Downloaded from https://iranpaper.ir
http://www.master-thesis.ir/Employment
shiraz.barg@gmail.com

Deep Reinforcement Learning-Based Dynamic Spectrum Access for D2D Communication Underlay Cellular Networks

Jingfei Huang, Yang Yang, Gang He, Yang Xiao, and Jun Liu, *Member, IEEE*

***Abstract*— This letter investigates a deep reinforcement learning (DRL)-based spectrum access scheme for device-to-device (D2D) communication underlay cellular networks. Specifically, cellular users (CUEs) and D2D pairs attempt to access the time slots (TSs) of a shared spectrum, and TSs are dynamically scheduled to CUEs in different frames. Based on the DRL theory, D2D pairs can be seen as a centralized agent which aims to learn an optimal spectrum access strategy to maximize the sum throughput without any prior information. In particular, with different locations of CUEs, the spectrum access manners for D2D communication are changed to ensure the communication quality of CUEs at the cell edge. Then, a double deep Q-network (DDQN) based D2D spectrum access (D^4SA) algorithm is proposed, which makes D2D pairs learn to decide whether to access the spectrum in different TSs. Moreover, to ensure the fairness of resource allocation among D2D pairs, we improve the proposed algorithm and incorporate fairness into the objective function. Simulation results show that our proposed algorithm can achieve an optimal sum throughput close to the theoretical upper bound, where the performance is significantly improved compared to the scheme based on base station cooperation.**

***Index Terms*— Device-to-device communication, dynamic spectrum access, time slots, deep reinforcement learning, double deep Q-network.**

I. Introduction

DEVICE-TO-DEVICE (D2D) communication has the advantages of high transmission rate and low latency, which makes two nearby mobile terminals comprise a D2D pair and communicate directly without traversing the base station (BS) [1]. By reusing cellular frequency resources, D2D communication can significantly improve the spectrum efficiency [2]. However, cellular users (CUEs) will suffer severe interference when D2D pairs share the resources [3], which seriously deteriorates the performance of the entire system.

Many previous works emerged to explore the application of D2D communication in cellular networks [4]–[9]. Literature [4]–[7] focus on the scenario where cellular resources are shared by both cellular and D2D communications (i.e., non-orthogonal access manner). Specifically, [4] proposes a Q-learning based spectrum allocation scheme to maximize the throughput while mitigating interference between D2D and CUEs. Further, a distributed spectrum allocation framework is designed in [5] which adopts actor-critic (AC) algorithm to handle intelligent decision making problems with high-dimensional state-action spaces. In addition, both [6] and [7] consider resource allocation and power control for D2D-assisted communication networks. [6] aims to maximize system capacity and spectrum efficiency with the interference to CUEs minimized, while [7] aims to maximize the sum throughput with constraints of reliability and latency. By contrast, literature [8] and [9] assume that D2D pairs share orthogonal spectrum with CUEs (i.e., orthogonal access manner). Considering the interference among D2D users, authors in [8] jointly consider communication mode selection and resource allocation to maximize the energy efficiency and [9] proposes a deep Q-network (DQN) based transmission scheme for D2D communication to maximize the sum rate of D2D network. Nevertheless, those existing researches do not consider the combination of two aforementioned access manners and the dynamic variation of time slot (TS) assignment for CUEs.

In this letter, we propose a hybrid spectrum access scheme that considers both orthogonal and non-orthogonal access manners in D2D communication underlay cellular networks. In particular, we consider a dynamic TDD network where time slots (TSs) are dynamically assigned to CUEs. D2D pairs learn to switch between two access manners based on a joint condition of the CUE's location and the D2D communication mode to achieve the objective of maximizing the sum throughput. The challenge is that we assume D2D pairs have no idea of the access principles of CUEs. Thus, based on deep reinforcement learning (DRL) theory, we design a double deep Q-network (DDQN) based D2D spectrum access (D^4SA) algorithm, which enables D2D pairs to autonomously learn an optimal access strategy to achieve the objective of maximizing the sum throughput without knowing any prior information of the system. Furthermore, compared with most of the existing literature which mainly regards Q-function as the objective function, this letter separates Q-function and objective function upon which actions are chosen, which provides an approach to generalize the Q-learning based framework so that more general objectives can be achieved. Specifically, we incorporate fairness into the objective function to guarantee the fairness of resource allocation among D2D pairs. Simulation at last shows our proposed algorithm can both arrive at an optimal sum throughput referring to theoretical upper bound and achieve the fairness objective, which significantly outperforms the spectrum access based on BS cooperation.

