Dynamic Video Delivery using Deep Reinforcement Learning for Device-to-Device Underlaid Cache-Enabled Internet-of-vehicle Networks

Minseok Choi, Myungjae Shin, and Joongheon Kim

Abstract: **This paper addresses an Internet-of-vehicle network that utilizes a device-to-device (D2D) underlaid cellular system, where distributed caching at each vehicle is available and the video streaming service is provided via D2D links. Given the spectrum reuse policy, three decisions having different timescales in such a D2D underlaid cache-enabled vehicular network were investigated: 1) The decision on the cache-enabled vehicles for providing contents, 2) power allocation for D2D users, and 3) power allocation for cellular vehicles. Since wireless link activation for video delivery could introduce delays, node association is determined in a larger timescale compared to power allocations. We jointly optimize these delivery decisions by maximizing the average video quality under the constraints on the playback delays of streaming users and the data rate guarantees for cellular vehicles. Depending on the channel and queue states of users, the decision on the cache-enabled vehicle for video delivery is adaptively made based on the frame-based Lyapunov optimization theory by comparing the expected costs of vehicles. For each cache-enabled vehicle, the expected cost is obtained from the stochastic shortest path problem that is solved by deep reinforcement learning without the knowledge of global channel state information. Specifically, the deep deterministic policy gradient (DDPG) algorithm is adopted for dealing with the very large state space, i.e., time-varying channel states. Simulation results verify that the proposed video delivery algorithm achieves all the given goals, i.e., average video quality, smooth playback, and reliable data rates for cellular vehicles.**

Index Terms: **Deep reinforcement learning, device-to-device underlaid network, vehicular networks, video delivery, wireless caching.**

I. INTRODUCTION

WITH the increasingly growing number of mobile devices, it is expected that tens of exabytes of global data traffic will be handled on a daily basis, 70% of which will consist of on-demand video streaming services [1]. In on-demand video streaming services, a small number of popular contents is requested at ultra-high rates, i.e., most of the requests are repeated, and thus, provision of the desired contents from the remote base stations (BSs) would waste resources. In this regard, the wireless caching technology discussed in [2], whereby the BS pushes popular contents to cache-enabled nodes with limited storage space during off-load time so that these nodes can provide popular contents directly to nearby mobile users, is advantageous for video streaming services.

As device-to-device (D2D) communication has become a promising technology for improving spectral efficiency, a D2D-assisted caching network has been studied [3], [4], where mobile devices can store popular contents and directly respond to the requests of neighboring users. Especially for delay-sensitive content delivery, the necessary decision on which cache-enabled device will deliver the desired content to the streaming user has been extensively researched. The simplest method is to let the cache-enabled node with the strongest channel condition deliver the content to the streaming user [5]. The advanced node association schemes have been developed for heterogeneous caching networks by jointly optimizing routing and caching [7], managing interference among D2D-assisted delivery links [8] and allowing cooperation between adjacent BSs [9]. However, these methods did not consider the different quality levels of contents and assume that all the cached contents have the same size.

Since multimedia contents (e.g., video files) can be encoded to multiple versions that differ in quality level (e.g., in their peak-signal-to-noise-ratio (PSNR) and spatial resolution) [10], each cache-enabled node can store the identical video contents of different quality levels. In this case, the decision on which of the cache-enabled nodes for content delivery becomes closely related to content quality that the user can enjoy [11]. There exist several studies on methods that dynamically select the quality level of the desired video [12]–[14] or maximize a network utility function of time-average video quality [15]. Whereas the video delivery policies presented in [12], [15] are operated at the BS side, this paper considers a scenario where users dynamically choose the desired content quality as in dynamic adaptive streaming over HTTP (DASH) [16]. In [11], [13], [14], the authors also considered video quality selection at the user side, but not resource allocations in the D2D underlaid cellular system. In addition, the above studies [10]–[15] requires global channel state information (CSI); however, in practical vehicular networks, it is difficult to track the time-varying channel gains due to high mobility of vehicle users.

The reinforcement learning (RL) algorithms have been recently employed to proactively cache popular contents in the scenario in the absence of global CSI in [17]–[19]; however,

Manuscript received April 12, 2020; revised January 08, 2020; approved for publication by Patrick Seeling, Division II Editor, February 01, 2021.

This research was supported in part by the National Research Foundation of Korea (2019R1A2C4070663) and in part by the National Research Foundation of Korea under Grant NRF-2020R1G1A1101164.

M. Choi is with the Department of Telecommunication Engineering, Jeju National University, Jeju 63243, Korea, email: ejaqmf@jejunu.ac.kr

M. Shin is with Seoul National University Hospital, Seoul 03080, Korea, email: asdfwv@naver.com.

J. Kim is with the School of Electrical Engineering, Korea University, Seoul 02841, Korea, email: joongheon@korea.ac.kr.

J. Kim is the corresponding author.

Digital Object Identifier: 10.23919/JCN.2021.000006

1229-2370/19/$10.00 © 2021 KICS

content delivery is not optimized in these works. In addition, the RL-based dynamic resource allocation method for edge computing networks is proposed in [20], but content delivery is not considered. Content placement and delivery are also jointly optimized in cache-enabled D2D networks, based on the deep-Q network (DQN) [21] and the deep deterministic policy gradient (DDPG) frameworks [22]. In addition, the RL-based content delivery policy of a mobile device with service delay constraint was proposed in [23]. However, the above studies [21]–[23] did not consider the cache-enabled D2D underlaid cellular system and differentiated quality requirements of multimedia contents.

In parallel, D2D underlaid cellular systems have been extensively researched for efficient uses of spectrum resources, in which frequency bands are shared for both cellular users (CUEs) and D2D users (DUEs). In general, when spectrum is allocated to CUEs, the newly generated D2D link reuses the spectrum of one of the CUEs; therefore, the DUE interferes with the existing cellular and other D2D links. In order to manage the interference as well as to maximize the network performances, the advanced power controls and resource allocations have been proposed for the D2D underlaid cellular system in [24]–[31]. Both the centralized and the distributed power control schemes that improve the signal-to-interference-plus-noise ratio (SINR) were proposed and analyzed in D2D underlaid cellular networks in [24], and the scheme proposed in [25] achieves the proportional fairness among users. As the D2D underlaid cellular system supports vehicle-to-vehicle (V2V) connections, the global CSI is difficult to be obtained due to high mobility of vehicle users [32]. To deal with this issue, power controls that reduce the requirement of global CSI were presented by utilizing the vehicles' geographic features [26], large-scale channel fading information [27], the statistical CSI [28], or the path loss rather than instantaneous CSI [29]. Furthermore, the deep learning-based and the deep reinforcement learning (DRL)-based power allocation schemes not requiring global CSI have been proposed in [30] and [31], respectively. However, all of the above studies do not consider the content delivery in cache-enabled D2D underlaid cellular networks. The content delivery in D2D underlaid cellular networks was researched in [33]; however, this work focuses on offloading the cellular data traffic by D2D links rather than node association for content delivery in caching networks. The cache-enabled D2D underlaid cellular networks have been studied in [34] and [35] that proposed a caching method and a incentive mechanism respectively but not content delivery.

In this context, this paper jointly optimizes the node association and resource allocations without global CSI for content delivery in the D2D underlaid cache-enabled Internet-of-vehicle system. This paper reflects the two characteristics of cellular system-assisted D2D communications to the multimedia content delivery policy in wireless vehicular caching networks: spectrum reuse and high mobility. In the absence of global CSI, we propose a deep reinforcement learning (DRL) based adaptive delivery scheme which learns a policy that makes following decisions: 1) The cache-enabled vehicle that will deliver the desired content to the streaming user, 2) power allocations for D2D-assisted delivery links, and 3) power allocations for cellular links. Specifically, the delivery policy is learned by deep deterministic policy gradient (DDPG), which is a model-free and off-policy algorithm. The main contributions of this paper are as follows:

- A framework of the compromising characteristics of the D2D underlaid cellular system, the vehicular network, and the wireless caching network is presented. For such a network, the joint optimization problem of three decisions having different timescales is formulated: 1) Association with the cache-enabled vehicle for delivering multimedia contents (e.g., video files), 2) power allocation of the cache-enabled vehicle delivering the content via the D2D link, and 3) power allocation of the CUE whose the spectrum is reused by the D2D-assisted delivery link. The optimization problem maximizes the time-average video quality under the constraints on the limited playback delay of the DUE and the minimum data rate of the CUE.
- The problem of dynamic power allocations for both cellular and D2D links sharing the identical spectrum without the knowledge of global CSI is formulated based on a Markov decision process and solved by using the DRL approach. In contrast to the approaches in [26] and [27], the proposed approach dynamically changes power allocations to control interference and to limit the playback delay based on channel statistics and queue states. In order to achieve efficient and improved learning of the delivery policy as well as to deal with the very large state space, we adopt a DDPG-based method because the state space is continuous and massively large.
- Considering the interference between the CUEs and DUEs, the decision on the cache-enabled vehicle that will delivery the content, i.e., the D2D transmitter, is made under the frame-based Lyapunov optimization theory [36], in larger timescale than power allocations of cellular and D2D links. With the help of the DRL-based power allocations determined in smaller timescale, the node associations for content delivery can be also completed without global CSI.
- We present an evaluation via data-intensive simulations to verify the proposed video delivery policy, as well as to show the advantages of Lyapunov optimization theory and DRL-based approaches.

