Deep Reinforcement Learning-Based Mobility-Aware UAV Content Caching and Placement in Mobile Edge Networks

Stephen Anokye, *Student Member, IEEE*, Daniel Ayepah-Mensah, Abegaz Mohammed Seid, Gordon Owusu Boateng, and Guolin Sun, *Member, IEEE*

***Abstract*—With the proliferation of smart mobile devices, there is now an ever increasing craving for higher bandwidth for end user satisfaction. Increasing mobile traffic leads to congestion of backhaul networks. One promising solution to this problem is the mobile edge network and consequently mobile edge caching. There is an emerging paradigm shift toward the use of unmanned aerial vehicles (UAVs) to assist the traditional cellular networks and also to provide connectivity in places where there are no small base stations or faulty ones as a result of some natural disaster such as flooding. Hence, UAVs can be used to assist in content caching as well. This work proposes the use of human centric features, random waypoint user mobility model, and deep reinforcement learning to predict the location of the UAVs and the contents to cache at the UAVs. We formulated our problem as a Markov decision problem (MDP) and proposed a dueling reinforcement learning-based algorithm to solve the MDP problem. Our simulation results prove that our algorithm converges to an optimal solution and performs better than other baseline reinforcement learning algorithms in terms of quality of experience satisfaction, transmit power, and cache resource utilization**

***Index Terms*—Mobile edge caching, mobile edge network (MEN), reinforcement learning (RL), unmanned aerial vehicle (UAV).**

I. Introduction

THE world is witnessing an astronomical growth of mobile traffic and an ever-increasing demand from the end users for a high bandwidth and quality of experience (QoE) because everything is now mobile and smart. Mobile data traffic has grown between 2011 and 2016 and is estimated to increase to 49.0 exabytes per month by 2021 [1]. Increasing mobile traffic leads to congestion of backhaul networks, which leads to a higher cost of operation and maintenance, a lower quality of service (QoS), and this also inhibits data delivery. The increasing demand for bandwidth coupled with greater QoE demands and performances is becoming too much for the current 4G technologies, and thus, the need for newer solutions in 5G. In order to meet the increasing data demands, small-cell networks will be widely deployed, which can achieve much higher throughput and energy efficiency [2]. Mobile edge network (MEN), with its architecture, is a promising solution to address the above-mentioned issues. By moving the network functions and resources closer to end users (i.e., the network edge), many benefits can be obtained, such as high data rates, low delay, improved energy efficiency, and flexible network deployment and management [3]. One innovative proposal to overcoming these challenges is caching the content. While the two common locations for caching the content are at the evolved packet core and the radio access network [4], certain popular contents (e.g., on-the-air TV series and popular music) are frequently requested. Such contents can be cached during off-peak times in the network edge, such as at base stations (BSs) and even on user devices [5]. Then, the contents are distributed to requesters through high-rate and low-cost MENs rather than transmitted through the backhaul network repeatedly.

There is an emerging paradigm shift toward the use of unmanned aerial vehicles (UAVs) to assist the traditional cellular networks in wireless communications to provide connectivity from the air-to-ground users. Such communication from the air is expected to be a major component of beyond 5G cellular networks. When mobile users move outside the cell coverage areas, the cached contents may not be effectively distributed to the users. In addition, when the user hands over to a new cell, the contents requested may not be cached, leading to extra delay and bandwidth consumption due to the caching on the new BS or long-distance fetch from the content server. Also, in drone cells, the limited front-haul capacity can hardly satisfy the demands of data-craving services [6]. To alleviate the pressure of small cells and reduce the cost of densely deployed small BSs (SBSs), UAVs can be exploited to assist small cells in providing high-speed transmission due to their low cost and high mobility. UAV-aided wireless networks can establish wireless connections without infrastructure, realize larger wireless coverage, and achieve higher transmission rate.

Manuscript received June 9, 2020; revised November 24, 2020; accepted May 11, 2021. Date of publication June 8, 2021; date of current version March 24, 2022. This work was supported in part by the National Natural Science Research Foundation of China, under Grant 61771098, in part by the Fundamental Research Funds for the Central Universities under Grant ZYGX2018J068, in part by the fund from the Department of Science and Technology of Sichuan province, under Grant 2017GFW0128, Grant 8ZDYF2265, Grant 2018JYO578, and Grant 2017JY0007, and in part by the ZTE Innovation Research Fund for Universities Program 2016. *(Corresponding author: Guolin Sun.)*

The authors are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: sanokye@umat.edu.gh; ayeps2000@gmail.com; mamsied2002@gmail.com; boatenggordon48@gmail.com; guolin.sun@uestc.edu.cn).

Digital Object Identifier 10.1109/JSYST.2021.3082837

There has been growing interests in research into the use of UAVs to assist in caching of popular contents in mobile networks because of the various advantages it brings to the network. In our scheme, we seek to use multiple UAVs while considering the mobility of users using a random waypoint user mobility model in a more interactive way through the use of a deep reinforcement learning (DRL) algorithm. This simulates our environment in a more realistic scenario. To the best of authors' knowledge, this is the first of such work. The main contributions of this article are based on the following.

1) We propose a generic algorithm to integrate the use of human-centric features, random waypoint user mobility model, and DRL to predict the location of the UAVs and the contents to cache at the UAVs to maximize the QoE satisfaction and reduce the transmit power of the UAVs. The proposed mobility model is much useful considering the scenario and algorithm employed because it is able to mimic a mobile user better in our simulation.
2) We model the content caching and placement problem as a Markov decision process (MDP) and propose a solution based on dueling DQN technique, which maximizes QoE satisfaction and minimizes UAVs' transmit power.
3) Extensive simulations are conducted to show that the proposed algorithm converges to an optimal solution. Also our proposed algorithm has a better performance in terms of the QoE satisfaction and the transmit power of the UAVs as compared to other reinforcement learning (RL) algorithms and the conceptor-based ecostate network (CBESN) in [6].

The rest of this article is organized as follows. We review related works in Section II. Section III presents system models, which consists of the transmission model, caching model, mobility model, and QoE model. We present the problem formulation in Section IV. Then, performance evaluation and simulation results are discussed in Section V. Finally, Section VI concludes this article.

II. Related Works

Many researchers have investigated the use of UAV(s) to assist in video content caching. The UAV(s) assist the traditional terrestrial cellular networks. In our related works, we take a look at papers that have indulged the use of UAVs in the caching of contents. The related works are grouped into two according to the methods used, machine learning (ML)-based and other optimization techniques. The second and third paragraphs of this section deals with the other optimization techniques, the fourth on the ML-based and the last paragraph of this section deals with a comparison of our article with similar ones.

The authors in the host of works done, employed the use of various forms of models and methods in solving the numerous problems associated with UAV caching. In [7], a multiUAV network was used to solve the joint performance optimization problem of drone-cell, edge caching, and edge computing using a geometric-based stochastic (GBS) channel modeling technology. In [8], the authors used a distributed algorithm to reduce the load of the wireless backhaul link and also save energy, by caching contents in a multi-UAV scenario. Amer *et al.* [9] proposed the use of coordinated multipoint transmission along with caching for providing seamless connectivity to aerial users. In particular, a network of clustered cache-enabled SBSs serving aerial users was considered in which a requested content by an aerial user is cooperatively transmitted from collaborative ground SBSs. For this network, a novel upper bound expression on the coverage probability was derived as a function of the system parameters. In [10], UAV assisted secure transmission for scalable videos in hyperdense networks via caching was studied. In the proposed scheme, UAVs can act as SBSs to provide videos to mobile users in some small cells. To reduce the pressure of wireless backhaul, UAVs and SBSs are both equipped with caches to store videos at off-peak time. To facilitate UAVs, a single antenna was equipped at each UAV, and thus, only the precoding matrices of SBSs were cooperatively designed to manage interference by exploiting the principle of interference alignment. On the other hand, the SBSs replaced by UAVs were idle. Thus, in order to guarantee secure transmission, the idle SBSs were further exploited to generate jamming signal to disrupt eavesdropping. Additionally, the authors in [11] employed the use of a statistical method to maximize the coverage of the UAV aided network by offloading traffic to the UAV while caching contents at the UAVs and used energy harvesting to enhance the power consumption of the UAV. In [12], the authors focused on a half-duplex decode-and-forward cache-enabled UAV relaying communication system in downlink scenario and obtain a semiclosed-form of the optimal UAV placement by maximizing the system average achievable rate. In [13], a novel scheme to guarantee the security of UAV-relayed wireless networks with caching via jointly optimizing the UAV trajectory and time scheduling was proposed. For every two users that have cached the required file for the other, the UAV broadcasts the files together to these two users, and the eavesdropping can be disrupted. For users without caching, minimum average secrecy rate was maximized by jointly optimizing the trajectory and scheduling, with the secrecy rate of the caching users satisfied. Due to the nonconvexity of the corresponding optimization problem, an iterative algorithm via successive convex optimization (IASCO) was proposed to solve it approximately.

