Deep Reinforcement Learning for Joint Channel Selection and Power Control in D2D Networks

Junjie Tan, *Student Member, IEEE*, Ying-Chang Liang, *Fellow, IEEE*, Lin Zhang, *Member, IEEE*, and Gang Feng, *Senior Member, IEEE*

***Abstract*— *Device-to-device* (D2D) technology, which allows direct communications between proximal devices, is widely acknowledged as a promising candidate to alleviate the mobile traffic explosion problem. In this paper, we consider an overlay D2D network, in which multiple D2D pairs coexist on several orthogonal spectrum bands, i.e., channels. Due to spectrum scarcity, the number of D2D pairs is typically more than that of available channels, and thus multiple D2D pairs may use a single channel simultaneously. This may lead to severe co-channel interference and degrade network performance. To deal with this issue, we formulate a joint channel selection and power control optimization problem, with the aim to maximize the *weighted-sum-rate* (WSR) of the D2D network. Unfortunately, this problem is non-convex and NP-hard. To solve this problem, we first adopt the state-of-art *fractional programming* (FP) technique and develop an FP-based algorithm to obtain a near-optimal solution. However, the FP-based algorithm requires instantaneous global *channel state information* (CSI) for centralized processing, resulting in poor scalability and prohibitively high signalling overheads. Therefore, we further propose a distributed *deep reinforcement learning* (DRL)-based scheme, with which D2D pairs can autonomously optimize channel selection and transmit power by only exploiting local information and outdated nonlocal information. Compared with the FP-based algorithm, the DRL-based scheme can achieve better scalability and reduce signalling overheads significantly. Simulation results demonstrate that even without instantaneous global CSI, the performance of the DRL-based scheme can approach closely to that of the FP-based algorithm.**

***Index Terms*— Device-to-device (D2D), channel selection, power control, deep reinforcement learning (DRL), fractional programming (FP).**

Manuscript received July 16, 2019; revised January 11, 2020 and June 26, 2020; accepted October 18, 2020. Date of publication October 29, 2020; date of current version February 11, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61631005, Grant U1801261, and Grant 61801101; in part by the Macau Science and Technology Development Fund (FDCT) under Grant 0009/2020/A1; in part by the Fundamental Research Funds for the Central Universities under Grant ZYGX2019Z022; and in part by the Programme of Introducing Talents of Discipline to Universities under Grant B20064. This article was presented in part at the IEEE Global Communications Conference (GLOBECOM) 2019. The associate editor coordinating the review of this article and approving it for publication was J. Gross. *(Corresponding author: Ying-Chang Liang.)*

Junjie Tan and Ying-Chang Liang are with the Center for Intelligent Networking and Communications (CINC), University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China (e-mail: tan@kuspot.com; liangyc@ieee.org).

Lin Zhang and Gang Feng are with the National Key Laboratory on Communications, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China (e-mail: linzhang1913@gmail.com; fenggang@uestc.edu.cn).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2020.3032991

I. Introduction

A. Background

NOWADAYS, rapidly growing mobile devices have incurred an unprecedented growth in mobile traffic demands, which brings a great challenge to current cellular networks. To meet the challenge, *device-to-device* (D2D) technology has been proposed by allowing direct communications between proximal devices. Compared with conventional cellular technologies, D2D technology achieves two main advantages [2]–[4]. Firstly, D2D technology can effectively alleviate the burden of base stations by traffic offloading. Secondly, D2D technology can enhance spectrum efficiency greatly by shortening the propagation distance of the signals between two devices.

Typically, D2D networks operate by reusing the spectrum of cellular networks, and there exist two main reusing modes in existing literature, namely underlay mode and overlay mode [5]. In the underlay mode, *cellular users* (CUs) and D2D pairs are allowed to transmit concurrently on the same spectrum [6]–[9]. In the overlay mode, a small portion of the cellular spectrum is reserved, and only the D2D pairs are allowed to transmit data on the reserved spectrum [5], [10]. Compared with the underlay mode, the overlay mode can eliminate the interference between the D2D pairs and the CUs due to the orthogonal spectrum partition. Therefore, for better protecting the CUs, we mainly focus on the overlay mode in this paper. Nevertheless, due to the spectrum scarcity, the reserved spectrum for the D2D network is usually rather limited. In other words, the available channels, i.e., the orthogonal spectrum bands, for the D2D network are typically much less than the needs of the D2D pairs, and multiple D2D pairs may use a single channel simultaneously. This could result in severe co-channel interference among D2D pairs and degrade the performance of the D2D network [11]. Hence, it is crucial to allocate proper channels and transmit power for the D2D pairs to enhance the performance of the D2D networks.

In fact, the channel selection and/or power control problem has been extensively studied for D2D networks in recent years. For example, in [12], a joint channel assignment and power control scheme is developed to improve both the overall throughput and energy efficiency of D2D pairs. In [13], the spatial reuse of a D2D network is maximized by optimizing the channel allocation for D2D pairs. In [14] and [15], two game theory based distributed algorithms are developed for the D2D

1536-1276 © 2020 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: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

networks. In particular, [14] improves the individual rate of each D2D pair by optimizing the power allocation, while [15] focuses on maximizing the sum-rate under the heterogenous rate requirements of D2D pairs.

B. Motivations

We notice that most of the existing algorithms on channel selection and/or power control in D2D networks have relatively high requirements on signalling exchanges. Specifically, the centralized optimization methods, e.g., [12] and [13], run with instantaneous global *channel state information* (CSI), which requires massive and instantaneous signalling exchanges. On the other hand, the distributed optimization methods, e.g., [14] and [15], need to repeatedly update some variables among D2D pairs until the convergence of the algorithms. For timeliness, the signallings containing those variables need to be exchanged instantaneously among D2D pairs. Therefore, it is typically infeasible to implement these algorithms in practical situations.

The root cause of the signalling issue is that the existing schemes solve optimization problems for each snapshot of network states, and the obtained solution will not be optimal once network states change, e.g., CSI varies. After carefully analyzing these algorithms, we find that there exist correlations among network information. For example, CSI is typically time-correlated according to some patterns. Furthermore, locally measured interference is determined by the global CSI and the decisions of other D2D pairs, implying that the network information also has spatial correlations. If D2D pairs are aware of these correlation patterns, they can leverage outdated and local information to predict future global network information and to make proper decisions on channel selection and power control accordingly. As a result, signalling overheads can be reduced significantly. However, those correlation patterns are hidden and cannot be used directly.

Fortunately, the emergence of *deep reinforcement learning* (DRL) makes it possible to learn and leverage those correlation patterns. DRL is an effective machine learning technique to solve decision-making problems in a dynamic and uncertain environment. The basic idea behind DRL is to learn hidden environmental patterns by continuously interacting with the environment and analyzing the historical data observed during the interactions. After that, decisions can be optimized according to the learnt patterns. Inspired by this idea, we adopt DRL to address the signalling issue in D2D networks. In the following, we introduce some successful applications of DRL in the area of wireless communications.