II. NETWORK MODEL

This section describes the D2D underlaid cache-enabled vehicular network that we considered, and the user queue model is introduced. Since we focus on the multimedia services (e.g., on-demand streaming), the delivered content chunks are accumulated in the user queue and the playback latency or stall events are closely related to the queue dynamics.

A. D2D Underlaid Cache-enabled Vehicular Network

This paper addresses the D2D underlaid vehicular caching network where a certain vehicle user can request a video file from one of the cache-enabled vehicles in its vicinity while some CUEs are communicating with the BS, as shown in Fig. 1. In Fig. 1, we can see that both cellular and D2D links coexist, and CUE 1 (or 2) and DUE 1 (or 2) share the identical frequency band. The server has already pushed popular video files during off-peak hours to cache-enabled vehicles, the storage size of

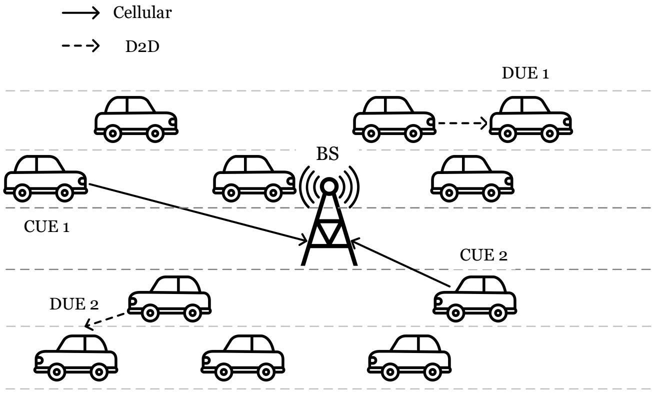


Fig. 1. D2D underlaid cache-enabled vehicular network.

which is finite. Since we focus on video delivery, caching policies are outside the scope of this paper and only cache-enabled vehicles that store the desired video are considered. Assume that the desired video has N quality levels and that there is no quality controller in the cache-enabled vehicles, and therefore, they can transmit only the video file of the fixed quality that the server pushes. In this case, the user can choose the quality level of the video file to be received; let $q \in \{1, \cdots, N\}$ denote the desired quality level. Therefore, the choice of the quality level of the video file to be received is consistent with the choice of the cache-enabled vehicle nearby for video delivery. When the video request of a certain user has been accepted at one of the nearby cache-enabled vehicles, a D2D link is activated between them for delivering the desired content. Therefore, the cache-enabled vehicle that is selected to deliver the requested content to the streaming user will be simply called the D2D transmitter.

Let CUEs send massive traffic data to nearby infrastructures, e.g., BSs or roadside units. These vehicles utilize high-capacity vehicle-to-infrastructure (V2I) communication via cellular links. However, there exist several pairs of highly mobile DUEs for video delivery via D2D links. For the D2D underlaid cellular system, the spectrum reuse model is utilized, i.e., both CUEs and DUEs share the spectrum resources. Whenever a D2D link is generated, one of the orthogonal resources of the CUEs is reused by the D2D link. This paper considers only one CUE and a pair of a D2D transmitter and a file-requesting user that reuse the spectrum of the considered CUE, and thus, the spectrum reuse policy for multiple CUEs and DUEs is outside its scope.

The cache-enabled vehicles are modeled using independent Poisson point processes (PPPs) with intensity λ. Assuming a probabilistic caching policy [37], let p_q be the caching probability for the desired video of quality q. Then, the PPP intensity of vehicles caching the desired video of quality q is λp_q. Suppose that the system does not allow any additional D2D link that reuses the identical spectrum of the considered CUE and the streaming user who is already downloading the desired content from certain cache-enabled vehicle. Then, the system can guarantee negligible interference between multiple D2D links. User mobility is also captured in the network model. The user is moving in a pre-determined direction and periodically searches for a cache-enabled vehicle from which to receive the desired video file continuously. As shown in Fig. 1, the geological distribution of cache-enabled vehicles in the vicinity of the user that store the desired file can vary at each time slot, and therefore, the decision on the cache-enabled vehicle to be used for video delivery should be appropriately updated.

Table 1. System description parameters.

N	Number of quality levels
$\alpha(t)$	Cache-enabled vehicle chosen at t
$P_c(t)$	Transmit power of cellular link at t
$P_d(t)$	Transmit power of D2D link at t
$Q(t)$	Queue backlog of DUE at t
$W(t)$	Virtual queue backlog at t
q	Video quality
$\mathcal{P}(q)$	Measure of video quality q
$S(q)$	Video file size of quality q
K	Number of frames
T	Time duration of a frame
t_c	Unit time slot duration
t_k	Beginning time of the k-th frame
$\mathcal{T}_k$	Time interval of the k-th frame
$R_c(t)$	Data rate of cellular link at t
$R_d(t)$	Data rate of D2D link at t
η_c	Minimum data rate of cellular link
P_d^0, P_c^0	Maximum power of cellular and D2D links
$\mathcal{L}(t)$	Lyapunov function
λ	Intensity of cache-enabled vehicle distribution
$\mathcal{B}$	Bandwidth
σ^2	Noise variance
V	Lyapunov coefficient
$\mathcal{C}_k$	Cache-enabled vehicle candidate set in $\mathcal{T}_k$
$\Theta(t)$	State of MDP at t
$\Xi(t)$	Action of MDP at t
$\mathcal{S}$	State space of MDP
$\mathcal{A}$	Action space of MDP
$r(t)$	Reward of MDP at t
$P_{s's}$	Transition probability from state s' to state s
π	Trained policy of DDPG algorithm

B. User Queue Model and Channel Model

A video content consists of many sequential chunks. The user receives the video content from the cache-enabled vehicle and processes data for video streaming services in units of chunks. Each chunk of a content is responsible for some playback time of the entire stream. Even if all the chunks are in the correct sequence, the quality of each chunk can differ in dynamic streaming. Therefore, users can dynamically choose the video quality levels in each chunk's processing time. It can be stated that, when a queue model is used, playback delay occurs when the chunk to be played has not yet arrived in the queue. Thus, the receiver queue dynamics collectively reflect the various factors that cause the playback delay. Therefore, we focus on limiting the queueing delay by dynamically adjusting the queue backlogs of the streaming user.

In general, a queue model has its own arrival and departure processes. The queue dynamics of the D2D user in each discrete time slot $t \in \{0, 1, \cdots\}$ can be represented as follows:

$$Q(t+1) = \max\{Q(t) + \mu(t) - c, 0\} \tag{1}$$

$$Q(0) = 0, \tag{2}$$

where $Q(t)$ and $\mu(t)$ represent the queue backlog and the arrival process of the DUE receiver at slot t, respectively. Since the playback rate of streaming is usually unchanging, the departure is assumed as a constant c here for simplicity. The queue states are updated in each unit time slot t, and the interval of each slot is determined to be the channel coherence time, t_c. Suppose a block fading channel, the channel gain of which is static during the processing of multiple chunks; then, $t_c = c\tau$, where τ is the chunk processing time and c is the positive integer.

In this study, the queue backlog $Q(t)$ counted the number of video chunks in the queue. $\mu(t)$ semantically represents the number of received chunks and clearly depends on the data rate of the D2D link between the streaming user and its associated cache-enabled vehicle and on the chunk size. The arrival process is:

$$\mu(t) = \left\lfloor \frac{R_d(\alpha(t), P_c(t), P_d(t), t) \cdot t_c}{S(q(\alpha(t)))} \right\rfloor, \tag{3}$$

where $\alpha(t)$, $P_c(t)$, and $P_d(t)$ denote the cache-enabled vehicle associated with the user (i.e., the D2D transmitter), the transmit power of the CUE, and the transmit power of $\alpha(t)$ at slot t, respectively, and $q(\alpha(t))$ is the quality level of the requested file that the D2D transmitter $\alpha(t)$ can provide. $R(\alpha(t), t)$ and $S(q(\alpha(t)), t)$ indicate the data rate of the D2D link and the chunk size of the requested file with the desired quality $q(\alpha(t))$ at slot t, respectively. Some video chunks can be only partially delivered as the channel condition varies at every time t. Because partial chunk transmission is meaningless in our algorithm, flooring is applied in (3).

