A Reinforcement Learning Based Joint Spectrum Allocation and Power Control Algorithm for D2D Communication Underlaying Cellular Networks

Wentai Chen and Jun Zheng(✉)

National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, Jiangsu, People's Republic of China
{wtchen, junzheng}@seu.edu.cn

Abstract. This paper studies the spectrum allocation and power control (SA-PC) problem in device-to-device (D2D) communication underlaying a cellular network. A distributed multi-agent reinforcement learning (MARL) based joint SA-PC algorithm is proposed for performing spectrum allocation and power control for each D2D user in the network. The proposed algorithm uses Q learning, a typical form of reinforcement learning (RL), to select the optimal resource block (RB) and power level for each D2D user. In the Q-learning algorithm, each D2D user is treated as an individual agent and maintains a single-state Q table. Each agent selects an RB and a power level according to its Q table in the learning process. Simulation results show that the proposed Q-learning based joint SA-PC algorithm can achieve good throughput performance.

Keywords: D2D communication · Spectrum allocation · Power control · Multi-agent reinforcement learning · Q learning

1 Introduction

Device to device (D2D) communication is one of the promising technologies for future mobile cellular networks. In D2D communication, two mobile devices directly communicate with each other without traversing a base station (BS), which can effectively improve spectral efficiency, increase the system throughput, and reduce the data transmission latency of a network [1]. Usually, D2D communication works in an underlay mode in which D2D users share the spectrum resources of cellular users. While this can effectively improve the network performance, it would on the other hand cause severe interference between D2D users and cellular users. Accordingly, the mitigation of such interference becomes a critical issue in D2D communication. An effective way to address this issue is through efficient resource management, including spectrum allocation and power control. In this context, extensive work has been conducted and a variety of spectrum allocation and/or power control algorithms have been proposed using traditional approaches [2–8]. With recent advances in artificial intelligence (AI), machine learning (ML) is arousing a widespread interest from the

S. Han et al. (Eds.): AICON 2019, LNICST 286, pp. 146–158, 2019.
https://doi.org/10.1007/978-3-030-22968-9_13

community of wireless communication. Considerable work has been conducted in applying this advanced approach to wireless communication in general and D2D communication in particular. But even so, relevant work on resource management for D2D communication is still limited. It is interesting to further explore the application of the advanced ML approach in spectrum allocation and power control for improving the performance of D2D communication, which motivated us to conduct this work.

In this paper, we study the spectrum allocation and power control (SA-PC) problem in D2D communication underlaying a cellular network using the ML approach. A distributed multi-agent reinforcement learning (MARL) based joint SA-PC algorithm is proposed for performing spectrum allocation and power control for each D2D user in the network. Specifically, the proposed algorithm uses Q learning, a typical form of reinforcement learning (RL), to select the optimal resource block (RB) and power level for each D2D user. In the algorithm, each D2D user is treated as an individual agent and maintains a single-state Q table. In the learning process, each agent selects an RB and a power level according to its Q table. Simulation results are shown to evaluate the performance of the proposed Q-learning based joint SA-PC algorithm in terms of the throughput performance.

The rest of the paper is organized as follows. Section 2 reviews related work in the literature. Section 3 describes the system model and formulates the SA-PC problem considered in this paper. Section 4 presents the proposed MARL-based SA-PC algorithm. Section 5 shows simulation results to evaluate the performance of the proposed algorithm. Section 6 concludes the paper.

2 Related Work

Spectrum allocation (SA) and power control (PC) have been extensively studied for D2D communication underlaying cellular networks. A variety of SA-PC algorithms have been proposed in the literature [2–8]. In [2], Cai et al. proposed a capacity oriented resource allocation algorithm (CORAL) for resource allocation in D2D communication. The proposed algorithm introduces the concept of a Capacity-Oriented REstricted (CORE) region for a D2D pair to determine a candidate cellular user set for the D2D pair in resource allocation, which can help increase the system capacity. In [3], Chen et al. proposed a time division scheduling (TDS) resource allocation algorithm to efficiently exploit the downlink spectrum resources of cellular users for D2D communication. In the D2D pair assignment for each timeslot, the proposed algorithm follows a location dispersion principle in order to reduce the interference from D2D users to cellular users and thus can increase the system throughput. In [4], Cai et al. proposed a graph-coloring resource allocation (GOAL) algorithm using a graph-coloring approach and introduced the concept of the interference negligible distance (INS) to identify those D2D pairs which can simultaneously share the same spectrum resources of cellular users, and the concept of the signal to interference ratio limited area (SLA) to identify a set of D2D pairs which cannot share the spectrum resources of a cellular user. In [5], Chen et al.proposeda service-aware resource allocation (SARA) scheme for D2D communication to improve the network performance, which takes into account the different service requirements of D2D users. In [6], Zulhasnine et al.