C. Related Work on DRL

Conventional *reinforcement learning* (RL) methods, such as Q-learning, can only deal with a small action-state space. DRL overcomes this limitation with the help of *deep neural networks* (DNNs) [16], [17]. Since wireless communication systems are usually sophisticated with numerous adjustable parameters and wireless environments are typically dynamic and stochastic, DRL is particularly suitable to apply in wireless communications [18], [19].

Thus far, there has been some successful examples, especially for channel selection problems [20]–[23] and power control problems [24], [25]. In particular, [20] investigates a dynamic multi-channel access problem, for which DRL is used to learn the variation pattern of channel quality and then select the optimal channel without knowing exact system dynamics. In [22], a channel allocation problem is studied in the context of multibeam satellite systems, and DRL is adopted to minimize the service blocking probability by learning the temporal correlation patterns of beam traffic and user distribution. [23] investigates a joint user association and channel selection problem for *heterogenous networks* (HetNets), where DRL enables multiple users to distributively maximize downlink utility while guaranteeing their own *quality-of-service* (QoS) requirements. In [24], a DRL-based power control algorithm is developed to maximize the overall throughput of a multi-cell cellular network by mitigating inter-cell interference. [25] considers a power control problem in a *cognitive radio* (CR) scenario, where a *secondary user* (SU) adopts DRL to learn the power control policy of a *primary user* (PU), and then to adaptively adjust the transmit power, such that the QoS of both PU and SU can be guaranteed.

In addition, DRL has also been successfully applied in *adaptive modulation and coding* (AMC) [26], *medium access control* (MAC) protocol design [27], wireless caching [28], and *unmanned aerial vehicle* (UAV) communications [29], [30].

D. Our Contributions

In this paper, we consider an overlay D2D network, where multiple D2D pairs coexist on several channels. To avoid severe co-channel interference, we formulate a joint channel selection and power control optimization problem, with the objective to maximize the *weighted-sum-rate* (WSR) of the D2D network. Unfortunately, the problem is difficult to solve globally because it is non-convex and NP-hard. To solve this problem, we first employ *fractional programming* (FP) to develop a near-optimal FP-based algorithm. While FP is a state-of-art optimization method that has been well studied with power control problems, e.g., [31], it is not straightforward to apply it to the considered joint channel selection and power control problem. The reasons are two-fold. Firstly, owing to the coupling of channel selection and power control, the considered problem is much more complicated than the power control problems. Secondly, the optimization of channel selection involves integer programming, which further increases the difficulty and requires new designs of the algorithm. However, as an optimization-based algorithm, the developed FP-based algorithm needs to be deployed in a centralized manner and requires instantaneous global CSI as the inputs, resulting in poor scalability and prohibitively high signalling overheads.

Hence, we further address this issue by developing a DRL-based scheme. Although DRL has been investigated for channel selection and power control separately in the

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

literature, e.g., [20]–[25], there still lacks of a joint design scheme, especially for D2D networks. Since D2D networks suffer complicated mutual interference and have massive CSI, it is challenging to develop an appropriate DRL-based scheme for them to achieve good scalability and reduce signalling overheads. To solve this issue, we first propose a distributed framework to allow D2D pairs to operate autonomously and independently. As such, the computational overheads of each D2D pair will not increase with the network size, achieving a good scalability. Then, we develop a DRL-based algorithm to reduce the signalling overheads by exploiting the aforementioned correlations among network information. In principle, enabled by DRL, each D2D pair can learn the correlation patterns by analyzing local information and partial outdated nonlocal information. By leveraging the learnt patterns, they can infer future global network information to optimize channel selection and transmit power accordingly. Eventually, the DRL-based scheme only requires a small amount of delay-tolerant signalling exchanges for some outdated nonlocal information, which reduces signalling overheads significantly compared with the FP-based algorithm. The main contributions of this work can be summarized as:

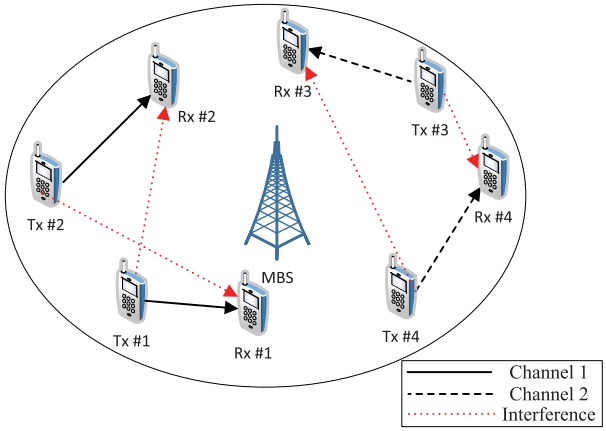
- We develop a centralized FP-based channel selection and power control algorithm, which can provide a near-optimal solution based on the instantaneous global CSI of the D2D network. To our best knowledge, it is the first work to apply FP to the joint channel selection and power control problem, and it can be used as a benchmark algorithm of the problem.
- We propose a distributed DRL-based scheme, which enables each D2D pair to autonomously optimize channel selection and transmit power with only local information and some outdated nonlocal information. Compared with the FP-based algorithm, the DRL-based scheme reduces signalling overheads significantly and has better scalability.
- Simulation results demonstrate that even without instantaneous global CSI, the proposed DRL-based scheme can still achieve approximate performance of the FP-based algorithm.

E. Organization of the Paper

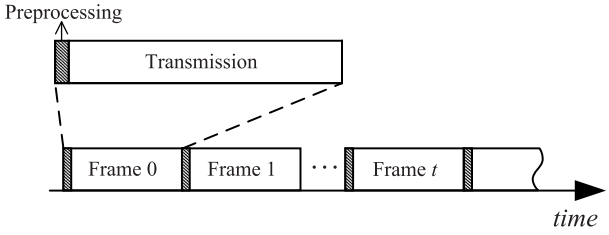
The rest of this paper is organized as follows. We begin with the system model in Section II, and then formulate a joint channel selection and power control problem for WSR maximization in Section III. To solve the problem, we develop an FP-based algorithm and a DRL-based algorithm in Section IV and Section V respectively. In Section VI, we present simulation results to evaluate the performance of the proposed algorithms. Finally, conclusions are drawn in Section VII.

II. System Model

In this paper, we consider an overlay multi-channel D2D network, a sample sketch of which is depicted in Fig. 1a. In particular, M D2D pairs share N channels, and M is typically larger than N, i.e, $M > N$. Each D2D pair is composed of



(a) A sample D2D network with four D2D pairs and two channels.



