kappa)\boldsymbol{\theta^{\pi'}}$;
15: $t \leftarrow t+1$;
16: $\Gamma = \epsilon^{\text{Min}} + (\epsilon^{\text{Max}} - \epsilon^{\text{Min}})/\exp(-\epsilon^{d}t)$;
17: **until** $\sigma_s(t) \leq \tau_s, \forall s$ or maximum iterations;
18: **end for**

Algorithm 1 presents the DRL-based cache-sharing algorithm for minimizing delay. In the initialization phase (Line 1), the algorithm creates the experience replay memory buffer $\mathcal{O}$ with size of Z and mini-batch $\tilde{\mathcal{O}}$ of size W. It also creates the critic network $Q(\boldsymbol{n}, i|\boldsymbol{\theta^{Q}})$ and the target critic network $Q'(.)$ with weights $\boldsymbol{\theta^{Q}}$ and $\boldsymbol{\theta^{Q'}} \leftarrow \boldsymbol{\theta^{Q}}$, respectively. Furthermore, it creates the actor network $\pi(\boldsymbol{n}|\boldsymbol{\theta^{\pi}})$ and the target actor network $\pi'()$ with weights $\boldsymbol{\theta^{\pi}}$ and $\boldsymbol{\theta^{\pi'}} \leftarrow \boldsymbol{\theta^{\pi}}$, and initializes the ϵ-greedy probability with ϵ^{Max}, ϵ^{Min} and ϵ^{d} values.

TABLE 6. Simulation parameters of system.

Notations	Descriptions	Values
$u \in \mathcal{U}$	Number of UAVs	4
$\max(V_u^{horizon})$	Maximum speed of UAVs	40 m/s
A_u	Altitude of UAVs	200 m
a and b	Environmental parameters (urban)	10, 0.15
μLoS	Additional path loss for LoS	2
$\mu NLoS$	Additional path loss for NLoS	100
ρ	Path loss exponent	2
$h \in \mathcal{H}$	Number of users	30, 50, 70, 80
f_c	Carrier frequency	2 GHz
B	Bandwidth	20 MHz
SP	Transmit power of UAV	23 dBm
σ^2	Variance of the Gaussian noise	-174 dBm/Hz
M_u	Cache size	20 MB, 40 MB
s^k	Size of the content $k \in \mathcal{K}$	1-16 MB

TABLE 7. Simulation parameters of DRL.

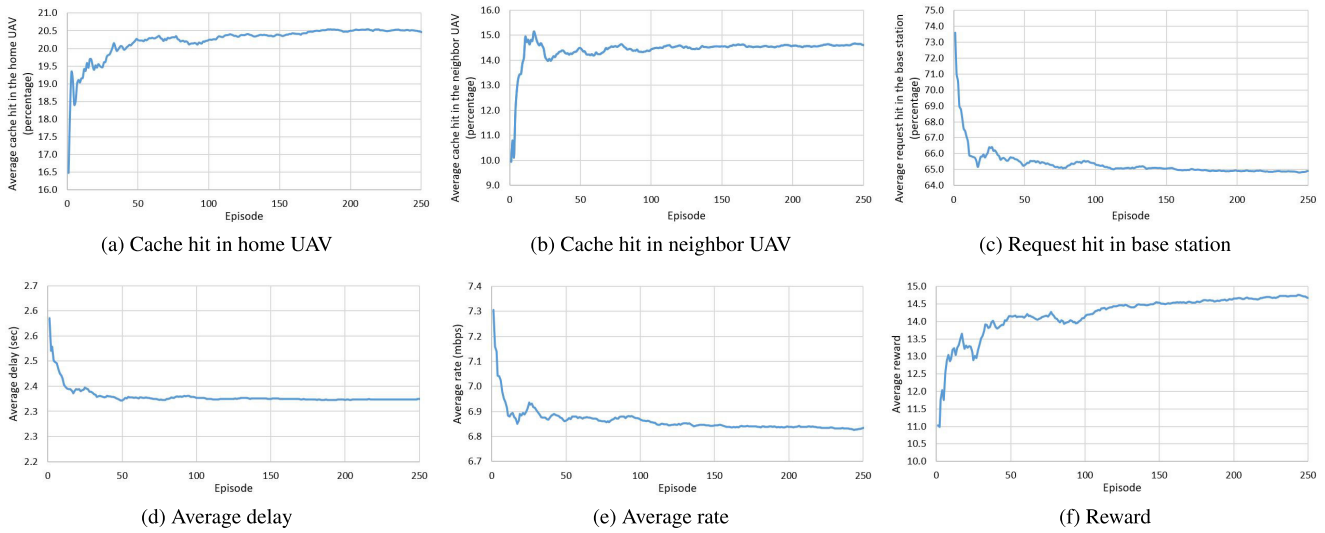
Notations	Descriptions	Values
$\pi(\boldsymbol{n}\|\boldsymbol{\theta^{\pi}})$	Actor network parameters	256, relu, 256, relu,256, relu, sigmoid
$Q(\boldsymbol{n}, \mathbf{x}\|\boldsymbol{\theta^{Q}})$	Critic network parameters	256, relu, 256, relu,256, relu, linear
κ	Actor and Critic LR	10^{-3}
	Soft Target update	10^{-2}
$\tilde{\mathcal{O}}$	Batch Size	256
$\mathcal{O}$	Memory size	10^{6}
λ	Discount factor	0.995