proposed a centralized heuristic algorithm considering the interference link gain from a D2D transmitter to the BS. The optimization problem is formulated as a Mixed Integer Non-Linear Programming (MINLP) with the synchronized resource allocation of the cellular users and D2D users. In [7], Esmat et al. proposed a two-phase optimization algorithm for adaptive resource allocation, which provides better system throughput than the traditional algorithm by computing a Lagrangian dual decomposition (LDD) problem. In [8], Hsu and Chen proposed a power control and channel allocation algorithm, in which the channel of each user device is reallocated after the first turn of power allocation. The channel reallocation and power control proceed until the transmission power no longer decreases. Simulation results show that the proposed algorithm outperforms the existing algorithms in terms of system capacity.

With recent advances in the ML area, considerable work has been conducted to explore the application of the ML approach in resource management for D2D communication [9–12]. In [9], a Q learning-based algorithm was proposed for the resource allocation in a single-cell scenario with two cellular users and several D2D users whose arrival follows a poison process. In [10], a centralized Q-learning algorithm and a distributed Q-learning algorithm were proposed to solve the power control problem for D2D communication underlying cellular networks. In [11], a power control method based on Classification and Regression Tree (CART) was proposed, which provides a faster convergence than the reinforcement learning methods. In [12], an adaptive resource allocation algorithm was proposed using cooperative reinforcement learning considering the neighboring factor of the D2D users. The proposed algorithm considers the coordination problem between different D2D users and can achieve a better system throughput than some existing reinforcement learning algorithms.

3 System Model and Problem Formulation

In this section, we first describe the system model and then formulate the joint spectrum allocation and power control (SA-PC) problem considered in this paper.

3.1 System Model

We consider a single-cell cellular system consisting of one base station (BS), M cellular users (CUs) and N D2D user (DU) pairs, where a DU pair consists of the transmitter (T_x) of one DU and the receiver (R_x) of another DU. The set of M CUs is denoted by $\mathcal{C} = \{C_1, C_2, \ldots, C_M\}$ and the set of N DU pairs is denoted by $\mathcal{D} = \{D_1, D_2, \ldots, D_N\}$. There are K orthogonal resource blocks (RBs) in the system, which are denoted by $\mathcal{B} = \{B_1, B_2, \ldots, B_K\}$. We assume that the DU pairs work in an underlay mode and are allowed to share the uplink transmission resources (i.e., RBs) of the CUs. Each RB is occupied by one CU and can be shared by one or more DU pairs. Either a CU or a DU pair is allowed to occupy only one RB. The system model is illustrated in Fig. 1.

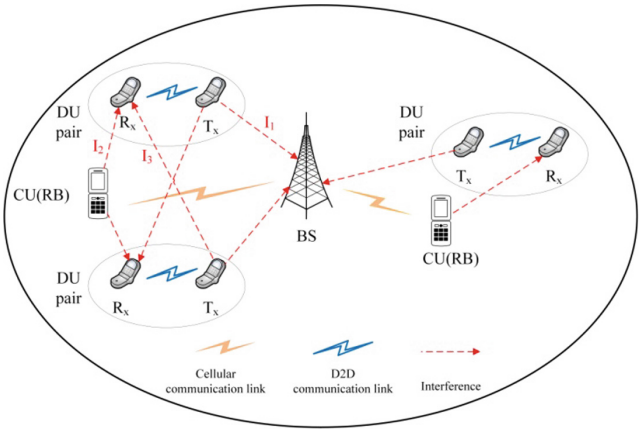


Fig. 1. System model

There exist three types of interferences in the system model, which are illustrated in Fig. 1:

(1) I_1: the interference from the transmitter T_x of a D2D pair to the BS;
(2) I_2: the interference from a CU to the receiver R_x of a D2D pair, where the CU and the DU pair share the same RB;
(3) I_3: the interference from the transmitter T_x of one D2D pair to the receiver R_x of another D2D pair, where the two D2D pairs share the same RB.