Furthermore, Zhang *et al.* [14] developed an optimization framework to maximize utility (delay, data rate) and optimized the transmission power of the vehicles and the trajectory of the UAV whilst caching social contents in a social Internet of vehicles scenario through the use of a single UAV. The authors of [15] in a vehicle-to-everything (V2X) scenario, proposed a novel UAV-assisted data dissemination protocol with proactive caching at the vehicles and an advanced file sharing strategy for revolutionizing communications. Specifically, in the proactive caching phase, UAVs were employed to act as flying BSs for information interactions. Considering the time-variant network topology, a spatial scheduling (SS) algorithm for the trajectory optimization of each UAV, which can expedite the caching process and boost the system throughput was proposed. Then, in the file sharing phase, based on the previous caching status, a relay ordering algorithm to enhance the network transmission performance was provided. Xu *et al.* used a greedy-based heuristic optimization approach to optimize the caching policy, UAV

trajectory, and communication scheduling in [16]. The authors proposed a novel scheme to overcome the UAV endurance issue, by utilizing the promising technique of proactive caching at the ground networks (GNs) where a single UAV is dispatched to serve a group of GNs with random and asynchronous requests. Some optimization methods were used in [17] and [18] to maximize throughput while minimizing the signal-to-interference-plus-noise-ratio (SINR) of the UAV in the former and minimizing the delay by optimizing the caching strategy and location of the UAVs in the latter.

The paper [19] investigated the problem of cache node placement and selection with the coexistence of UAVs cache and device-to-device cache in mobile networks. Different from the conventional caching approaches assuming ground users remain static, the dynamic movement design of UAV to maximize the cache-aided throughput taking into account the movement of ground users was considered in this article. As the formulated optimization problem was NP-hard, a mobility-aware probabilistic caching algorithm in which K-means clustering is utilized to obtain the partition of ground users was proposed. Chen *et al.* used multi-UAVs connected to core network and some mobile users to assist in content caching. ML algorithms were used to solve their respective problems in [6] and [20]. Chen *et al.* used a CBESN to maximize the QoS and minimize the transmission power of the UAVs in [6] compared to the liquid sate machine (LSM) used in [20] to maximize the queue stability requirements of users. The authors in [21] proposed an online UAV-assisted wireless caching design via jointly optimizing UAV trajectory, transmission power, and caching content scheduling, formulated the joint optimization of online UAV trajectory and caching content delivery as an infinite-horizon ergodic MDP problem to obtain a QoE-optimal solution based on the concept of request queues in wireless caching networks. Based on this, an actor–critic based online RL algorithm was proposed to solve the problem.

Although most of the caching problems have been solved using various optimization techniques, it introduces a lot of complexities in the solutions. Most of the solutions either used a single UAV, did not consider the mobility of the users or did not use ML techniques in their solutions. The scenario of V2X and the method of SS algorithm used in [15] are in contrast with our UAV network scenario and dueling RL method. The works in [6] and [20] though employed a conceptor ecostate network and liquid state ML, respectively, using multiple UAVs whilst taking into account the human centric features such as mobility of users in caching the contents at the UAVs, the methods used differ from our proposed dueling RL algorithm and our random waypoint mobility model mimics mobile users very well in our simulations. Lastly, even though Chai and Lau [21] used the RL (actor–critic) method in its solution, the proposed scheme did not consider the mobility of the users and our proposed dueling technique can learn which states are (or are not) valuable, without having to learn the effect of each action for each state, which translates in a better convergence rate than actor–critic. It can be observed from the aforementioned works that this is the first article to deploy multiple UAVs taking into account users' mobility based on random waypoint user mobility model and integrating it with dueling DRL technique to maximize QoE satisfaction of users and minimize the transmit power of UAVs. Table I gives a summary of the related works

TABLE I
SUMMARY OF RELATED WORKS

Approach	Methods	Algorithms	Literature
ML based	RL	actor-critic	[21]
	Other Learning Techniques	k-means, LSM, CBESN	[6],[19],[20]
Optimization and other methods		GBS channel modeling , distributed algorithm, statistical method, IASCO, SS algorithm.	[7],[8],[9], [10],[11],[12], [13],[14],[15], [16],[17],[18]

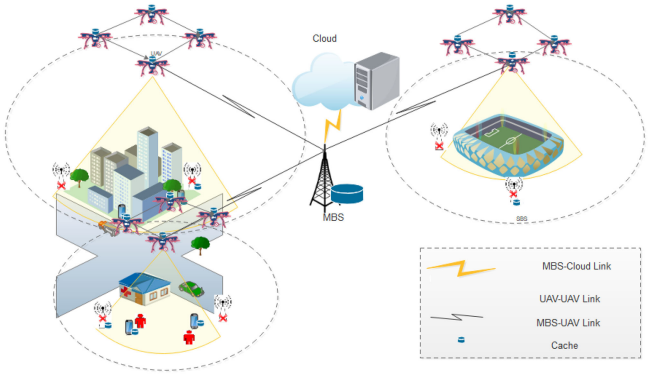


Fig. 1. Network architecture.

III. SYSTEM MODEL

In our scheme, we consider the downlink of a MEN, which services a set of $i \in \{1, 2, \ldots, I\}$ mobile users connected to a macrobase station (MBS). A set $k \in \{1, 2, \ldots, K\}$ cache-enabled UAVs acting as flying BSs (relays) are deployed to serve the mobile ground users. We consider air-to-ground transmission using mmWave frequency spectrum due to the high altitudes of the UAVs. The MBS is connected to the core network via a fiber cable. Fig. 1 shows the network architecture of our scenario. In this model, the content server stores N contents of equal sizes that are required by all users. The UAVs would be used to store popular contents that the users request. Caching at the UAVs allows the servicing of mobile users whose QoE requirements cannot be satisfied by the SBSs. We denote the set of C cached contents of the UAV k by C_k where $C \leq N$ and $k \in K$. For simplicity, we assume that each user can request at most one content during a specified time slot τ. The UAVs would remain static during each content transmission. Their locations would be updated according to the mobility patterns of the users after each content transmission is complete at its current location. A list of notations is provided in Table II. The UAVs, which are connected to the MBS are also connected to the cloud. We assume there are no or faulty SBSs and mobile users $i \in I$, therefore, need the UAVs to stay connected. The aim is for the agent to use the human centric features of the users and the other devices

TABLE II
LIST OF NOTATION

Symbol	Meaning	Symbol	Meaning
I	Number of users	C	Number of contents stored at UAV cache
P_M	Transmit power of MBS	F	Number of intervals in each time slot
M	MBS	H	Number of time slots to collect user mobility
K	Number of UAVs	P_k	Transmitted power of UAV
N	Number of contents	$d_{t,ki}$	Distance between UAVs or SBSs and users
L_{FS}	Free space path loss	$\chi\sigma_{LoS}$ and $\chi\sigma_{NLoS}$	Shadowing random variables
f_c	Carrier frequency	$C^H_{\tau,si}$	Rate of SBS-user link
μ_{LoS} and μ_{NLoS}	Path loss exponents	τ, $\Delta\tau$	Time slot, time slot duration
$\gamma^V_{t,ki}$	SINR of user i	σ^2	Variance or gaussian noise
$C^V_{\tau,ki}$	Rate of UAV-user link	d_o	Free space distance reference
t, Δt	Interval, interval duration	ϕ_t	Elevation angle
$D_{\tau,i,n}$	Delay	$l^{LoS}_{t,ki}$, $l^{NLoS}_{t,ki}$	Path loss from UAV k to user i
$C^V_{\tau,ki}$	Rate of UAV-user link	$L^{LoS}_{t,ki}$, $L^{NLoS}_{t,ki}$	Path loss from MBS to UAV k
$l_{t,ki}$	Path loss of UAVs-users	$C^F_{\tau,ki}$	Fronthaul rate of UAV
$w_{\tau,t,k}=[x_{\tau,k},y_{\tau,k},h_{\tau,k}]$	Coordinate of UAVs	B_k	Total available bandwidth to UAVs

to determine the location of the UAVs and also the contents to cache at UAVs since the users are mobile.