II. Scenario Description

As illustrated in Fig. 1, we consider a scenario of D2D communication underlay cellular networks, where multiple D2D pairs share the uplink cellular frequency resources and the BS locates at the center of the cell. Both cellular and D2D

Manuscript received April 15, 2021; revised May 7, 2021; accepted May 9, 2021. Date of publication May 13, 2021; date of current version August 12, 2021. This work was supported by Beijing Natural Science Foundation (L182024 and 4182055) and National Natural Science Foundation of China (61801035 and 61701027). The associate editor coordinating the review of this letter and approving it for publication was L. Liang. *(Corresponding author: Yang Yang.)*

The authors are with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China, and also with the Center for Data Science, Beijing 100876, China (e-mail: buptyy1015@gmail.com).

Digital Object Identifier 10.1109/LCOMM.2021.3079920

1558-2558 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO ESPIRITO SANTO. Downloaded on February 22,2022 at 12:03:38 UTC from IEEE Xplore. Restrictions apply.

Downloaded from https://iranpaper.ir
http://www.master-thesis.ir/Employment
shiraz.barg@gmail.com

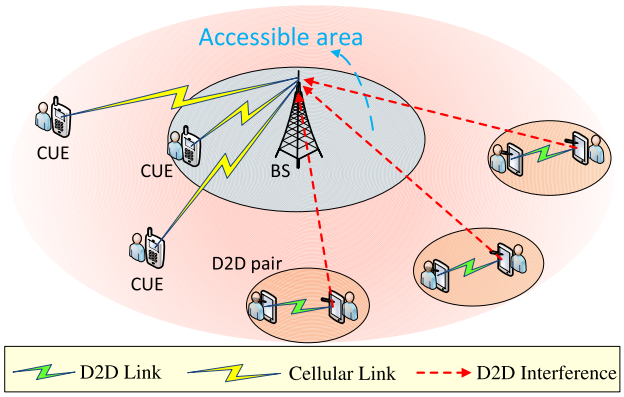

Fig. 1. System of D2D underlay cellular networks.

communications are performed in a TDD way. Specifically, each frame of cellular communication consists of Y ($Y \geq 2$) TSs, while CUEs transmit in X ($X \leq Y$) specific TSs within each frame in a repetitive manner from frame to frame and TSs allocation among CUEs is orthogonal. By observing the environment, D2D pairs learn to access the shared spectrum in proper TSs to achieve a certain objective, where cellular communication should be ensured as the top priority.

For simplicity, we claim that at most one D2D pair can be allowed to transmit signals at each TS. When the D2D pair transmits at the TS occupied by the CUE, the signal to interference plus noise ratio (SINR) of the CUE can be described as

$$\gamma = \frac{P_c G_c^B}{\sigma^2 + P_d G_d^B}, \tag{1}$$

where σ^2 is the power of the Additive White Gaussian Noise (AWGN), P_c and P_d are the transmission power of CUE and D2D transmitter (DT), respectively. G_c^B is the channel gain of cellular communication and G_d^B denotes the channel gain between DT and BS. With the top priority of ensuring cellular communication quality, the SINR of CUE has to be above a certain threshold γ_{th}, which is associated with G_c^B due to the dynamic variation of the CUE's location. Then, we can obtain $G_c^B \geq \frac{\gamma_{\text{th}}\left(\sigma^2 + P_d G_d^B\right)}{P_c}$. Due to the fact that $G_c^B \propto d^{-n}$ (e.g., in free space, $n = 2$) where d denotes the distance between CUE and BS, there is a threshold d_{th} for the distance between CUE and BS. Specifically, if the CUE is close enough to the BS (i.e., $d \leq d_{th}$), D2D pairs can reuse the CUE's resources without affecting cellular communication quality (i.e., non-orthogonal manner). If the CUE is far away from BS (i.e., $d > d_{th}$), the SINR of the CUE is weak and thus D2D pairs can only share orthogonal resources with the CUE to guarantee cellular communication (i.e., orthogonal manner). As shown in Fig. 1, we simply expect that the cell coverage is a circular area with radius R meters and a circular area within d_{th} meters away from the BS is regarded as an "accessible area". Only if the CUE moves to this "accessible area" can D2D pairs be allowed to access the spectrum in the CUE's TS. For example, if CUE1 locates in the "accessible area" while CUE2 is out of the "accessible area", the spectrum access of D2D pairs within a frame of 10 TSs is shown in Fig. 2 where the CUE's location is fixed within a frame. In fact, we explore the scenario where the CUE's location dynamically changes over time in this letter.