Line 3-5, the environment gets reset in each episode and environment reset provides first state of the environment. At the start of each episode (Line 3-5), the environment is reset and provides the initial state of the environment. Then, the actor network takes the state as input and maps it to an action (Line 7). The agent performs the action $\boldsymbol{i}(t)$, which leads to the next state $\boldsymbol{n}(t+1)$ and the provided reward $r(t)$ (Line 8). The tuple of transition $(\boldsymbol{n}(t), \boldsymbol{i}(t), r(t), \boldsymbol{n}(t+1))$ is then saved in the memory buffer $\mathcal{O}$ (Line 9).

Once there is enough experience stored in the memory buffer, the agent takes a random sample from it (Line 10). The parameters of the actor, target actor, critic, and target critic networks are then updated to new weights (Line 11-14), and the algorithm loops to another time slot t. The noise added to the actions for exploration is denoted by Γ, and it gets reduced over time (Line 16). The terminal state of the episode is reached when the requested content of user $h \in \mathcal{H}$ is fully downloaded (Line 17). In the next section, we discuss our simulation setting and results to evaluate the effectiveness of our approach.

V. SIMULATION

We conducted a simulation study of our proposed cache-sharing approach in a service area of 500 meters, where a base station is connected to a content center and cache-enabled UAVs serve ground users. The locations of the UAVs and users were generated randomly based on a Homogeneous Poisson Point Process (HPPP) in the service area. The simulation parameters for both the system and DRL algorithm are provided in Tables 6 and 7, respectively.



(a) Cache hit in home UAV (b) Cache hit in neighbor UAV (c) Request hit in base station

(d) Average delay (e) Average rate (f) Reward

FIGURE 4. System performance with respect to the number of episodes.

A. SIMULATION RESULTS

To evaluate the effectiveness of our proposed cache-sharing solution among UAVs, we conducted extensive simulations, comparing it with two baselines. The first baseline (baseline 1) assumes that no caching is available in the home UAV $u \in \mathcal{U}$ or the neighbor UAV $u' \in \mathcal{U}$, and the content is directly downloaded from the base station. The second baseline (baseline 2) assumes that both the home UAV $u \in \mathcal{U}$ and neighbor UAV $u' \in \mathcal{U}$ are cache-enabled, but no cache sharing is considered between them. To compare the performance, we focused on two metrics, namely the average delay and average rate usage of the system, which also define the load on the system.

We also conducted simulations with different numbers of users, ranging from 30 to 80, and two different cache sizes of 20 MB and 40 MB.

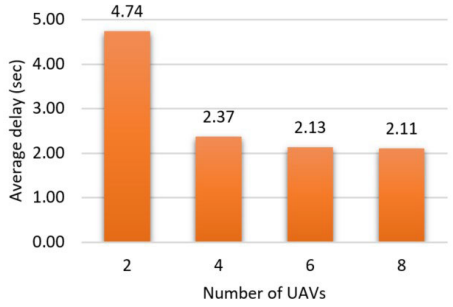
B. SYSTEM PERFORMANCE

Figure 4 shows the system's performance in terms of the number of episodes. Figures 4a, 4b, and 4c show the percentages of cache hits for users' requested content in their home UAVs, neighbor UAVs, and base station requests, respectively.