3.2 Problem Formulation

In this paper, we consider the joint spectrum allocation and power control problem in D2D communication underlaying a single-cell cellular system shown in Fig. 1. There are two aspects in the SA-PC problem: RB allocation for the DU pairs (i.e., SA) and power assignment for the DU pairs (i.e., PC). For simplicity, we assume that $M = K$ and the RB allocation for the CUs is fixed. Each CU is allocated a different RB.

Before we formulate the power control problem, we first analyze the signal to interference plus noise ratio (SINR) at a CU and at the receiver of a DU, respectively. For a CU that occupies the rth RB, the SINR at the CU is given by

$$SINC_{C_i}^r = \frac{p_{C_i}^r \cdot G_{C_i}^r}{\sigma^2 + \sum_{D_j \in \mathcal{D}^r} p_{D_j}^r \cdot G_{D_j}^r}, \quad i=1,2,\ldots,M; j=1,2,\ldots,N \tag{1}$$

where C_i denotes the ith CU, D_j denotes the jth DU pair, $\mathcal{D}^r$ denotes a set of DU pairs that share the rth RB, $p_{C_i}^r$ and $p_{D_j}^r$ denote the transmission power of C_i and D_j which share the rth RB, respectively, $G_{C_i}^r$ and $G_{D_j}^r$ denote the channel gains on the rth RB from the BS to C_i and D_j, respectively, and σ^2 is the noise variance.

Similarly, for a DU pair that shares the rth RB, the SINR at the receiver of the DU pair is given by

$$SINC_{D_j}^r = \frac{p_{D_j}^r \cdot G_{D_jD_j}^r}{\sigma^2 + p_{C_i}^r \cdot G_{C_iD_j}^r + \sum\limits_{\substack{D_k \in \mathcal{D}^r \\ k \neq j}} p_{D_k}^r \cdot G_{D_kD_j}^r}, \tag{2}$$

where $G_{D_jD_j}^r$, $G_{C_iD_j}^r$, and $G_{D_kD_j}^r$ denote the channel gain on the link from the transmitter of D_j to the receiver of D_j, the channel gain from C_i to the receiver of D_j, and the channel gain from the transmitter of D_k to the receiver of D_j, respectively.

Next we formulate the joint SA-PC problem. Given a set of RBs $\mathcal{B} = \{B_1, B_2, \ldots, B_K\}$ and a set of power levels $\mathcal{P} = \{p_1, p_2, \ldots, p_L\}$, the SA-PC problem under consideration is to jointly find a set of optimal RBs $\mathcal{B}_o^* = \{B_{D_1}^*, B_{D_2}^*, \cdots, B_{D_N}^*\}$ and a set of optimal power levels $\mathcal{P}_o^* = \{p_{D_1}^*, p_{D_2}^*, \cdots, p_{D_N}^*\}$ for all the DU pairs so that the overall system throughput is maximized, i.e.,

$$\text{Objective:} \quad \max \sum_{r=1}^{K} \{\log_2(1 + SINR_{C_i}^r) + \sum_{D_j \in \mathcal{D}^r} \log_2(1 + SINR_{D_j}^r)\} \tag{3}$$

$$\text{subject to} \quad SINR_{C_i}^r \geq \tau_0, \tag{4}$$

$$p_1 \leq p_{D_j}^r \leq p_L, \qquad \forall j, r, \tag{5}$$

where τ_0 denotes the minimum SINR requirement of a CU, constraint (4) ensures the SINR requirement of each CU, and constraint (5) ensures that the transmission power of each DU is limited to the range $[p_1, p_L]$.

4 MARL-Based SA-PC Algorithm

In this section, we first introduce the concept of reinforcement learning and then present an MARL-based algorithm to solve the SA-PC problem formulated in Sect. 3.

4.1 Reinforcement Learning

Reinforcement learning is an important branch of machine learning. A typical RL problem can be modeled as a Markov Decision Process (MDP), which is defined as a decision process that satisfies the Markov property, i.e., the environment's response at time $t + 1$ depends only on the state and action at t, and does not rely on the previous states. An MDP can be represented as a tuple of 5 elements, denoted as $\{\mathcal{S}, \mathcal{A}, \mathcal{T}, R, \gamma\}$:

- $\mathcal{S}$: a finite set of all possible states;
- $\mathcal{A}$: a finite set of actions that can be selected by an agent;
- $\mathcal{T}$: a set of transition probabilities from one state to another;
- R: a reward function to evaluate the action chosen by an agent;
- γ: a discount factor to balance the effect of the future reward and the immediate reward.