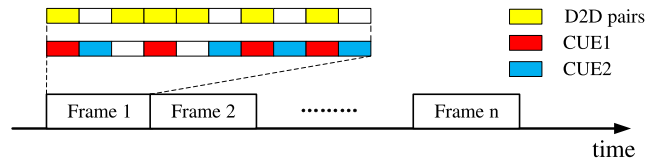

Fig. 2. Spectrum access within a frame of 10 TSs.

According to the above analysis, we can know that d_{th} is determined by the threshold of the CUE's SINR when P_c, P_d and G_d^B are not changed, which means d_{th} is an adjustable parameter in the environment. By interacting with the environment, D2D pairs are expected to perceive the existence of the "accessible area" autonomously, which plays an important role when D2D pairs learn to switch between two access manners based on the CUE's location. Furthermore, we assume that D2D pairs have two communication modes: D2D mode and cellular mode (when two devices become too far to communicate with each other directly), and the concept of D2D communication probability is defined to indicate the probability of a D2D pair in D2D mode. Only if the D2D pair is in D2D mode can it be allowed to reuse CUEs resources, which should be autonomously learned by D2D pairs through the perception of the environment.

III. DRL Based Dynamic Spectrum Access

A. Dynamic D2D Spectrum Access Using DDQN

In RL, the agent interacts with the environment in discrete time steps [10]. At time t, the agent observes the environment state $\mathrm{s}_t \in S$ and takes an action $\mathrm{a}_t \in A$ according to a policy π, where A and S are the action set and the state set, respectively. Then, the agent will receive a reward r_{t+1} and the environment transforms to the next state s_{t+1}. The goal of the agent is to learn an optimal policy π^* to maximize the cumulative discounted reward defined as $R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+1+k}$, where $\gamma \in [0, 1)$ is a discount factor to indicate the importance of future rewards. For our framework, each D2D pair should be considered as an agent and thus a multi-agent problem is presented. However, the training of multi-agent is difficult due to more requirements of training time and competition among D2D pairs. Hence, to avoid conflicts when D2D pairs make decisions independently, we utilize a centralized agent to coordinate the transmission of D2D pairs. Specifically, the centralized agent decides whether to transmit and which one to transmit at current TS. If the agent selects a D2D to transmit, the selected D2D will receive feedback after transmitting and then communicate with the agent with this information. The coordination information can be sent through a control channel which can be implemented as a short TS before each TS of information transmission. Other implementations are also possible and we omit the discussion here since the focus of this letter is not on implementation details. The detailed definitions of "action", "state" and "reward" are given as follows.

1) Action: The action taken by the agent decides whether to transmit and which one to transmit. Therefore, the action set can be defined as $\{0, 1, 2, \ldots, L\}$, where L denotes the number of D2D pairs in the cell. The action $a_t = 0$ means that D2D pairs do not transmit, while $a_t = i$ ($i = 1, 2, \ldots, L$) means that the ith D2D pair transmits signals.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO ESPIRITO SANTO. Downloaded on February 22,2022 at 12:03:38 UTC from IEEE Xplore. Restrictions apply.