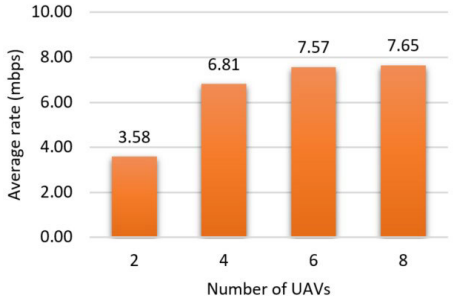
At the start of the simulations, the cache hit rate in the home UAVs was 16.4%, with 9.9% in the neighbor UAVs, and the remaining 73.5% of requests were fulfilled via the base station. Over time, the home and neighbor UAVs learned to cache new content, increasing the cache hit rate to 20.5% and 14.9%, respectively. This led to a reduction in the average request from the base station to 64.5.

Figure 4d shows that the average delay at the start of the simulations was 2.6 seconds, which was directly proportional to the cache miss rate in the home and neighbor UAVs. However, as the cache miss rate decreased, the average delay also decreased to 2.3 seconds. Figure 4e shows the total average rate used in the system, which includes user-to-UAV, UAV-to-UAV, and UAV-to-ground base station links. The total average rate was directly proportional to the power consumption and average delay. As the cache hit rate increased in UAVs, the total average rate decreased from 7.3 Mbps to 6.8 Mbps.

Figure 4f shows the reward of the system, which is inversely proportional to the average delay and average rate while being directly proportional to the cache hits of the home and neighbor UAVs. The system received a higher reward of 10 and 5 when there were cache hits in the home and neighbor UAVs, respectively, and a penalty of -10 when a request went to the base station. At the start of the simulations, the reward was lower when the cache hit rate was lower in the home and neighbor UAVs, and the average delay and average rate were higher. However, as the DRL model learned, the reward of the system increased from 11 to 14.7. Next, we aimed to assess how well our approach scales by increasing the number of UAVs in the system. Additionally, we sought to determine the optimal number of UAVs required in a specific area for a given set of users. To explore this, we conducted the following simulation. In our simulation, we tested our proposed solution with a 20 MB storage capacity for each UAV. We varied the number of UAVs from 2 to 8 in a fixed service area containing 80 users in figure 5. Surprisingly, our findings suggest that beyond four UAVs, there is no significant difference in terms of average data rate and average delay, as illustrated in the figure. It's crucial to note that this observation is based on a fixed number of users and a specific service area. The fixed service area ensures ample capacity to serve users efficiently. Additionally, a practical consideration is that increasing the number of UAVs in the system would lead to higher overall costs. It's worth emphasizing that our proposed solution is designed to be scalable, accommodating any number of UAVs, and we have chosen four as a balanced and cost-effective configuration for the simulations presented in our study. As Figure 5 indicates, the point where the system performs optimally is with 4 UAVs.

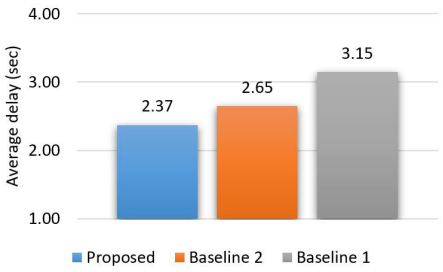


(a) Average delay, 20 MB cache size

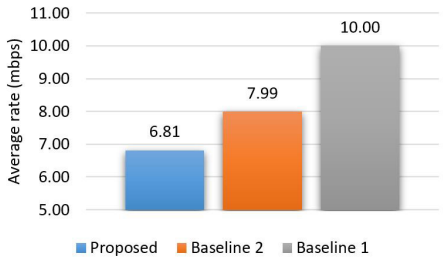


(b) Average rate, 20 MB cache size

FIGURE 5. Average delay and rate performance with respect to the number of UAVs.