(b) Frame structure for the D2D network.

Fig. 1. System model.

a *transmitter* (Tx) and a *receiver* (Rx). We denote the set of the M D2D pairs as $\mathcal{M} = \{1, \ldots, M\}$, and denote the set of the N channels as $\mathcal{N} = \{1, \ldots, N\}$. Besides, transmitter m and receiver m are defined as the transmitter and the receiver of D2D pair m respectively. In the following, we illustrate the channel model, the frame structure, and the signal model of the considered network.

A. Channel Model

Each channel between a transmitter and a receiver in the considered network is composed of three parts, namely path loss, shadowing, and Rayleigh fading. In particular, pass loss and shadowing are large-scale components of the channels, and they remain the same for long time, while Rayleigh fading is a small-scale component that is assumed to be block-fading. Jake's model is adopted to describe the time-correlation of Rayleigh fading [32]. Accordingly, if we denote by $g^{(t)}$ and $g^{(t+1)}$ the Rayleigh fading at block t and $t+1$ respectively, we have

$$g^{(t+1)} = \rho g^{(t)} + \sqrt{1-\rho^2}\delta, \tag{1}$$

where $\rho \in [0, 1]$ is the correlation factor accounting for the correlations of the channel realizations in two successive blocks. Besides, both δ and $g^{(0)}$ in (1) are complex Gaussian random variables that follow the distribution $\mathcal{CN}(0, 1)$.

We denote by $\chi_{m,k}$ and $\beta_{m,k}$ the pass loss and the shadowing from transmitter m to receiver k respectively, and denote by $g_{m,k,n}$ the Rayleigh fading from transmitter m to receiver k on channel n. Therefore, at block t, the channel gain between transmitter m and receiver k on channel n can be written as $h^{(t)}_{m,k,n} = \chi_{m,k}\beta_{m,k}\left|g^{(t)}_{m,k,n}\right|^2$.

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

B. Frame Structure and Signal Model

The D2D pairs in the considered network employ the frame structure depicted in Fig. 1b. In particular, each frame is composed of two phases: the preprocessing phase and the transmission phase. During the preprocessing phase, each D2D pair collects necessary information to choose a channel and a transmit power. During the transmission phase, the transmitter of each D2D pair transmits data to the corresponding receiver according to the decision. Cellular-assisted synchronization is adopted, in which a *macro base station* (MBS) broadcasts synchronization signals to synchronize the D2D pairs [33]. For simplicity, the transmissions of frames are assumed to be synchronized with the channel variations, i.e., frame t occupies block t. Besides, the MBS assists to allocate pilots to the D2D pairs orthogonally. The pilot allocation results are broadcasted and thus each D2D pair knows both its own pilot and the pilots of all other D2D pairs.

A D2D pair may suffer from the co-channel interference if there exists another D2D pair using the same channel. We use $\alpha_{m,n}^{(t)}$ to indicate whether D2D pair m transmits on channel n or not in frame t. If yes, we have $\alpha_{m,n}^{(t)}=1$, otherwise, $\alpha_{m,n}^{(t)}=0$. We consider that all transmitters and receivers are equipped with a single antenna. For fairness and simplicity, each D2D pair is considered to choose at most one channel to transmit in a frame, i.e., $\sum_{n\in\mathcal{N}}\alpha_{m,n}^{(t)}\leq 1, \forall m\in\mathcal{M}$, which is also adopted in existing literature, e.g., [34]. If we denote by $p_m^{(t)}$ the transmit power of D2D pair m in frame t, the corresponding received SINR on channel n can be expressed as

$$\gamma_{m,n}^{(t)}=\frac{\alpha_{m,n}^{(t)}h_{m,m,n}^{(t)}p_m^{(t)}}{\sigma^2+\sum\limits_{k\in\mathcal{M},k\neq m}\alpha_{k,n}^{(t)}h_{k,m,n}^{(t)}p_k^{(t)}}, \tag{2}$$

where σ^2 is the *additive white Gaussian noise* (AWGN) power.

III. Problem Formulation

Let w_m be the weight of D2D pair m. We consider to maximize the WSR of the D2D network, for which the channel selection and the transmit power of the D2D pairs should be jointly optimized in each frame. Denoting by $\boldsymbol{p}^{(t)}=\{p_1^{(t)},p_2^{(t)},\ldots,p_M^{(t)}\}\in\mathbb{R}_{1\times M}$ and $\boldsymbol{\alpha}^{(t)}=\{\alpha_{1,1}^{(t)},\alpha_{1,2}^{(t)},\ldots,\alpha_{M,N}^{(t)}\}\in\mathbb{R}_{1\times MN}$ the power control vector and the channel selection vector in frame t respectively, we can formulate the problem as

Problem 1:

$$\max_{\boldsymbol{\alpha}^{(t)},\boldsymbol{p}^{(t)}} \quad \sum_{m=1}^{M}w_m\sum_{n=1}^{N}r_{m,n}^{(t)} \tag{P1.Obj}$$

$$\text{s.t.} \quad 0\leq p_m^{(t)}\leq\bar{P}, \quad \forall m\in\mathcal{M}, \tag{P1.C1}$$

$$\alpha_{m,n}^{(t)}\in\{0,1\}, \quad \forall m\in\mathcal{M},\ \forall n\in\mathcal{N}, \tag{P1.C2}$$

$$\sum_{n=1}^{N}\alpha_{m,n}^{(t)}\leq 1, \quad \forall m\in\mathcal{M}, \tag{P1.C3}$$

where $r_{m,n}^{(t)}$ is the transmission rate achieved by D2D pair m on channel n in frame t and is defined as $r_{m,n}^{(t)}=\log_2\left(1+\gamma_{m,n}^{(t)}\right)$.

In **Problem 1**, (P1.C1) restricts the transmit power of each D2D pair to be nonnegative and no larger than a maximum transmit power $\bar{P}$, while (P1.C2) and (P1.C3) constrain that each D2D pair is allowed to select at most one channel in each frame. For the D2D pairs that have zero transmit power or do not select a channel, they will not be scheduled to transmit. This phenomenon may happen to some D2D pairs with very strong channel gains among them. In this case, part of the D2D pairs will keep silent for the maximization of the WSR.

Unfortunately, it is challenging to solve **Problem 1** directly because it is a non-convex and *mixed integer nonlinear programming* (MINLP) problem. In fact, even for a given $\boldsymbol{\alpha}^{(t)}$, this problem is NP-hard due to the coupling of $\boldsymbol{p}^{(t)}$ [35]. Recently, FP has been proposed and widely acknowledged to solve the power control problem near-optimally [31], i.e., **Problem 1** with a given $\boldsymbol{\alpha}^{(t)}$. Following the idea of FP, we develop an FP-based algorithm to solve **Problem 1**.

