Self-Adaptive Power Control with Deep Reinforcement Learning for Millimeter-Wave Internet-of-Vehicles Video Caching

Dohyun Kwon, Joongheon Kim, David A. Mohaisen, and Wonjun Lee

Abstract: **Video delivery and caching over the millimeter-wave (mmWave) spectrum is a promising technology for high data rate and efficient frequency utilization in many applications, including distributed vehicular networks. However, due to the short handoff duration, calibrating both optimal power allocation of each base station toward its associated vehicles and cache allocation are challenging for their computational complexity. Heretofore, most video delivery applications were based on on-line or off-line algorithms, and they were limited to compute and optimize high dimensional objectives within low-delay in large scale vehicular networks. On the other hand, deep reinforcement learning is shown for learning such scale of a problem with an optimized policy learning phase. In this paper, we propose deep deterministic policy gradient-based power control of mmWave base station (mBS) and proactive cache allocation toward mBSs in distributed mmWave Internet-of-vehicle (IoV) networks. Simulation results validate the performance of the proposed caching scheme in terms of quality of the provisioned video and playback stall in various scales of IoV networks.**

Index Terms: **Deep reinforcement learning, Internet-of-vehicle caching, video caching.**

I. INTRODUCTION

THE millimeter-wave (mmWave) is a promising technology for provisioning high-end resolution of video contents, with superior data rate and improved efficient frequency utilization [1]–[5]. Based on current global trends, the ratio of video traffic among mobile data traffic is expected to increase, where 78% of the mobile traffic will be composed of video contents in 2021 [6]. As such, video caching in mmWave networks has been highlighted by both industry and academia [7]–[12].

In particular, it is expected that most traffic of forthcoming mobile networks would consist of mmWave-based video chunks. Meanwhile, among plentiful use cases of mmWave based video provisioning scenarios, Internet-of-vehicle (IoV) networks are faced with multiple challenges [13]. For example, the user equipment (UE) installed in vehicle has an intrinsic feature: Very high mobility. Considering that propagation in mmWave is quite directive and with a comparatively short coverage region [14], [15], the mmWave propagation of distributed mmWave base station (mBS) for UEs is constrained for short association time.

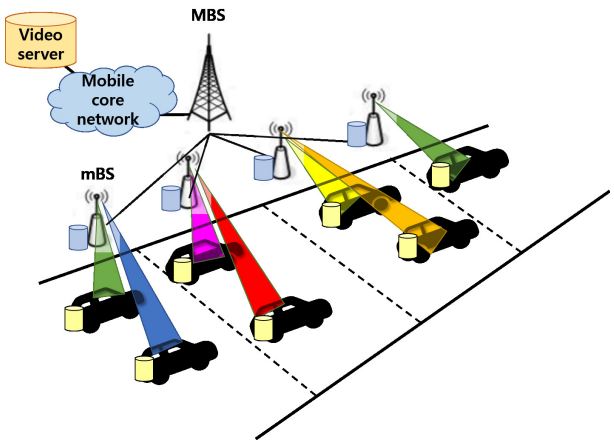

Fig. 1. Considered power-cache aware video caching scheme in distributed IoV networks.

In realistic mmWave IoV networks, the caching scheme should also consider proactive cache size allocation towards distributed mBSs for preventing playback stall. In addition, power control of each mBS for energy efficiency and the number of requested chunks from media servers for minimizing the number of dropped video chunks are investigated [16], [17]. That is, the edge node (i.e., mBS) is responsible for providing cache size and power allocation for supporting associated vehicles. Moreover, if the caching scheme should reflect more optimization objectives or is considered even in larger IoV networks, the classical caching scheme is limited to calibrate such optimal point within certain amounts of delay bounds for avoid video streaming stall events [5], [18]. As such, a novel caching scheme for such mmWave based IoV networks is required. To this end, and to address those issues, we propose a deep reinforcement learning (DRL) based caching scheme for learning an optimal power control of each mBS in the considered IoV networks, and cache allocation towards mBS with a realistic caching scenario. The reason why DRL is used among various optimization and learning based algorithms is that it is one of the emerging sequential

Manuscript received Nov. 21, 2020; revised June 15, 2020; approved for publication by Tim O'Shea, Guest Editor, July 15, 2020.

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2017-0-01637) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation); and also by MSIT, Korea, under ITRC support program (IITP-2018-0-01396) supervised by IITP.

D. Kwon is with Hyundai-Autoever, Seoul, Korea, email: kdh1102@cau.ac.kr.

J. Kim is with the School of Electrical Engineering, Korea University, Seoul, Korea, e-mail: joongheon@korea.ac.kr.

D. A. Mohaisen is with the Department of Computer Science, University of Central Florida, Orlando, FL, USA, e-mail: mohaisen@ucf.edu.

W. Lee is with the School of Cybersecurity, Korea University, Seoul, Korea, e-mail: wlee@korea.ac.kr.

J. Kim, D.A. Mohaisen, and W. Lee are corresponding authors.

Digital Object Identifier: 10.1109/JCN.2020.000022

1229-2370/18/$10.00 © 2020 KICS

Creative Commons Attribution-NonCommercial (CC BY-NC).
This is an Open Access article distributed under the terms of Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided that the original work is properly cited.

decision-making algorithms for solving time-varying systems under unexpected observations. As the agent of DRL learns the optimal action policy, the caching scheme can derive optimal point of power and cache size allocation for UEs of each edge node as soon as the agent observes the state of environment.

Contributions. This paper proposes DRL-based video caching scheme in mmWave IoV networks, as illustrated in Fig. 1, for enabling an optimal video provisioning service under highly dynamic and multiple dimensional learning objectives. Note that the real-world velocity data of vehicles is utilized so that a realistic simulation of IoV networks is available [19]. The contribution of algorithm in this paper can be summarized as follows:

- **Extended action space:** Among various kinds of DRL algorithms, the deep deterministic policy gradient (DDPG) is adopted and enables us to learn continuous and a multi-dimensional action space. DDPG learns an optimal point of power and cache size for each mBS and for each observed IoV network. The learning agent is considered as a macro base station (MBS), which learns and controls the constrained optimal power allocation towards distributed edge nodes (i.e., multiple mBSs) so that their associated vehicular user equipment (VUE) can experience qualitative video provisioning service with optimized quality of video and seamless playback.
- **Model free and off policy:** Specifically, the advantage of the model-free property of DDPG is that the MBS does not need to know the complete information of the vehicular network, while the model-based DRL algorithms needs such a knowledge. The MBS interacts with the environment and accumulates the experience of the interactions and utilizes them for learning the optimal caching policy. In addition, the advantage of the off-policy property of DDPG is that even though the caching policy is updated by the learning process, it can utilize the experience, collected from the previous caching policy, so that the data efficiency is much more improved than the on-policy based algorithms, such as SARSA [21].

Organization. The rest of this paper is organized as follows: Section II introduces various caching schemes including the DRL based approaches and the classical one. Next, Section III summarizes an overview of reinforcement learning, including DDPG. Section IV proposes the system model and description of our caching scheme. Section V discusses the simulation settings and our proposed scheme's performance for power-cache allocation learning in mmWave IoV networks. Finally, Section VI concludes this paper.

II. RELATED WORK

In this section, classical video caching schemes, which includes optimization formula-based approaches and the DRL based-methods, are introduced. First, we summarize the classical optimization formula-based approaches. Next, we review the DRL-based video caching schemes, especially the deep-Q network (DQN)-based approach.

A. Optimization for Caching

The [20] proposed vehicular content centric networks (VCCN) with mobility prediction capabilities for efficiently electing caching nodes. Among multiple vehicles located in a specific hot spot region, the representative vehicle is selected based on the sojourn time. The elected caching node in the hot spot region taking a role of mediating the caching procedure, and the rest of the vehicles are serviced from the caching node. The dynamic cache algorithm (DCA) is proposed in [22], which enables adaptive bitrate (ABR) video streaming service in vehicular networks. The authors considered the mobility issue of vehicular networks, which induces time-varying state of wireless channel. The proposed DCA algorithm based the caching scheme addressed the issue derived from the mobility feature by jointly considering the quality adaptation, cache placement, and bandwidth allocation. A distributed content caching architecture was proposed in [23] focusing on reducing delay of content delivery. Specifically, minimizing the delay due to advent of layered-video encoding techniques such as the scalable video coding (SVC) is NP-hard, so the authors transformed the caching problem and derived pseudopolynomial-time optimal caching solution. Moreover, several video caching schemes in vehicular networks and mobile edge networks are proposed [24] under various scenarios and caching optimizations with classical approaches in information centric network (ICN) were proposed in [25] and [26].