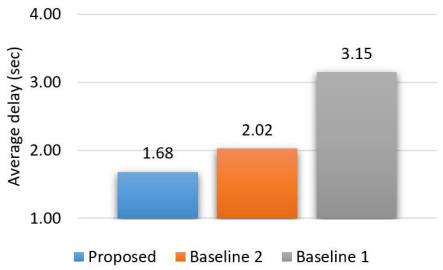


(a) Average delay comparison, 20 MB cache size

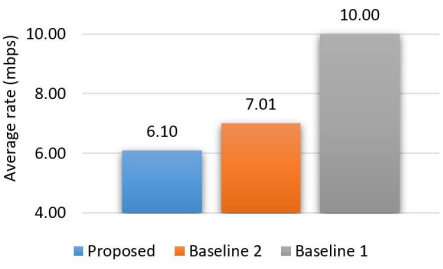


(b) Average rate comparison, 20 MB cache size

FIGURE 6. Comparison of the proposed approach with baselines, 20 MB cache size.



(a) Average delay comparison, 40 MB cache size



(b) Average rate comparison, 40 MB cache size

FIGURE 7. Comparison of the proposed approach with baselines, 40 MB cache size.

C. COMPARISON

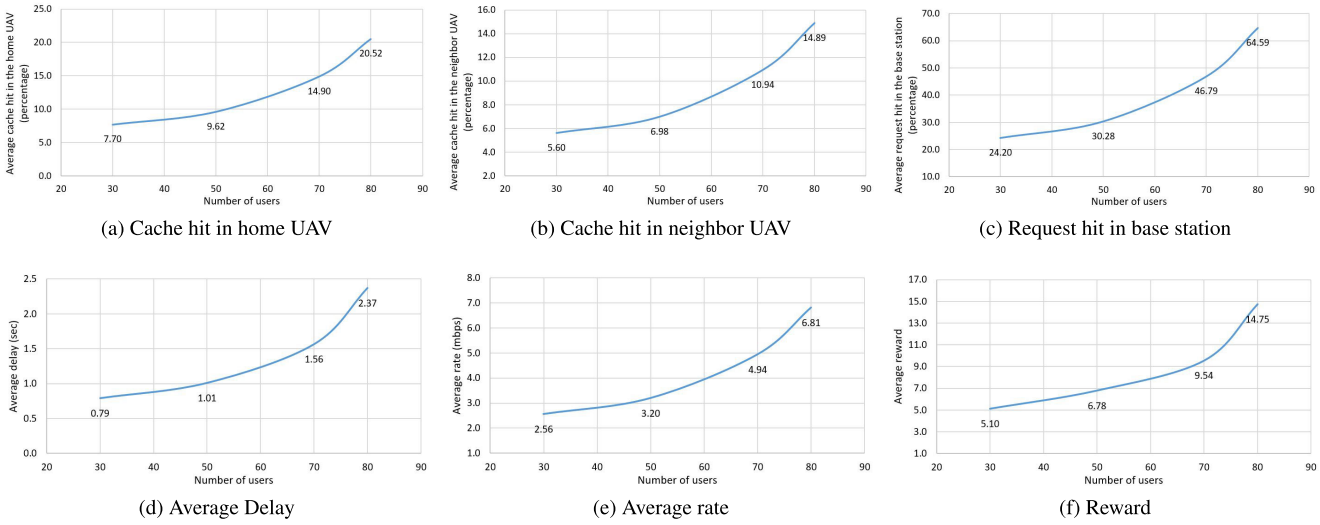
After evaluating the performance of our proposed solution with respect to the number of episodes, we conducted additional simulations to investigate the impact of cache size and the number of users associated with UAVs on the system's performance. Specifically, we performed simulations with two different cache sizes of 20 MB and 40 MB, and we varied the number of users in the system from 30 to 80. We analyzed the system's average delay, average rate usage, and cache hit percentages for home UAVs, neighbor UAVs, and the base station. These simulations allowed us to gain insights into the system's performance under different conditions and to evaluate the effectiveness of our proposed solution in varying scenarios.

1) SMALL CACHE SIZE

In this experiment, we evaluated our proposed solution with 20 MB cache size per UAV and a total of 80 users in the system. We compared our proposed solution with two baselines. Baseline 1 represents caching-disabled UAVs, and the requested content is downloaded via the base station. Baseline 2 represents caching-enabled UAVs but with no cache sharing between UAVs.