A. Transmission Model

1) User-UAV Links: The line-of-sight (LoS) and non-line-of-sight (NLoS) path loss of UAV k located at $(x_{\tau,k}, y_{\tau,k}, h_{\tau,k})$ transmitting a content to user i at interval t of time slot τ is expressed as

$$l^{\text{LoS}}_{t,ki}\left(w_{\tau,t,k}, w_{\tau,t,i}\right) = L_{\text{FS}}\left(d_o\right) + 10\mu_{\text{LoS}} \log\left(d_{t,ki}\left(w_{\tau,t,k}, w_{\tau,t,i}\right)\right) + \chi\sigma_{\text{LoS}} \tag{1}$$

$$l^{\text{NLoS}}_{t,ki}\left(w_{\tau,t,k}, w_{\tau,t,i}\right) = L_{\text{FS}}\left(d_o\right) + 10\mu_{\text{NLoS}}\log\left(d_{t,ki}\left(w_{\tau,t,k}, w_{\tau,t,i}\right)\right) + \chi\sigma_{\text{NLoS}} \tag{2}$$

where $w_{\tau,t,k}=[x_{\tau,k}, y_{\tau,k}, h_{\tau,k}]$ is the coordinate of the UAV k during time slot τ, $w_{\tau,t,i}=[x_{t,i}, y_{t,i}]$ is the time-varying coordinate of user i at interval t, $L_{\text{FS}}(d_o) = 20\log(d_o f_c 4\pi/c))$ is the free-space path loss, d_o denotes the free-space reference distance, f_c is the carrier frequency, c is the speed of light, $d_{t,ki}(w_{\tau,t,k}, w_{\tau,t,i}) = \sqrt{(x_{t,i}-x_{\tau,k})^2 + (y_{t,i}-y_{\tau,k})^2 + {h_{\tau,k}}^2}$ is the distance between user i and UAV k, μ_{LoS} and μ_{NLoS} are the path loss exponents for LoS and NLoS links, $\chi\sigma_{\text{LoS}}$ and $\chi\sigma_{\text{NLoS}}$ are the shadowing random variables.

The LoS probability $P_r(l^{\text{LoS}}_{t,ki})$ is calculated as

$$P_r(l^{\text{LoS}}_{t,ki}) = (1 + X_{\text{exp}}(-Y[\phi_t - X]))^{-1} \tag{3}$$

where X and Y are constants depending on the environment (rural, urban, dense, etc), $\phi_t = \text{Sin}^{-1}(h_{\tau,k}/d_{t.ki}(w_{\tau,t,k}, w_{\tau,t,i}))$ is the angle of elevation.

The average path loss from the UAV k to user i at interval t is

$$\bar{l}_{t,ki}\left(w_{\tau,t,k}, w_{\tau,t,i}\right) = P_r\left(l^{\text{LoS}}_{t,ki}\right) \times l^{\text{LoS}}_{t,ki} \times P_r\left(l^{\text{NLoS}}_{t,ki}\right) \times l^{\text{NLoS}}_{t,ki} \tag{4}$$

where $P_r(l^{\text{NLoS}}_{t,ki}) = 1 - P_r(l^{\text{LoS}}_{t,ki})$.

The average SINR of user i located at $(w_{\tau,t,i})$ from the associated UAV k at an interval t is given by

$$\gamma^V_{t,ki} = \frac{P_{t,ki}}{10^{\bar{l}_{t,ki}(w_{\tau,t,k}, w_{\tau,t,i})/10}\sigma^2} \tag{5}$$

where $P_{t,ki}$ is the transmit power of UAV k to user i at time interval t and σ^2 is the variance of the Gaussian noise.

The channel capacity between UAV k and user i is

$$C^V_{\tau,ki} = \frac{1}{F_{\tau,i}} \sum_{t=1}^{F_{\tau,i}} \frac{B_k}{U_k} \log_2(1 + \gamma^V_{t,ki}). \tag{6}$$

The total bandwidth available for each UAV, which is equally divided among the associated users is B_k. The number of users associated with UAV k is U_k and $F_{\tau,i}$ is the number of intervals that user i uses to receive a content during time slot τ.

2) MBS-UAV links: The LoS and NLoS path loss from the MBS to UAV k at time interval t within time slot τ can be given by

$$L^{\text{LoS}}_{t,ki} = d_{t,ki}\left(w_{\tau,t,k}, w_{\tau,t,M}\right)^{-\beta} \tag{7}$$

$$L^{\text{NLoS}}_{t,ki} = \eta d_{t,ki}\left(w_{\tau,t,k}, w_{\tau,t,M}\right)^{-\beta} \tag{8}$$

where $w_{\tau,t,M}=[x_M, y_M]$ is the location of the MBS and β is the path loss exponent. The LoS connection probability and the average SNR of the link between the MBS and the UAV k can be calculated using (3)–(5).

B. Caching Model

In this system, we consider that the file database containing N files, which denotes as $n \in \{1, 2, 3, \ldots, N\}$. For simplicity and tractability, we assume that the total N contents are with the same normalized size. The content popularity is denoted by $f = \{f_1, f_2, f_3, \ldots, f_N\}$, where f_n is the requested probability for the nth file and the constraint is $0 \le f_n \le 1$ for all $n = \{1, 2, 3, \ldots, N\}$, which are sorted in terms of the decreasing order of the content popularity f_n. In this article, we assume that the file popularity f_n satisfies the Zipf law [22], [23]. The

popularity of the nth ordered content is written as

$$f_n = \frac{1/n^{\alpha}}{\sum_{j=1}^{N} 1/j^{\alpha}} \tag{9}$$

where α denotes the skewness of the file popularity. A large α denotes the content items are of concentrated distribution. For simplicity, we further assume that each UAV has a limited cache storage of capacity C, and thus, each UAV is able to store at most C_k popular contents. We employ the probability caching scheme [24] where each device independently stores content f_n in a certain probability p_n. The content caching probability of the content is denoted as $P = \{p_1, p_2, p_3, \ldots, p_N\}\ n \in [1, N]$. The cache capacity restriction for each UAV can be given by $\sum_{n=1}^{N} p_n \leq C_k$ owing to the limited cache capacity. Considering that the UAVs cannot store all required files, appropriate content placement in UAV is fairly important. The data caching is able to increase the transmission rates, thereby, improving the delay.

C. Mobility Model

We employ the random waypoint user mobility model. In this model, a user begins by staying in one position for a specified period of time after which the user moves in a direction determined by an angle uniformly distributed between $[0,\ 2\pi]$ and the user is assigned a random speed of a pedestrian between $[0,\ C_{\max}]$. Upon arrival at its destination, the user pauses for a specified time period before starting the process again. The mobility pattern for each user will then be used to determine the content that must be cached as well as the optimal location of each UAV, which will naturally impact the QoE of each user. In this model, the associations of the mobile users with the UAVs can change depending on the QoE requirement. Since the users are moving continuously, the locations of the UAVs must change accordingly so as to serve the users effectively. However, for tractability, we assume that the UAVs will remain static during each content transmission. In essence, the UAVs will update their locations according to the mobility of the users after each content transmission is complete at a current location.

D. QoE Model

In the considered system, contents can be transmitted to the users via the following two types of links: (a) content server-MBS-user, and (b) UAV cache-user. The backhaul link connecting the MBS to the core network is assumed to be fiber and, therefore, its delay is neglected. Thus, the delay of a user i receiving content n over the two types of links at each time slot τ can be written as

$$D_{\tau,i,n} = \begin{cases} \frac{L}{C^F_{\tau,k}} + \frac{L}{C^V_{\tau,ki}}, & \text{link (a)} \\ \frac{L}{C^V_{\tau,ki}}, & \text{link (b)} \end{cases} \tag{10}$$

where $C^F_{\tau,k}$ is the rate of content transmission from the MBS to UAV k. The lower bound of the delay for each user i receiving content n is given by $\min\{\frac{L}{v_{U_k}}, \frac{L}{C^{\max}_K}\} \leq D_{\tau,i,n}$, where $C^{\max}_K = B_k \log_2(1 + \frac{P_{\max}}{10(L_{\text{FS}}(d_o)+10\mu_{\text{LoS}}\log(h_{\min})-4\sigma_{\text{LoS}})/10\sigma^2})$ with $P_{\max}$ being the maximum transmit power of each UAV, v_{U_k} the fronthaul rate of each user receiving contents from UAVs and $h_{\min}$ being the minimum altitude of the UAV. From [6], we can see that the minimum delay of each user depends on the rate of the fronthaul links and the maximum transmit power of the UAVs. Therefore, we can improve the QoE of each user by adjusting the UAV's transmit power. In particular, as the number of users increases and the rate of fronthaul links decreases, the QoE requirement of users can be satisfied by adjusting the UAVs' transmit power. Note that, the upper bound of the delay τ is set based on the desired system requirement. We can categorize the sensitivity to the delay into five groups using the popular mean opinion score (MOS) model [25], which is often used to measure the QoE of a wireless user. The mapping between delay and MOS model is given by

$$\overline{D}_{\tau,i,n} = \frac{\Delta_\tau - D_{\tau,i,n}}{\Delta_\tau - \min\left\{\frac{L}{v_{U_k}}, \frac{L}{C^{\max}_K}\right\}} \tag{11}$$

which is shown in Table III. The QoE of each user i receiving content n at time slot τ can be given by [26]

$$Q_{\tau,i,n} = \zeta \overline{D}_{\tau,i,n}. \tag{12}$$

TABLE III
MEAN OPINION SCORE MODEL

QoE	Poor	Fair	Good	Very Good	Excellent
Interval scale	0.0-0.2	0.2-0.4	0.4-0.6	0.6-0.8	0.8-1.0

IV. PROBLEM FORMULATION

The caching problem can be modeled as an MDP and solved using the dueling RL algorithm with the objective of maximizing the user's QoE and minimizing the average total transmit power of the UAVs. This section briefly presents the MDP, RL, and dueling DQN.