Besides, prefetching-based data dissemination in vehicular cloud systems (VCSs) has been widely studied for vehicular ad-hoc networks (VANETs) to satisfy various wireless communication capabilities such as multimedia streaming, vehicle information and autonomous navigation services [27]. The authors focused on how to exploit the local data storages (i.e., content cache) of roadside wireless access points (APs) within VCS for efficient data dissemination. That is, the prefetching approach takes a role in proactively caching contents for efficient data dissemination in VCS. Such data dissemination can enhance the local access to popular Internet contents via proxy servers. In [28] proactive caching for various wireless network applications was studied. In [28], a content prefetching technique for named data networking (NDN), which is one of the ICN framework, was proposed and showed to maximize the probability that a user retrieves the desired content in a vehicle-to-infrastructure (V2I) scenario. The authors leveraged an integer linear programming formulation of optimally distributing content in the network nodes while also accounting for the available storage and link capacities.

In [29], the deployment of unmanned aerial vehicles for video caching is discussed and the conceptor-based echo state network is used for solving the quality-of-experience (QoE) optimization. This approach is novel and well-discussed, however the problem in the paper is not equivalent to ours because we assume the existence of fixed infrastructure mBS for more reliable service provisioning.

B. DRL-based Caching

The DRL-based caching schemes aim to find the optimal policy in a learning phase. The agent observes the system state. As

the agent acquires the system state information, the agent follows its policy and interacts with the system. After the environment interacts with the actions of the agent, the environment returns the corresponding reward value to the agent, and the agent learns a better policy based on the reward value. The deep Q-learning approach based mobility-aware caching and computational offloading scheme was proposed in [30]. The authors formulated a joint optimal caching and computing resource allocation problem to minimize the overall system cost under hard deadline delay, dynamic storage capacities and computation resources constraints with deep Q-learning approach. In addition, deep reinforcement learning based caching schemes for variety of application areas including interference alignment, software-defined networks (SDNs), and 5G mobile edge computing were proposed in [31]–[33]. An integrated framework that can dynamically orchestrate networking, caching, and computing resources is proposed in [31] to enhance the performance of services for smart cities. Based on the framework, a mobile edge computing and caching scheme with SDN and network functions virtualization (NFV) is proposed with deep Q-learning based approach. Similarly, [32] proposed deep Q-learning based resource allocation strategy for next generation vehicular networks. The authors formulated the resource allocation problem and jointly considered an orchestration of content caching with ICN, networking (e.g., SDN and NFV), and computing (e.g., cloud/edge computing) for optimizing the network. In addition, cache-enabled interference alignment strategy for next generation wireless networks is proposed in [33]. Unlike most of the previous interference alignment (IA) techniques, which assumed the channel is invariant, the transition model of channel state is designed as a finite state Markov channel (FSMC).

III. DEEP REINFORCEMENT LEARNING FOR CACHING

In this section, we review deep reinforcement learning based video caching in mmWave based IoV networks. First, we sum up the preliminaries of reinforcement learning. Next, DQN based caching schemes are introduced as well. Finally, the DDPG algorithm, which is appropriate for large scale of action and state space, is introduced for proposed power-cache aware control policy.

A. Preliminaries

A Markov decision process (MDP) is defined as $M = \{\mathcal{S}, \mathcal{A}, T, r\}$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ denotes the set of possible actions, T denotes the transition model and r denotes the reward value. Based on the MDP, the goal of the reinforcement learning is to train a policy $\pi_\theta \in \Pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$. The policy π maps the state of the environment to the action to maximize the expected reward $\mathcal{J}(\pi)$. With a finite T process, the expected reward $\mathcal{J}(\pi)$ can be described as the accumulation of the reward at each time step as $\mathcal{J}(\pi) = \mathbb{E}\left[\sum_{t=0}^{T} \delta^t r_t | \pi\right]$, where δ is a discount factor which adjusts the effect of future rewards to the current decision. The optimal policy π^* is then described as follows:

$$\pi^* = \arg\max_{\pi} \mathcal{J}(\pi). \tag{1}$$

Based on this equation, the objective of reinforcement learning is described as $\arg\max_\theta \mathbb{E}_{s\sim\pi_\theta}\left[r(s, \pi_{\theta(s)})\right]$.

B. DQN

DQN utilizes a neural network to approximate state-action functions (Q-functions). The Q-functions, which is approximated by neural network, allows the DQN to learn the policy even in a high dimensional system state space. The concept of DQN is based on a classical Q-learning algorithm. In the classical Q-learning, the Q-value of a state-action pair is estimated through iterative updates based on multiple interaction with the environment. Therefore, in DQN the immediate reward we receive and the expected Q-value of the new state are used to update the Q-functions. Therefore, the objective of DQN is described as follows:

$$\arg\min_{\theta} L_{DQN}(\theta) = \arg\min_{\theta}(Q(s_t, a_t; \theta) - \bar{Q}(s_t, a_t; \theta))^2, \tag{2}$$

where s_t is the state at time t, a_t is the selected action at s_t and θ is the parameters of Q-functions. $Q(s_t, a_t; \theta)$ is the target Q-value which is derived from the current Q-functions at time t. Therefore, $Q(s_t, a_t; \theta) = r_t + \delta \text{max}_{\acute{a}} Q(s_{t+1}, \acute{a}; \theta)$.

C. DDPG

Although the DQN based approaches can only handle discrete and low-dimensional action spaces, environments in many realistic applications have continuous and high dimensional action spaces (i.e., proactive caching, resource management, etc). Moreover, the DQN algorithms cannot be straightforwardly applied to continuous actions since DQN depends on choosing the best action that maximizes the Q-value function. When there is a finite number of discrete actions, the action that makes the Q value maximal is chosen, because possible Q values at the state can be computed directly for each action. However, when the action space is continuous, it is hard to exhaustively evaluate the Q values. DDPG is an algorithm which concurrently learns the Q-value function and the policy. The action-value Q function is learned and used to learn the policy. In the DDPG, the optimal Q-function $Q^*(s, a)$ is approximated by neural network, similar to DQN. Therefore, because the action space in continuous, the function $Q^*(s, a)$ can be differentiable in terms of the action. Based on that fact, a policy π_θ can be updated efficiently. The $Q_\phi(s, a)$ which is approximated with the parameters in ϕ is updated based on minimizing the mean-squared Bellman error (MSBE) as $L(\phi, \mathcal{D}) = \mathbb{E}\left[(Q(s_t, a_t; \phi) - \bar{Q}(s_t, a_t; \phi))^2\right]$, where $\mathcal{D}$ is a set of transitions (s, a, r, s'). DDPG aims to learn a deterministic policy $\pi_\theta(s)$ which provides an action that maximizes $Q_\phi(s, a)$. Because the action space is continuous, the Q-function is differentiable in terms of the action. With respect to the policy parameters θ, gradient ascent is performed to update the policy π_θ as $\max_\theta \mathbb{E}_{s\sim\mathcal{D}}\left[Q_\phi(s, \pi_\theta(s))\right]$.

IV. DDPG-BASED POWER-STORAGE-AWARE CACHING

In this section, we propose the overall architecture of a power-cache aware video caching scheme with a DDPG algorithm, which is introduced in the previous section. In IoV networks,

the classical caching scheme, which needs to compute the optimal video caching options toward RSU (i.e., mBS cache of edge node) for every time step through extensive calculation, is an unrealistic caching option due to the time domain overhead. As the duration of association time between mBS and vehicles is short, the effect of overhead may severely affect the video provisioning service, which results in degradation of QoS. Thus, we introduce DDPG-based power and cache storage aware proactive caching scheme for meeting the requirements of the considered scenario by using two ideas: (i) Calculation of the optimal caching action through a learning process so that the optimal caching option can be derived for seamless services after the learning process and (ii) scale-adaptable IoV networks with satisfying optimal power and preemptive cache allocation of mBS for qualitative video provisioning service. System description and design of DDPG based caching scheme are proposed in the following subsections.

A. Assumptions

Before introducing the overall system description, several assumptions regarding elements of the proposed caching scheme of mmWave IoV network are denoted. The assumptions are defined for the following components of the caching scheme: The mBS, the vehicle, the MBS, and the video contents. Note that the components fully satisfy the corresponding assumptions for quality-cache aware video caching scheme.