As shown in figure 6, our proposed approach outperforms both baselines in terms of average delay. Specifically, the average delay in our proposed approach is 2.37 seconds, while baseline 1 and baseline 2 have average delays of 3.15 seconds and 2.65 seconds, respectively. This indicates that our proposed approach is more efficient in delivering content to users with lower delay times. It is worth noting that the cache sharing mechanism between UAVs in our proposed approach contributes significantly to the reduction of delay. By sharing cached content between UAVs, the system can avoid redundant downloads of the same content from the base station, thus reducing the overall delay.

Fig. 6b illustrates the comparison of the average rate used in the system with the two baselines. It shows that our proposed solution achieved an average rate of 6.81 Mbps, which is better than the average rates achieved by Baseline 1 and Baseline 2, which were 10 Mbps and 7.99 Mbps,



(a) Cache hit in home UAV (b) Cache hit in neighbor UAV (c) Request hit in base station

(d) Average Delay (e) Average rate (f) Reward

FIGURE 8. System performance with respect to the number of users.

respectively. In other words, lower average rate depicts higher power efficiency and higher cache hits both at the home and the neighbor UAVs, thus, reducing the network traffic to retrieve the required content.

2) LARGE CACHE SIZE

In this simulation, we investigated the scenario where UAVs are equipped with a larger cache size of 40MB, and a total of 80 users are present in the system. The results obtained in this scenario are compared with the previously simulated scenario where UAVs were equipped with 20MB of cache storage. The comparison results show an improvement in the values of average delay and average rate.

Fig. 7 illustrates the comparison between our proposed solution and baseline 2 in terms of average delay and average rate. As shown in Fig. 7a, the average delay time of our proposed solution and baseline 2 reduced from 2.37 and 2.65 to 1.68 and 2.02, respectively. This improvement can be attributed to the larger cache size in UAVs, which can store more content and reduce the number of requests that need to be fulfilled from the base station.

Similarly, Fig. 7b shows that the average rate used in our proposed solution and baseline 2 reduced from 6.81 and 7.99 to 6.1 and 7.01, respectively. This reduction in average rate is due to the increased cache hit ratio in UAVs, which reduces the number of requests that need to be transmitted between the UAVs and the base station.

Overall, these results suggest that the system performance improves when the cache size in the UAVs is increased. Therefore, increasing the cache size of UAVs can be an effective way to improve the system performance in UAV-enabled communication networks.

Furthermore, Fig, 7b shows the average rate used in our proposed solution and baseline 2 reduced from 6.81 and 7.99 to 6.1 and 7.01. In other words, the system performance improves when the cache size in the UAV is increased.

3) DIFFERENT NUMBER OF USERS

In this simulation, we considered different numbers of users in the system, starting from the 30 up to 80. As Fig. 8 illustrates, the system's performance with respect to the number of users. It shows when the number of the users are small, the system is under utilized but as the number of users increases the system performance also increases. Figures 8a, 8b, and 8c show the corresponding cache hit ratios of the home UAVs, neighbor UAVs, and the base station. When UAVs are associated with 30 users, the system is not fully utilized, and most requests are served via the base station. However, as the number of associated users increases, the cache hit ratios of the home and neighbor UAVs improve, while that of the base station reduces. Specifically, the cache hit ratios in home UAVs, neighbor UAVs, and content served via the base station increase from 7.7%, 5.6%, and 24.2% to 20.5%, 14.8%, and 64.5%, respectively, when the number of associated users increases from 30 to 80.

Fig. 8d shows that the total average delay of the system increases as the number of associated users increases. When UAVs are associated with 30 users, the total average delay is 0.79 seconds. However, as the number of associated users increases to 80, the total average delay increases to 2.37 seconds. This is because as the number of users increase, traffic generated also increases, which incurs higher traffic load on the shared communication infrastructure causing higher delays in the system.

Fig. 8e illustrates the total average rate utilization of the system. When UAVs are associated with 30 users, the total average rate used in the system is 2.5 Mbps. However, as the UAVs start serving all 80 users, the total average rate increases to 6.8 Mbps which can be explained by higher utilization of communication infrastructure.