A. Markov Decision Process

Let $S = \{s_1, s_2, \ldots s_l\}$ denote the state space and let $A = \{a_1, a_2, \ldots a_m\}$ denote the action set. The agent takes an action $a(t)(a(t) \in \text{A})$ based on the current state $s(t)(s(t) \in \text{S})$. Then, the system transfers to a new state $s(t+1)(s(t+1) \in \text{S})$ with the transition probability $P_{s(t)(s+1)}(a)$ and obtains the immediate reward $r(s(t),\ a(t))$. For the long-term consideration, the target is the future reward that is characterized by a discount factor $\epsilon (0 < \epsilon < 1)$. The RL agent aims to determine an optimal policy $a^*(a^* = \pi^*(s) \in \text{A})$ for each state s, which maximizes the expected time-average reward. This quantity is expressed as

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \epsilon^t r(s(t), a(t)) | s(0) = s\right] \tag{13}$$

where $\mathbb{E}$ denotes the expectation. The environment is formulated as an MDP. Hence, the value function can be rewritten as

$$V^{\pi}(s) = R\left(s, \pi(s)\right) + \epsilon \sum_{s' \in \text{S}} P_{ss'}(\pi(s)) V^{\pi}(s') \tag{14}$$

where $R(s,\pi(s))$ is the mean value of the immediate reward $r(s,\pi(s))$and $P_{ss'}(\pi(s))$ is the transition probability from s to s', when the action $\pi(s)$ is executed. The optimal policy π^* follows Bellman's criterion as:

$$V^{\pi^*}(s) = \max_{a\in \mathrm{A}} \left[R(s,a) + \sum_{s'\in \mathrm{S}} P_{ss'}(a) V^{\pi^*}(s') \right]. \tag{15}$$

In the model-based learning mechanism, the reward R and the transition probability P are given. Thus, the optimal policy can be derived. We now consider the model-free RL, where both R and P are unknown. Note that Q-learning is one of the strategies to determine the best policy π^*. A state-action function, namely Q-function, is defined as

$$Q^{\pi}(s,a) = R(s,a) + \epsilon \sum_{s'\in \mathrm{S}} P_{ss'}(a) V^{\pi}(s') \tag{16}$$

which represents the discounted cumulative reward, when action a is performed at state s and continues optimal policy from that point on. The maximum Q-function is expressed as

$$Q^{\pi^*}(s) = \ R(s,a) + \epsilon \sum_{s'\in \mathrm{S}} P_{ss'}(a) V^{\pi^*}(s'). \tag{17}$$

The discounted cumulative state function can be written as

$$V^{\pi^*}(s) = \max_{a\in \mathrm{A}} Q^{\pi^*}(s,a). \tag{18}$$

We now aim to estimate the best Q-function instead of finding the best policy. The Q-function can be obtained by using the recursive method [35]

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha(r + \epsilon \max_{a'} Q_t\,(s',a') - Q_t(s,a)\ ) \tag{19}$$

where α is the learning rate. Furthermore, $Q_t(s,a)$ definitely converges to $Q^*(s,a)$, when a proper α is selected [36]. One way to estimate the Q-function is to use a function approximation such as the neural network $Q(s,a;\theta) \approx Q^*(s,a)$, where the parameter θ is the weight of the neural network [27]. In particular, θ is adjusted at each iteration during the Q network training in order to reduce the mean square error. Note that the neural network acts as a function approximation. Hence, we can represent any number of possible states as a vector. Then, the system learns to map these states to the Q-values. Although the combination of the neural networks and the Q-learning makes the RL algorithms more flexible and efficient, it still exhibits some instabilities. Therefore, deep Q-learning is introduced to deal with this issue as well as to enhance the RL algorithms. This novel mechanism will be presented the following section.

B. Overview of RL

RL aims to find a balance between exploration (of uncharted territory) and exploitation [28]. In RL, the environment is typically formulated as an MDP, which is described by four tuples (S, A, R, P), set of possible state S, set of available actions A, and reward function $R : S \times A \to R$ and transition probability $P(s' \mid s,a) \to [0,1]$. However, the state space, the explicit transition probability, and the reward function are not necessarily required [28], [27]. The agent interacts with an unknown environment through the repeated observations, actions, and rewards to construct the optimal strategy. It is a promising approach to deal with tasks with high complexity in the real world [27]. There are two kinds of RL, namely model-free and model-based RL, depending on whether the transition probability is given or not. Moreover, a variety of model-free and model-based algorithms [29] can be used to approximate the reward functions. We now briefly describe the dueling DQN as follows.

C. Dueling DQN

Dueling architecture is a model-free algorithm developed by Wang *et al.* [30] and draws its inspiration from residual RL and the concept of advantage learning and updating by Baird [31]. The key insight behind our new dueling architecture is that for many states, it is unnecessary to estimate the value of each action choice because in those states the choice of action has no repercussion on what happens. It takes advantage of all the features of DQN. DQN is developed to make the RL applicable to the real applications. It requires two improvements to transform the regular Q-learning to deep Q-learning. The first one is an experience replay. At each time instant t, an agent stores its interaction experience tuple, $e(t) = (s(t), a(t), r(t), s(t+1))$ into a replay memory, $D(t) = \{e(1), \ldots, e(t)\}$. Different from the traditional Q-learning, where the arrived samples are used to train neural network's parameters, DQN randomly selects the samples experience pool to train the deep neural network's parameters. The second modification is that DQN updates the weight every N time steps, instead of updating it every time step. By doing so, the learning process becomes more stable. The deep Q-function is trained toward the target value by minimizing the loss function, Loss(θ), at each iteration. The loss function can be written as

$$\mathrm{Loss}\,(\theta) = \ \mathbb{E}\left[(y - Q(s,a,\theta)^2) \right] \tag{20}$$

where the target value y is expressed as $y = r + \max_{a'} Q(s',a',\theta_i^-)$. In the Q-learning, the weights $\theta_i^- = \theta_{i-1}$, whereas in deep Q-learning $\theta_i^- = \theta_{i-N}$.

Dueling DQN uses two sequences (or streams) of fully connected layers. The streams are constructed such that they have the capability of providing separate estimates of the value and advantage functions, $V(s;\theta,\beta)$ and $A(s,\ a;\theta,\alpha)$, respectively. Finally, the two streams are combined to produce a single output Q function

$$Q\,(s,a;\theta,\alpha,\beta) = \ V\,(s;\theta,\beta) + \ A(s,\ a;\theta,\alpha). \tag{21}$$

To make (21) identifiable, it is made to implement forward mapping

$$Q\,(s,a;\theta,\alpha,\beta) = \ V\,(s;\theta,\beta) + \left(A\,(s,\ a;\theta,\alpha) - \max_{a'\in|\mathcal{A}|} A(s,\ a';\theta,\alpha) \right). \tag{22}$$

An alternative module replaces the max operator with an average

$$Q\,(s,a;\theta,\alpha,\beta) = \ V\,(s;\theta,\beta)$$

$$+\left(A\left(s,\ a;\theta,\alpha\right)-\frac{1}{|\mathcal{A}|}\sum_{a'}A(s,\ a';\theta,\alpha)\right). \tag{23}$$

This has the advantage of increasing the stability of the optimization and the disadvantage of losing the original semantics of V and A.