A Rayleigh fading channel is assumed for the wireless links from all vehicles to infrastructures and vehicles. Denote the channel by $h = \sqrt{X}\beta g$, where $X = A/d^{\gamma}$ controls path loss with d being the server-user distance, the path loss component of A, and the decay exponent γ. In addition, β is a log-normal shadowing random variable with standard deviation ξ, and g represents the fast fading component with complex Gaussian distribution $g \sim CN(0, 1)$. In vehicular networks, fast fading components are not easily estimated at the receiver side owing to users' high mobility. Thus, in this study we considered the situation in which only the slow fading components, i.e., X and β, are known in advance, but the fast fading components are not known.

Consider a typical streaming user reusing the spectrum of a certain CUE. Then, the cellular link and the D2D link for streaming that share the spectrum are interfering with each other. Therefore, the data rate of the cellular link from the CUE to the BS is given by

$$R_d(t) = \mathcal{B} \log_2 \Big(1 + \frac{P_c(t)|h_{c,B}(t)|^2}{P_d(t)|h_{d,B}(\alpha(t), t)|^2 + \sigma^2}\Big), \tag{4}$$

where $\mathcal{B}$ is the bandwidth and σ^2 is the normalized noise variance. $h_{c,B}(t)$ and $h_{d,B}(\alpha(t), t)$ are respectively the channel fading gains from the CUE and the D2D transmitter $\alpha(t)$ to the BS. Similarly, the data rate of the D2D link between the streaming user and the D2D transmitter $\alpha(t)$ is given by

$$R_d(t) = \mathcal{B} \log_2 \Big(1 + \frac{P_d(t)|h_d(\alpha(t), t)|^2}{P_c(t)|h_{c,d}(t)|^2 + \sigma^2}\Big), \tag{5}$$

where $h_d(\alpha(t), t)$ and $h_{c,d}(t)$ are the channel gains of the D2D link for video delivery and of the interference link from the CUE to the streaming user, respectively. Note that the data rate of the video delivery link in (5) limits the number of chunks that can be delivered to the streaming user as given by (3), which depends on the associated cache-enabled vehicle as well as power allocations for cellular and D2D users.

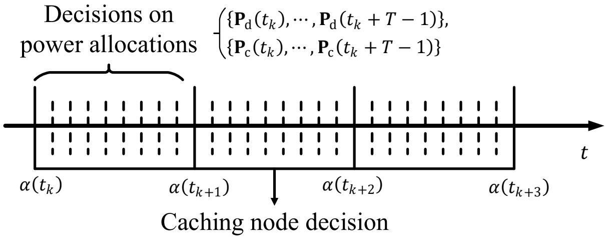

Decisions on
power allocations
$\{\mathbf{P}_d(t_k), \cdots, \mathbf{P}_d(t_k + T - 1)\},$
$\{\mathbf{P}_c(t_k), \cdots, \mathbf{P}_c(t_k + T - 1)\}$
t
$\alpha(t_k)$
$\alpha(t_{k+1})$
$\alpha(t_{k+2})$
$\alpha(t_{k+3})$
Caching node decision

Fig. 2. Different timescales for decisions on $\alpha(t)$, $P_d(t)$, and $P_c(t)$.

III. DYNAMIC NODE ASSOCIATIONS AND POWER ALLOCATIONS

This section introduces how the timescales of decisions on node association and power controls are different and formulates the optimization problem the maximizes the average video quality with the constraints of limited playback latency for DUEs and data rate guarantees for CUEs.

A. Decisions at Different Timescales

The goal of this study is to determine the appropriate decisions for video delivery at each slot t in the network model described in Section II: 1) The cache-enabled vehicle for video delivery $\alpha(t)$, 2) the transmit power of the cache-enabled vehicle $P_d(t)$, and 3) the transmit power of the cellular vehicle $P_c(t)$. The last decision is not directly related to video delivery, but very significantly affects video delivery in terms of interference. Here, the association with one of the cache-enabled vehicles in the vicinity of can be made after it has been determined which vehicles store the desired file, the delivery has been requested, and the request accepted. Therefore, the decision on the cache-enabled vehicle $\alpha(t)$ takes longer than that on the dynamic power allocation. Thus, we consider the scenario where the decision on $\alpha(t)$ is made at a larger timescale than that on $P_d(t)$ and $P_c(t)$.

In this context, the different timescales of decisions on $\alpha(t)$, $P_d(t)$, and $P_c(t)$ are shown in Fig. 2. The D2D transmitter and the CUE allocate $P_d(t)$ and $P_c(t)$ at time slots $t \in \{0, 1, 2, \cdots\}$, but the update of association with the D2D transmitter is performed at a larger timescale, $t \in \{1, T, 2T, \cdots\}$, where T is the time interval for the decision on $\alpha(t)$. The time slot for the k-th association is denoted by $t_k = (k-1)T$ for $n \in \{1, 2, \cdots\}$. Let the k-th frame for updates of association with the D2D transmitter $\alpha(t)$ be $T_k = \{t_k, t_k + 1, \cdots, t_k + T - 1\}$. Hereinafter, we use the term α_k rather than $\alpha(t)$ for $t \in \mathcal{T}_k$, because α_k remains unchanged over the frame $\mathcal{T}_k$. As shown in Fig. 2, after $\alpha(t)$ has been determined at time t_k, decisions on the transmit power levels of the D2D transmitter $\alpha(t)$ and the CUE, i.e., $P_d(t)$ and $P_c(t)$, are made over $t \in \mathcal{T}_k$ to maximize the average streaming quality while guaranteeing the data rate of the CUE, as well as limiting the playback delay of the streaming user.

The user can create a candidate set of nearby cache-enabled vehicles storing the desired video, denoted by $\mathcal{C}_k$; $\alpha(t_k) \in \mathcal{C}_k$. In existing studies, the nearest vehicle was usually chosen, because it can provide the best channel condition. However, because the streaming user is moving and there exists an interfering cellular vehicle, an association with the nearest vehicle is not always the best choice. Therefore, the user collects up to N candidates of cache-enabled vehicles located in its vicinity for video delivery, and selects that which can provide the best video quality and allows the user to avoid the playback delay during the frame $\mathcal{T}_k$.

B. Problem Formulation

For determining the appropriate video delivery policy, three performance metrics are considered: The average video quality, the playback delay of the streaming user, and the data rate of the CUE, the spectrum of which is reused by the streaming user. Based on these goals, we can formulate the optimization problem that maximizes the long-term average video quality constrained by the need to avert queue emptiness and guarantee the data rate of the CUE

$$\max_{\mathbf{P_d},\mathbf{P_c},\boldsymbol{\alpha}} \lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} \mathbb{E}[\mathcal{P}(q(\alpha_k))] \tag{6}$$

$$\text{s.t.} \lim_{K\to\infty} \frac{1}{KT}\sum_{t=1}^{KT} \mathbb{E}[Z(t)] < \infty \tag{7}$$

$$\lim_{K\to\infty} \frac{1}{KT}\sum_{t=1}^{KT} \mathbb{E}[R_c(P_c, P_d, t)] \geq \eta_c \tag{8}$$

$$0 \leq P_d \leq P_0^d \tag{9}$$

$$0 \leq P_c \leq P_0^c, \tag{10}$$

where $\mathcal{P}(q(\alpha_k))$ is the quality measure of $q(\alpha_k)$ and η_c is the minimum data rate for the CUE. P_0^d and P_0^c are the power budgets of all the cache-enabled vehicles and the CUE, respectively. The decision vectors are represented as $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_K]$, $\mathbf{P}_d = [P_d(0), P_d(1), \cdots, P_d(KT-1)]$, and $\mathbf{P}_c = [P_c(0), P_c(1), \cdots, P_c(KT-1)]$. Specifically, the expectation of (6) is with respect to random channel realizations and stochastic distributions of vehicles. The constraint (7) limits the playback delay of the streaming service, and the constraint (8) guarantees the minimum data rate of the CUE.

As mentioned previously, playback delay occurs when the next chunk to be played has not arrived in the queue, and therefore, the role of the constraint (7) is to avoid queue emptiness, where $Z(t) = \tilde{Q} - Q(t)$. Here, $Z(t)$ is introduced so that $Q(t)$ is sufficiently large to avert playback delay, and $\tilde{Q}$ is a sufficiently large parameter that affects the maximal queue backlog. From (2), the queue dynamics of $Z(t)$ can be represented as

$$Z(t+1) = \min\{Z(t) + c - \mu(t), \tilde{Q}\} \text{ and } Z(0) = \tilde{Q}. \tag{11}$$

Although the update rules of $Q(t)$ and $Z(t)$ are different, both queue dynamics refer to the same video chunk processing. Therefore, playback delay due to emptiness of $Q(t)$ can be explained by the queueing delay of $Z(t)$. By Littles' theorem [38], the expected value of $Z(t)$ is proportional to the time-averaged queueing delay. Our aim is to limit the queuing delay by addressing (7), and it is well known that Lyapunov control-based time-average optimization with (7) can make $Z(t)$ bounded [39].