IV. Fractional Programming Approach

In **Problem 1**, the constraint (P1.C2) makes the problem an MINLP problem that cannot be well handled by FP. Hence, we first relax the constraint by allowing each element in $\boldsymbol{\alpha}^{(t)}$ to be continuous between 0 and 1. By dropping the time index (t), we can relax **Problem 1** as

Problem 2:

$$\max_{\boldsymbol{\alpha},\boldsymbol{p}} \quad \sum_{m=1}^{M}w_m\sum_{n=1}^{N}r_{m,n} \tag{P2.Obj}$$

$$\text{s.t.} \quad 0\leq p_m\leq\bar{P}, \quad \forall m\in\mathcal{M}, \tag{P2.C1}$$

$$\alpha_{m,n}\geq 0, \quad \forall m\in\mathcal{M},\ \forall n\in\mathcal{N}, \tag{P2.C2}$$

$$\sum_{n=1}^{N}\alpha_{m,n}\leq 1, \quad \forall m\in\mathcal{M}. \tag{P2.C3}$$

Then, we apply the FP technique and propose **Theorem 1** below.

Theorem 1: We first introduce the following auxiliary variables, namely $\boldsymbol{\lambda}=\{\lambda_{1,1},\lambda_{1,2},\ldots,\lambda_{M,N}\}\in\mathbb{R}_{1\times MN}$, $\boldsymbol{\vartheta}=\{\vartheta_{1,1},\vartheta_{1,2},\ldots,\vartheta_{M,N}\}\in\mathbb{R}_{1\times MN}$, $\boldsymbol{\varphi}=\{\varphi_{1,1},\varphi_{1,2},\ldots,\varphi_{M,N}\}\in\mathbb{R}_{1\times MN}$, and $\boldsymbol{\nu}=\{\nu_1,\nu_2,\ldots,\nu_M\}\in\mathbb{R}_{1\times M}^{+}$. Then, $\boldsymbol{\lambda}$, $\boldsymbol{\vartheta}$, $\boldsymbol{p}$, $\boldsymbol{\varphi}$, $\boldsymbol{\nu}$, and $\boldsymbol{\alpha}$ are updated iteratively as follows

1)

$$\lambda_{m,n}=\frac{A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})}{B_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})}, \quad \forall m\in\mathcal{M},\ n\in\mathcal{N}, \tag{3}$$

where $A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})=\alpha_{m,n}h_{m,m,n}p_m$ and $B_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})=\sigma^2+\sum\limits_{k\neq m}h_{k,m,n}p_k\alpha_{k,n}$.

2)

$$\vartheta_{m,n}=\frac{\sqrt{J_{m,n}p_m\alpha_{m,n}}}{\sigma^2+\sum_{k=1}^{M}h_{k,m,n}p_k\alpha_{k,n}}, \quad \forall m\in\mathcal{M},\ n\in\mathcal{N}, \tag{4}$$

where $J_{m,n}=w_m(1+\lambda_{m,n})h_{m,m,n}$.

3)

$$p_m=\min\left\{\bar{P},p_m'\right\}, \quad \forall m\in\mathcal{M}, \tag{5}$$

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

where $p'_m = \left[\frac{\sum_{n=1}^{N}\vartheta_{m,n}\sqrt{J_{m,n}\alpha_{m,n}}}{\sum_{n=1}^{N}\sum_{k=1}^{M}h_{m,k,n}\alpha_{m,n}\vartheta_{k,n}^2}\right]^2$.

4)

$$\varphi_{m,n}=\frac{\sqrt{J_{m,n}p_m\alpha_{m,n}}}{\sigma^2+\sum_{k=1}^{M}h_{k,m,n}p_k\alpha_{k,n}},\quad \forall m\in\mathcal{M},\ n\in\mathcal{N}. \tag{6}$$

5) Bisection search over ν_m to equalize

$$\nu_m\left[1-\sum_{N=1}^{N}\left(\frac{\varphi_{m,n}\sqrt{J_{m,n}p_m}}{\varphi_{m,n}^2h_{m,m,n}p_m+\nu_m}\right)^2\right]=0,\quad \forall m\in\mathcal{M}. \tag{7}$$

6)

$$\alpha_{m,n}=\left(\frac{\varphi_{m,n}\sqrt{J_{m,n}p_m}}{\varphi_{m,n}^2h_{m,m,n}p_m+\nu_m}\right)^2,\quad \forall m\in\mathcal{M},\ n\in\mathcal{N}. \tag{8}$$

Eventually, $\boldsymbol{p}$ and $\boldsymbol{\alpha}$ will converge to a locally optimal solution to **Problem 2**.

Proof: Please refer to Appendix. ■

According to **Theorem 1**, we develop **Algorithm 1** to solve **Problem 2**. Since $\boldsymbol{\alpha}$ has been relaxed in **Problem 2**, the $\boldsymbol{\alpha}$ obtained by **Algorithm 1** should be rounded to satisfy the constraint (P1.C2). A straightforward scheme is to let them select the channel with the largest value in their channel selection vectors. Nevertheless, this scheme may make neighboring D2D pairs select the same channel, resulting in high co-channel interference. To deal with this issue, we propose a successive rounding procedure as shown in Step 2-7 of **Algorithm 2**. In each outer loop, i.e., Step 3-6, each D2D pair successively selects the best channel that maximizes (P1.Obj), given that the channel selection of other D2D pairs is fixed. As such, each outer loop can increase (P1.Obj) monotonically. Since (P1.Obj) is finite for a given $\boldsymbol{p}$, the successive rounding procedure can eventually stop at a set of channel selection with which (P1.Obj) cannot be further improved. After the rounding of $\boldsymbol{\alpha}$, all D2D pairs can be divided into N subsets, each of which occupies a single channel dedicatedly. As the single-channel power control problem has been well solved by the *FP power control* (FPPC) algorithm [31, Algorithm 3], we can further refine the power control vector $\boldsymbol{p}$ by executing the FPPC algorithm respectively for each channel. We summarize the above procedures as the successive rounding and power refinement algorithm shown in **Algorithm 2**. Finally, **Algorithm 1** and **Algorithm 2** constitute the FP-based joint channel selection and power control algorithm.

Since **Problem 1** is defined for an instantaneous frame, the FP approach needs to repeatedly collect instantaneous global CSI and calculate immediate solutions in the practical implementation. The required massive signalling exchanges are challenging to realize. Moreover, the centralized execution also limits its scalability. Thus, we alternatively propose a DRL approach to achieve higher scalability at reduced signalling overheads.