We first find the minimum rate required to meet the QoE requirement of each user associated with the UAVs. Next, we determine the minimum transmit power of each UAV required to meet the QoE threshold of the associated users. Finally, we formulate the minimization problem. From Table III, we can see that, for $0.8 \leq \overline{D}\tau, i, n \leq 1$, the MOS of delay will be "Excellent," which means that the delay is minimized. In this case, $\overline{D}\min = 0.8$ is the minimum value that maximizes the delay component of user $i's$ QoE. Consider the transmission between an UAV k located at $w_{\tau,t,k}$ and a user i located at coordinates $w_{\tau,t,i}$. From (10), the delay requirement for a given UAV k that is sending content n to user i at time slot τ is

$$C^{R}_{\tau,ki,n}=\left\{\begin{array}{c}\frac{L}{\left(\Delta_\tau-\overline{D}_{\min}\left(\Delta_\tau-\min\left\{\frac{L}{v_{U_k}},\frac{L}{C^{\max}_{K}}\right\}\right)-\frac{L}{C^{F}_{\tau,k}}\right)}\cdot \\ n\notin C_k \\ \frac{L}{\left(\Delta_\tau-\overline{D}_{\min}\left(\Delta_\tau-\min\left\{\frac{L}{v_{U_k}},\frac{L}{C^{\max}_{K}}\right\}\right)\right)}\cdot \\ n\in C_k\end{array}\right\}. \tag{24}$$

From (24), we can see that, by storing content n at cache of UAV k, the delay requirement for minimizing delay decreases. Based on (5), the minimum transmit power needed to guarantee the QoE requirement of user i receiving content n at interval t is

$$P^{\min}_{t,ki}\left(w_{\tau,t,k},\delta^{R}_{i,n},n\right)=\left(2^{\delta^{R}_{i,n}U_k/B_k}-1\right)$$
$$\sigma^2 10^{\bar{l}_{t,ki}(w_{\tau,t,k},w_{\tau,t,i})/10} \tag{25}$$

where $\delta^{R}_{i,n}$ is the minimum rate required to maximize the user's QoE.

Our problem can be solved with a DRL algorithm where an agent in a given state takes an action, gets a reward and moves to the next state. This DRL is characterized by a set of states, actions, and rewards. At first, the controller collects the status from each UAV, each user and each user's mobility. Then, it constructs the system states, which are the user's mobility and communication channel information as well as the caching contents of the SBSs and UAVs. The agent receives the system states and determines the optimal action a$*$. This action includes the sets of the UAVs, and the users as well as their caching resources for the requesting user. Finally, the control system receives this action and forwards it to the user. The steps are presented in algorithm 1 and a pictorial view of the DRL framework shown in Fig. 2.

1) *States:* The states of the available UAVs k, the available users i and the available caches c for user i in time slot t (with duration τ), are all determined by the realization of the states of the random variables $\gamma^{V}_{t,ki}$ and the realization of the states of the random variables ς_c and the content request probabilities P. Furthermore, the input data also include the number of contacts per time slot (contact rate) and the contact times for every user i-UAV k

$$S_i(t)=\left\{P,\gamma^{V}_{t,ki}\right\}. \tag{26}$$

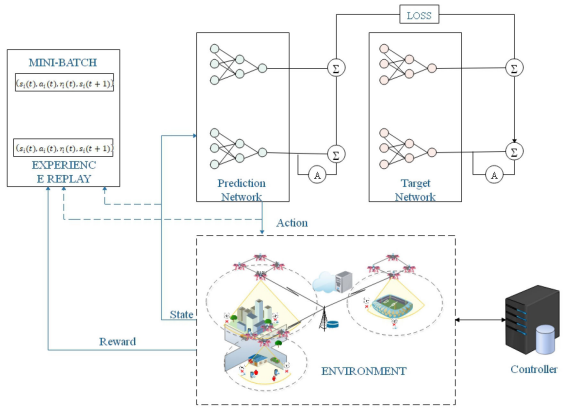


Fig. 2. DRL framework.

2) *Actions:* In the system, the agent has to decide which UAV is assigned to the requesting user, whether or not the requested content should be cached in the UAV. The current composite action $a_i(t)$ is denoted by

$$a_i(t)=\{a^{\text{ca}}_i(t)\} \tag{27}$$

where $a^{\text{ca}}_i(t)$ is defined and interpreted as follows: For the caching control of UAVs, define $K \times C$ matrix $a^{\text{ca}}_i(t) = a^{\text{ca}}_{KXC}$, where each element $a^{\text{ca}}_{kxc}(t)$ represents the cache control of the cth content cached at UAV k for user i and $a^{\text{ca}}_{Kxc}(t) \in \{0, 1\}$. Here, $a^{\text{ca}}_{kxc}(t) = 0$ means that the cth content is not cached at UAV k or UAV k is out of the communication range of user i during time slot t, while $a^{\text{ca}}_{kxc}(t) = 1$ means that the cth content is cached and UAV k is a contact candidate with user i at t. It implies that at every epoch, a^{ca}_{KXC} determines the subset of UAVs that are the contact candidates with user i as well as the subset of contents that are cached at these UAVs.

3) *Reward:* To maximize the reward, we aim to minimize the cost of caching and delay. We define the reward for a specific user i as

$$r_i(t)=\{Q_{\tau,i,n}+\xi c_i\}. \tag{28}$$

Then, ξc is the cache resource utilization, where ξ is the caching coefficient such that $0 \leq \xi \leq 1$. In our scenario, we assume only UAVs have the caching storage capability while users do not have content caching capability.

D. Optimal Locations and Contents to Cache at UAVs

1) User Association: We find the user-UAV association based on the predicted users' locations at the next time interval. Clearly, the prediction accuracy of the users' locations will directly affect the users' association. A user is associated with an UAV if the following condition is satisfied:

Algorithm 1 Dueling Mobility-Aware Content Caching and Placement

Input Content Request Probability P, SINRs between users and UAVs, $\gamma^V_{t,ki}$
Output Optimum location of the UAVs, Optimum contents to cache at UAVs, QoE of the system

while system is running **do**
  Let the users move for time t with the given user's mobility model
  Calculate the QoE for the new system
  Initialize the experience replay buffer $\mathcal{D}$ with capacity N.
  Initialize the action-value function Q with random weights
  for episode $= 1, M$ do
  Receive the initial observation O_1, and pre-process O_1 to be the beginning state s_1.
    for $t = 1, T$ **do**
  Choose a random probability p
    if $p \leq \varepsilon$ **then**
      randomly select an action $a_i(t)$
    else
      $a_i(t) = Q^*(s(t),\ a_i(t);\ \theta)$
    end if
    Execute action $a_i(t)$ in the system, and obtain the reward $r_i(t)$, and the next observation O_{i+1}.
    Process $O_{i+1}$to be the next state s_{i+1}.
    Store the experience $(s_i(t), a_i(t), r_i(t), s_i(t+1))$ into the experience replay buffer $\mathcal{D}$.
    Get a batch of $\mathcal{B}$ samples (s_i, a_i, r_i, s_{i+1}) from the replay buffer.
    Compute $Q(s_i(t), a_i(t) \mid \theta, \alpha, \beta) = V(s_i(t) \mid \theta, \beta) + A(s_i(t), a_i(t) \mid \theta, \alpha) - \frac{1}{|A|}\sum_{a_i(t)} A(s_i(t)|\theta, \alpha)$
    Set $y_i = r_i$ if s_{i+n} as the terminal state or $y_i = r_i + \gamma Q(s_{i+1}, \text{argmax}_{a_{i+1}} Q(s_{i+1}, a_{i+1;\theta_i}); \theta_{i+1})$ otherwise
    Calculate the optimal location of the UAVs using (32)
    Calculate the optimal contents to cache at the UAVs using (31)
    Calculate transmit power of the UAVs using (25)
    Perform a gradient descent step on $(y_i - Q(s_i, a_i, \theta))^2$
  end for
end while

$$C^V_{\tau,ki} \geq \frac{L}{\left(\Delta_\tau - \overline{D}_{\min}\left(\Delta_\tau - \min\left\{\frac{L}{v_k} - \frac{L}{C^{\max}_K}\right\}\right) - \frac{L}{v_{U_k}}\right)} \tag{29}$$

where v_k is the capacity of the link between the UAV and the MBS. From [6, Th. 1], we can see that the user-RRH association depends on the fronthaul rate of each user, the delay requirement, and the device rate requirement. From (24), we can see that the fronthaul rate of each user decreases as the number of the users associated with the UAVs increases. Clearly, the decrease of the fronthaul rate for each user will improve the delay requirement.