Fig. 8f shows the system rewards based on the number of users served by the UAVs. The results indicate that the system reward increases as the number of associated

users increases, since now UAVs serve higher number of users resulting in harvesting higher rewards. Overall, our simulation shows that the system performance improves as the number of users increases, but this also leads to an increase in the average delay of the system due to higher utilization of the shared infrastructure. In addition, the total average rate utilization also increases, indicating that the system is efficiently utilizing its resources.

VI. CONCLUSION

In conclusion, this paper addresses the issue of data traffic growth in future wireless networks and its potential impact on transmission delay. We proposed to use cache-enabled UAVs as aerial base stations to tackle this problem and presenting a novel framework based on DRL actor-critic, specifically deep deterministic policy gradient (DDPG), which leverages UAV collaboration and cache-sharing to optimize system performance. Our simulation results demonstrate that our proposed solution outperforms the two baseline solutions, which are cache-disabled UAVs and cache-enabled UAVs with no collaboration. We also investigated different scenarios, including varying the number of users associated with the UAVs and different cache sizes in the UAVs. Overall, our proposed solution significantly improves the system's performance, providing a promising solution for future wireless networks. As a future extension, we aim investigating the use of multiple-agent deep deterministic policy gradient (MADDPG) to further improve the system performance in a cooperative environment with multiple agents. MADDPG is expected to be more efficient than DDPG, which is limited to a single agent. This approach holds great potential in addressing the future challenges of wireless networks with the increasing demand for data traffic. Despite the rigorous methodology employed in this study, it's crucial to acknowledge certain limitations that will be addressed in our future work. Firstly, while we considered UAV-to-UAV collision as a constraint, there are other potential scenarios during UAV operations, such as collisions with objects like buildings, trees, walls, and other flying entities. Investigating the process of replacing UAVs after such incidents could be an intriguing avenue for further exploration. Secondly, we did not delve into specific methods aimed at optimizing the overall service time of UAVs to enhance energy efficiency. Our primary focus was minimizing transmission delay, considering transmission power as a constraint. However, recognizing one of the key challenges for UAVs lies in their endurance, future work may need to explore dedicated methods to optimize overall service time and enhance energy efficiency.

REFERENCES

[1] H. Wang, G. Ding, F. Gao, J. Chen, J. Wang, and L. Wang, "Power control in UAV-supported ultra dense networks: Communications, caching, and energy transfer," *IEEE Commun. Mag.*, vol. 56, no. 6, pp. 28–34, Jun. 2018.

[2] W. Lee, H. Lee, and H.-H. Choi, "Deep learning-based network-wide energy efficiency optimization in ultra-dense small cell networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 6, pp. 8244–8249, Jun. 2023.

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

[4] J. Qiao, Y. He, and X. S. Shen, "Proactive caching for mobile video streaming in millimeter wave 5G networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 10, pp. 7187–7198, Oct. 2016.

[5] A. Haghrah, A. Haghrah, J. M. Niya, and S. Ghaemi, "Handover triggering estimation based on fuzzy logic for LTE-A/5G networks with ultra-dense small cells," *Soft Comput.*, vol. 27, no. 22, pp. 17333–17345, Nov. 2023.

[6] T. Naous, M. Itani, M. Awad, and S. Sharafeddine, "Reinforcement learning in the sky: A survey on enabling intelligence in NTN-based communications," *IEEE Access*, vol. 11, pp. 19941–19968, 2023.

[7] Q. Zhu, J. Zheng, and A. Jamalipour, "Coverage performance analysis of a cache-enabled UAV base station assisted cellular network," *IEEE Trans. Wireless Commun.*, vol. 22, no. 11, pp. 8454–8467, Nov. 2023.

[8] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, "Survey on UAV cellular communications: Practical aspects, standardization advancements, regulation, and security challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3417–3442, 4th Quart., 2019.

[9] M. Maboudi, M. Homaei, S. Song, S. Malihi, M. Saadatseresht, and M. Gerke, "A review on viewpoints and path planning for UAV-based 3D reconstruction," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 5026–5048, 2023.

[10] M. Chen, W. Saad, and C. Yin, "Liquid state machine learning for resource allocation in a network of cache-enabled LTE-U UAVs," in *Proc. IEEE Global Commun. Conf.*, Dec. 2017, pp. 1–6.