Algorithm 1 The FP Algorithm for Solving **Problem 2**

1: Initialize $\boldsymbol{\alpha}$ and $\boldsymbol{p}$ arbitrarily to satisfy constrains (P2.C1)-(P2.C3).
2: **repeat**
3: Update $\boldsymbol{\lambda}$ according to (3).
4: Update $\boldsymbol{\vartheta}$ according to (4).
5: Update $\boldsymbol{p}$ according to (5).
6: Update $\boldsymbol{\varphi}$ according to (6).
7: Update $\boldsymbol{\nu}$ by solving (7) with bisection search.
8: Update $\boldsymbol{\alpha}$ according to (8).
9: **until** (P2.Obj) converges.

Algorithm 2 The Successive Rounding and Power Refinement Algorithm

1: Obtain $\boldsymbol{\alpha}$ and $\boldsymbol{p}$ after executing **Algorithm 1**.
2: **repeat**
3: **for** $m=1:M$ **do**
4: Compute (P1.Obj) for each $n\in\mathcal{N}$ by letting $\alpha_{m,n}=1$ and $\alpha_{m,i}=0,\forall i\neq n$..
5: Find n^* that maximizes (P1.Obj). Let $\alpha_{m,n^*}=1$ and $\alpha_{m,i}=0,\forall i\neq n^*$.
6: **end for**
7: **until** (P1.Obj) converges.
8: **for** $n=1:N$ **do**
9: Execute the FPPC algorithm for $\{m|\alpha_{m,n}=1,\forall m\in\mathcal{M}\}$.
10: **end for**

V. Deep Reinforcement Learning Approach

In this section, we first propose a distributed framework and illustrate the information acquisition procedure. After that, we introduce a general RL framework, and develop a DRL-based algorithm within the RL framework. Finally, we discuss about the nonstationarity issue and summarize the advantages of the DRL approach over the FP approach.

A. Distributed Framework and Information Acquisition

In order to yield a scalable scheme, we first propose a distributed framework, in which each D2D pair autonomously makes decisions on channel selection and power control. As such, by distributing computations to each D2D pair, the individual computational overheads will not increase as the network scales up. However, in theory, each D2D pair still requires plenty of network information to solve **Problem 1**.

To analyze the information flow in the network, we categorize the network information into local information and nonlocal information according to whether it can be obtained directly by the transmitter or receiver of a D2D pair. In particular, the time for obtaining local information can be negligible due to the direct link within each D2D pair. In contrast, the signallings among D2D pairs are exchanged via a common control channel, which causes additional delay to the acquisition of nonlocal information [24]. For simplicity, we assume that it takes the time duration of a frame for a D2D pair to fetch the required nonlocal information. Since a D2D

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

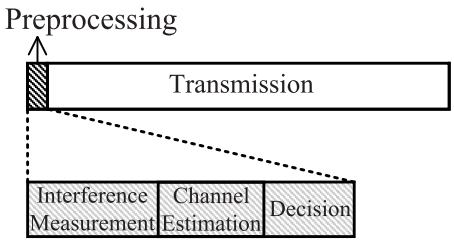


Fig. 2. The details of the preprocessing phase of a frame.

pair is mostly affected by its neighboring D2D pairs, signalling exchanges can be reduced if each D2D pair only considers U neighboring D2D pairs. We define two kinds of neighbors respectively from the angle of transmitters and receivers, i.e., interfering or being interfered. The receiver-neighbors of D2D pair m contain U D2D pairs with the largest large-scale component in the channels from their transmitters to receiver m, while transmitter-neighbors of D2D pair m contain U D2D pairs with the largest large-scale component in the channels from transmitter m to their receivers. The receiver-neighbor set and the transmitter-neighbor set of D2D pair m are denoted by $\mathcal{M}^m_{R,U}$ and $\mathcal{M}^m_{T,U}$, respectively. Upon joining the network, the D2D pairs perform *channel estimation* (CE) for multiple times in a period of time and take the average to estimate the large-scale components of the channels among them. After that, the neighbor sets can be determined accordingly.

Next, we refine the frame structure design and take D2D pair m in frame t for example to describe the information acquisition procedure. As Fig. 2 shows, we divide the preprocessing time into three sub-phases, namely the *interference measurement* (IM) phase, the CE phase, and the decision phase. During the IM phase, D2D pairs transmit with previous decisions although CSI has already changed. Then, D2D pair m measures the received interference on all channels, i.e., $\{\sigma^2 + \sum_{j\neq m} \alpha^{(t-1)}_{j,n} p^{(t-1)}_j h^{(t)}_{j,m,n} | \forall n \in \mathcal{N}\}$. During the CE phase, each D2D pair transmits pilots on all channels. For simplicity, we assume perfect CE here. In this way, D2D pair m obtains the local CSI between its own transmitter and receiver, denoted by $\boldsymbol{h}^{(t)}_m = \{h^{(t)}_{m,m,n} | \forall n \in \mathcal{N}\}$, as well as the CSI from the transmitters of the D2D pairs in $\mathcal{M}^m_{R,U}$ to receiver m, denoted by $\{\boldsymbol{h}^{(t)}_{k,m} | \forall k \in \mathcal{M}^m_{R,U}\}$, where $\boldsymbol{h}^{(t)}_{k,m} = \{h^{(t)}_{k,m,n} | \forall n \in \mathcal{N}\}$. By listening to the control channel, at the beginning of the decision phase, each D2D pair obtains the outdated nonlocal information that its neighbors transmitted in the previous frame. In particular, the outdated information that D2D pair m fetches from a neighboring D2D pair j in $\mathcal{M}^m_{T,U}$ includes weight w_j, transmit power $p^{(t-1)}_j$, channel $o^{(t-1)}_j$, transmission rate $C^{(t-1)}_j$, local CSI $\boldsymbol{h}^{(t-1)}_j$, and CSI from transmitter m to receiver j, denoted by $\boldsymbol{h}^{(t-1)}_{m,j}$. Similarly, D2D pair m also fetches weight w_k, $p^{(t-1)}_k$, $C^{(t-1)}_k$, and $\boldsymbol{h}^{(t-1)}_k$ from D2D pair k in $\mathcal{M}^i_{R,U}$. Since weights are constant, they are only exchanged once. Then, the D2D pairs make decisions and broadcast the information required by their neighbors on the control channel using some token mechanisms [24].

From the above, the information available to each D2D pair can only provide a local and outdated view of the network, which makes traditional optimization methods inapplicable. Fortunately, there exist some correlations among the network information. For example, the CSI and the decisions of D2D pairs are time-correlated. Besides, the locally measured interference is also correlated with the CSI and decisions involving all D2D pairs, implying the existence of spatial correlations in the information. Although these correlation patterns are not exactly known to the D2D pair, they are typically hidden in the historical data. It is possible for the D2D pairs to learn the hidden patterns by analyzing historical data. With the learnt knowledge, they can infer future global network information to optimize channel selection and transmit power proactively.