Downloaded from https://iranpaper.ir
http://www.master-thesis.ir/Employment
shiraz.barg@gmail.com

2) State: The agent is expected to listen to the channel, so we first define the channel state observed by the agent after taking action a_t as $c_t \in \{\mathcal{S}, \mathcal{I}, \mathcal{F}, \mathcal{R}\}$, where $\mathcal{S}$ means only one user transmits at current TS; $\mathcal{I}$ denotes no user transmits; $\mathcal{F}$ means that D2D pairs reuse current TS and cause severe interference to cellular communication; $\mathcal{R}$ presents that D2D pairs share current TS with a CUE successfully. Plus, the BS can obtain CUEs location through some wireless positioning technologies and save the information in the database. Then, the agent can obtain the location information of CUEs from the BS through a control channel and thus maintains a distance vector $\boldsymbol{d} = [d_1, d_2, \ldots, d_K]$ to indicate the distance information between CUEs and BS, where K denotes the number of CUEs in the cell. And D2D pairs share the mode information with the centralized agent. Therefore, there is a mode vector $\boldsymbol{m} = [m_1, m_2, \ldots, m_L]$, where m_i $(i = 1, 2, \ldots, L)$ is 1 or 0 indicating the ith D2D pair under D2D mode or not, respectively. Then, the agent's observation of the environment at time $t+1$ is described as $\boldsymbol{o}_{t+1} = [a_t, c_t, \boldsymbol{d}_{t+1}, \boldsymbol{m}_{t+1}]$. The environment state at time $t+1$ is defined as $s_{t+1} = [\boldsymbol{o}_{t-M+2}, \ldots, \boldsymbol{o}_t, \boldsymbol{o}_{t+1}]$, where M is the state history length.

3) Reward: The reward function is determined according to the channel state. The reward r_{t+1} received by the agent after taking action a_t is given as

$$r_{t+1} = \begin{cases} 0, & c_t = \mathcal{I} \quad or \quad \mathcal{F}, \\ 1, & c_t = \mathcal{S}, \\ 2, & c_t = \mathcal{R}. \end{cases} \tag{2}$$

The description of reward here is associated with the objective of maximizing the sum throughput.

Q-learning, a typical RL algorithm, utilizes an action-value function named Q-function as the objective function. The Q-function for an action a at a state s under a particular policy π is defined as $Q^{\pi}(s,a) = E_{\pi}\left[R_t \mid s_t = s, a_t = a\right]$. Q-learning maintains a Q-value table to find an optimal policy π^*, i.e., $Q^*(s,a) = \max_{\pi} Q^{\pi}(s,a)$. However, when the state space and/or the action space is large, it will take up a lot of time and resources for Q-learning to converge. DQN introduces a neural network as Q neural network (QNN) to approximate the Q-value instead of maintaining a Q-table. Specifically, QNN gets an input state s_t and outputs the estimated Q-values for all possible actions, i.e., $q(s_t, a_t; \boldsymbol{\theta}) \approx Q(s_t, a_t), a_t \in A$, where $\boldsymbol{\theta}$ denotes the QNN parameter. In DQN, the agent learns an optimal policy to optimize the Q-function by minimizing the loss function:

$$Loss(\boldsymbol{\theta}) = E\left[\left(y_t - q(s_t, a_t; \boldsymbol{\theta})\right)^2\right], \tag{3}$$

where y_t is the target Q-value described as

$$y_t = r_{t+1} + \gamma \max_a q(s_{t+1}, a, \boldsymbol{\theta}). \tag{4}$$

The agent trains the QNN to optimize the approximation of Q-values by updating the parameter iteratively as follows:

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \alpha E\left[\left(y_t - q(s_t, a_t; \boldsymbol{\theta})\right) \nabla q(s_t, a_t; \boldsymbol{\theta})\right], \tag{5}$$

where $\alpha \in [0, 1)$ is the learning rate. Moreover, DQN has an experience reply pool to store the experience data collected by the agent in the form of $(s_t, a_t, r_{t+1}, s_{t+1})$, from which a batch of experiences will be sampled randomly to train the QNN as defined in (5). To improve the algorithm stability, a 'double networks' method is adopted, which means that there is another target network with the parameter $\boldsymbol{\theta'}$ in the training process. The target Q-value y_t defined in (4) is replaced by

$$y_t = r_{t+1} + \gamma \max_a q(s_{t+1}, a; \boldsymbol{\theta'}), \tag{6}$$

where $\boldsymbol{\theta'}$ should be updated by $\boldsymbol{\theta}$ after every C training steps.

DQN still has some problems such as overestimation, which means the estimated Q-value may be higher than the true value. Therefore, we adopt DDQN to design our proposed D^{4}SA algorithm, which decouples the action selection from the estimation of target Q-value to prevent overestimation. The target Q-value defined in (6) is replaced by the following:

$$y_t = r_{t+1} + \gamma q(s_{t+1}, a; \boldsymbol{\theta'}), \tag{7}$$

where $a = \arg\max_a q(s_{t+1}, a; \theta)$. The agent adopts ε-greedy policy when it selects action based on Q-values [10].