From the optimization problem in (6)–(10), we can intuitively see the manner in which decisions are made according to $Z(t)$. Suppose that the queue is almost empty; $Q(t) \approx 0$ and $Z(t) \approx \tilde{Q}$. In this case, the user prefers the cache-enabled vehicle that has a strong channel condition and stores a low-quality file, and the associated cache-enabled vehicle prefers to allocate more transmit power to reduce $Z(t)$. However, the large transmit power of the D2D transmitter will significantly interfere with to the CUE. In addition, the geological location of the D2D transmitter $\alpha(t)$ influences the data rate of the CUE. However, when the chunks already accumulated in the queue are sufficient to avoid the playback delay, i.e., $Q(t) \approx \tilde{Q}$ and $Z(t) \approx 0$, the streaming user will want to associate with the vehicle caching a high-quality video. In addition, it is preferable that the transmit power of the D2D transmitter be small so that the CUE is provided with a large data rate.

IV. DECISION ON CACHE-ENABLED VEHICLE FOR VIDEO DELIVERY

For avoiding the emptiness of the queue $Q(t)$, i.e., for pursuing the stability of $Z(t)$, the optimization problem of (6)–(10) is solved based on the Lyapunov optimization theory. We first transform the inequality constraint in (8) into the form of queue stability. Specifically, we define the virtual queue $W(t)$ with the update equation:

$$W(t+1) = \max\{W(t) + \eta_c - R_c(P_c, P_d, t), 0\}. \tag{12}$$

The strong stability of the virtual queue $W(t)$ pushes the average of $R_c(P_c, P_d, t)$ such that the constraint in (8) is satisfied.

In addition, because the timescale of the decision on $\boldsymbol{\alpha}$ is larger than that of the decisions on $\boldsymbol{P}_v$ and $\boldsymbol{P}_c$, the frame-based Lyapunov optimization theory [36] is used for the decision on the cache-enabled vehicle to be used for video delivery. Let $\Theta(t) = [Z(t), W(t)]$ be the concatenated vector of actual and virtual queue backlogs. Then, the quadratic Lyapunov function $L[\Theta(t)]$ is defined as

$$L(\Theta(t)) = \frac{1}{2}\Big\{Z(t)^2 + \gamma W(t)^2\Big\}, \tag{13}$$

where γ is a coefficient for adjusting the scales of $Z(t)$ and $W(t)$. Then, let $\Delta(.)$ be a frame-based conditional Lyapunov function that can be formulated as $\mathbb{E}[L(t_k + T) - L(t_k)|\Theta(t_k)]$, i.e., the drift over the time interval T. The dynamic policy is designed to solve the given optimization problem of (6)–(10) by observing the current queue states of $\Theta(t_k)$ and determining the cache-enabled vehicle such that the upper bound on the frame-based *drift-plus-penalty* is minimized [36]:

$$\Delta(t_k) - V\mathbb{E}[\mathcal{P}(q(\alpha_k))|\Theta(t_k)], \tag{14}$$

where V is the importance weight for quality improvement.

The upper bound on the Lyapunov drift can be found in the Lyapunov function:

$$\begin{aligned}&L[\boldsymbol{\Theta}(t+1)]-L[\boldsymbol{\Theta}(t)]\\&=\frac{1}{2}\Big\{Z(t+1)^2-Z(t)^2+\gamma(W(t+1)^2-W(t)^2)\Big\}\\&\le\frac{1}{2}\Big\{c^2+\mu(t)^2+\gamma(\eta_c-R_c(P_c,P_d,t))^2]\Big\}\\&\quad+\{Z(t)(c-\mu(t))+\gamma W(t)(\eta_c-R_c(P_c,P_d,t))\}. \qquad (15)\end{aligned}$$

By summing (15) over $t\in\mathcal{T}_k$ and taking the expectation with respect to random channel generalizations, the upper bound on the frame-based conditional Lyapunov drift is obtained by

$$\begin{aligned}\Delta(\boldsymbol{\Theta}(t_k))\le BT+\mathbb{E}\Bigg[\sum_{t=t_k}^{t_k+T-1}\Big\{&Z(t)(c-\mu(t))\\&+\gamma W(t)(\eta_c-R_c(P_c,P_d,t))\Big\}\Bigg], \qquad (16)\end{aligned}$$

where we assume that the departure and arrival rates of all queues are bounded, and B is a constant such that

$$\frac{1}{2}\mathbb{E}\Big[c^2+a(t)^2+\gamma(\eta_c-R_c(P_c,P_d,t))^2\Big]\le B. \qquad (17)$$

According to (14), minimizing the bound on frame-based drift-plus-penalty is consistent with minimizing

$$\begin{aligned}&\mathcal{D}(\alpha_k,\Theta(t_k),\mathbf{P}_{d,k},\mathbf{P}_{c,k})\\&=\mathbb{E}\Bigg[\sum_{t=t_k}^{t_k+T-1}\Big\{Z(t)(c-\Big\lfloor\frac{t_cR_d(\alpha_k,P_d,P_c,t)}{S(q(\alpha_k))}\Big\rfloor)\\&\quad+\gamma W(t)(\eta_c-R_c(P_c,P_d,t))\\&\quad-VT\cdot\mathcal{P}(q(\alpha_k))\Big\}\Big|\Theta(t_k)\Bigg], \qquad (18)\end{aligned}$$

because BT is a constant. Let $\mathbf{P}_{d,k}=[P_d(t_k),P_d(t_k+1),\cdots,P_d(t_k+T-1)]$ and $\mathbf{P}_{c,k}=[P_c(t_k),P_c(t_k+1),\cdots,P_c(t_k+T-1)]$. Note that P_c and P_d change over time slots $t\in\mathcal{T}_k$. This frame-based drift-plus-penalty algorithm was shown in [36] to satisfy the queue stability constraints of (7)–(8) while maximizing the objective function of (6). For a given spectrum reuse policy, the minimum bound on frame-based drift-plus-penalty can be obtained by

$$\mathcal{D}(\alpha_k,\Theta(t_k))=\max_{\mathbf{P}_{d,k},\mathbf{P}_{c,k}}\mathcal{D}(\alpha_k,\Theta(t_k),\mathbf{P}_{d,k},\mathbf{P}_{c,k}). \qquad (19)$$

In Section V, we describe the determination of $\mathbf{P}_{d,k}$ and $\mathbf{P}_{c,k}$ by using the deep reinforcement learning approach, which is aimed to minimize (18) for a given pair of a streaming user and a CUE sharing the spectrum.

System parameter V in (18) is the weight factor for the term representing the video quality. The value of V is important for controlling the queue backlogs and the quality level of the desired video, i.e., for choosing α_k at every frame. The appropriate initial value of V needs to be obtained empirically, because it depends on the distributions of cache-enabled vehicles and channel environments. In addition, $V\ge 0$ should be satisfied. If $V<0$, the optimization goal is converted into minimizing the video quality. Moreover, in the case of $V=0$, vehicle users aim only at stacking the queue backlogs of $Z(t)$ and do not pursue a high-quality video. In contrast, when $V\to\infty$, vehicle users do not consider the queue state and thus simply associate with the cache-enabled vehicle that stores the highest-quality video. V can be regarded as the parameter to control partially the trade-off between the video quality, the data rate of the CUE, and the queueing delay of the streaming user, which reflects the fact that the selection of the cache-enabled vehicle for video delivery explicitly adjusts the mutual interference effects between the CUE and DUEs.

With the initial condition of $\Theta(t_k)$, the system can compute $\mathcal{D}(\alpha_k,\Theta(t_k))$ for all possible $\alpha_k\in\mathcal{C}_k$. Then, the optimal association policy of α_k^* that minimizes $\mathcal{D}(\alpha_k,\Theta(t_k))$ can be obtained by

$$\alpha^*(t_k)=\arg\max_{\alpha\in\mathcal{C}_k}\mathcal{D}(\alpha(t_k),\Theta(t_k)). \qquad (20)$$

In practice, since the number of cache-enabled vehicles in the vicinity of the streaming user is finite, the user can easily find a suitable vehicle for video delivery, i.e., make the decision on α_k, via a greedy search.

V. DYNAMIC POWER ALLOCATIONS BY DEEP REINFORCEMENT LEARNING

A. Modeling of Markov Decision Process

According to (15), we can formulate the drift-plus-penalty algorithm of the k-th frame as

$$\begin{aligned}\{\mathbf{P}^*_{d,k},\mathbf{P}^*_{c,k}\}&=\arg\min\mathcal{D}(\alpha_k,\Theta(t_k),\mathbf{P}_{d,k},\mathbf{P}_{c,k}) &(21)\\ \text{s.t. }&0\le P_d\le P_0^d &(22)\\&0\le P_c\le P_0^c. &(23)\end{aligned}$$

The problem of (21)–(23) can be modeled by a Markov decision process (MDP). In the network model, α_k is given, and $\Theta(t)$ for $t\in\mathcal{T}_k$ can be observed before making decisions on $P_c(t)$ and $P_v(t)$ at every time $t\in\mathcal{T}_k$.