- **mBS:** Considering most of typical highway is constructed in rural regions and signal propagation of mBS is quite directive, we assume that mBSs of the considered IoV networks directly orient toward the highway. Specifically, the mBSs make beam alignment toward vehicles on highway within range of azimuth angle, which is an appropriate for servicing their own coverage region. In addition, each mBS is assumed that its data transmission is not affected by others. That is, the mBSs on the highway are independent and identically distributed (i.i.d.) over highway with distance of their non-overlapped coverage region so that IA is out of scope of this paper. Finally, because the case of at least two vehicles associated with the same mBS are located on the same position of a highway at the same time is illogical and does not exist, we assumed that the downlink (DL) of each mBS is enough to transmit the entire video chunks of its cache for each associated vehicle for a unit time step no matter the quality of the video. That is, the capacity of DL from mBS is sufficient to provision video toward each associated vehicle because i.i.d. setting of mBSs and non-overlapping position of associated vehicles with fully available bandwidth of the air interface for each vehicle.
- **Vehicle:** For practical reasons, it can be envisioned that the vehicles on the highway *only* move forward, i.e., a vehicle can enter the coverage region of the following mBS or stay within the coverage region of the currently associated mBS after a time step. We assume that the vehicles move forward with probability of FSMC transition model with the value of ρ in Fig. 2, which represents transition probability of vehicles given that the average velocity of vehicles of IoV networks and distance of mBS cell coverage region are both available. As presented in [34], the FSMC has been widely utilized to represent the dynamic variation of vehicular network channel. Because the channel is established between roadside mBS and UEs in the vehicular network, the simplified position transition of vehicles can be modeled with the FSMC. Note that the u_i and x_j in Fig. 2 represent the ith vehicle and the jth mBS, respectively. In addition, the request of vehicle is collected by the MBS, so that currently associated mBS can provide the corresponding video chunks, while the following mBS proactively allocates cache size and video chunks from the media server to prepare the handoff.
- **MBS:** In the proposed mmWave IoV networks, the MBS takes a role of learning agent for power and preemptive cache allocation of mBSs for seamless video provisioning service among the vehicles on the highway. Based on the assumption of full knowledge of channel state information (CSI) of MBS in vehicular network in [35] and the signaling state of IoV networks through backhaul communications between MBS and mBSs within low cost [36], we assumed that the MBS has full knowledge of the considered IoV networks in four aspects: (i) Association information between mBSs and vehicles, (ii) cache occupancy state of each mBS, (iii) buffer occupancy state of each vehicle, and (iv) history of provisioned average quality of video for vehicles along with their trajectory. These four states consist of the state of the MBS, which calculates the corresponding caching action, and is derived from the neural network of the DDPG algorithm. In other words, the MBS can learn the optimal caching policy through trial and error. The detailed procedure of learning process is given in Algorithm 1. In addition, the MBS can be assumed that video chunks toward an mBS for each vehicle is limited up to $\overline{m}$ unit size, which is the upper bound (UB) of the video size cached at the corresponding mBS, for satisfying *fairness* of caching service considering the limited storage of mBS cache.
- **Contents:** Guided by previous research work [37], we assume that the popularity of video contents among vehicles follows Zipf distribution [38] where all chunks during a video session are deterministically requested in sequence. Moreover, for each video chunk, it is assumed that the data rate of corresponding quality of video determines the unit size of a chunk. For example, suppose that there exists two video chunks with 360p and 720p quality, where the required data rate for supporting them is 1 Mbps and 5 Mbps, respectively. Then, we assume that the **unit size** of each corresponding single chunk for those quality is 1 and 5. That is, in case of a vehicle requesting three chunks of 720p quality video, 3×5 unit size of vehicle's buffer is increased, while the associated mBS's cache loses corresponding unit size. Finally, each vehicle is assumed to watch a video, which it firstly requested, throughout the entire sojourn time on the highway.

B. System Description

In the following, descriptions of system elements including vehicles, cache, buffer, and video are provided.

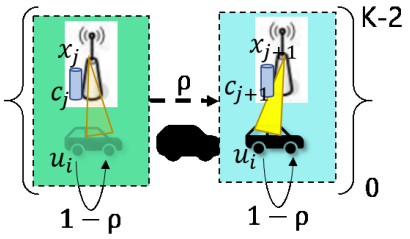

Fig. 2. The finite state Markov chain for vehicle association transition model. The system average velocity of vehicles affects the value of transition probability ρ.

B.1 Vehicles on Highway

There exists vehicles up to N and K fixed mBSs on the roadside of the considered highway mmWave IoV networks. The set of vehicles and mBSs are denoted by $\mathcal{U} = \{u_0, u_1, \cdots, u_i, \cdots, u_{N-1}\}$ and $\mathcal{X} = \{x_0, x_1, \cdots, x_j, \cdots, x_{K-1}\}$, respectively. The u_i and x_j represent ith vehicle and jth mBS, where $\forall u_i \in \mathcal{U}$, $\forall x_j \in \mathcal{X}$, $i \in [0, N)$, and $j \in [0, K)$. We assume that every u_i can be associated with only one mBS at the same time, which follows hard handoff mechanism. In addition, the u_i can only move forward on the highway, i.e., if the u_i is associated with x_j at time step t, the u_i can only associate with x_j or x_{j+1} at time step $t+1$, not x_{j-1} such that $j \in (0, K-1)$. In addition, the association information or the discrete position of $\forall u_i$ can be represented as a matrix $\mathbb{P}_{N\times K}$, where

$$P_{N\times K} = \begin{Vmatrix} p_{0,0} & \cdots & p_{0,K} \\ \vdots & p_{i,j} & \vdots \\ p_{N-1,0} & \cdots & p_{N-1,K-1} \end{Vmatrix} \tag{3}$$

and each element $p_{i,j}$ of $P_{N\times K}$ represents whether u_i belongs to the coverage of x_j or not. For example, the $p_{i,j}$ is -1 if u_i is associated with x_j. Moreover, if the $p_{i,j}$ is 1, u_i is associated with x_{j-1}. Otherwise, the value is set to 0.

The value $p_{i,j} \in P_{N\times K}$ can be equal or changed over time by following the FSMC transition probability model described in Fig. 2, where the transition probability set is represented as ϱ, which is derived from system average velocity vector $\mathcal{V}$ and mBS cell range $\mathcal{O}$. For example, if u_i is associated with x_j such that $i \in [0, N)$, and $j \in [0, K-1)$, the value of $p_{i,j} = -1$ and $p_{i,j+1} = 1$ and the rest of elements $p_{i,l}$ such that $l \in [0, K-1)$ except j and $j+1$ is 0. The mBS cell range $\mathcal{O}$ is assumed as 150 m, and the system average velocity $\mathcal{V}$ of $\forall u_i \in \mathcal{U}$ is set to $\mathcal{V}$ =80 km/h [19]. Suppose that each time step is one second and considering the $\mathcal{V}$ and $\mathcal{O}$ settings, the ϱ can be calculated as 0.143, which is the probability of each vehicle to move forward to associate following mBS for the next time step. That is, each $u_i \in \mathcal{U}$, which is associated with x_j, transits its position over time toward the cell of the following mBS x_{j+1} with the FSMC transition probability given that average velocity of users in the IoV networks is available and $\mathcal{O}$ =150 m for $i \in [0, N)$, and $j \in [0, K-1)$, respectively.

Table 1. System description parameters.

Parameter	Description
N	The number of vehicles
K	The number of mBS
$\mathcal{X}$	The set of mBS
$\mathcal{U}$	The set of vehicles
$\mathcal{C}$	The set of mBS cache
$\mathcal{B}$	The set of vehicular buffer
$\mathcal{M}$	The macro base station
$\mathcal{V}$	The vector of vehicle velocity
$\overline{c}$	The UB of mBS cache storage
$\overline{b}$	The UB of vehicular buffer
$\overline{m}$	The UB of cached video at x_j for u_i
ϱ	The vector of FSMC transition probabilities
R	The total reward of IoV network
γ	The learning rate of $\mathcal{M}$
$C_{N\times K}$	The state of mBS cache
$B_{N\times K}$	The state of vehicular buffer
$H_{N\times K}$	The state of average quality history
$P_{N\times K}$	The state of vehicle position
$V_{N\times K}$	The action of mBS power allocation
$L_{N\times K}$	The action of proactive cache allocation

B.2 Cache, Buffer, and Video

There exists a set of video caches $\mathcal{C} = \{c_0, c_1, \cdots, c_j, \cdots, c_{K-1}\}$ on the highway and each c_j is equipped with mBS x_j such that $j \in [0, K)$. Each video cache c_j of x_j stores video chunks for vehicles. Suppose that c_j is requested to provide video chunks from u_i, where $p_{i,j} = -1$, then x_j immediately provides the cached video chunks or request the media server. In addition, the following mBS, denoted by x_{j+1}, notices the request of the vehicle and proactively allocates cache size and contents from the media server to prepare the position transition of u_i. In addition, the spatial upper bound of c_j is denoted as $\overline{c}$, and the video contents, which are cached in c_j, are transmitted towards the associated set of vehicles $u_{y,j}$, such that $y \in [0, N)$. We assumed that the capacity of the mmWave link between u_i and x_j is sufficient so that x_j can provision the entire video chunks towards a set of vehicles, which are associated with x_j. Moreover, video buffer set $\mathcal{B} = \{b_0, b_1, \cdots, b_i, \cdots, b_{N-1}\}$ represents the set of buffer which is equipped within each vehicle. The buffer b_i is mounted on u_i and the spatial upper bound of b_i and buffer playback rate is denoted by $\overline{b}$ and $\mathcal{F}$, respectively. Meanwhile, there is a set of video qualities, which are denoted by $\mathcal{Q}$. Moreover, it is assumed that each u_i can be served with each quality of video chunk in the quality set $\mathcal{Q}$. The $\mathcal{Q}$ can be defined as $\mathcal{Q}$ =[360p, 480p, 720p, 1080p, 4K], for example. Each quality level in $\mathcal{Q}$ requires an average link capacity of 1, 3, 5, 8, and 40 Mbps, respectively, in ascending order, which determines the QoS of u_i associated with x_j. Per the aforementioned assumption regarding video contents, the unit size of a chunk is determined by the quality of the video.