Thanks to the powerful representation capability of DNNs, DRL can efficiently learn the correlation patterns hidden in historical data and use the learnt knowledge to make decisions in a dynamic environment. Thus, we apply DRL to enable each D2D pair to learn the required knowledge and make the optimal decisions. Since DRL stems from RL, we first introduce a general RL framework in the next subsection.

B. General RL Framework

The core of RL is to learn the optimal decision-making policy in a dynamic environment by trial and error. In RL, the decision maker is called the agent. Generally, the interactions between the agent and the environment can be modeled as a *Markov decison process* (MDP). Next, we provide a summary of the basic elements included in an MDP.

- Action space: Each decision that the agent is allowed to make is called an action. All the available actions constitute the action space, which is represented by $\mathcal{A}$.
- State space: States are defined as the observations of the environmental status by the agent. The state space, denoted as $\mathcal{S}$, is composed of all the possible states.
- Transition probabilities: Transition probabilities describe the changes of the environmental status during the interactions with the agent. Supposing that the agent observes the state to be s and makes a decision a at a step, the probability that the state becomes s' at the next step is defined as the transition probability $P_a(s, s')$.
- Reward: Once an action is taken, the environment feeds back a reward to score how good the designed goal has been achieved. Assuming that the agent takes action a at state s, the resulted reward is denoted as $d_a(s)$.
- Policy: Policy π is the decision rule of the agent. Denote by $\pi(s, a)$ the probability of the agent taking action a on the condition of state being s, where we have $a \in \mathcal{A}$, $s \in \mathcal{S}$, and $\sum_{a\in\mathcal{A}} \pi(s, a) = 1, \forall s \in \mathcal{S}$.

RL aims to obtain an optimal policy to maximize the long-term discounted cumulative reward

$$D = \sum_{t=0}^{\infty} \tau^t d_{a_t}(s_t), \tag{9}$$

where $\tau \in [0, 1)$ is the discount factor, representing the importance of future rewards; s_t denotes the state at step t; a_t denotes the action taken by the agent at step t according to the policy π. Hence, RL can be seen as a tool to solve the optimal control problem of the MDP. According to [16], if the

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

transition probabilities among the states in an MDP are fully known, we can apply *dynamic programming* (DP) to analyze the MDP. However, it is generally difficult to acquire the transition probabilities. Alternatively, model-free RL methods can be applied, and Q-learning is the most popular technique among them.

In Q-learning, Q-values are used to estimate the expected accumulative reward of every state-action pair $\langle s, a\rangle$ under policy π, denoted by $\boldsymbol{Q}^{\pi}(s,a)$. Assuming that policy π^* maximizes the long-term reward, the corresponding Q-values should be the highest for the same state-action pairs. From Bellman's equation, we have

$$\boldsymbol{Q}^{\pi^*}(s,a) = d_a(s) + \tau \sum_{s' \in \mathcal{S}} P_a(s,s') \max_{a'} \boldsymbol{Q}^{\pi^*}(s',a'). \quad (10)$$

Since transition probabilities are unavailable to the agent, we cannot derive $\boldsymbol{Q}^{\pi^*}$ and π^* directly from (10). Alternatively, Q-learning obtains them in an iterative manner, where the RL agent continuously updates the Q-values and takes a new action to interact with the environment. Specifically, if s_t and a_t denote the state and the adopted action at the step t respectively, the update equation of the Q-values is given by

$$\boldsymbol{Q}(s_t,a_t) = (1-\eta)\boldsymbol{Q}(s_t,a_t) + \eta \left[d_{a_t}(s_t) + \tau \max_{a \in \mathcal{A}} \boldsymbol{Q}(s_{t+1},a) \right], \quad (11)$$

where η stands for the learning rate. Having the updated Q-values, the agent takes a new action by ε-greedy strategy. In particular, the agent chooses the action with the highest Q-value at the probability of $1-\varepsilon$ and takes a random action at the probability of ε. The ε-greedy strategy enables the agent to explore new but potentially better actions to escape from local optima. The policy obtained by Q-learning will eventually converge to the globally optimal policy π^* [16].

C. DRL-Based Algorithm

In this part, we develop a DRL-based algorithm to solve the studied problem. Within the proposed distributed framework, each D2D pair operates independently by treating other D2D pairs as a part of its own environment. Hence, each D2D pair is modeled as an agent. Next, we map the key elements from the RL framework to our problem by taking D2D pair m for example. In the sequel, the terms "agent" and "D2D pair" are used interchangeably for convenience.

1) Action Space: During the decision phase of each frame, the agent should decide whether or not to transmit in the current frame. Further, if the agent chooses to transmit, it needs to select a channel and a transmit power. By discretizing the maximum transmit power $\bar{P}$ into L levels, we can define the action space $\mathcal{A}$ of the agent as

$$\mathcal{A} = \left\{ \{o,p\} \,\middle|\, \forall o \in \mathcal{N}, p \in \left\{ \frac{\bar{P}}{L}, \frac{2\bar{P}}{L} \ldots, \bar{P} \right\} \right\} \cup \{n_0, p_0\}, \quad (12)$$

where $\{n_0, p_0\}$ denotes that the agent does not transmit. Thus, the dimension of the action space is $NL+1$. We denote by $a_m^{(t)} = \{o_m^{(t)}, p_m^{(t)}\}$ the action taken by agent m in frame t, where $o_m^{(t)}$ and $p_m^{(t)}$ are the selected channel and transmit power respectively.

2) Reward: For the distributive and autonomous operation of D2D pairs, the global objective function, i.e, the global WSR shown in (P1.Obj), needs to be decoupled into individual reward functions. A feasible design is proposed in [24] for a single-channel power control problem. Inspired by its success, we extend it to the multi-channel scenario considered in this paper. To be specific, for D2D pair m in frame t, its individual weighted rate is defined as $w_m C_m^{(t)}$, where

$$C_m^{(t)} = \sum_{n=1}^{N} r_{m,n}^{(t)} = \log_2 \left(1 + \frac{h^{(t)}_{m,m,o_m^{(t)}} p_m^{(t)}}{\sigma^2 + \sum\limits_{j \neq m} \alpha^{(t)}_{j,o_m^{(t)}} h^{(t)}_{j,m,o_m^{(t)}} p_j^{(t)}} \right), \quad (13)$$

and its resulted WSR reduction is defined as $\sum\limits_{k \in \mathcal{M}^m_{T,U}} w_k (C^{(t)}_{k\backslash m} - C^{(t)}_k)$, where

$$C^{(t)}_{k\backslash m} = \log_2 \left(1 + \frac{\alpha^{(t)}_{k,o_k^{(t)}} h^{(t)}_{k,k,o_k^{(t)}} p_k^{(t)}}{\sigma^2 + \sum\limits_{j \neq m,k} \alpha^{(t)}_{j,o_k^{(t)}} h^{(t)}_{j,k,o_k^{(t)}} p_j^{(t)}} \right). \quad (14)$$