The MDP is defined as $M=\{\mathcal{S},\mathcal{A},T,r\}$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ denotes the action space, T denotes the transition model and r denotes the reward structure. The queue backlog set of $\Theta(t)$ represents the current state that satisfies the Markov property. The state space is $\mathcal{S}=\mathcal{Z}\times\mathbb{R}^+$, where $Z(t)\in\mathcal{Z}=\{0,1,\cdots,\tilde{Q}\}$ and $W(t)\in\mathbb{R}^+$. $\mathbb{R}^+$ represents a set of nonnegative real numbers. The action set consists of power allocations for the D2D transmitter and the CUE, i.e., $P_d(t)$ and $P_c(t)$ for $t\in\mathcal{T}_k$. Denote the action at slot t by $\Xi(t)=[P_c(t),P_d(t)]$. By letting the action space be $\mathcal{A}\in[0,P_0^d]\times[0,P_0^c]$, the constraints of (22) and (23) are satisfied. Let power allocations for both the CUE and the DUE be uniformly discretized into N_A+1 levels, and the finite action space is represented by $\mathcal{A}=\{0,P_0^d/N_A,\cdots,(N_A-1)P_0^d/N_A\}\times\{0,P_0^c/N_A,\cdots,(N_A-1)P_0^c/N_A\}$. The action decisions are made over the k-th frame, i.e., $t\in\mathcal{T}_k$, and according to (18), the reward (i.e., incurred cost with the negative sign) at each slot $t \in \mathcal{T}_k$ is represented by

$$r(\Theta(t), \Xi(t)) = Z(t)\Big(c - \Big\lfloor \frac{t_c R_d(\alpha_k, P_d, P_c, t)}{S(q(\alpha_k))} \Big\rfloor\Big) + \gamma W(t)(\eta_c - R_c(P_c, P_d, t)) - VT \cdot \mathcal{P}(q(\alpha_k)); \quad (24)$$

therefore, the reward r is the cost (24) multiplied by the negative sign. At every slot t, channel gains are randomly generated and state transitions occur according to random network events and the current queue state of $\Theta(t)$. The transition from $\Theta(t)$ to $\Theta(t+1)$ is defined as

$$P_{s's}(\xi) = \Pr\big\{\Theta(t+1) = s' | \Theta(t) = s, \Xi(t) = \xi\big\}, \quad (25)$$

for all states $s, s' \in \mathcal{S}$ and $\xi \in \mathcal{A}$.

According to Bellman optimality equation, the minimum incurred cost at $\Theta(t_0) = s_0$ is given by

$$\begin{aligned} &\min_{\Xi} \mathbb{E}\Bigg[\sum_{t=t_0}^{T} r\big(\Theta(t), \Xi(t)\big)\Big|\Theta(t_0) = s_0\Bigg] \\ &= \min_{\Xi} \mathbb{E}\Big[r(s_0, \xi) + G\big(\Theta(t_0+1)\big|\Theta(t_0) = s_0, \Xi(t_0) = \xi\big)\Big] \\ &= \min_{\Xi} \mathbb{E}\Big[r(s_0, \xi) + \sum_{s \in \mathcal{S}} P_{s,s_0}(\xi) G(s) \\ &\qquad \Big|\Theta(t_0) = s_0, \Xi(t_0) = \xi\Big]. \end{aligned} \quad (26)$$

Note that the channel information is not known, and the state transition probabilities are not given; therefore, we solve the problem (21)–(23) by using a DRL algorithm. Based on the finite MDP, the goal of reinforcement learning is to train a policy $\pi \in \Pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ which gives all action candidates at every state the probability values in $[0, 1]$. The policy π maps the state of the environment to the action to maximize the expected reward. Denote the expected reward under the policy π by $\mathcal{J}(\pi)$. With finite T steps, $\mathcal{J}(\pi)$ can be described as the accumulation of the reward at each time step.

$$\mathcal{J}(\pi) = \mathbb{E}\left[\sum_{t=0}^{T} \delta^t r\big(\Theta(t), \Xi(t)\big)\Big|\pi\right], \quad (27)$$

where δ is a discount factor that adjusts the effect of future rewards to the current decision. The optimal policy π^* is

$$\pi^* = \arg\max_{\pi} \mathcal{J}(\pi). \quad (28)$$

In deep reinforcement learning, the policy π is approximated by parameter θ. The state sequence of s that is generated according to the policy π_θ is the distribution. Then, the expected reward obtained by the state sequence s and the policy π_θ can be denoted as $\mathcal{J}(\mathrm{s}, \pi_\theta(\mathrm{s}))$, and the objective reinforcement learning is formulated as:

$$\arg\max_{\theta} \mathbb{E}_{s \sim \pi_\theta}\left[\mathcal{J}(\mathrm{s}, \pi_\theta(\mathrm{s}))\right]. \quad (29)$$

The following subsections describe the deep reinforcement learning algorithms that can solve (29) by finding the optimal policy for every state at each time step t. In the following subsections, s_t, a_t, and r_t are used for state, action, and reward at time step t for simplicity, rather than $\Theta(t)$, $\Xi(t)$, and $r(\Theta(t), \Xi(t))$.

B. Deep Q-network (DQN)

The deep *Q*-network (DQN) is one of the breakthrough deep reinforcement algorithms for applying a neural network to reinforcement learning. A neural network is used to approximate state-action functions (*Q*-functions). The *Q*-functions approximated by the neural network allow the DQN to go straight from the high dimensional state space. The concept of the DQN is based on the classic *Q*-learning algorithm. In classical *Q*-learning, the *Q*-value of a state-action pair is estimated through iterative updates based on multiple interactions with the environment. Therefore, in a DQN, with every action taken in a state, the immediate reward received and the expected *Q*-value of the new state are used to update the *Q*-functions. Accordingly, the objective of a DQN is described as

$$\arg\min_{\theta} \ell_{DQN}(\theta) = \arg\min_{\theta}(\mathcal{Q}(s_t, a_t; \theta) - \bar{\mathcal{Q}}(s_t, a_t; \theta))^2, \quad (30)$$

where $s_t \in \mathcal{S}$ state at time t, $a_t \in \mathcal{A}$ selected action at s_t, and θ is the parameter set of *Q*-functions. $\mathcal{Q}(s_t, a_t; \theta)$ is the target *Q*-value that is derived from the current *Q*-functions at time *t*. Therefore, $\mathcal{Q}(s_t, a_t; \theta) = r_t + \delta \max_{\acute{a}}(s_{t+1}, \acute{a}; \theta)$. Although DQNs show a successful and high performance in many domains, however, because a DQN approximates *Q*-functions with neural networks, it can be used only for a discrete action space.

C. Deterministic Policy Gradient (DPG)

Policy gradient methods attempt to learn a policy function directly, in contrast to a DQN, which attempts to learn action-value *Q* functions. The basic idea is to increase the probabilities of actions that lead to high rewards and reduce the probabilities of actions that lead to low rewards until the optimal policy is trained. In deterministic policy gradient (DPG) methods, a neural network is used to approximate the policies. The policy is trained by updating the parameters of the policies via stochastic gradient optimization. The objective of the DPG method is described as

$$\arg\max_{\theta} \ell_{DPG}(\theta) = \arg\max_{\theta} \mathbb{E}\left[\log \pi_\theta(a_t | s_t; \theta)\hat{r}_t\right], \quad (31)$$

where $\hat{r}_t$ is the reward that is returned from the environment. The DPG method is an on-policy algorithm and can be used for environments having either discrete or continuous action spaces.

D. Deep Deterministic Policy Gradient (DDPG)

Although DQNs can solve problems with high-dimensional state spaces, they can handle only discrete and low-dimensional action spaces. However, the action spaces of the environments of many applications (i.e., proactive caching, resource management, etc.) are continuous and high dimensional. As mentioned previously, DQN algorithms cannot be straightforwardly applied to continuous actions, because a DQN depends on the best action that maximizes the *Q*-value function being selected. When there exists a finite number of discrete actions, the action that causes the *Q* value to be maximized can be selected, because the possible *Q* values at the state can be computed directly for each action. However, when the action space is continuous, it is difficult to evaluate Q values exhaustively.