C. DDPG-based Caching

The system can be represented in terms of reinforcement learning, where the agent in the system is $\mathcal{M}$. The $\mathcal{M}$ controls the overall power and proactive cache allocation toward each x_j on a highway from the remote media server $\mathcal{Z}$, with specific power allocation level v_k of the video and cache size $c_{i,j}$ and $c_{i,j+1}$ for u_i where $k \in [0, 2]$, $i \in [0, N)$, and $j \in [0, K-1)$, respectively. In the following, the learning process of *power allocation* and *proactive cache allocation* toward mBSs are introduced in terms of the state space, action space, reward, and algorithmic description, in details.

C.1 **State space**

The state space of the caching system consists of the following elements: Preloaded unit size of video for each u_i along with the entire mBS $\mathcal{X}$, buffer occupancy of each u_i, average quality history of the provisioned video for each u_i, and the position of each vehicle u_i over time. The elements are denoted by $C_{N\times K}$, $B_{N\times K}$, $H_{N\times K}$, and $P_{N\times K}$, respectively. The state of a position is represented as in (3) and the rest of the elements of the state space are represented as follows:

$$C_{N\times K} = \begin{Vmatrix} c_{0,0} & \cdots & c_{0,K-1} \\ \vdots & c_{i,j} & \vdots \\ c_{N-1,0} & \cdots & c_{N-1,K-1} \end{Vmatrix}, \tag{4}$$

$$B_{N\times K} = \begin{Vmatrix} b_{0,0} & \cdots & b_{0,K-1} \\ \vdots & b_{i,j} & \vdots \\ b_{N-1,0} & \cdots & b_{N-1,K-1} \end{Vmatrix}, \tag{5}$$

$$H_{N\times K} = \begin{Vmatrix} h_{0,0} & \cdots & h_{0,K-1} \\ \vdots & h_{i,j} & \vdots \\ h_{N-1,0} & \cdots & h_{N-1,K-1} \end{Vmatrix}. \tag{6}$$

First, $c_{i,j}$ in (4) represents the cache occupancy of x_j which is the preloaded unit size of the video for satisfying u_i's request. The maximum storage size of each $c_{i,j}$ for $i \in [0, N)$ and $j \in [0, K)$ is limited to $\overline{m}$ for vouching fair video transmission toward vehicles and $\sum_k c_{k,j} \leqslant \bar{c}$, where $k \in [0, N)$ and $\overline{m} \times N \leqslant \bar{c}$.

Next, the $b_{i,j}$ of (5) represents the buffer occupancy of u_i associated with x_j. Each $b_{i,j}$ for $\forall j \in [0, K)$ has UB of $\bar{b}$ and *packet drop* can occur when $b_{i,j} + c_{i,j} - \mathcal{F} \geqslant \bar{b}$. Moreover, video playback service can be *stalled* if $b_{i,j} + c_{i,j} - \mathcal{F} \leqslant 0$.

Finally, the average quality state of the provisioned video at $u_{i,j}$ can be calculated by the cumulative average quality of the provisioned video history through trajectory of u_i from x_0 to x_j. The average quality state of $u_{i,j}$ can be denoted by $h_{i,j}$, and is utilized for learning the policy of $\mathcal{M}$ which aims to provision an enhanced quality of the video toward u_i. Suppose that $s_{i,j}$ represents the sojourn time step of u_i associated with x_j, the $h_{i,j}$ can be calculated as follows:

$$h_{i,j} = \sum_{k=0}^{j} \frac{\sum_{t=0}^{s_{i,k}-1} q_i^{k,t}}{s_{i,k}}, i \in [0, N), j \in (0, K). \tag{7}$$

Moreover, $h_{i,j} = 0$ when $p_{i,j} = 1$ and $j = 0$, i.e., we only consider the history of quality provisioned at u_i, which has the drive experience on the highway. The $q_i^{k,t}$ in (7) represents tth quality index of video chunks, where it is provisioned at u_i associated with x_k.

C.2 **Action Space**

The $\mathcal{M}$ can learn the optimal action, which proactively requests the optimal power and cache allocation toward x_j and x_{j+1} (i.e., vicinity of the u_i) from $\mathcal{Z}$ for seamless video retrieval, given that the state of mmWave IoV networks can be observed. Here, the action space of $\mathcal{M}$ consists of $V_{N\times K}$ and $L_{N\times K}$, where each of them represents the amount of power allocation matrix and cache allocation matrix, respectively, and they can be denoted as follows:

$$V_{N\times K} = \begin{Vmatrix} v_{0,0} & \cdots & v_{0,K-1} \\ \vdots & v_{i,j} & \vdots \\ v_{N-1,0} & \cdots & v_{N-1,K-1} \end{Vmatrix}, \tag{8}$$

$$L_{N\times K} = \begin{Vmatrix} l_{0,0} & \cdots & l_{0,K-1} \\ \vdots & l_{i,j} & \vdots \\ l_{N-1,0} & \cdots & l_{N-1,K-1} \end{Vmatrix}. \tag{9}$$

First, the $v_{i,j}$ of $V_{N\times K}$ represents the amount of power allocation of mBS, where $\mathcal{M}$ requests specific quality of video with respect to the power $v_{i,j}$ to $\mathcal{Z}$ to serve video at x_j for u_i. In addition, $l_{i,j}$ of $L_{N\times K}$ stands for the size of the allocated cache size at x_j for u_i by $\mathcal{M}$. The $\mathcal{M}$ requests cache size up to two neighboring mBSs for accomplishing two missions as follows: (i) Secure seamless current video provisioning service for x_j and (ii) preemptive cache allocation at x_{j+1} for enabling seamless services where $j \in [0, K-1)$. For example, if u_i on a highway is associated with x_j, the $\mathcal{M}$ allocates cache storage with unit size of $l_{i,j}$ and $l_{i,j+1}$ at x_j and x_{j+1}, respectively, for supporting current video service for u_i and preemptive video caching for handoff of u_i. When u_i is not yet on the highway, i.e., $p_{i,j} = 1$ and $j = 0$, the $\mathcal{M}$ only allocates cache size toward x_0 for u_i for proactive video caching. If the u_i is associated with x_{K-1}, the $\mathcal{M}$ requests x_{K-1} to allocate cache size of l_{K-1} for satisfying current video service of u_i.

C.3 **Algorithm for Learning The Proactive Caching**

The $\mathcal{M}$ learns the proactive caching policy and accomplishes power and preemptive cache allocation toward mBSs for seamless video retrieval by utilizing the proposed DDPG based algorithm as shown in Algorithm 1. The overall caching policy learning procedures are as follows. First, the parameters of the actor and critic network, which activate and evaluate action of $\mathcal{M}$, are initialized (line 1). Then, the target networks regarding both actor and critic network, $Q^{'}$ and $\mathcal{A}^{'}$, are initialized with the origin's one (line 2). By iterating each episode, the $\mathcal{M}$ repeats following procedures to learn optimal caching policy which is power-cache aware:

i) For every episode, the transition pairs, attained by an arbitrarily generated set of states s of size φ, corresponding actions generated by the actor network with input s, reward value for s and a, and the next state space $s^{'}$, are paired and stored at replay buffer Φ (lines 5–7).