Term $(C^{(t)}_{k\backslash m} - C^{(t)}_k)$ is defined as the rate reduction that D2D pair m causes to D2D pair k, and it quantifies the negative effects on the rate of D2D pair k due to the interference from D2D pair m. In other words, if D2D pair m has larger interference to D2D pair k, the resulted rate reduction is higher. From the above, we can design the reward of agent m in frame t as

$$d_m^{(t)} = w_m C_m^{(t)} - \sum_{k \in \mathcal{M}^m_{T,U}} w_k^{(t)} (C^{(t)}_{k\backslash m} - C^{(t)}_k), \quad (15)$$

which can be calculated locally. In particular, the individual weighted rate can be obtained directly based on its own throughput measured during the transmission phase, while the WSR reduction caused to neighboring agents can be calculated based on the acquired information illustrated in Section V-A, despite some delay. For example, in frame t, after fetching w_k, $p_k^{(t-1)}$, $o_k^{(t-1)}$, $C_k^{(t-1)}$, $\boldsymbol{h}_k^{(t-1)}$, and $\boldsymbol{h}_{m,k}^{(t-1)}$ from agent $k \in \mathcal{M}^m_{T,U}$, agent m can derive $C^{(t-1)}_{k\backslash m}$ according

$$C^{(t-1)}_{k\backslash m} = \log_2 \left(1 + \frac{\alpha^{(t-1)}_{k,o_k^{(t-1)}} h^{(t-1)}_{k,k,o_k^{(t-1)}} p_k^{(t-1)}}{\frac{2^{C_k^{(t-1)}}-1}{\alpha^{(t-1)}_{k,o_k^{(t-1)}} h^{(t-1)}_{k,k,o_k^{(t-1)}} p_k^{(t-1)}} - \alpha^{(t-1)}_{m,o_k^{(t-1)}} h^{(t-1)}_{m,k,o_k^{(t-1)}} p_m^{(t-1)}} \right) \quad (16)$$

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

to (16), shown at the bottom of the previous page, with which $w_k(C_{k\backslash m}^{(t-1)} - C_k^{(t-1)})$ can be obtained. By applying similar procedures to other transmitter-neighbors, agent m can obtain its resulted weight sum-rate reduction in frame $t-1$. Hence, rewards are only available to the agents after the delay by a frame.

3) State Space: As the basis for decision-making, states should provide enough information for agents to maximize their rewards. By analyzing (15), we can decompose the reward function of an agent into following components: its own received signal power, its own total received interference plus noise, its interference caused to transmitter-neighbors, and the received signal power and total received interference plus noise of transmitter-neighbors. Particularly, the total interference plus noise received by the agent mainly consists of the interference from its receiver-neighbors. Among all the components, during the decision phase of a frame, the agent can only know accurately its own received signal power of the current frame, because the others involve the unavailable nonlocal information. For example, the interference that an agent causes to its transmitter-neighbors or receives from its receiver-neighbors is determined by the nonlocal CSI and the decisions of other agents, and they all are unavailable to the agent until the next frame.

Fortunately, as mentioned before, the temporal and spatial correlations in the network information make it possible for an agent to predict the required information based on local and outdated information. In particular, future CSI can be predicted based on the outdated CSI, while the future decisions of other agents can be predicted by analyzing their historical behaviours. Having the prediction, each agent can make appropriate decisions to maximize their own rewards. Hence, we design states to include not only the present information that provides accurate description about the current frame, but also the outdated information that helps to predict.

For clearness, we define three sets of information available to agent m in frame t, including the local information, denoted by $\boldsymbol{\phi}_{\mathrm{L},m}^{(t)}$, the nonlocal information from agent k in $\mathcal{M}_{R,U}^{i}$, denoted by $\boldsymbol{\phi}_{\mathrm{R},m,k}^{(t)}$, and the nonlocal information from agent j in $\mathcal{M}_{T,U}^{i}$, denoted by $\boldsymbol{\phi}_{\mathrm{T},m,j}^{(t)}$, as follows.

- $\boldsymbol{\phi}_{\mathrm{L},m}^{(t)}$: $\boldsymbol{h}_m^{(t)}, \{\boldsymbol{h}_{k,m}^{(t)} | \forall k \in \mathcal{M}_{R,U}^m\}, p_m^{(t-1)}, o_m^{(t-1)}, C_m^{(t-1)}, \{\sigma^2 + \sum_{i\neq m} \alpha_{i,n}^{(t-1)} p_i^{(t-1)} h_{i,m,n}^{(t)} | \forall n \in \mathcal{N}\}, p_m^{(t-1)} h_{m,m,o_m^{(t-1)}}^{(t-1)}, \sigma^2 + \sum_{i\neq m} \alpha_{m,o_m^{(t-1)}}^{(t-1)} p_i^{(*)(t-1)} h_{i,m,o_m^{(t-1)}}^{(t-1)}$.
- $\boldsymbol{\phi}_{\mathrm{R},m,k}^{(t)}$: $p_k^{(t-1)}, o_k^{(t-1)}, C_k^{(t-1)}, C_{m\backslash k}^{(t-1)}, \boldsymbol{h}_k^{(t-1)}, \alpha_{k,o_m^{(t-1)}}^{(t-1)} p_k^{(*)(t-1)} h_{k,m,o_m^{(t-1)}}^{(t-1)}, \sigma^2 + \sum_{i\neq k} \alpha_{i,o_k^{(t-1)}}^{(t-1)} p_i^{(*)(t-1)} h_{i,k,o_k^{(t-1)}}^{(t-1)}$.
- $\boldsymbol{\phi}_{\mathrm{T},m,j}^{(t)}$: $p_j^{(t-1)}, o_j^{(t-1)}, C_j^{(t-1)}, C_{j\backslash m}^{(t-1)}, \boldsymbol{h}_{m,j}^{(t-1)}, \boldsymbol{h}_j^{(t-1)}, \alpha_{m,o_j^{(t-1)}}^{(t-1)} p_m^{(*)(t-1)} h_{m,j,o_j^{(t-1)}}^{(t-1)}, \sigma^2 + \sum_{i\neq j} \alpha_{i,o_j^{(t-1)}}^{(t-1)} p_i^{(*)(t-1)} h_{i,j,o_j^{(t-1)}}^{(t-1)}$.