The deep deterministic policy gradient (DDPG) algorithm concurrently learns the Q-value function and the policy. The action-value Q function is learned and it is also used to learn the policy. In DDPG, the function $\mathcal{Q}^*(s,a)$ is approximated by the neural network, as in a DQN. Therefore, because the action space is continuous, the function $\mathcal{Q}^*(s,a)$ can be differentiable in terms of the action. Thus, the policy π_θ can be updated efficiently. $\mathcal{Q}_\phi(s,a)$, which is approximated using the parameter set of ϕ, is updated by minimizing the mean-squared Bellman error (MSBE).

$$\ell(\phi,\mathcal{D}) = \mathbb{E}\left[\left(\mathcal{Q}(s_t,a_t;\phi) - \bar{\mathcal{Q}}(s_t,a_t;\phi)\right)^2\right], \tag{32}$$

where $\mathcal{D}$ is a set of transitions (s,a,r,s',d). DDPG is aimed to learn a deterministic policy $\pi_\theta(s)$, which gives the action that maximizes $\mathcal{Q}_\phi(s,a)$.

$$\max_\theta \mathbb{E}_{s\sim\mathcal{D}}\left[\mathcal{Q}_\phi(s,\pi_\theta(s))\right]. \tag{33}$$

In summary, the DPG is used for finding the optimal deterministic policy by using the gradient method, and the DQN provides the optimal stochastic policy by using the deep Q-network. Lastly, the DDPG employs the deep learning framework for deriving the DPG, especially when the state and action spaces are continuous and/or the state transition model is not tractable.

VI. SIMULATION RESULTS

This section presents the simulation results to verify the advantages of the proposed dynamic node association and power allocation policy for content delivery in cache-enabled D2D underlaid cellular systems. We leveraged TensorFlow in our simulations to implement our proposed DDPG-based scheme. As shown in (6)–(10), the main performance metrics of streaming users (i.e., DUEs) are the average content quality and the delay incidence, and that of CUEs is the average data rate. Therefore, we first show the convergence of the proposed DDPG algorithm in the D2D underlaid cache-enabled vehicular network, and compare the performance metrics of the proposed scheme with other techniques.

A. Simulation environments

In the simulation, the scenario of the D2D underlaid cache-enabled vehicular network shown in Fig. 1 was considered, in which vehicles are moving in traffic lanes and the BS is located between the incoming and outgoing lanes. Some vehicles are communicating with the BS via cellular links, and streaming users (vehicles) receive the desired video chunks from nearby cache-enabled vehicles via D2D links. We assume that all vehicles can activate only one type of link, i.e., a vehicle communicating via a cellular link cannot activate a D2D link and vice versa. Since streaming users having the desired videos in their own storage do not activate D2D link, we consider only the streaming users requesting contents from another cache-enabled devices. For the D2D underlaid cellular model, the spectrum of one CUE can be reused by one D2D link. Since this paper focuses on node association and power controls for content delivery in the cache-enabled D2D underlaid cellular system, spectrum allocation is out of scope; therefore, suppose that the spectrum reuse policy is given, i.e., the pairs of the cellular and D2D links sharing the spectrum have already been determined. For simplicity, we also suppose that the maximum transmit power and noise variance of both cellular and D2D links are the same, i.e., $P_0^c = P_0^d = 23$ dB and $\sigma^2 = -114$ dBm. In addition, $\eta_c = 30$ Mbps and $\mathcal{B} = 2$ MHz are used. The shadowing standard deviations for cellular and D2D links are 8 dB and 3 dB, respectively.

As shown in Fig. 1 and Table 1, the rectangular road, the size of which is $r_x \times r_y$, consists of four lanes in each direction. $r_x = 1$ km and $r_y = (N_l+1)w_l$ where $w_l = 4$ m, because there are eight lanes and one lane between the two four-lane parts of the road for the BS. The BS is located at (0,0). For simplicity, the location of the CUE is fixed at (200, 8); however, the location of the DUE (i.e., streaming user) is randomly chosen. Assume that a constant speed $v = 60$ km/h for every vehicle and cache-enabled vehicles are distributed with $\lambda = 0.01$. We consider three quality levels whose PSNR measures and file sizes are $\mathcal{P}(q) = [34, 36.64, 39.11]$ dB and $S(q) = [2621, 5073, 10658]$ kbits, respectively. In addition, $T = 10$, $t_c = 1$ ms, $V = 0.3$, $c = 15$, $\bar{Q} = 500$ and $\gamma = 10^{-6}$ are used in this section.

For simulation, we employed a NVIDIA DGX station equipped with 4× Tesla V100 GPUs (a total of 128 GB memory available) and an Intel Xeon E5–2698 v3 2.2 GHz CPU with 20 cores (256 GB system memory available in total). In addition, Pythos version 3.6 on Ubuntu 16.04 LTS is utilized to build the DDPG-based node association and power control scheme. Also, we used a Xavier initializer [40] to avoid the occurrence of vanishing gradient descent during the learning phase. The neural network was constructed with a fully connected deep neural network (DNN), and the number of nodes in the hidden layer was 200. The discount factor for reward is $\delta = 10^{-6}$. The RL model was trained through a total of 100,000 iterations. Here, note that the DNN is used to approximate the Q-function $Q(s,a)$ of this system because the state space is continuous and the random event distribution (i.e., channel statistics in this setup) is not known. Therefore, the inputs of the DNN are the current state and action of the agent and the output becomes its Q-function.

To verify the advantages of the proposed dynamic video delivery policy, we compared the proposed scheme with four other schemes:

- "Genie-aided": Through knowledge of the fast fading gains of all links, the optimal transmit powers of both the CUE and the DUE are optimally obtained. The decision on the cache-enabled vehicle to be used for video delivery is not considered.
- The scheme presented in [27]: The power allocations for the CUE and the DUE are achieved based on ergodic capacity and are aimed to satisfy the constraint of the probability that delay occurs. This approach is not dynamic, and therefore, a fixed power allocation is applied for each frame. Because no association algorithm for video delivery is included in the method in [27], the decision on the cache-enabled vehicle for video delivery is not considered.

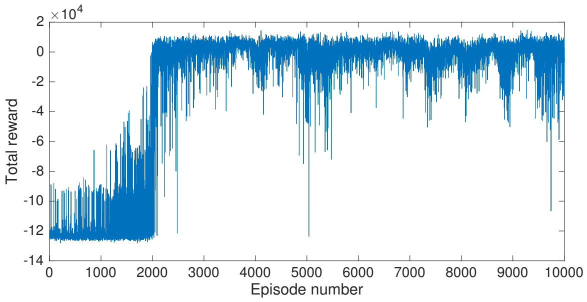


Fig. 3. Learning traces with learning rate of 3×10^{-6}.

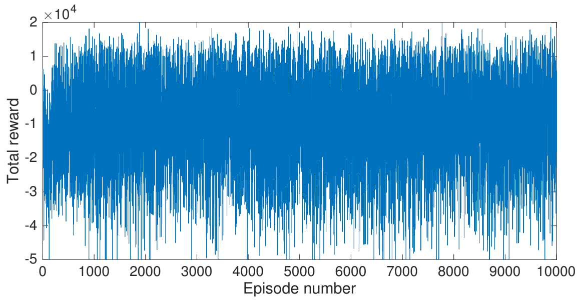


Fig. 4. Learning traces with learning rate of 4×10^{-6}.

- "Nearest": The streaming user associates with the nearest cache-enabled vehicle that is likely to provide the strongest channel condition. The power allocations based on deep reinforcement learning are the same as those of the proposed scheme.
- "Highest-Qual": The streaming user associates with the cache-enabled vehicle that caches the highest quality version of the requested file. If there are many vehicles caching the highest-quality version of video, the user chooses the nearest. The power allocations based on deep reinforcement learning are the same as those of the proposed scheme.

In summary, a comparison of our scheme with the "Genie-aided" scheme and the scheme presented in [27] can provide specific insight into the effectiveness of deep reinforcement learning-based power allocations. The performance comparison of the proposed cache-enabled vehicle association for video delivery based on frame-based Lyapunov optimization with the "Nearest" and "Highest-Qual" schemes shows its advantages.

B. Learning Traces

Fig. 3 shows the traces of learning the power allocations of $\mathbf{P}_d$ and $\mathbf{P}_c$ at a learning rate of 3×10^{-6} to minimize the upper bound on the drift-plus-penalty in (18). The learning trace in Fig. 3 is obtained with an example scenario of a certain randomly generated vehicular network; however, the learning traces of most of the random generations are similar to that in this figure. We can easily see that a dramatic increase in the reward is obtained after around 2000 episodes; 4000 episodes were used in the simulation to provide a sufficient margin to converge the learning process. However, although the reward seems to converge to a certain degree, the trace is in general unsteady. This means that the learning process is not quite stable, because the fast fading gains are completely unknown. Therefore, we restrict the action space of the power allocations to the finite set instead of the nonnegative real number set. In practice, the system is usually unable to change the transmit power continuously, and the finite N_A levels of transmit power are used. Thus, each transmit power becomes $P_0^d \in \{0, P_0^d/N_A, \cdots, (N_A - 1)P_0^d/N_A]$ and $P_0^c \in \{0, P_0^c/N_A, \cdots, (N_A - 1)P_0^c/N_A]$. In addition, this deep reinforcement learning is very sensitive to changes in the learning rate. We can see in Fig. 4 that the total rewards do not converge even as the episode proceeds when the learning rate is 4×10^{-6}.