Algorithm 1: DDPG based joint power-cache aware learning algorithm for mmWave IoV networks

1 Initialize the critic and actor network $Q(s,a|\theta^{Q})$ and $\mathcal{A}(s|\theta^{\mathcal{A}})$ with weights θ^{Q} and $\theta^{\mathcal{A}}$
2 Initialize the target network $Q^{'}$ and $\mathcal{A}^{'}$ with weights $\theta^{Q^{'}} \leftarrow \theta^{Q}$, $\theta^{\mathcal{A}^{'}} \leftarrow \theta^{\mathcal{A}}$ **for** *episode = 1,* $\mathcal{E}$ **do**
3 **Initialize the replay buffer** Φ **with following steps**
4 **for** *mini batch = 1, c* **do**
5 ▷ Randomly generate a state space S with size of minibatch ϕ and calculate corresponding action space A with weights $\theta^{\mathcal{A}^{'}}$, i.e., ϕ: $\mathbf{R}^{\varphi}$
6 ▷ Input the pair of S and A in size of φ to **IoV network environments** and observe the set of next state space $S^{'}$ and reward set R for each state-action pair
7 ▷ Store the each transition pair $\xi = (s, a, r, s^{'})$
8 **end**
9 **Then, update the target networks iteratively**
10 **for** *time step = 1,* $\mathcal{T}$ **do**
11 ▷ Sample a random minibatch κ **refresh** κ in Φ
12 ▷ Set $y_i = r_i + \delta Q^{'}(s_{i+1}, \mathcal{A}^{'}(s_{i+1}|\theta^{\mathcal{A}^{'}})|\theta^{Q^{'}})$
13 ▷ Update critic network by minimizing the loss: $L = \frac{1}{\varphi}\sum_i (y_i - Q(s_i, a_i|\theta^{Q}))^2$
14 ▷ Update the actor policy $\pi_{\mathcal{A}}$ using the sampled policy gradient: $\nabla_{\theta^{\mathcal{A}}} J \approx \frac{1}{\varphi}\sum_i \nabla_a Q(s, a|\theta^{Q})$
15 ▷ Utilize soft update to the target networks $Q^{'}$ and $\mathcal{A}^{'}$:

$$\theta^{Q^{'}} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q^{'}} \tag{10}$$

$$\theta^{\mathcal{A}^{'}} \leftarrow \tau\theta^{\mathcal{A}} + (1-\tau)\theta^{\mathcal{A}^{'}} \tag{11}$$

16 **end**
17 **end**

ii) After the Φ is fully calculated, the minibatch of transitions κ is randomly sampled from the replay buffer Φ. Then, for ith transition pair of κ, it is utilized for calculating the difference between target value y_i and model value $Q(s_i, a_i|\theta^{Q})$ to update the critic network with the gradients obtained from the difference. In addition, stochastic policy gradient is utilized to update parameters of the actor network as per line 15.

iii) Overall, the updated parameters of critic and actor networks are utilized to update the target parameters of $Q^{'}$ and $\mathcal{A}^{'}$ with soft update weight τ for efficient and stable learning (i.e., (10) and (11)). Note that the sampled κ is refreshed with another *trained* transition pairs for better learning procedure after it is sampled. The computational complexity of this algorithm depends on the stochastic policy gradient method for minimizing loss; and our proposed algorithm does not exceed the complexity of stochastic policy gradient.

C.4 Reward

A comprehensive revenue of $\mathcal{M}$ is used as our caching scheme's reward. The $\mathcal{M}$ takes composite action $\{V_{N\times K}, L_{N\times K}\}$, when it observes the state of mmWave IoV networks as $\{C_{N\times K}, B_{N\times K}, H_{N\times K}, P_{N\times K}\}$, and get the next state of IoV networks $\{C^{'}_{N\times K}, B^{'}_{N\times K}, H^{'}_{N\times K}, P^{'}_{N\times K}\}$ and weighted average reward sum R of the considered system reward including (i) quality variation, (ii) packet drop occurrence, and (iii) playback stall. These sub-rewards are denoted as r^q, r^p, and r^s, respectively. The total reward R of each episode is calculated as the average of transitions' rewards of κ sampled from Φ. Note that the sub-rewards are calculated in vehicle-by-vehicle manner, and each of them is added together for each reward domain as R^q, R^p, and R^s, respectively. That is, the episode reward R is equal to $R^q + R^p + R^s$.

First, in the case of the reward of quality r^q, it is determined by the action taken by $\mathcal{M}$, which is $V_{N\times K}$. The IoV network environment, e, which interacts with $\mathcal{M}$, calculates the r^q by comparing $H^{'}_{N\times K}$ and $H_{N\times K}$. Then e compares the cumulative average quality of video among them and gives a weighted reward to $\mathcal{M}$ if the expected quality corresponding to the allocated power $v_{i,j}$ originates an enhancement of the provisioned video quality and vice versa. The $\mathcal{M}$ can get reward if the action for u_i results in a higher quality of video than its previous average video quality. However, if not, $\mathcal{M}$ gets negative reward of r^q as penalty, which represents the degradation of QoS of u_i. Specifically, the quality of video is determined by the data rate, which can be calculated by:

$$g_{i,j} = \frac{g^{TX}_{i,j} g^{RX}_{i,j} \mu^2}{16\pi^2 (\frac{d_{i,j}}{d_0})^{\eta}}, \tag{12}$$

$$SINR_{i,j} = \frac{v_{i,j} g_{i,j}}{\sum_{k\in\mathcal{U}} v_{k,j} g_{k,j} + \sigma^2}, \tag{13}$$

$$a_{i,j} = \frac{W}{K_j}\log_2(1 + SINR_{i,j}), \forall j \in \mathcal{X}. \tag{14}$$

The $g_{i,j}$ in (12) represents the power gain from jth mBS to ith user. In addition, $g^{TX}_{i,j}$ and $g^{RX}_{i,j}$ stand for transmit antenna gain and receive antenna gain from jth mBS to ith user. Moreover, the μ represents the wavelength and $d_{i,j}$ and d_0 represents distance from jth mBS to ith user and far field reference distance, respectively. Lastly, the η represents the path-loss exponent. The $v_{i,j}$ in (13) represents the transmit power from jth mBS to ith user, σ^2 is the variance of additive white Gaussian noise (AWGN). According to Shannon's capacity formula, the achievable rate for ith user from jth mBS is as (14). The W stands for the system bandwidth, and K_j is the total number of users associated with jth mBS. Thus, each user can utilize $1/K_j$ of the total frequency bandwidth of each mBS. Based on the $a_{i,j}$, each user can receive the corresponding quality of video chunks from associated mBS.

Next, the r^p is calculated by observing the current buffer occupancy of each u_i, allocation action $L_{N\times K}$ of $\mathcal{M}$, and the buffer saturation rate $\mathcal{F}$. If the difference between the sum of buffer occupancy of u_i with cache x_j and $\mathcal{F}$ exceeds $\bar{b}$, then the packet drop occurs at u_i throughout the video provisioning service. This is, $\mathcal{M}$ gets punished by attaining minus rewards of r^p because the action of $\mathcal{M}$ originated the spectrum waste and power consumption of the corresponding mBS. By exploiting this reward structure, the MBS learns to cache the video chunks in a way that computational overhead and communication loss derived from unnecessary delivery service are dismissed.

Finally, the r^s can be computed by subtracting $\mathcal{F}$ from $b_{i,j}$ and adding $c_{i,j}$. If the result is positive, the u_i can playback pro-

visioned video chunks without any stall. On the other hand, if the result is less than zero, then the video playback at u_i can be stall, which results in a deteriorating QoS for u_i. Therefore, we define R as:

$$\begin{aligned} R =& R^q + R^p + R^s \\ =& \sum_{j=0}^{K-1} \sum_{i=0}^{N-1} \left(\psi r_{i,j}^q + \Xi r_{i,j}^p + \aleph r_{i,j}^s \right) \\ =& \sum_{j=0}^{K-1} \sum_{i=0}^{N-1} \left(\psi \cdot p_{i,j} \cdot r^q \left(\left| \left(\frac{h_{i,j}}{q_{i,j}} \right)^+ \right| - \left| \left(\frac{h_{i,j}}{q_{i,j}} \right)^- \right| \right) \right. \\ & + \Xi \cdot p_{i,j} \cdot r^p \left(\left| \left(\frac{b_{i,j}+l_{i,j}-\mathcal{F}}{\bar{b}} \right)^+ \right| - \left| \left(\frac{b_{i,j}+l_{i,j}-\mathcal{F}}{\bar{b}} \right)^- \right| \right) \\ & \left. + \aleph \cdot p_{i,j} \cdot r^s \left(\left| \left(\frac{1}{b_{i,j}+l_{i,j}-\mathcal{F}} \right)^\triangleleft \right| - \left| \left(\frac{1}{b_{i,j}+l_{i,j}-\mathcal{F}} \right)^\triangleright \right| \right) \right), \end{aligned} \tag{15}$$

where ψ, Ξ, and $\aleph$ represent the reward weights of r^q, r^p, and r^s, respectively. The $(a/b)^+$ is a function that returns 1 if $a < b$ or 0 otherwise. Moreover, $(a/b)^-$ is a function that returns 1 if $a > b$ and 0 otherwise. The $(1/c)^\triangleleft$ function returns 0 if $c \to -\infty$ and returns 1 otherwise. Lastly, The $(1/c)^\triangleright$ function returns 1 if $c \to -\infty$ and returns 0 otherwise. By utilizing these functions, the (15) represents the total system reward calculation procedure with respect to the quality variation, packet drop occurrence, and playback stall.

V. PERFORMANCE EVALUATION

We utilized various simulations for verifying the performance of our proposed power-cache aware caching scheme in mmWave based IoV networks. The caching scheme is evaluated with these simulations by measuring the corresponding results with respect to the aforementioned interest of rewards, given that the state of IoV network is observable by the agent. We leveraged TensorFlow [39] in our simulations to implement our proposed DDPG based caching scheme. We first present the simulation settings and then discuss the results. As many reinforcement learning literature has shown results based on empirical convergence without complexity analysis [40]–[44], we propose simulation results based on this approach.