2) Optimal Content Caching for UAVs: Based on the UAV association, we find the optimal contents to cache at each UAV. The content caching will reduce the transmission delay and, hence, decrease the delay requirement. From (24), we can see that, the optimal contents to store at the UAV cache lead to maximum reduction of the UAV's transmit power. The reduction of UAV transmit power is caused by the decrease of the delay requirement. Let $P_{j,i} = [p_{j,i1},\ p_{j,i2}, \ldots, p_{j,iN}\ ]$ be the content request distribution of user i during period j that consists of H time slots. The optimal contents that will be stored at each UAV cache can be determined based on the following:

$$C_k = \arg\max_{c_k} \sum_{j=1}^{T/H}\sum_{\tau=1}^{H}\sum_{i\epsilon\mathcal{U}_{\tau,k}}\sum_{n\epsilon C_k}(P_j, i, n\ \Delta P_{j,\tau,ki,n}) \tag{30}$$

where $\Delta P_{j,\tau,ki,n} = P^{\min}_{t,ki}(C^R_{\tau,ki})_{n\notin C_k} - P^{\min}_{t,ki}(C^R_{\tau,ki})_{n\in C_k}$. From [6, Th. 2], we can see that, when the fronthaul rates of all users are the same, the transmit power reduction $\Delta P_{j,\tau,ki,n}$ will be a constant. Subsequently, the optimal content caching becomes

$$C_k = \arg\max_{c_k} \sum_{j=1}^{T/H}\sum_{\tau=1}^{H}\sum_{i\epsilon\mathcal{U}_{\tau,k}}\sum_{n\epsilon C_k}(P_j, i, n). \tag{31}$$

From [6, Th. 2], we can see that the content caching depends on the preknowledge of users association as well as the content request distribution of each user. Therefore, by predicting the mobility pattern and content request distribution for each user, we can determine the optimal content to cache.

1) *Optimal locations of the UAVs:* Here, we determine the optimal UAVs' locations where the UAVs can serve their associated users using minimum transmit power. Once each UAV selects the suitable contents to cache, the transmission link (MBS-UAV-user or UAV-user) for each content and the delay requirement $C^R_{\tau,ki,n}$ in (24) are determined. In this case, the rate $\delta^R_{i,n}$ which is used to meet the QoE requirement of each user is also determined. Next, we derive a closed-form expression for the optimal location of UAV k during time slot τ in the following two special cases.
   a) UAVs positioned at low altitudes compared to the size of its corresponding coverage; $h^2_{\tau,k} \ll (x_{t,i} - x_{\tau,k})^2 + (y_{t,i} - y_{\tau,k})^2$ and $\mu_{\text{NLoS}} = 2$.
   b) UAVs positioned at high altitudes compared to the size of its corresponding coverage; $h^2_{\tau,k} \gg (x_{t,i} - x_{\tau,k})^2 + (y_{t,i} - y_{\tau,k})^2$

$$x_{\tau,k} = \frac{\sum_{i\epsilon\mathcal{U}_{\tau,k}}\sum_{t=1}^{F_{\tau,i}} x_{t,i}\Psi_{t,ki}}{\sum_{i\epsilon\mathcal{U}_{\tau,k}}\sum_{t=1}^{F_{\tau,i}}\Psi_{t,ki}}$$

$$y_{\tau,k} = \frac{\sum_{i\epsilon\mathcal{U}_{\tau,k}}\sum_{t=1}^{F_{\tau,i}} y_{t,i}\Psi_{t,ki}}{\sum_{i\epsilon\mathcal{U}_{\tau,k}}\sum_{t=1}^{F_{\tau,i}}\Psi_{t,ki}} \tag{32}$$

TABLE IV
SIMULATION PARAMETERS

Parameters	Settings
Loss and NLos Constants X, Y	11.9, 0.13
Shadowing random variable for LoS and NLoS $\chi\sigma_{LoS}$, $\chi\sigma_{NLoS}$	5.3, 5.27
Number of contents N	100
Number of users U	120
Path Loss for LoS and NLoS μ_{LoS}, μ_{NLoS}	2,2.4
χ	15
Number of UAVs K	5
No of time slots T	120
Cost of caching storage ξ_c	2units/MB
Free space distance reference d_o	5m
Discount factor λ	0.9
Path Loss Exponent β	2
Skewness of the content popularity α	0.56
Transmit power of MBS P_M	30dBm
Caching capacity of UAV C	100
Transmit power of UAV P_k	20dBm
Variance of the Gaussian noise σ^2	-95dMn
Minimum UAV altitude h_{min}	100m
B	1MHz
Total 2D area	4Km^2
newline Carrier frequencyf_c	38GHz
Total bandwidth available for each UAV B_k	1GHz
Learning rate ρ	0.1

where and $\Psi_{t,ki} = (2^{\delta^R_{i,n}/B} - 1)\sigma^2 10(L_{\text{FS}}(d_0) + x_\sigma)/10$

$$\sigma = \begin{cases} \sigma_{\text{NLoS}} \text{ for case } (a) \\ \sigma_{\text{LoS}} \text{ for case } (b). \end{cases}$$

V. PERFORMANCE EVALUATION

A. Experimental Settings

The experiment was set up using TensorFlow. We limit ourselves to five UAVs and within an area of 4 km^2 with a total user count of about 120. Our algorithm is a dueling deep Q network-based algorithm because dueling network represents two separate estimators: one for the state value function and one for the state-dependent action advantage function. The main benefit of this factoring is to generalize learning across actions without imposing any change to the underlying RL algorithm. The dueling architecture consists of two streams that represent the value and advantage functions, while sharing a common convolutional feature learning module. The two streams are combined via a special aggregating layer to produce an estimate of the state-action value function Q. Intuitively, the dueling architecture can learn which states are (or are not) valuable, without having to learn the effect of each action for each state. This is particularly useful in states where its actions do not affect the environment in any relevant way. The algorithm is presented in Algorithm 1. The simulations were run for 2000 episodes. Detailed simulation parameters are given in Table IV. In our graphs, 10 episodes make 1 epoch. The performance of our algorithm is evaluated by comparing it with other RL algorithms (precisely, Q-learning, nature-DQN, and Deep Deterministic Policy Gradient (DDPG)) and the conceptor ecostate network algorithm in [6]. A performance evaluation of cache resource utilization with and without mobility for two mobility models (random waypoint user mobility model and random walk mobility model) is done. The experimental findings are presented in the following sections.

B. Convergence Analysis

Our algorithm is evaluated in comparison with other RL algorithms (Q-learning, nature-DQN, and DDPG). We tested them for the rate of convergence in terms of reward, satisfaction, and loss. Fig. 3 shows the convergence performance for all the algorithms. Q-learning as shown in Fig. 3(a), nature-DQN in Fig. 3(b), the proposed dueling algorithm in Fig. 3(c), and DDPG in Fig. 3(d). It shows that all these methods can converge with dueling outperforming DDPG, nature-DQN, and Q-learning in that order in all the test cases (loss, satisfaction, and reward). For loss, dueling converges after about 100 episodes at a loss of about 0.2, DDPG also converges after about 250 episodes at a loss of about 0.4. Even though nature-DQN converges at a loss of about 0.1, which is much better, it takes a longer time (about 400 episodes) to converge and finally, Q-learning converges after about 1250 episodes at a loss of about 0.2. From the above-mentioned analysis, it shows that our proposed dueling algorithm performs better because it converges at 100 episodes and at that same number of episodes, all the other algorithms had not converged and was giving a much higher loss than 0.2. Likewise, for reward, dueling as shown in Fig. 3 (c) reveals that at zero (0) episode was a high reward of about 0.75, then as episodes increased to about 250 the reward increased to about 0.8, which interestingly was more or less stable with an increase from 250 and above. As at 2000 episodes the pattern was still stable. This shows a saturated pattern of reward from 250 and above of episodes. It took the other algorithms also about 250 episodes to converge but at a reward of 0.7, 0.52, and 0.48 for DDPG, nature-DQN, and Q-learning, respectively. Finally, the results of the analysis with regards to satisfaction show some fluctuations with stipulated probabilities as episodes' increase. These fluctuations vary around the mean which shows some low variabilities within the datasets. Hence, volatility pattern is lowly volatile due to low variability. The volatility rate is better with dueling than with the other algorithms. From the above-mentioned, dueling performs better for convergence in all test cases (loss, satisfaction, and reward) than DDPG, nature-DQN, and Q-learning.

C. Satisfaction

Fig. 4 shows the rate needed for satisfying the QoE requirement of each user versus the wireless fronthaul rate of each user. We can see that the rate required to maximize the users QoE decreases as the wireless fronthaul rate increases. Caching increases the fronthaul rate, which reduces the rate required to satisfy the user' QoE. Clearly, the use of caching at the UAVs can significantly reduce the rate required to reach the QoE threshold of each user when the wireless fronthaul rate for each user is low.