[11] R. Amer, W. Saad, H. ElSawy, M. M. Butt, and N. Marchetti, "Caching to the sky: Performance analysis of cache-assisted CoMP for cellular-connected UAVs," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2019, pp. 1–6.

[12] C. Wu, S. Shi, S. Gu, L. Zhang, and X. Gu, "Deep reinforcement learning-based content placement and trajectory design in urban cache-enabled UAV networks," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–11, Aug. 2020.

[13] Y. Wang, S. Fu, C. Yao, H. Zhang, and F. R. Yu, "Caching placement optimization in UAV-assisted cellular networks: A deep reinforcement learning-based framework," *IEEE Wireless Commun. Lett.*, vol. 12, no. 8, pp. 1359–1363, Aug. 2023.

[14] G. T. Maale, G. Sun, N. A. E. Kuadey, T. Kwantwi, R. Ou, and G. Liu, "DeepFESL: Deep federated echo state learning-based proactive content caching in UAV-assisted networks," *IEEE Trans. Veh. Technol.*, vol. 72, no. 9, pp. 12208–12220, Sep. 2023.

[15] R. Wang, X. Peng, J. Zhang, and K. B. Letaief, "Mobility-aware caching for content-centric wireless networks: Modeling and methodology," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 77–83, Aug. 2016.

[16] J. Yang, S. Xiao, B. Jiang, H. Song, S. Khan, and S. U. Islam, "Cache-enabled unmanned aerial vehicles for cooperative cognitive radio networks," *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 155–161, Apr. 2020.

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

[18] S. Fu, X. Feng, A. Sultana, and L. Zhao, "Joint power allocation and 3D deployment for UAV-BSs: A game theory based deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 23, no. 1, pp. 736–748, Jan. 2024.

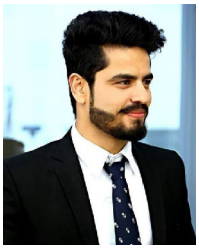
[19] T. Zhang, Z. Wang, Y. Liu, W. Xu, and A. Nallanathan, "Caching placement and resource allocation for cache-enabling UAV NOMA networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12897–12911, Nov. 2020.

[20] K. Song, J. Zhang, Z. Ji, J. Jiang, and C. Li, "Energy-efficiency for IoT system with cache-enabled fixed-wing UAV relay," *IEEE Access*, vol. 8, pp. 117503–117512, 2020.

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

[22] J. Ji, K. Zhu, D. Niyato, and R. Wang, "Joint cache placement, flight trajectory, and transmission power optimization for multi-UAV assisted wireless networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5389–5403, Aug. 2020.