A. Setup

In the following, we elaborate the implementation details of the proposed DDPG learning based video caching scheme in mmWave IoV network. First, we introduce hardware configuration for our simulation, and then show the overall design and implementation details of the software.

Hardware. For hardware, we used an NVIDIA DGX station equipped with 4 $\times$ Tesla V100 GPUs (total of 128GB of available memory) and Intel Xeon E5–2698 v4 2.2 GHz with 20 cores (total of 256 GB of available system memory) CPU.

Software. We also used Python with version 3.6 on Ubuntu 16.04 LTS to build the DDPG based caching scheme. In addition, we used Xavier initializer to avoid occurrence of vanishing gradient descent during the learning phase. The neural network is constructed with fully connected deep neural network, and the number of nodes in the hidden layer was 200.

Table 2. Simulation parameters setting.

Parameter	Value
Total episode $\mathcal{E}$	500
Time step T	100
Minibatch size	64
Discounting factor γ	0.95
Initial epsilon	0.9
Size of $\mathcal{D}$	1000
Optimizer	AdamOptimizer
Activation function	ReLU

We implemented both the DDPG based caching algorithm and the customized mmWave IoV networks in highway scenario. The agent in DDPG based caching algorithm continuously interacts with the dynamic IoV network environment and attains pairs of state transition. Accordingly, and in turn, the optimal caching policy can be acquired with policy gradient after the learning phase has converged. In addition, simulation parameters are summarized in Table 2.

B. Converged Performance for Each Learning Rate

First, the caching scheme is evaluated with three different values of learning rate γ. Fig. 3 to Fig. 5 show the tendency of convergence of learning phase throughout the episodes. Note that Fig. 3 to Fig. 5 have the same simulation setting of (K, N) =(20, 200) and $\rho = 0.143$ with different learning rates. For each learning rate simulation run, learning tendency for each reward is represented. For example, in case of Fig. 4, the impact of each reward category can be obtained from the gap between other measured values of the mixed reward. As the value of the green-lined graph—which represents the reward value without (w.o.) the packet drop occurrence reward—is getting higher, it can be considered that the $\mathcal{M}$ makes an optimal policy which considers the playback stall and quality of provisioned video to be more important than the packet drop for maximizing the QoS. Similarly, the red-lined graph in Fig. 4 is getting lower and is converged at specific value. It can be considered that the total reward value of caching scheme can be underestimated without the quality reward value, which means the importance of the quality reward on the learning phase is not negligible.

When $\gamma = 10^{-4}$, an interesting learning tendency can be observed in Fig. 5. While the red-lined graph in Fig. 5 does not quite change over the entire learning phase, other graphs are dramatically increased and finally converged at the optimal point. That is, the $\mathcal{M}$ learns the caching policy to maximize the quality reward than other criteria. The red-lined graph, which is the mixed reward value, consists of packet drop occurrence reward and playback stall reward, and does not change, while the other two graphs are sharply increased indicating that the quality reward is dramatically increased.

In summary, the total reward can be illustrated as in Fig. 6. Throughout the learning phase, the $\mathcal{M}$ with different learning rate γ learns its caching policy, and the policy can be evaluated by the system reward criteria as mentioned earlier. In case of

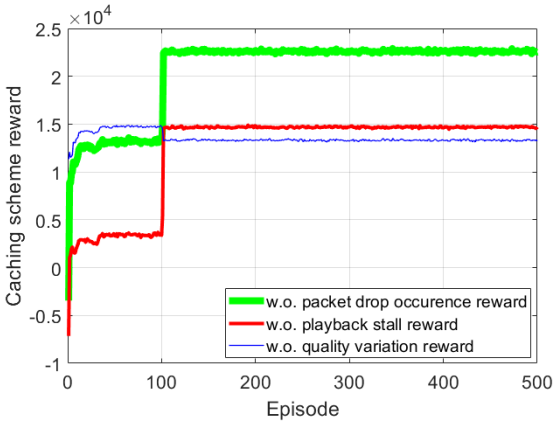

Fig. 3. Learning phase of proposed scheme ($\gamma = 10^{-3}$).

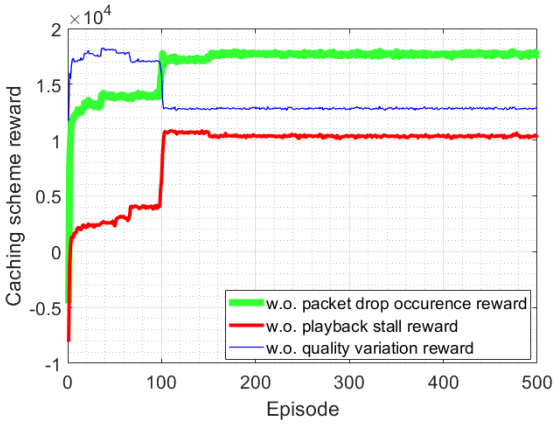

Fig. 4. Learning phase of proposed scheme ($\gamma = 5 \times 10^{-4}$).

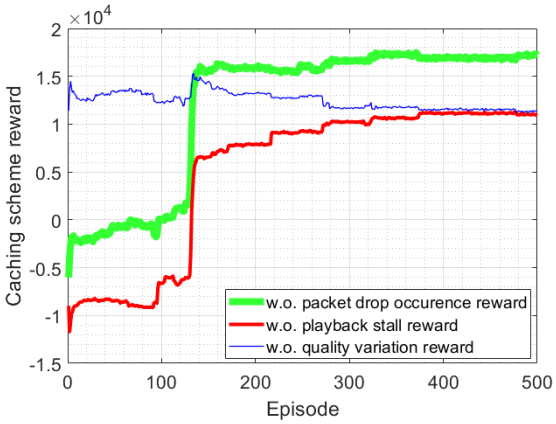

Fig. 5. Learning phase of proposed scheme ($\gamma = 10^{-4}$).

$\gamma = 10^{-3}$ and $\gamma = 5 \times 10^{-4}$, the $\mathcal{M}$ gets a converged caching policy around the 100th episode. However, in case of $\mathcal{M}$ with smaller γ, the $\mathcal{M}$ optimized its policy later. Therefore, our proposed power-cache aware video caching scheme accomplished stable and optimal video provisioning service towards vehicles in mmWave based distributed IoV networks.

As the optimal caching policy is attained, the $\mathcal{M}$ can immediately allocate power and cache units toward the distributed

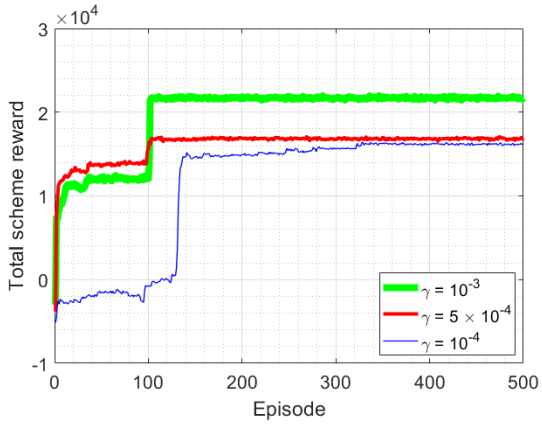

Fig. 6. Total reward convergence for each learning rate.

mBS as the system state is observed by $\mathcal{M}$ and thus caching scheme maximizes the QoS of the users. This caching policy differs from the classical caching scheme's policy, which needs to calculate the optimal caching strategy for each observation of IoV networks over time. Thus, the proposed caching scheme is highly affordable for optimal power and cache allocation of mBSs to provision superior quality and playback experience while seamless service is possible.

C. Robustness on Scalability

In the following, we argue for the importance of scalability in IoV networks. As the scale of the considered IoV networks gets larger, calibrating the optimal caching policy for seamless video services is hard to accomplish with classical approaches. Moreover, when the number of objective to optimize becomes larger calculating the optimal point for seamless video services.

Fig. 7 illustrates the total reward value convergence tendency throughout the learning phase. Note that the total reward of each case is proportional to the scale of the IoV networks. In addition, the ρ of FSMC model is set to 0.186, where the system average velocity of vehicles is 100 km/h. Moreover, the learning rate γ was set to 10^{-3}. Originally, the action space of Fig. 3 to Fig. 6 was $4000 = 20 \times 200$, where the IoV networks in Fig. 7 is 5000, 7500, and 20000. That is, the robustness of the proposed caching scheme with respect to scalability is validated through simulation in Fig. 7. Each scale of IoV networks in Fig. 7 showed converged performance for provisioning optimal quality of video and mitigated playback stall phenomenon through learning power and cache allocation toward mBSs.