B. Fairness Objective

For original D^{4}SA algorithm discussed above, the reward function defined in (2) is designed to represent the overall transmission result of all users and thus the Q-function is related to the objective of maximizing the sum throughput. However, the agent can learn an optimal policy to maximize the sum throughput but this objective can not instruct the agent to learn an optimal policy about how to allocate TSs among D2D pairs. For example, if the current TS is available for multiple D2D pairs, the agent will not know which one is the optimal choice, thus causing the unfairness of resource allocation among D2D pairs, which may affect the user experience. To solve this problem, we put forward a new objective function: $\sum_{i=1}^{L+1} \log(v^{(i)})$, where $v^{(i)}$ $(i = 1, 2, \ldots, L)$ is the throughput of the ith D2D pair and $v^{(L+1)}$ is the sum throughput of CUEs. The detailed introduction is given below.

We first reformulate the reward function for D^{4}SA algorithm. The reward received by the agent here is an $L+1$ dimension vector denoted by $\left[r^{(i)}\right]_{i=1}^{L+1}$, where $r^{(L+1)}$ represents the overall reward of CUEs and $r^{(i)}$ $(i = 1, 2, \ldots, L)$ is the reward of the ith D2D pair. Only if the ith $(i = 1, 2, \ldots, L+1)$ user transmit successfully can $r^{(i)}$ be assigned to 1. More specifically, $r_{t+1}^{(L+1)}$ is 1 in two cases: a) $a_t = 0$ and $c_t = \mathcal{S}$; b) $c_t = \mathcal{R}$. $r_{t+1}^{(i)}$ $(i = 1, 2, \ldots, L)$ is assigned to 1 in two cases: a) $a_t = i$ and $c_t = \mathcal{S}$; b) $c_t = \mathcal{R}$. Accordingly, the Q-function maintained by the agent is also an $L+1$ dimension vector denoted by $\left[Q^{(i)}(s,a)\right]_{i=1}^{L+1}$, where $Q^{(i)}(s,a)$ is the cumulative discounted reward of the ith user and thus it corresponds to the throughput of the ith user. As for the new objective function, due to the fact that the real throughput is hard to know, we replace $v^{(i)}$ with the estimated Q-value $q^{(i)}(s, a; \boldsymbol{\theta})$. Hence, for the ε-greedy algorithm, the agent chooses action a_t which can maximize $\sum_{i=1}^{L+1} \log(q^{(i)}(s_t, a_t; \boldsymbol{\theta}))$ with probability $1 - \varepsilon$. The loss function and target Q-value defined in (3), (7)

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO ESPIRITO SANTO. Downloaded on February 22,2022 at 12:03:38 UTC from IEEE Xplore. Restrictions apply.

Downloaded from https://iranpaper.ir
http://www.master-thesis.ir/Employment
shiraz.barg@gmail.com

Algorithm 1 D^{4}SA Algorithm With Fairness Objective

Initialize replay memory D to capacity N
Initialize current QNN with random parameters $\boldsymbol{\theta}$
Initialize target QNN with random parameters $\boldsymbol{\theta'}$
for $t = 1, 2, \ldots, T$ **do**
  Input s_t to QNN and output $Q = \{q^{(i)}(s_t, a; \boldsymbol{\theta}) | a \in A, i = 1, 2, \ldots, L+1\}$
  Select action a_t using ϵ-greedy algorithm
  Execute action a_t and observe reward $\left[r_{t+1}^{(i)}\right]_{i=1}^{L+1}$ and next state s_{t+1}
  Store experience tuple $\left(s_t, a_t, \left[r_{t+1}^{(i)}\right]_{i=1}^{L+1}, s_{t+1}\right)$ in D
  Every C steps reset $\boldsymbol{\theta'}$ as $\boldsymbol{\theta}$
  TRAIN QNN
end for
procedure TRAIN QNN
  Randomly sample N_E experience tuples from D as E
  for each sample $e = \left(s, a, \left[r^{(i)}\right]_{i=1}^{L+1}, s'\right)$ in E **do**
    $a' = \arg\max_a \sum_{i=1}^{L+1} \log(q^{(i)}(s', a; \boldsymbol{\theta}))$
    Calculate $y_e^{(i)} = r^{(i)} + \gamma q^{(i)}(s', a'; \boldsymbol{\theta'})$
  end for
  Perform Gradient Descent to update $\boldsymbol{\theta}$:

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \frac{\alpha}{N_E} \sum_{e \in E} \frac{1}{L+1} \sum_{i=1}^{L+1} \times (y_e^{(i)} - q^{(i)}(s, a; \boldsymbol{\theta})) \nabla q^{(i)}(s, a; \boldsymbol{\theta})$$

end procedure

should be replaced by the following:

$$Loss(\boldsymbol{\theta}) = E[\frac{1}{L+1} \sum_{i=1}^{L+1} (y_t^{(i)} - q^{(i)}(s_t, a_t; \boldsymbol{\theta}))^2], \quad (8)$$

$$y_t^{(i)} = r_{t+1} + \gamma q^{(i)}(s_{t+1}, a; \boldsymbol{\theta'}), \quad (9)$$

where $a = \arg\max_a \sum_{i=1}^{L+1} \log(q^{(i)}(s_{t+1}, a; \boldsymbol{\theta}))$. The update of $\boldsymbol{\theta}$ is given by

$$\boldsymbol{\theta} = \boldsymbol{\theta} + \alpha E[\frac{1}{L+1} \sum_{i=1}^{L+1} (y_t^{(i)} - q^{(i)}(s_t, a_t; \boldsymbol{\theta})) \nabla q^{(i)}(s_t, a_t; \boldsymbol{\theta})]. \quad (10)$$

The training process pseudo code for improved D^{4}SA algorithm with fairness objective is shown in Algorithm 1.

IV. Performance Evaluation

This section evaluates the performance of the proposed algorithm. We begin by presenting the simulation setup and then showcase the simulation results under different conditions.

A. Simulation Setup

We employed Google TensorFlow to establish the experimental environment and adopted 4 NVIDIA Tesla V100 boards as the DRL training platform to train the QNN. For the system, the location of CUE is randomly selected within the cell coverage at each TS and thus the probability of CUEs in the "accessible area" is evaluated by $\frac{\pi * d_{th}^2}{\pi * R^2}$ which is related to the radius of the "accessible area" denoted by d_{th}. In addition, residual units [11] are introduced to construct the QNN which includes 6 hidden layers and each hidden layer has 64 neurons with ReLU function as the activation function. Moreover, we adopt Adam optimizer to update $\boldsymbol{\theta}$ with a mini-batch of 64 experience samples. As for other parameters, the state history length M is set to 20, the learning rate α used in Adam is 0.01, and the discount factor γ is 0.9. The exploration ε in ε-greedy algorithm is set to 1 at the beginning and decreases to 0.005 with a decay rate of 0.995. The update frequency for target network C is set to 200.

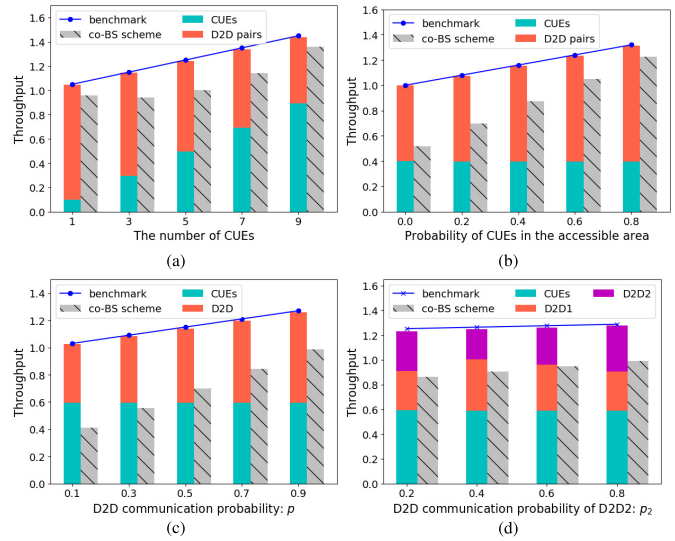

Fig. 3. Sum throughput and individual throughput for different cases. The "co-BS scheme" denotes the sum throughput of the scheme based on BS cooperation.