In a standard RL process, an agent interacts with the environment in a sequence of episodes, which are denoted by $t = 0, 1, 2, \ldots$ There are three steps in each episode, which are shown in Fig. 2:

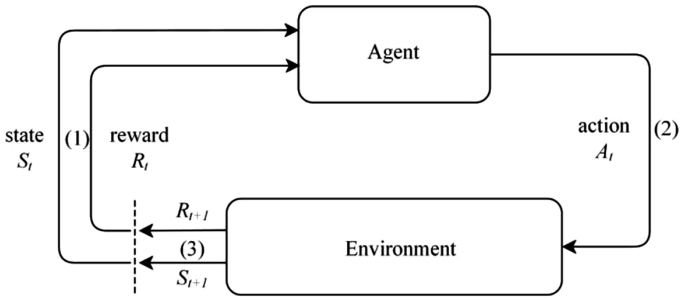


Fig. 2. Standard reinforcement learning process

(1) The agent receives the current state S_t and reward signal R_t.
(2) The agent selects and executes the action A_t.
(3) The environment generates a new reward R_{t+1} and transfers to another state S_{t+1}.

An agent starts learning from an initial state S_0, and continues the episodes until the learning process converges.

In the above learning process, an agent selects its action in each episode according to a policy π, which is given by

$$\pi_t(a \mid s) = P(A_t = a \mid S_t = s), \quad a \in \mathcal{A}, \ s \in \mathcal{S}, \tag{6}$$

where $P(A_t = a \mid S_t = s)$ is the probability of selecting action a in state s. By selecting different actions and updating the current policy in each episode, the agent can make a better decision and reaches the optimal policy π^* after a number of episodes.

Next we introduce a typical form of reinforcement learning: Q learning, which will be used to solve the SA-PC problem. Like other RL methods, Q learning needs no prior knowledge about the environment. In Q learning, an agent learns how to behave based on the previous experience, which is traced by a Q function. The Q function is used to determine the value of an action a in a given state s and is defined

$$Q_t(s, a) = E_\pi[\sum_{i=1}^{\infty} \eta^{i-1} R_{t+i} \mid S_t = s, A_t = a], \tag{8}$$

where $E_\pi(\cdot)$ denotes the expected value of a random variable given that the agent follows the policy π, t denotes the index of the current episode, η is the discount rate, and R_{t+i} is the reward in the $(t + i)$th episode.

In Q learning, the Q function is represented by a two-dimensional table, in which each row represents a state of the environment, and each column represents an action of an agent. The learning process starts with an initial state, and the initial Q table is usually set to all zeros. At each episode, assuming that the current state is $S_t = s$, the

agent needs to select the best action $A_t = a$ according to the learned policy. After performing a, the agent will receive a reward R_{t+1}, and the environment will transfer to the next state S_{t+1}. It can be proved by induction that Q learning is able to converge to the optimal values if all the states can be visited infinitely as the learning process proceeds [13].

In the SA-PC problem, each D2D user is treated as an independent agent. In this way, it becomes a multi-agent Q learning problem, in which all DU pairs on different RBs need to learn a global optimal policy. In the next section, we will present a distributed Q-learning algorithm for solving the problem.

4.2 Distributed Q-learning Based Joint SA-PC Algorithm

In this section, we present the proposed distributed Q-learning joint SA-PC algorithm for D2D communication.

A. Component definitions

We first define the components of distributed Q learning, namely agent, action and reward. In the distributed Q learning algorithm, an agent is defined as a DU pair in the cellular system. Thus, there are N agents in the whole system. An action of an agent is defined as the action that a DU pair takes to select a RB from $\mathcal{B} = \{B_1, B_2, \ldots, B_K\}$ and a power level p from $\mathcal{P} = \{p_1, p_2, \ldots, p_L\}$. A reward is defined as the overall system throughput including all CUs and all DU pairs. The value of the reward is determined using the following reward function:

$$R = \sum_{r=1}^{K} R^r, \tag{8}$$

where R^r is the reward on the rth RB, which is given by

$$R^r = \begin{cases} \log_2(1 + SINR^r_{C_i}) + \sum\limits_{D_j \in \mathcal{D}^r} \log_2(1 + SINR^r_{D_j}), & SINR^r_{C_i} \geq \tau_0 \\ -1 & \text{otherwise} \end{cases}. \tag{9}$$

According to Eq. (9), if the SINR requirement of C_i cannot be satisfied, i.e., $SINR^r_{C_i} < \tau_0$, the reward is set to -1 as a penalty term, which ensures the priority of C_i.