[23] S. Chai and V. K. N. Lau, "Online trajectory and radio resource optimization of cache-enabled UAV wireless networks with content and energy recharging," *IEEE Trans. Signal Process.*, vol. 68, pp. 1286–1299, 2020.
[24] A. A. Khuwaja, Y. Zhu, G. Zheng, Y. Chen, and W. Liu, "Performance analysis of hybrid UAV networks for probabilistic content caching," *IEEE Syst. J.*, vol. 15, no. 3, pp. 4013–4024, Sep. 2021.
[25] Y. Wang, C. Feng, T. Zhang, Y. Liu, and A. Nallanathan, "QoE based network deployment and caching placement for cache-enabling UAV networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.
[26] S. W. Kang, K. Thar, and C. S. Hong, "Unmanned aerial vehicle allocation and deep learning based content caching in wireless network," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2020, pp. 793–796.
[27] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "Wireless networks with cache-enabled and backhaul-limited aerial base stations," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7363–7376, Nov. 2020.
[28] B. Jiang, J. Yang, H. Xu, H. Song, and G. Zheng, "Multimedia data throughput maximization in Internet-of-Things system based on optimization of cache-enabled UAV," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3525–3532, Apr. 2019.
[29] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.
[30] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.
[31] J. Schulman, S. Levine, P. Abbeel, M. I. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. ICML*, 2015, pp. 1889–1897.
[32] K. Zheng, Y. Sun, Z. Lin, and Y. Tang, "UAV-assisted online video downloading in vehicular networks: A reinforcement learning approach," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–5.
[33] C. T. Cicek, H. Gultekin, B. Tavli, and H. Yanikomeroglu, "UAV base station location optimization for next generation wireless networks: Overview and future research directions," in *Proc. 1st Int. Conf. Unmanned Vehicle Syst.-Oman (UVS)*, Oman, Feb. 2019, pp. 1–6.
[34] O. Kalinagac, S. S. Kafiloglu, F. Alagoz, and G. Gur, "Caching and D2D sharing for content delivery in software-defined UAV networks," in *Proc. IEEE 90th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2019, pp. 1–5.
[35] M. Chen, W. Saad, and C. Yin, "Echo-liquid state deep learning for 360° content transmission and caching in wireless VR networks with cellular-connected UAVs," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 6386–6400, Sep. 2019.
[36] S. M. A. Kazmi, T. N. Dang, I. Yaqoob, A. Manzoor, R. Hussain, A. Khan, C. S. Hong, and K. Salah, "A novel contract theory-based incentive mechanism for cooperative task-offloading in electrical vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 8380–8395, Jul. 2022.
[37] S. M. A. Kazmi, S. Otoum, R. Hussain, and H. T. Mouftah, "A novel deep reinforcement learning-based approach for task-offloading in vehicular networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.
[38] T. M. Ho and K.-K. Nguyen, "Joint server selection, cooperative offloading and handover in multi-access edge computing wireless network: A deep reinforcement learning approach," *IEEE Trans. Mobile Comput.*, vol. 21, no. 7, pp. 2421–2435, Jul. 2022.
[39] S. M. A. Kazmi, T. M. Ho, T. T. Nguyen, M. Fahim, A. Khan, M. J. Piran, and G. Baye, "Computing on wheels: A deep reinforcement learning-based approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 22535–22548, Nov. 2022.



HAMIDULLAH MUSLIH received the BCS degree in communication and operating system from Herat University, Herat, Afghanistan, in 2014, and the MCS degree in security and network engineering from Innopolis University, Innopolis, Russia, in 2021, where he is currently pursuing the Ph.D. degree. He is also an Instructor with the Department of Security and Network Engineering, Innopolis University. His current research interests include applying machine learning based solution to 6G and wireless communication networks. Additionally, he is also working toward optimizing UAVs planning and deployment for future networks.

S. M. AHSAN KAZMI received the Ph.D. degree from the Department of Computer Science and Engineering, Kyung Hee University, South Korea, in 2017. He was as a Postdoctoral Fellow with the Department of Computer Science and Engineering, Kyung Hee University, from 2017 to 2018. He is currently a Senior Lecturer of data science with the Department of Computer Science and Creative Technologies, University of the West of England, Bristol. Prior to this, he was an Assistant Professor with the Institute of Information Security and Cyber-Physical System, Innopolis University, Russia. His research interests include applying analytical techniques of optimization, machine learning, and game theory to radio resource management for future networks. He received the Best KHU Thesis Award in engineering in 2017 and several best paper awards from prestigious conferences.

MANUEL MAZZARA received the Ph.D. degree in computer science from the University of Bologna, Italy. He is currently a Professor of computer science with Innopolis University, Russia. He has published many relevant and highly cited articles, in particular in the field of service engineering and software architectures and has collaborated with European and U.S. industries, and governmental and inter-governmental organizations, such as United Nations, always at the edge between science and software production. His research interests include software engineering, service-oriented architecture, concurrency theory, formal methods, and software verification. The work conducted by him and his team in recent years focuses on the development of theories, methods, tools, and programs covering the two major aspects of software engineering, such as the process side, related to how we develop software; and the product side, concerning the results of this process.

GASPARD BAYE received the B.S. degree in information and communication technology (ICT) from The ICT University, Yaounde, Cameroon, and the M.Sc. degree (Hons.) in security and network engineering from Innopolis University. He is currently pursuing the Ph.D. degree with the Department of Computer and Information Science. He is a Research Assistant with the Cybersecurity Center, University of Massachusetts Dartmouth, and an NSA/DHS designated Center of Academic Excellence in Cyber Defense Research (CAER) for USA. He is a Prolific Researcher with a focus on applying OpenSet-based deep learning algorithms/large language models (LLMs) to improve security operations. He was the overall best-graduating student at The ICT University and received more than seven awards, including a special award from the President.