The throughput is regarded as the performance metrics, which is defined as the number of packets successfully transmitted per TS, and the average throughput at time t is calculated over N TSs by the formulation: $\sum_{\tau=t-N+1}^{t} r_\tau / N$. In our scheme, D2D pairs have to learn an optimal access strategy without any prior scheduling information of the system. For benchmark, it is a theoretical upper bound of sum throughput when D2D pairs are regarded as model-aware users and execute the optimal access strategy. Further, we compare our scheme with a spectrum access scheme where D2D pairs lack learning ability and can only obtain partial information of the environment by the cooperation of BS.

B. Analysis of Simulation Results

With the objective of maximizing the sum throughput, the optimal strategy is that D2D pairs catch all the unused TSs within a frame and reuse the CUE's TS when the CUE is in the "accessible area" and at least one D2D pair is in D2D mode. In the following analysis, q denotes the probability of CUEs in the "accessible area" which is calculated by $\frac{\pi * d_{th}^2}{\pi * R^2}$ and p is the D2D communication probability.

In Fig. 3(a) and (b), we first consider the cases when D2D communication probability p is 1, where all D2D pairs have the same opportunity to access current TS and thus multiple D2D pairs can be represented by one virtual D2D pair. For simplicity, we assume that each CUE is assigned ten percent

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO ESPIRITO SANTO. Downloaded on February 22,2022 at 12:03:38 UTC from IEEE Xplore. Restrictions apply.

Downloaded from https://iranpaper.ir
http://www.master-thesis.ir/Employment
shiraz.barg@gmail.com

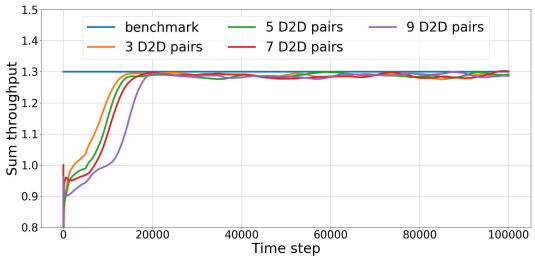

Fig. 4. Sum throughput when six CUEs coexist with multiple D2D pairs.

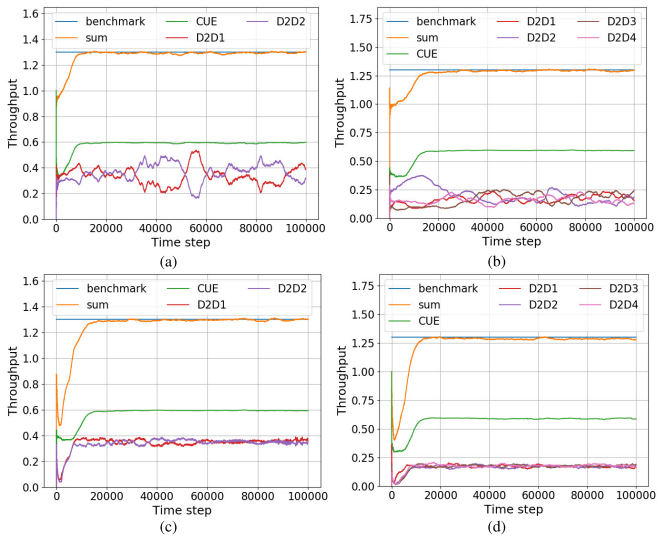

Fig. 5. Sum throughput and individual throughput with different objective functions. (a) and (b) are the results with the objective of maximizing the sum throughput. (c) and (d) are the results with fairness objective.

of the TSs within a frame. Fig. 3(a) presents the throughput results when the number of CUEs increases from 1 to 9 with a fixed $q = 0.5$, and Fig. 3(b) shows the results when q is changed from 0.1 to 0.9 in the case of 4 CUEs. Then, we investigate the performance with the D2D communication probability $p \in (0, 1)$, where D2D pairs with different p have different opportunities to access current TS. For convenience, variable q is fixed to 0.5. With the coexistence of six CUEs and one D2D pair, we change p from 0.1 to 0.9 and the throughput results are given in Fig. 3(c). After that, we add a D2D pair to the system and D2D communication probabilities of two D2D pairs (D2D1 and D2D2) are denoted by p_1 and p_2, respectively. Fig. 3(d) presents the results when p_1 is 0.8 and p_2 is changed from 0.2 to 0.8. All the results show that our algorithm can achieve an optimal sum throughput referring to theoretical upper bound and the performance gets significantly improved compared to the scheme based on BS cooperation.