In a standard Q learning algorithm, an agent needs to transfer between different states by selecting different actions, which usually takes a large number of episodes to converge. Furthermore, it is difficult to define the states in the Q learning algorithm that matches the physical states in the single-cell cellular system [14]. According to [14], we use a single state in the distributed Q learning algorithm and the state formulation is not needed.

B. Algorithm description

The distributed Q learning based SA-PC algorithm is proposed for performing spectrum allocation and power control for the DU pairs in the system. In the algorithm, each agent maintains a single-state Q table of size $(1, K \times L)$. In the learning process, the Q table for a D2D pair is initialized to all zeros. In each episode, each agent (i.e., each DU pair) selects RBs and power levels simultaneously, and then receives a reward and updates its Q table. The learning process continues until all Q tables converge to the same optimal values.

In each episode, an action is selected based on an ε-greedy strategy, which is described as follows:

- Select a random action with a probability ε;
- Select an action according to the maximum Q value of the current state with a probability $(1 - \varepsilon)$.

Here, ε is the threshold of the probability, which decays with the number of episodes as

$$\varepsilon = \varepsilon_{\min} + (\varepsilon_{\max} - \varepsilon_{\min}) \cdot \exp(-h \cdot t), \tag{10}$$

where $\varepsilon_{\max}$ and $\varepsilon_{\min}$ denote the upper limit and the lower limit of ε; h is a decay rate within [0, 1]; t is the index of the current episode. At the beginning of learning, ε is set to a value close to 1. Thus, the agent is likely to select a random action that it has not selected before to find more new states of the environment. As the learning process continues, the value of ε decreases accordingly, and thus the agent relies more on the learned policy. The ε-greedy strategy helps an agent explore more states and actions at the beginning of the learning process so that the convergence of Q learning can be ensured [9]. After an action is selected, the Q table is updated based on the following function:

$$Q_{t+1}^{j}(s,a) = \max\{Q_t^j(s,a),\ r_{t+1} + \gamma \max_{a' \in \mathcal{A}} Q_t^j(s',a')\}, \tag{11}$$

where the Q value for D_j in the $(t + 1)$th episode $Q_{t+1}^{j}(s,a)$ will be updated only when the newly arrived Q value exceeds $Q_{t+1}^{j}(s,a)$ and γ is the discount factor which varies from 0 to 1. The newly arrived Q value corresponds to the second item in Eq. (11), i.e.,

$$r_{t+1} + \gamma \cdot \max_{a \in \mathcal{A}} Q_i^j(s',a'). \tag{12}$$

The pseudo codes of the proposed distributed Q learning based SA-PC algorithm is described in Algorithm 1.

Algorithm 1 Distributed Q-learning based SA-PC algorithm

Input:

$\mathcal{B}$= {B_1, B_2, ..., B_K} {a set of K resource blocks}

$\mathcal{C}$= {C_1, C_2, ..., C_M} {a set of M cellular users}

$\mathcal{D}$= {D_1, D_2, ..., D_N} {a set of N D2D users}

$\mathcal{P}$= {p_1, p_2, ..., p_L} {a set of available power levels}

Output:

$\mathcal{B}_b^* = \{B_{D_1}^*, B_{D_2}^*, \cdots, B_{D_K}^*\}$ {optimal RBs for all DU pairs}

$\mathcal{P}_b^* = \{p_{D_1}^*, p_{D_2}^*, \cdots, p_{D_N}^*\}$ {optimal power levels for all DU pairs}

Function:

$Q_t^j(s,a)$, {Q table for D_j in the tth episode under state s and action a}

Initialize:

for D_j , $j \in \{1,2, ..., N\}$

initialize $Q_t^j(s,a)=0$, $a \in \mathcal{A}$

Learning:

for:

select the rth RB, $r \in \{1, 2,..., K\}$

for:

select D_j, for all the DUs on the rth RB.

for:

select action $a \in \mathcal{A}$ according to the ε greedy strategy

execute a and calculate the reward r_{t+1}

update $Q_t^j(s,a)$ according to Eq. (11)

end for

end for

end for

5 Simulation Results

In this section, we evaluate the performance of the proposed distributed Q learning based SA-PC algorithm through simulation results. The simulation experiment was conducted on a simulator developed using python. We consider a single-cell cellular system where the CUs and DU pairs are uniformly distributed. The parameters used in the simulation experiment are listed in Table 1.