Next, the learning tendency of average quality level with respect to the controlled power of mBS's transmitter and unit size of mBS's cache in various scale of IoV networks are proposed in Figs. 8 and 9. For power control aspects, the $\mathcal{M}$ with scale of $(K, N) = (20, 250)$ and $(K, N) = (25, 300)$ learns optimal power allocation toward mBSs on the road side, which results in sufficient data rate and can be supported toward users so that maximized quality of video (i.e., 4K resolution) can be provisioned. Besides, as the scale of IoV networks is $(K, N) = (40, 500)$, which is $5\times$ more dense compared to setting of Fig. 3 to Fig. 5, the $\mathcal{M}$ learned to allocate power corresponding to 720p resolution of video toward users with limited spectrum availabil-

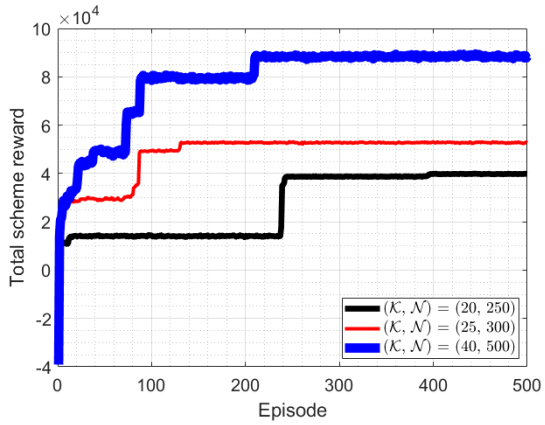

Fig. 7. Total reward convergence for each scale of IoV networks.

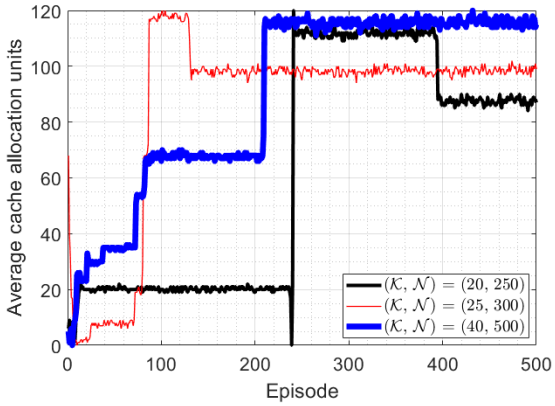

Fig. 9. Average size of proactively allocated cache of mBSs.

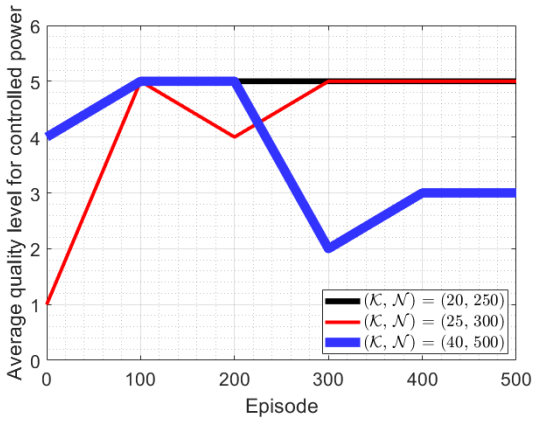

Fig. 8. Average quality of provisioned video toward vehicles with respect to controlled power of mBS.

ity.

Finally, $\mathcal{M}$ learns to allocate cache size toward mBS for supporting seamless video retrieval at neighboring users. As in Fig. 9, the $\mathcal{M}$ with scale of $(K, N) = (20, 250)$ and $(K, N) = (25, 300)$ learns to allocate smaller cache size than scale of $(K, N) = (40, 500)$. That is, the $\mathcal{M}$ with larger scale learns caching policy to allocate low power utilization strategy. However, the $\mathcal{M}$ stabilizes the distributed IoV networks with more flourish cache size for each user so that playback stall problem at user can be mitigated. Besides, for smaller scales, $\mathcal{M}$ aims to learn the caching scheme to achieve a maximized average quality level of provisioned video (i.e., higher power allocation of mBS). Therefore, proposed power-cache aware video caching scheme in distributed mmWave IoV networks enables us to learn the optimal caching policy, which accomplishes an optimal power and cache allocation toward mBSs and attains stabilized performance even for an enlarged scale of IoV networks.

VI. CONCLUSION AND FUTURE WORK

We proposed a deep reinforcement learning based video caching scheme in mmWave IoV networks to optimize power consumption and cache allocation of mBS with minimum number of stall events for seamless services. With our proposed caching scheme, stabilized and optimized caching options in a large-scale distributed IoV networks can be achieved as the system state is observed. Through an extensive set of simulations, the proposed caching scheme is shown to be appropriate for learning a massive scale of action space and stabilized learning performance, even when the scale of the considered distributed IoV networks is enlarged.

As future work directions, real-world implementation and its corresponding prototype-based performance evaluation will be considered. Furthermore, additional performance evaluations in order to compare with the other reinforcement learning algorithms will be intensively conducted. Lastly, the extension of our work with multi-agent deep reinforcement learning algorithms is worthy to consider in order to build scalable large-scale systems with multiple distributed base stations. To guarantee the convergence in multi-agent deep reinforcement learning, we need more sophisticated and well-designed reward functions and action spaces.

ACKNOWLEDGEMENTS

This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2018-0-00170, Virtual Presence in Moving Objects through 5G) and also by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2017-0-01637) supervised by the IITP (Institute for Information & Communications Technology Promotion). J. Kim, A. Mohaisen, and W. Lee are the corresponding authors of this paper.

REFERENCES

[1] L. Wei, R. Q. Hu, Y. Qian, and G. Wu, "Key elements to enable millimeter wave communications for 5G wireless systems," *IEEE Wireless Commun.*, vol. 21, no. 6, pp. 136–143, Dec. 2014.

[2] M. A. Salkuyeh and B. Abolhassani, "Optimal video packet distribution in multipath routing for urban VANETs," *J. Commun. Netw.*, vol. 20, no. 2, pp. 198–206, Apr. 2018.

[3] J. G. Andrews *et al.*, "What will 5G be?," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, June 2014.

[4] T. E. Bogale and L. B. Le, "Massive MIMO and mmWave for 5G wireless HetNet: Potential benefits and challenges," *IEEE Veh. Technol. Magazine*, vol. 11, no. 1, pp. 64–75, Mar. 2016.