Fig. 4 shows the results when the number of D2D pairs is 3, 5, 7, 9 with 6 CUEs. We can see that the algorithm can still achieve an optimal sum throughput but the convergence performance may decrease a little as the number of D2D pairs increases. Fig. 5(a) and (b) exhibit the throughput results of original D^{4}SA algorithm only with sum throughput objective when six CUEs coexist with 2 and 4 D2D pairs, respectively, which show that the originally proposed D^{4}SA algorithm can achieve an optimal sum throughput but the individual throughput of each D2D pair is unstable and stochastic. Then, we incorporate fairness into the objective function, as discussed in Section III.B. And we evaluate the performance of D^{4}SA algorithm with fairness objective in the same situation with that in Fig. 5(a) and (b). The results are presented in Fig. 5(c) and (d), from which we can see that not only can the agent achieve an optimal sum throughput, but it can achieve the fairness of resource allocation among D2D pairs.

V. Conclusion

This letter investigated the dynamic spectrum access problem for D2D communication underlay cellular networks. Specifically, we considered two spectrum access manners for D2D communication based on the locations of CUEs and introduced an "accessible area" to guarantee the communication quality of CUEs at the cell edge. Based on DRL theory, we proposed a D^{4}SA algorithm which enables D2D pairs to autonomously learn an optimal spectrum access strategy to maximize the sum throughput while prior information is completely unknown. Further, we discovered that resource allocation among D2D pairs is stochastic and unfair, and thus we reformulated the objective function to instruct the agent to achieve both the maximum of sum throughput and the fairness among D2D pairs. Last, simulation results show that the proposed algorithm can achieve an optimal performance, which significantly outperforms the scheme based on BS cooperation. For further research, we plan to explore the scenario where CUEs execute more complex access principles and adopt distributed multi-agent algorithms to improve the performance of the proposed method.

References

[1] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 4, pp. 1801–1819, Apr. 2014.

[2] D. Cotton and Z. Chaczko, "GymD2D: A device-to-device underlay cellular offload evaluation platform," 2021, *arXiv:2101.11188*. [Online]. Available: http://arxiv.org/abs/2101.11188

[3] Y. Kai, J. Wang, H. Zhu, and J. Wang, "Resource allocation and performance analysis of cellular-assisted OFDMA device-to-device communications," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 416–431, Jan. 2019.

[4] K. Zia, N. Javed, M. N. Sial, S. Ahmed, A. A. Pirzada, and F. Pervez, "A distributed multi-agent RL-based autonomous spectrum allocation scheme in D2D enabled multi-tier HetNets," *IEEE Access*, vol. 7, pp. 6733–6745, Jan. 2019.

[5] Z. Li and C. Guo, "Multi-agent deep reinforcement learning based spectrum allocation for D2D underlay communications," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1828–1840, Feb. 2020.

[6] D. Wang, H. Qin, B. Song, K. Xu, X. Du, and M. Guizani, "Joint resource allocation and power control for D2D communication with deep reinforcement learning in MCC," *Phys. Commun.*, vol. 45, Apr. 2021, Art. no. 101262.

[7] C. Guo, L. Liang, and G. Y. Li, "Resource allocation for vehicular communications with low latency and high reliability," *IEEE Trans. Wireless Commun.*, vol. 18, no. 8, pp. 3887–3902, Aug. 2019.

[8] T. Zhang, K. Zhu, and J. Wang, "Energy-efficient mode selection and resource allocation for D2D-enabled heterogeneous networks: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1175–1187, Feb. 2021.

[9] A. Moussaid, W. Jaafar, W. Ajib, and H. Elbiaze, "Deep reinforcement learning-based data transmission for D2D communications," in *Proc. 14th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2018, pp. 1–7.

[10] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DO ESPIRITO SANTO. Downloaded on February 22,2022 at 12:03:38 UTC from IEEE Xplore. Restrictions apply.