In the performance evaluation, we compare the proposed Q-learning based SA-PC algorithm with a Q-learning based PC algorithm and a random allocation algorithm. The PC algorithm only considers the power control for the DU pairs based on Q learning, while assuming that the RBs are randomly allocated to all DU pairs. In the random allocation algorithm, both RBs and power levels are randomly allocated to all DU pairs. Moreover, we use the system throughput and the D2D throughput as the

Table 1. Simulation parameters

Parameter	Value
M	20
N	10–100
K	20
L	5
Cell radius	500 m
p_1, p_2, p_3, p_4, p_5	{1, 6.5, 12, 17.5, 23} dBm
p_c	24 dBm
Noise power	−116 dBm/Hz
Resource block bandwidth	180 kHz
Gain model between user and BS	15.3 + 37.6 lg(d(km)) dB
Gain model between two users	128 + 40 lg(d(km)) dB
Learning rate α	0.9
Discount factor γ	0.9
τ_0	6 dB

performance metrics. The system throughput is defined as the throughput of all CUs and DU pairs in the system, and the D2D throughput is defined as the throughput of all DU pairs in the system.

Figure 3 compares the convergence of the optimal Q values with the distributed Q learning based SA-PC algorithm under $M = 20$, $N = 10$. It can be observed that all the Q values converge to the same optimal values after around 1200 episodes, which proves that an optimal policy can be obtained for the proposed Q learning algorithm. Meanwhile, the updates of different DU pairs are asynchronous, due to the fact that they select their actions independently during the learning process.

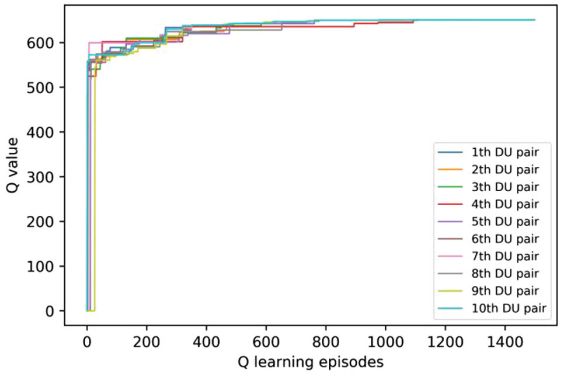
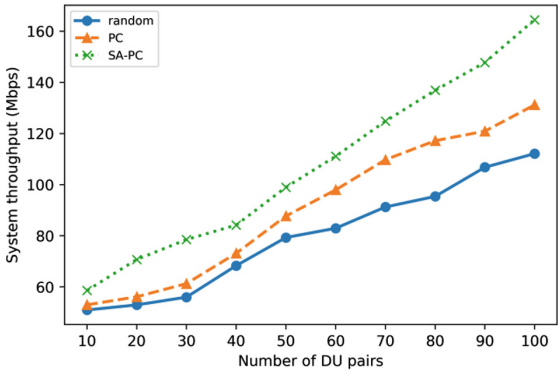
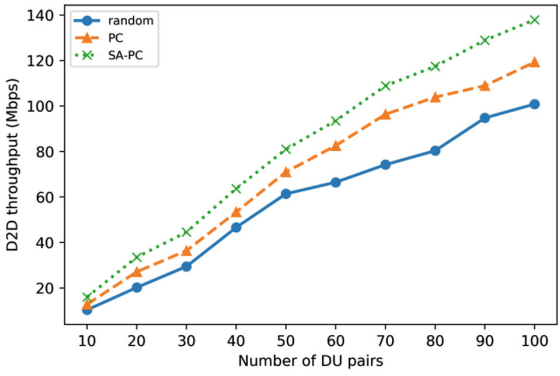


Fig. 3. Convergence of the Q values (M = 20, N = 10)

Figure 4 shows the system throughput and D2D throughput with the proposed SA-PC algorithm, the PC algorithm and the random allocation algorithm, respectively. It is observed that both the system throughput and the D2D throughput increase as the number of DU pairs increases, and all the three algorithms have the same trends over the number of DU pairs. On the other hand, both the system throughput and the D2D throughput with the proposed SA-PC algorithm are larger than those with the PC algorithm and the random allocation algorithm, which demonstrates the superior performance of the proposed SA-PC algorithm.