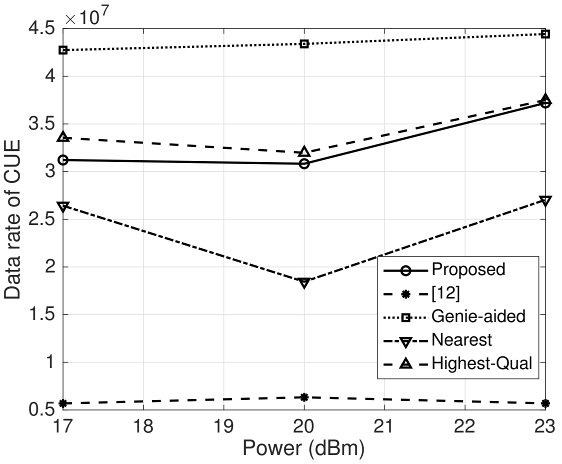


Fig. 5. Data rate of CUE R_c vs. P_0.

C. Performances of Cellular and D2D Links

We used three performance metrics: 1) The data rate of the CUE, 2) the playback delay at the DUE, and 3) the time-average video quality. Figs. 5–7 show the plots of these performance metrics versus the transmit power budgets of both cellular and D2D users, i.e., P_0^d and P_0^c. Because we assume $P_0^d = P_0^c = P_0$, the power of the target signal of the CUE, as well as the interfering signal from the D2D link, increase as P_0 grows. Therefore, the data rate of the CUE does not dramatically change with P_0, as shown in Fig. 5. We can see that the "Genie-aided" scheme definitely shows the largest data rates among the compared techniques. However, the data rate of the scheme presented in [27] is the smallest, because it endeavors to reduce the queuing delay of the DUE rather than to raise the data rate.

The effect of the proposed step for selecting the cache-enabled vehicle for video delivery based on the frame-based Lyapunov optimization can be seen by comparing its performance with that of the "Nearest" and "Highest-Qual" schemes. Because in the "Nearest" scheme the streaming user must associate with the nearest cache-enabled vehicle, the distance between the D2D transmitter and the BS is usually shorter than that produced by the proposed algorithm and "Highest-Qual." Therefore, the data rate of "Nearest" is smaller than that of the proposed scheme and "Highest-Qual," because the interference power that it introduces to the BS is large.

Similarly, since the distribution density of cache-enabled vehicles that store the highest-quality content is considerably less

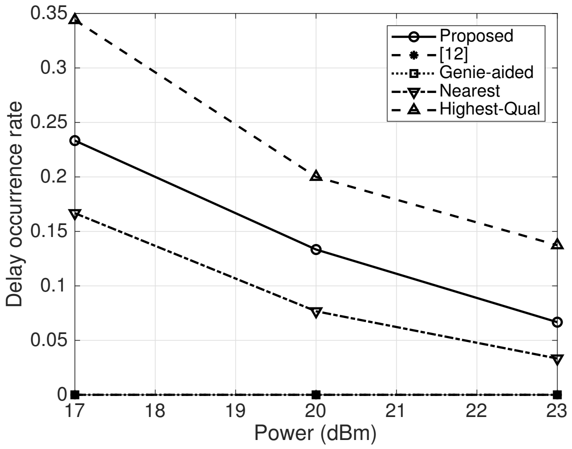


Fig. 6. Delay occurrence rates vs. P_0.

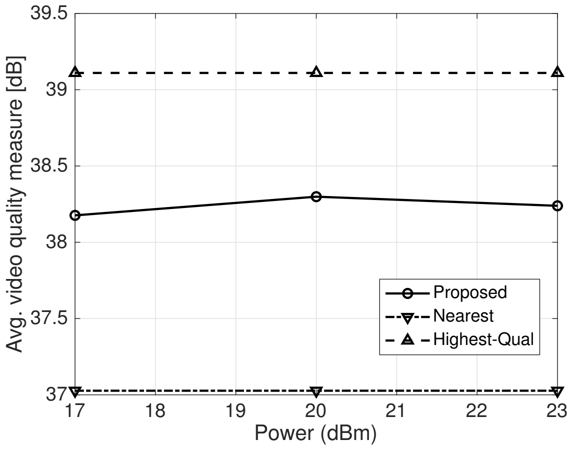


Fig. 7. Average video quality vs. P_0.

than that of vehicles that store the desired content at lower qualities, “Highest-Qual" can provide a larger data rate for the CUE than the proposed scheme. However, by considering the minimum data rate of the CUE (i.e., η_c), the proposed algorithm adequately satisfies this constraint, whereas “Nearest" does not. In addition, as already mentioned concerning the unstable learning process in Fig. 3, the data rate performances of all methods that utilize deep reinforcement learning-based power allocation are not monotonic; however, the fluctuation of their trends is not very severe.

Fig. 6 shows the plots of playback delay incidences versus P_0. As explained previously, when there is no chunk in the queue when the streaming service is being employed, playback delay occurs. Therefore, the playback delay incidence reflects the extent of queue emptiness that occurs over the total playback time. In Fig. 6, it can be seen that the “Genie-aided" scheme and the scheme presented in [27] result in no delay. However, the other power allocation algorithms do not result in delay. A comparison of Figs. 5 and 6, reveals that the proposed algorithm endeavors to satisfy the constraint of the minimum data rate of the CUE at the expense of its delay performance.

Among the comparison schemes that utilize power allocation based on the deep reinforcement learning approach, “Nearest" shows the lowest playback delay occurrence rates and the proposed scheme shows the second lowest, whereas the “Highest-Qual" scheme is expected to yield rather long buffering times. As P_0 increases, more chunks can be delivered to the streaming user with the increased data rate of the D2D link; thus, the delay occurrence rates of all the schemes decrease. “Nearest" in particular provides the strongest channel condition to the D2D link, and therefore, its delay performance is better than that of the proposed and the “Highest-Qual" scheme.

In Fig. 7, we can see the time-average video qualities of all the comparison schemes. Since “Genie-aided" scheme and the scheme presented in [27] do not consider the decision on the cache-enabled vehicle for video delivery, they are not shown in this figure. Meanwhile, “Nearest" and “Highest-Qual" show the trends of quality performance that differ from those of the proposed scheme. Obviously, the “Highest-Qual" scheme gives the best video quality. Because the “Nearest" scheme does not pursue video quality enhancement, it is obvious that its performance is the lowest among the techniques compared in Fig. 7. The performance of the proposed algorithm is better than that of “Nearest" and poorer than that of “Highest-Qual".

Overall, we can state that, although the proposed algorithm results in a few playback delays, it can achieve both the minimum data rate of the CUE and quite good time-average video quality. The scheme in [27] shows almost no playback delay, but it cannot satisfy the minimum data rate constraint. Moreover, in contrast to the method in [27], which does not consider wireless caching and video delivery, the proposed algorithm can nicely achieve average video quality. Similarly, “Nearest” provides a better delay performance than the proposed scheme, but it also cannot satisfy the minimum data rate constraint and achieve the video quality as effectively as the proposed algorithm. Finally, “Highest-Qual" can deliver the highest-quality streaming service, but its delay performance is poorer than that of the proposed scheme. Thus, we can conclude that the proposed algorithm smooths the trade-off between the data rate of the CUE, the playback delay incidence, and the time-average video quality.

VII. CONCLUSION AND FUTURE WORK

In this paper, a method for the joint optimization of three decisions having different timescales in D2D underlaid cellular and vehicular caching networks was proposed: 1) Association with a cache-enabled vehicle to allow video delivery, 2) power allocation for the DUE, and 3) power allocation for the CUE. The proposed algorithm maximizes the long-term time averaged video quality while limiting the playback delay and guaranteeing the data rate of the cellular user, given the spectrum reuse policy. The decision on the cache-enabled vehicle to be used for video delivery is achieved by using the frame-based Lyapunov optimization theory under consideration of the interference signal from the CUE. The dynamic power allocations of both the CUEs and DUEs are obtained by using the deep reinforcement learning approach in the absence of knowledge of channel fast fading. Our intensive simulation results verify that the proposed algorithm effectively achieves a balanced trade-off between the data rate of the cellular user, the playback delay occurrence of video streaming, and the average video quality. As future work, extension to dynamic content delivery and routing optimization in multi-hop wireless networks, e.g., vehicular ad hoc networks (VANETs), is considerable.

REFERENCES

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 20162021 White Paper", Cisco. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/serviceprovider/visual-networking-index-vni/mobile-white-paper-c11–520862.html

[2] E. Bastug, M. Bennis and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, Aug. 2014.

[3] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[4] M. Ji, G. Caire and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176–189, Jan. 2016.

[5] S. H. Chae and W. Choi, "Caching placement in stochastic wireless caching helper networks: Channel selection diversity via caching," *IEEE Trans. Wireless Commun.*, vol. 15, no. 10, pp. 6626–6637, Oct. 2016.