[5] J. Kim, G. Caire, and A. F. Molisch, "Quality-aware streaming and scheduling for device-to-device video delivery," *IEEE/ACM Trans. Netw.,* vol. 24, no. 4, pp. 2319–2331, Aug. 2016.
[6] Cisco, "Cisco visual networking index: Global mobile data traffic forecast, 2016–2021 Q&A," https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-forecast-qa.html, July 2018, [Accessed: 2019-01-08].
[7] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5G: RAN, core network and caching solutions," *IEEE Commun. Surveys Tuts.,* vol. 20, no. 4, pp. 3098–3130, Fourth Quarter 2018.
[8] Y. Niu, C. Gao, Y. Li, L. Su, and D. Jin, "Exploiting multi-hop relaying to overcome blockage in directional mmwave small cells," *J. Commun. Netw.*, vol. 18, no. 3, pp. 364–374, June 2016.
[9] J. Kim, Y. Tian, S. Mangold, and A. F. Molisch, "Joint scalable coding and routing for 60 GHz real-time live HD video streaming applications," *IEEE Trans. Broadcasting*, vol. 59, no. 3, pp. 500–512, Sept. 2013.
[10] M. Baianifar, S. M. Razavizadeh, H. Akhlaghpasand, and I. Lee, "Energy efficiency maximization in mmWave wireless networks with 3D beamforming," *J. Commun. Netw.*, vol. 21, no. 2, pp. 125–135, Apr. 2019.
[11] J. Kim and A. F. Molisch, "Fast millimeter-wave beam training with receive beamforming," *J. Commun. Netw.*, vol. 16, no. 5, pp. 512–522, Oct. 2014.
[12] J. Kim, Y. Tian, S. Mangold, and A. F. Molisch, "Quality-aware coding and relaying for 60 GHz real-time wireless video broadcasting," in *Proc. IEEE ICC*, June 2013, pp. 5148–5152.
[13] S. Park, B. Kim, H. Yoon, and S. Choi, "RA-eV2V: Relaying systems for LTE-V2V communications," *J. Commun. Netw.*, vol. 20, no. 4, pp. 198–206, Aug. 2018.
[14] T. S. Rappaport *et al.*, "Millimeter wave mobile communications for 5G cellular: It will work!," *IEEE Access,* vol. 1, no. 1, pp. 335–349, 2013.
[15] J. Kim and A. F. Molisch, "Quality-aware millimeter-wave device-to-device multi-hop routing for 5G cellular networks," in *Proc. IEEE ICC*, June 2014, pp.5251–5256.
[16] S. Zhang, N. Zhang, X. Fang, P. Yang, and X. S. Shen, "Self-sustaining caching stations: Toward cost-effective 5G-enabled vehicular networks," *IEEE Commun. Mag.,* vol. 55, no. 11, pp. 202–208, Nov. 2017.
[17] N. Magaia, Z. Sheng, P. R. Pereira, and M. Correia, "REPSYS: A robust and distributed incentive scheme for in-network caching and dissemination in vehicular delay-tolerant networks," *IEEE Wireless Commun.,* vol. 25, no. 3, pp. 65–71, June 2018.
[18] H. Ahlehagh and S. Dey, "Video-aware scheduling and caching in the radio access network," *IEEE/ACM Trans. Netw.,* vol. 22, no. 5, pp. 1444–1462, Oct. 2014.
[19] *Highway Data Explorer* [Online]. Available: http://dtdapps.coloradodot.info/otis/HighwayData
[20] L. Yao, A. Chen, J. Deng, J. Wang, and G. Wu, "A cooperative caching scheme based on mobility prediction in vehicular content centric networks," *IEEE Trans. Veh. Technol.,* vol 67, no. 6, pp. 5435–5444, June 2017.
[21] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," MIT press, 2018.
[22] Y. Guo, Q. Yang, F. R. Yu, and V. C. Leung, "Cache-enabled adaptive video streaming over vehicular networks: A dynamic approach," *IEEE Trans. Veh. Technol.,* vol. 67, no. 6, pp. 5445–5459, June 2018.
[23] K. Poularakis, G. Iosifidis, A. Argyriou, I. Koutsopoulos, and L. Tassiulas, "Caching and operator cooperation policies for layered video content delivery", in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–9.
[24] Y. Huang, X. Song, F. Ye, Y. Yang, and X. Li, "Fair caching algorithms for peer data sharing in pervasive edge computing environments," in *Proc. IEEE ICDCS,* June 2017, pp. 605–614.
[25] S. Fu, P. Duan, and Y. Jia, "Content-exchanged based cooperative caching in 5G wireless networks," in *Proc. IEEE GLOBECOM,* Dec. 2017, pp. 1–6.
[26] S. Arabi, E. Sabir, and H. Elbiaze, "Information-centric networking meets delay tolerant networking: Beyond edge caching," in *Proc. IEEE WCNC,* Apr. 2018, pp. 1–6.
[27] R. Kim, H. Lim, and B. Krishnamachari, ""Prefetching-based data dissemination in vehicular cloud systems," *IEEE Trans. Veh. Technol.*, vol. 65, no. 1, pp. 292–306, Jan. 2015.
[28] G. Mauri, M. Gerla, F. Bruno, M. Cesana, and G. Verticale, "Optimal content prefetching in NDN vehicle-to-infrastructure scenario," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2513–2525, June 2016.
[29] M. Chen *et al.*, "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.
[30] T. T. Le and R. Q. Hu, "Mobility-aware edge caching and computing in vehicle networks: A deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10190–10203, Nov. 2018.
[31] Y. He, F. R. Yu, N. Zhao, V. C. Leung, and H. Yin, "Software-defined networks with mobile edge computing and caching for smart cities: A big data deep reinforcement learning approach," *IEEE Commun. Mag.,* vol. 55, no. 12, pp. 31–37, Dec. 2017.
[32] Y. He, N. Zhao, and H. Yin, "Integrated networking, caching, and computing for connected vehicles: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.,* vol. 67, no. 1, pp. 44–55, Jan. 2017.
[33] Y. He *et al.*, "Deep reinforcement learning-based optimization for cache-enabled opportunistic interference alignment wireless networks," *IEEE Trans. Veh. Technol.,* vol. 66, no. 11, pp. 10433–10445, Nov. 2017.
[34] S. Lin *et al.*, "Fast simulation of vehicular channels using finite-state markov models," *IEEE Wireless Commun. Letters, early access*, 2019.
[35] Z. Ning, X. Wang, F. Xia, and J. J. Rodrigues, "Joint computation offloading, power allocation, and channel assignment for 5G-enabled traffic management systems," *IEEE Trans. Ind. Inf.*, vol. 15, no. 5, pp. 3058–3067, May 2019.
[36] N. Wang, E. Hossain, and V. K. Bhargava, "Joint downlink cell association and bandwidth allocation for wireless backhauling in two-tier HetNets with large-scale antenna arrays," *IEEE Trans. Wireless Commun.,* vol. 15, no. 5, pp. 3251–3268, Jan. 2016.
[37] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory,* vol. 59, no. 12, pp. 8402–8413, Sept. 2013.
[38] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc. IEEE INFOCOM,* Mar. 1999, pp. 126–134.
[39] Y. J. Mo, J. Kim, J.-K. Kim, A. Mohaisen, and W. Lee, "Performance of deep learning computation with TensorFlow software library in GPU-capable multi-core computing platforms," in *Proc. IEEE ICUFN*, July 2017, pp. 240–242.
[40] J. Clausen, W. L. Boyajian, L. M. Trenkwalder, V. Dunjko, and H. J. Briegel, "On the convergence of projective–simulation–based reinforcement learning in Markov decision processes," in *arXiv preprint arXiv:1910.11914,* 2019.
[41] V. Mnih *et al.*, "Asynchronous methods for deep reinforcement learning," in *Proc. ICML*, June 2016, pp. 1928–1937.
[42] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," in *Proc. ICLR*, May 2016, pp. 1–14.
[43] M. Hessel *et al.*, "Rainbow: Combining improvements in deep reinforcement learning," in *Proc. AAAI*, Feb. 2018, pp. 1–8.
[44] H. V. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. AAAI*, Feb. 2016, pp. 1–7.

Dohyun Kwon is currently a Research Engineer at Hyundai-Autoever, Seoul, Republic of Korea. He received his B.S. and M.S. degrees in Computer Science and Engineering from Chung-Ang University, Seoul, Republic of Korea, in 2018 and 2020, respectively. His research focus includes deep reinforcement learning for mobile networks.

Joongheon Kim (M'06–SM'18) is currently an Assistant Professor of Electrical Engineering at Korea University, Seoul, Korea. He received the B.S. and M.S. degrees in Computer Science and Engineering from Korea University, Seoul, Korea, in 2004 and 2006, respectively; and the Ph.D. degree in Computer Science from the University of Southern California (USC), Los Angeles, CA, USA, in 2014. Before joining Korea University as an Assistant Professor in 2019, he was with LG Electronics as a research engineer (Seoul, Korea, 2006–2009), InterDigital as an intern (San Diego, CA, USA, 2012), Intel Corporation as a systems engineer (Santa Clara in Silicon Valley, CA, USA, 2013–2016), and Chung-Ang University as an Assistant Professor (Seoul, Korea, 2016–2019).

He is a Senior Member of the IEEE. He was a recipient of Annenberg Graduate Fellowship with his Ph.D. admission from USC (2009), Intel Corporation Next Generation and Standards (NGS) Division Recognition Award (2015),

Haedong Young Scholar Award by KICS (2018), IEEE Veh. Technol. Society (VTS) Seoul Chapter Award (2019), Outstanding Contribution Award by KICS (2019), Gold Paper Award from IEEE Seoul Section Student Paper Contest (2019), and IEEE Systems J. Best Paper Award (2020).

David Aziz Mohaisen earned his M.Sc. and Ph.D. degrees from the University of Minnesota in 2012. Currently, he is an Associate Professor of Computer Science at the University of Central Florida. Prior to joining Central Florida, he was an Assistant Professor at SUNY Buffalo (2015–2017), a Senior Research Scientist at Verisign Labs (2012–2015), and a Researcher at ETRI (2007–2009). He was awarded the Summer Faculty Fellowship from the US AFOSR (2016), the Best Student Paper at ICDCS (2017), the Best Paper Award at WISA (2014), the Best Poster Award at IEEE CNS (2014), and a Doctoral Dissertation Fellowship from the University of Minnesota (2011). He is in the editorial board of IEEE Trans. Mobile Computing. He is a Senior Member of ACM and a Senior Member of IEEE.

Wonjun Lee received the B.S. and M.S. degrees in Computer Engineering from Seoul National University, Seoul, South Korea, in 1989 and 1991, respectively, the M.S. degree in Computer Science from the University of Maryland at College Park, College Park, MD, USA, in 1996, and the Ph.D. degree in Computer Science and Engineering from the University of Minnesota, Minneapolis, MN, USA, in 1999. In 2002, he joined the faculty of Korea University, Seoul, where he is currently a Professor with the Department of Computer Science and Engineering. He has authored or co-authored over 180 papers in refereed international journals and conferences. His research interests include communication and network protocols, optimization techniques in wireless communication and networking, security and privacy in mobile computing, and radio frequency powered computing and networking. Dr. Lee has served as a Technical Program Committee member for the IEEE International Conference on Computer Communications from 2008 to 2018. He was associated with the Computing Machinery International Symposium on Mobile Ad Hoc Networking and Computing from 2008 to 2009 and the IEEE International Conference on Computer Communications and Networks from 2000 to 2008 and over 118 international conferences.