(a) System throughput



(b) D2D throughput

Fig. 4. Comparison of the throughput performance

6 Conclusion

This paper studied the SA-PC problem in D2D communication underlaying a cellular network. A distributed Q-learning based joint SA-PC algorithm was proposed for performing spectrum allocation and power control for each D2D user in the network. The proposed algorithm uses Q learning, a typical form of RL, to select the optimal RB

and power level for each D2D user. In the Q-learning algorithm, each DU pair is treated as an individual agent and maintains a single-state Q table. Each agent selects an RB and a power level according to its Q table in the learning process. The objective is to select the optimal RB and power level for each D2D user. Simulation results shows that the proposed Q-learning based joint SA-PC algorithm can achieve better performance than a Q-learning based PC algorithm and a random allocation algorithm in terms of the system throughput and the D2D throughput..

References

1. Asadi, A., Wang, Q., Mancuso, V.: A survey on device-to-device communication in cellular networks. IEEE Commun. Surv. Tutor. **16**(4), 1801–1819 (2014)
2. Cai, X., Zheng, J., Zhang, Y., Murata, H.: A capacity oriented resource allocation algorithm for device-to-device communication in mobile cellular networks. In: Proceedings of IEEE ICC 2014, Sydney, Australia, June 2014
3. Chen, B., Zheng, J., Zhang, Y.: A time division scheduling resource allocation for D2D communication in cellular networks. In: Proceedings of IEEE ICC 2015, London, UK, June 2015
4. Cai, X., Zheng, J., Zhang, Y.: A graph coloring based resource allocation algorithm for D2D communication in cellular networks. In: Proceedings of IEEE ICC 2015, London, UK, June 2015
5. Chen, B., Zheng, J., Zhang, Y., Murata, H.: SARA: a service-aware resource allocation scheme for device-to-device communication underlaying cellular networks. In: Proceedings of IEEE Globecom 2014, Austin, USA, December 2014, pp. 4916–4921 (2014)
6. Zulhasnine, M., Huang, C., Srinivasan, A.: Efficient resource allocation for device-to-device communication underlaying LTE networks. In: Proceedings of 2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2010), Niagara Falls, Canada, pp. 11–13, October 2010
7. Esmat, H., Elmesalawy, M., Ibrahim, I.: Adaptive resource sharing algorithm for device-to-device communications underlaying cellular networks. IEEE Commun. Lett. **20**(3), 530–533 (2016)
8. Hsu, C., Chen, W.: Joint power control and channel assignment for green device-to-device communication. In: Proceedings of 2018 16th International Conference on Pervasive Intelligence and Computing (PiCom 2018), Athens, Greece, pp. 881–884 (2018)
9. Luo, Y., Shi, Z., Zhou, X., Liu, Q., Yi, Q.: Dynamic resource allocations based on q-learning for D2D communication in cellular networks. In: Proceedings of 2014 11th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 19–21 December 2014
10. Nie, S., Fan, Z., Zhao, M., Gu, X., Zhang, L.: Q-learning based power control algorithm for D2D communication. In: Proceedings of 2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC 2016), Valencia, Spain, pp. 1–6 (2016)
11. Fan, Z., Gu, X., Nie, S., Chen, M.: D2D power control based on supervised and unsupervised learning. In: Proceedings of 2017 3rd IEEE International Conference on Computer and Communications (ICCC 2017), Chengdu, China, pp. 558–563 (2017)

12. Khan, M.I., Alam, M.M., Le Moullec, Y., Yaacoub, E.: Cooperative reinforcement learning for adaptive power allocation in device-to-device communication. In: Proceedings of 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), Singapore, pp. 476–481 (2018)
13. Sutton, R.S., Barto, A.G.: Reinforcement learning: an introduction. IEEE Trans. Neural Netw. **9**(5), 1054 (1998)
14. Lauer, M., Riedmiller, M.: An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In: Proceedings of 2000 17th International Conference on Machine Learning, San Francisco, CA, pp. 535–542 (2000)