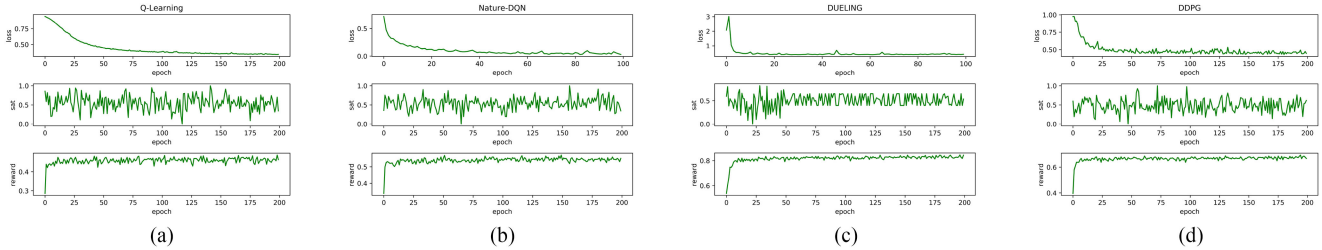


Fig. 3. Convergence analysis. (a) Q-learning. (b) Nature-DQN. (c) Dueling. (d) DDPG.

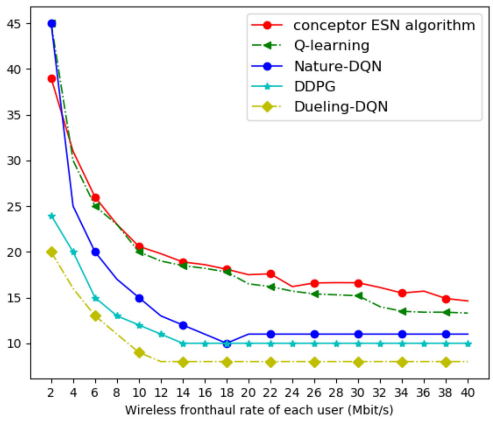


Fig. 4. Rate requirements.

Our dueling algorithm gives a rate of about 3 Mb/s at a fronthaul rate of about 14 Mb/s. Clearly our proposed algorithm performs better than all the other algorithms including the conceptor ecostate network algorithm. Fig. 5(a) shows how the percentage of users with satisfied QoE requirement remains almost constant from 0 to 60 users and changes as the number of the users varies from then onward. We can see that the percentage of the satisfied users decreases as the number of the users increase. This is because as the number of data craving users increase, competition for bandwidth increases, and ultimately reduces the satisfaction. In all cases our algorithm performed better than all the other algorithms and yielded a gain of up to about 5% more than the next performing algorithm (DDPG) and about 40% better than all the other algorithms in terms of the percentage of satisfied users when the number of users rose to 120 users. It is an expected result for the UAVs to maximize the QoEs of the users in the absence of SBSs.

D. Transmit Power

From Fig. 5(b), we can also see that as the number of UAVs increase, the average transmit power decreases. This is because increasing the number of UAVs, increases the total cache capacity, which increases the hit rate, and therefore, the UAV would need less transmit power for transmission. When the number of UAVs is 1, our algorithm has an average minimum transmit power of 20 compared to about 24 for DDPG, about 29 for natrue-DQN, about 32 for Q-learning and 35 for the conceptor ecostate network. The average minimum transmit power continues to decrease as the number of UAVs increase. At number of UAVs of 5, the average minimum transmit power has decreased significantly to about 5 for conceptor ecostate network, 4 for Q-learning, 3 for nature-DQN, 2 for DDPG, and 1 for our proposed dueling-DQN. It also shows that our algorithm performs better than the other algorithms. Fig. 5(c) shows how the total transmit power of the UAVs in a time period changes as the number of the users varies. We can see that the total UAV transmit power of all algorithms increases as the number of the users increase. This is due to the fact that the capacity of the wireless fronthaul link of UAVs are limited. Therefore, the UAVs need to increase their transmit power to be able to satisfy the QoE requirement of each user. In that regard, we can see our algorithm giving better total transmit power of about 10 for 10 users and increasing to about 45 for 70 users, which in all cases performed better than all the other algorithms.

E. Cache Resource Utilization

Fig. 6(a) shows the cache resource utilization. As the number of contents increase, the cache resource utilization also increases. This is because increasing the number of contents, increases the hit rate which translates into a good cache resource utilization. Amazingly from our observations, it is clear that for this particular metric of cache resource utilization, apart from dueling and DDPG, the conceptor ecostate network performs better for this metric than the nature-DQN and the Q-learning algorithms. It is clear from Fig. 6(a) that dueling performs better than the other algorithms. Additionally, two mobility models (random waypoint user mobility model and random walk user mobility model) were compared using our proposed dueling algorithm and the conceptor ESN in [6] in Fig. 6(b) and (c). In Fig. 6(b), we compared the two algorithms (dueling and conceptor ecostate network) with and without mobility. We can observe that the resource utilization for both algorithms with mobility is better than that without mobility. Both algorithms with and without mobility converges at about 100 epochs and at the peak of 200 epochs, dueling with mobility has a resource utilization of 0.79–0.75 without mobility. At the peak of 200 epochs, conceptor ecostate with mobility has a resource utilization of 0.70–0.69 without mobility. In Fig. 6(c), we compared the two algorithms (dueling and conceptor ecostate network) with and without mobility. We observed that the resource utilization

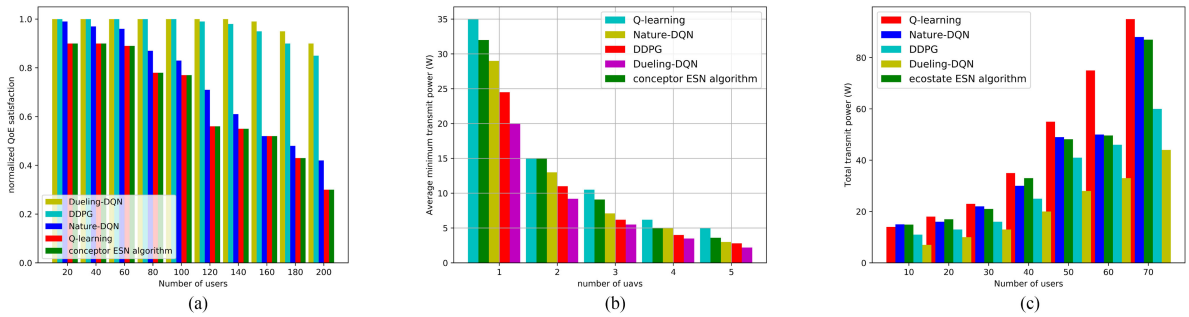


Fig. 5. Satisfaction and power. (a) QoE satisfaction. (b) Average minimum transmit power. (d) Total transmit power.

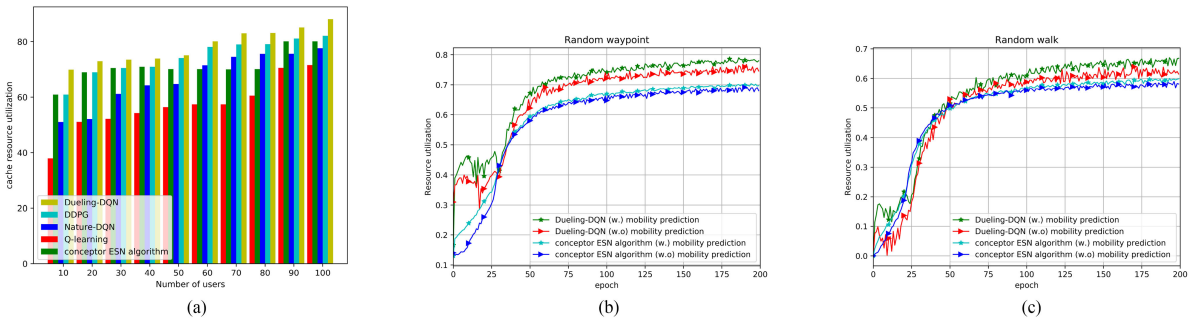


Fig. 6. Cache resource utilization and mobility. (a) Cache resource utilization. (b) Random waypoint mobility model. (c) Random walk mobility model.

for both algorithms with mobility is better than that without mobility. Both algorithms with and without mobility also converges at about 100 epochs and at the peak of 200 epochs, dueling with mobility has a resource utilization of 0.68–0.62 without mobility. At the peak of 200 epochs, conceptor ecostate with mobility has a resource utilization of 0.60–0.59 without mobility. It emerges that our proposed dueling algorithm outperforms the conceptor ESN algorithm and the random waypoint mobility model performs better than the random walk model.