[6] C. Yang, Y. Yao, Z. Chen and B. Xia, "Analysis on cache-enabled wireless heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 131–145, Jan. 2016.

[7] K. Poularakis, G. Iosifidis and L. Tassiulas, "Approximation algorithms for mobile data caching in small cell networks," *IEEE Trans. Commun.*, vol. 62, no. 10, pp. 3665–3677, Oct. 2014.

[8] L. Zhang, M. Xiao, G. Wu and S. Li, "Efficient scheduling and power allocation for D2D-assisted wireless caching networks," *IEEE Trans. Commun.*, vol. 64, no. 6, pp. 2438–2452, June 2016.

[9] W. Jiang, G. Feng and S. Qin, "Optimal cooperative content caching and delivery policy for heterogeneous cellular networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1382–1393, May 2017.

[10] K. Poularakis, G. Iosifidis, A. Argyriou and L. Tassiulas, "Video delivery over heterogeneous cellular networks: Optimizing cost and performance," in *Proc. IEEE INFOCOM*, 2014, pp. 1078-1086.

[11] M. Choi, J. Kim and J. Moon, "Wireless video caching and dynamic streaming under differentiated quality requirements," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1245–1257, June 2018.

[12] 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.

[13] M. Choi, A. No, M. Ji and J. Kim, "Markov decision policies for dynamic video delivery in wireless caching networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 12, pp. 5705–5718, Dec. 2019.

[14] M. Choi, A. F. Molisch and J. Kim, "Joint distributed link scheduling and power allocation for content delivery in wireless caching networks," *IEEE Trans. Wireless Commun.*, Early Access, Aug. 2020.

[15] D. Bethanabhotla, G. Caire, and M. J. Neely, "Adaptive video streaming for wireless networks With multiple users and helpers," *IEEE Trans. Commun.*, vol. 63, no. 1, pp. 268–285, Jan. 2015.

[16] T. Stockhammer, "Dynamic adaptive streaming over HTTP - standards and design principles, *Proc. ACM MMSys2011*, Feb. 2011.

[17] S. O. Somuyiwa, A. György and D. Gündüz, "A reinforcement-learning approach to proactive caching in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 6, pp. 1331–1344, June 2018.

[18] W. Jiang, G. Feng, S. Qin, T. S. P. Yum and G. Cao, "Multi-agent reinforcement learning for efficient content caching in mobile D2D networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1610–1622, Mar. 2019.

[19] V. Kirilin, A. Sundarrajan, S. Gorinsky and R. K. Sitaraman, "RL-cache: Learning-based cache admission for content delivery," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 10, pp. 2372–2385, Oct. 2020.

[20] S. Deng et al., "Dynamical resource allocation in edge for trustable Internet-of-things systems: A reinforcement learning method," *IEEE Trans. Industrial Informatics*, vol. 16, no. 9, pp. 6103–6113, Sept. 2020.

[21] L. Li et al., "Deep reinforcement learning approaches for content caching in cache-enabled D2D networks," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 544–557, Jan. 2020.

[22] G. Qiao, S. Leng, S. Maharjan, Y. Zhang and N. Ansari, "Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 247–257, Jan. 2020.

[23] Z. Nan, Y. Jia, Z. Chen and L. Liang, "Reinforcement-learning-based optimization for content delivery policy in cache-enabled HetNets," in *Proc. IEEE GLOBECOM*, 2019.

[24] N. Lee, X. Lin, J. G. Andrews and R. W. Heath, "Power control for D2D underlaid cellular networks: Modeling, algorithms, and analysis," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 1, pp. 1–13, Jan. 2015.

[25] X. Li, R. Shankaran, M. A. Orgun, G. Fang and Y. Xu, "Resource allocation for underlay D2D communication With proportional fairness," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6244–6258, July 2018.

[26] Y. Ren, F. Liu, Z. Liu, C. Wang, and Y. Ji, "Power control in D2D-based vehicular communication networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, pp. 5547–5562, Dec. 2015.

[27] L. Liang, G. Y. Li and W. Xu, "Resource allocation for D2D-enabled vehicular communications," *IEEE Trans. Commun.*, vol. 65, no. 7, pp. 3186–3197, July 2017.

[28] P. Sun, K. G. Shin, H. Zhang and L. He, "Transmit power control for D2D-underlaid cellular networks based on statistical features," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 4110–4119, May 2017.

[29] N. Cheng et al., "Performance analysis of vehicular device-to-device underlay communication," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5409–5421, June 2017.

[30] W. Lee, M. Kim and D. Cho, "Deep learning based transmit power control in underlaid device-to-device communication," *IEEE Systems J.*, vol. 13, no. 3, pp. 2551–2554, Sept. 2019.

[31] I. Budhiraja, N. Kumar and S. Tyagi, "Deep reinforcement learning based proportional fair scheduling control scheme for underlay D2D communication," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3143–3156, Mar. 2021.

[32] Y. Liu, Z. Tan, and X. Chen, "Modeling the channel time variation using high-order-motion model," *IEEE Commun. Lett.*, vol. 15, no. 3, pp. 275–277, Mar. 2011.

[33] C. Xu, C. Gao, Z. Zhou, Z. Chang and Y. Jia, "Social network-based content delivery in device-to-device underlay cellular networks using matching theory," *IEEE Access*, vol. 5, pp. 924–937, Nov. 2017.

[34] Y. Wang, X. Tao, X. Zhang and Y. Gu, "Cooperative caching placement in cache-enabled D2D underlaid cellular network," *IEEE Commun. Lett.*, vol. 21, no. 5, pp. 1151–1154, May 2017.

[35] L. Shi, L. Zhao, G. Zheng, Z. Han and Y. Ye, "Incentive design for cache-enabled D2D underlaid cellular networks using stackelberg game," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 765–779, Jan. 2019.

[36] M. J. Neely and S. Supittayapornpong, "Dynamic Markov decision policies for delay constrained wireless scheduling," *IEEE Trans. Automatic Control*, vol. 58, no. 8, pp. 1948–1961, Aug. 2013.

[37] B. Blaszczyszyn and A. Giovanidis, "Optimal geographic caching in cellular networks," in *Proc. IEEE ICC*, 2015.

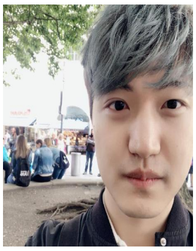
[38] Dimitri Bertsekas and Robert G. Gallager, *Data networks* (2nd edition) Prentice Hall, 1992.

[39] Michael J. Neely, "Stochastic network optimization with application to communication and queueing systems", *Morgan & Claypool Synthesis Lectures Commun. Netw.*, vol. 3, no. 1, pp. 1–211, 2010.

[40] X. Glorot and Y. Bengio, "Understanding the difficulty of tranining deep feedforward neural networks," in *Proc. AISTATS*, 2010.

[41] H. Gao, C. Liu, Y. Li and X. Yang, "V2VR: Reliable hybrid-network-oriented V2V data transmission and routing considering RSUs and connectivity probability," *IEEE Trans. Intelligent Trans. Sys.*, Early Access, Apr. 2020.

Minseok Choi received the B.S., M.S., and Ph.D. degrees from the School of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2011, 2013, and 2018, respectively. He was a Visiting Postdoctoral Researcher in electrical and computer engineering with the University of Southern California (USC), Los Angeles, CA, USA, and a Research Professor in electrical engineering with Korea University, Seoul, South Korea. He has been an Assistant Professor with Jeju National University, Jeju, South Korea, since 2020. His research interests include wireless caching networks, stochastic network optimization, non-orthogonal multiple access, and 5G networks.



Myungjae Shin received the B.S. and M.S. degrees in Computer Science and Engineering (summa cum laude) from the Chung-Ang University (CAU), Seoul, South Korea, in 2018 and 2020, respectively. From March to August 2020, he was an AI researcher at Seoul National University Hospital (SNUH), Seoul, South Korea. He is an AI Researcher at Korea University, working with Prof. Joongheon Kim. His research interests include deep reinforcement learning, machine learning, and robotics. Mr. Shin was the recipient of the National Science & Technology Scholarship (2016–2017).

Joongheon Kim (M'06–SM'18) has been with the School of Electrical Engineering, Korea University, Seoul, Korea, since 2019, where he is currently an Assistant Professor. 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, he was with LG Electronics (Seoul, Korea, 2006–2009), InterDigital (San Diego, CA, USA, 2012), Intel Corporation (Santa Clara in Silicon Valley, CA, USA, 2013–2016), and Chung-Ang University (Seoul, Korea, 2016–2019). He serves as an associate editor for IEEE TRANS. VEHICULAR TECHNOLOGY. He internationally published more than 80 journals, 110 conference papers, and 6 book chapters. He also holds more than 50 granted patents. 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 the Korea Institute of Communication and Information Sciences (KICS) (2018), IEEE Vehicular Technology 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 JOURNAL Best Paper Award (2020).