VI. Conclusion

In this article, we have proposed a dueling reinforcement-based algorithm to predict the location of the cache enabled UAVs and the contents to cache at the UAVs taking into consideration human centric features such as user mobility while maximizing the QoE satisfaction of users and reducing the transmit power of the UAVs. The random waypoint mobility model was employed in this article. Our algorithm was compared with Q-learning, nature-DQN, and DDPG and conceptor ESN using the metrics of convergence, satisfaction, transmit power, and cache resource utilization. Additionally, the two algorithms of dueling and conceptor ecostate algorithms were compared both with and without mobility for cache resource utilization. The experimental results showed that our dueling algorithm gives significant gains in terms of convergence, QoE satisfaction, transmit power, and the cache resource utilization as compared to all the other algorithms.

References

[1] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2015 - 2020," *Growth Lakel.*, vol. 2017, pp. 2015–2020, 2016.
[2] N. Wang, E. Hossain, and V. K. Bhargava, "Backhauling 5G small cells: A radio resource management perspective," *IEEE Wireless Commun.*, vol. 22, no. 5, pp. 41–49, Oct. 2015.
[3] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.
[4] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, Feb. 2014.
[5] Z. Su, Q. Xu, F. Hou, Q. Yang, and Q. Qi, "Edge caching for layered video contents in mobile social networks," *IEEE Trans. Multimed.*, vol. 19, no. 10, pp. 2210–2221, Oct. 2017.
[6] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.
[7] N. Cheng *et al.*, "Air-ground integrated mobile edge networks: Architecture, challenges, and opportunities," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 26–32, Aug. 2018.
[8] F. Cheng *et al.*, "Caching UAV assisted secure transmission in small-cell networks," in *Proc. Int. Conf. Comput. Netw. Commun.*, 2018, pp. 696–701.
[9] 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.*, 2019, pp. 1–6.
[10] N. Zhao *et al.*, "Caching UAV assisted secure transmission in hyper-dense networks based on interference alignment," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2281–2294, May 2018.
[11] H. Wu, X. Tao, N. Zhang, and X. Shen, "Cooperative UAV cluster-assisted terrestrial cellular networks for ubiquitous coverage," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2045–2058, Sep. 2018.
[12] F. Liang, J. Zhang, B. Li, Z. Yang, and Y. Wu, "The optimal placement for caching UAV-assisted mobile relay communication," in *Proc. IEEE Int. Conf. Commun. Technol.*, 2019, pp. 540–544.

[13] F. Cheng, G. Gui, N. Zhao, Y. Chen, J. Tang, and H. Sari, "UAV-relaying-Assisted secure transmission with caching," *IEEE Trans. Commun*., vol. 67, no. 5, pp. 3140–3153, May 2019.
[14] L. Zhang, Z. Zhao, Q. Wu, H. Zhao, H. Xu, and X. Wu, "Energy-aware dynamic resource allocation in UAV assisted mobile edge computing over social Internet of vehicles," *IEEE Access*, vol. 6, pp. 56700–56715, 2018.
[15] R. Lu, R. Zhang, X. Cheng, and L. Yang, "UAV-assisted data dissemination with proactive caching and file sharing in V2X networks," in *Proc. IEEE Global Commun. Conf.*, 2019, pp. 1–6.
[16] X. Xu, Y. Zeng, Y. L. Guan, and R. Zhang, "Overcoming endurance issue: UAV-Enabled communications with proactive caching," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1231–1244, Jun. 2018.
[17] 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.
[18] 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.
[19] Y. J. Chen, K. M. Liao, M. L. Ku, and F. P. Tso, "Mobility-aware probabilistic caching in UAV-assisted wireless D2D networks," in *Proc. IEEE Global Commun. Conf.*, 2019, pp. 1–6.
[20] 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.*, 2018, pp. 1–6.
[21] 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, Feb. 2020.
[22] J. Song, H. Song, and W. Choi, "Optimal content placement for wireless Femto-caching network," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4433–4444, Jul. 2017.
[23] W. C. Ao and K. Psounis, "Fast content delivery via distributed caching and small cell cooperation," *IEEE Trans. Mobile Comput.*, vol. 17, no. 5, pp. 1048–1061, May 2018.
[24] J. Elias and B. Blaszczyszyn, "Optimal geographic caching in cellular networks with linear content coding," in *Proc. 15th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wireless Netw.*, WiOpt, 2017, pp. 1–6.
[25] K. Mitra, A. Zaslavsky, and C. Ahlund, "Context-aware QoE modelling, measurement, and prediction in mobile computing systems," *IEEE Trans. Mobile Comput.*, vol. 14, no. 5, pp. 920–936, May 2015.
[26] H. Li, T. Wei, A. Ren, Q. Zhu, and Y. Wang, "Deep reinforcement learning: Framework, applications, and embedded implementations: Invited paper," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des.*, 2017, pp. 847–854.
[27] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
[28] H. Y. Ong, K. Chavez, and A. Hong, "Distributed deep Q-learning," 2015, *arXiv:1508.04186*.
[29] G. Shixiang *et al.*, "Continuous deep Q-learning with model-based acceleration," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 2829–2838.
[30] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Frcitas, "Dueling network architectures for deep reinforcement learning," in *Proc. 33rd Int. Conf. Mach. Learn.*, 2016, pp. 1995–2003.
[31] M. E. Harmon and L. C. Baird III, "Multi-player residual advantage learning with general function approximation," *Wright Lab. WL/AACF*, Wright-Patterson Air Force Base, OH, USA, Rep. No. WL-TR-1065, 1996.

Stephen Anokye (Student Member, IEEE) received the B.Sc. degree in computer science from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana, in 2004 and the M.Eng. degree in computer science from Hunan 876 University, Changsha, China, in 2009. He is currently working toward the Ph.D. degree with the University of Electronic Science and Technology of China, Chengdu, China, since 2017.

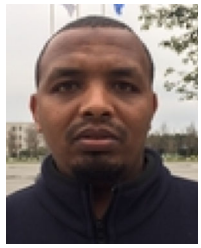
Between 2010 and 2012, he was a Lecturer with the Garden City University College, Kumasi, Ghana. He has been a Lecturer with the Department of Computer Science and Engineering, University of Mines and Technology, Tarkwa, Ghana, since 2012. His research interests are security in wireless sensor, mobile/cloud computing, and big data.

Mr. Anokye is also a member of the Mobile Cloud-Net Research Team, UESTC.

Daniel Ayepah-Mensah received the bachelor's degree in computer engineering from the Kwame Nkrumah University of Science and Technology (KNUST), Kumasi, Ghana, in 2014 and the master's degree in computer science and engineering in 2019 from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, where he is currently working toward the Ph.D. degree in computer science and engineering.

From 2014 to 2017, he was a Software Developer. His research interest includes generally wireless networks, big data, and cloud computing.

Mr. Ayepah-Mensah is a member of the Mobile Cloud-Net Research Team, UESTC.



Abegaz Mohammed Seid received the B.Sc. degree in computer science from Ambo University, Ambo, Ethiopia, in 2010, and the M.Sc. degree in computer science from Addis Ababa University, Addis Ababa, Ethiopia, in 2015. He is currently working toward the Ph.D. degree in computer science and technology with the University of Electronic Science and Technology of China (UESTC), Chengdu, China.

From 2010 to 2016, he was with the Dilla University, Dila, Ethiopia, as a Graduate Assistant and a Lecturer. He has authored or coauthored six scientific conference and journal papers. His research interests include wireless network, mobile edge computing, fog computing, UAV network, IoT, and 5G wireless network.

Mr. Seid was with the College of Engineering and Technology as a Member of the Academic Committee and Associate Registrar. He is currently a member of the Mobile Cloud-Net Research Team, UESTC.

Gordon Owusu Boateng received the bachelor's degree in telecommunications engineering from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana, in 2014 and the master's degree in computer science in 2019, from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, where he is currently working toward the Ph.D. degree.

From 2014 to 2016, he worked under subcontracts for Ericsson (Ghana) and TIGO (Ghana). His interests include mobile/cloud computing, 5G wireless networks, data mining, D2D communications, blockchain, and SDN.

Mr. Boateng is also a member of the Mobile Cloud-Net Research Team, UESTC.

Guolin Sun (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in communication and information system from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2000, 2003, and 2005, respectively.

After his Ph.D. graduation in 2005, he has got eight-year industrial work experience in wireless research and development for LTE, Wi-Fi, IoT, cognitive radio, localization, and navigation. Before he joined UESTC as an Associate Professor in August 2012, he was with the Huawei Technologies Sweden. He has filed more than 30 patents and authored or coauthored more than 30 scientific conference and journal papers. His general research interests are software defined networks, network function virtualization, and radio resource management.

Dr. Sun is currently a Vice-Chair of the 5G oriented cognitive radio SIG of the Technical Committee on Cognitive Networks (TCCN) of the IEEE Communication Society. He is currently a TPC member of several conferences.