The above items without $(*)$ can be obtained directly via the information acquisition procedure described in Section V-A, while the others marked by $(*)$ can be derived based on the acquired information. For WSR maximization, weights should also be put into the state. In particular, we define $\boldsymbol{\omega}_m$ for agent m to be $\boldsymbol{\omega}_m = \{w_m, \{w_k | k \in \mathcal{M}_{R,U}^m\}, \{w_j | k \in \mathcal{M}_{T,U}^m\}\}$ and put it into the states. Intuitively, more historical data may help the agent to learn patterns and make prediction more easily. Thus, we design states to cover the information obtained in several frames. With trace-back depth ς, the state of agent m in frame t is given by (17), shown at the bottom of the page.

Note that each agent has an infinitely large state space due to the continuous variables in (17). If we adopt the Q-learning algorithm, we need to establish a table to store an infinitely large number of Q-values. This is obviously infeasible and makes Q-learning inapplicable.

To deal with the issue, DRL is proposed to use $\boldsymbol{Q}(s,a;\boldsymbol{\theta})$ to approximate the Q-value $\boldsymbol{Q}(s,a)$ by a DNN, where $\boldsymbol{\theta}$ is the weights of the DNN. The established DNN is also called *deep Q-network* (DQN). Denote by $\boldsymbol{\theta}^*$ the optimal weights of the DQN. Then, $\boldsymbol{Q}(s,a;\boldsymbol{\theta}^*)$ represents the maximum expected long-term reward for state-action pair $\langle s,a\rangle$. To obtain the optimal weights $\boldsymbol{\theta}^*$ and the corresponding Q-values, DRL adopts an interactive procedure similar to Q-learning. In particular, the DRL agent first updates $\boldsymbol{\theta}$ with a historical experience. We define an experience of agent m in frame t as a tuple $e_m^{(t)} = \left\langle s_m^{(t)}, a_m^{(t)}, d_m^{(t)}, s_m^{(t+1)} \right\rangle$. As explained before, $d_m^{(t-1)}$ is not available until frame t. Hence, agent m constructs experience $e_m^{(t-1)}$ at the end of frame t. With experience $e_m^{(t-1)}$, the DRL agent can update the $\boldsymbol{\theta}$ of frame t, denoted by $\boldsymbol{\theta}^{(t)}$, by minimizing the loss function $L(\boldsymbol{\theta})$, i.e.,

$$\boldsymbol{\theta}^{(t+1)} = \arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \left[y_t^{\text{target}} - \boldsymbol{Q}(s_m^{(t-1)}, a_m^{(t-1)}; \boldsymbol{\theta}) \right]^2, \tag{18}$$

where

$$y_t^{\text{target}} = d_m^{(t-1)} + \tau \max_a \boldsymbol{Q}(s_m^{(t)}, a; \boldsymbol{\theta}^{(t)}). \tag{19}$$

Then, the agent estimates the Q-values $\boldsymbol{Q}(s,a;\boldsymbol{\theta}^{(t+1)})$, takes a new action $a_m^{(t+1)}$ by the ε-greedy strategy, and interacts with the environment to receive a new experience. By repeating above procedures until the convergence of the weights $\boldsymbol{\theta}$, the DRL agent can obtain the optimal policy.

Apart from the basic DRL framework, "experience replay" and "quasi-static target network" are adopted to enhance the performance [17]. In "experience replay", a memory $\mathcal{R}$ with a maximum size of R experiences is created to store historical

$$s_m^{(t)} = \left\{ \boldsymbol{\omega}_m, \boldsymbol{\phi}_{\mathrm{L},m}^{(t)}, \left\{ \boldsymbol{\phi}_{\mathrm{R},m,k}^{(t)} | k \in \mathcal{M}_{R,U}^m \right\}, \left\{ \boldsymbol{\phi}_{\mathrm{T},m,j}^{(t)} | j \in \mathcal{M}_{T,U}^m \right\}, \ldots, \boldsymbol{\phi}_{\mathrm{L},m}^{(t-\varsigma+1)}, \left\{ \boldsymbol{\phi}_{\mathrm{R},m,k}^{(t-\varsigma+1)} | k \in \mathcal{M}_{R,U}^m \right\}, \left\{ \boldsymbol{\phi}_{\mathrm{T},m,j}^{(t-\varsigma+1)} | j \in \mathcal{M}_{T,U}^m \right\} \right\} \tag{17}$$

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

Algorithm 3 The DRL-Based Channel Selection and Power Control Algorithm for Agent m

Create a trained DQN and a target DQN with weights $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ respectively. Initialize $\boldsymbol{\theta}$ randomly, let $\boldsymbol{\theta}' = \boldsymbol{\theta}$ and $t = 1$.
repeat
  if $t - 1 < R$ or $rand() < \epsilon$ **then**
    Randomly select an action $a_m^{(t)} \in \mathcal{A}$.
  else
    Select the action $a_m^{(t)} = \arg\max_a \boldsymbol{Q}(s_m^{(t)}, a; \boldsymbol{\theta}^{(t)})$.
  end if
  Obtain $s_m^{(t+1)}$ by (17).
  if $t > 1$ **then**
    Obtain $d_m^{(t-1)}$ and $e_m^{(t-1)}$. Insert $e_m^{(t-1)}$ into $\mathcal{R}$.
  end if
  if $t - 1 \geq R$ **then**
    Sample $\mathcal{B}$ from $\mathcal{R}$. Minimize (20) to obtain $\boldsymbol{\theta}^{(t+1)}$.
  end if
  Let $t = t + 1$.
  if $t \bmod K == 0$ **then**
    Let $\boldsymbol{\theta}' = \boldsymbol{\theta}^{(t)}$.
  end if
until $t > t_{\max}$

experiences. For each training, b experiences are sampled from $\mathcal{R}$ to be a minibatch $\mathcal{B}$. In "quasi-static target network", a target DQN is created to predict the target values. The target DQN is synchronized with the DQN under training every K frames. With the modifications, (18) and (19) can be respectively replaced by

$$\boldsymbol{\theta}^{(t+1)} = \arg\min_{\boldsymbol{\theta}} \frac{1}{b} \sum_{e \in \mathcal{B}^{(t)}} \left[y_e^{\text{target}} - \boldsymbol{Q}(s_e, a_e; \boldsymbol{\theta}) \right]^2, \quad (20)$$

and

$$y_e^{\text{target}} = d_e + \tau \max_{a'} \boldsymbol{Q}(s'_e, a'; \boldsymbol{\theta}'), \quad (21)$$

where $\boldsymbol{\theta}'$ denotes the weights of the target DQN; $\mathcal{B}^{(t)}$ denote the minibatch sampled in frame t; $e = \langle s_e, a_e, d_e, s'_e \rangle$ is a sample experience in the minibatch.

The proposed DRL-based algorithm is summarized in **Algorithm 3**, by taking agent m for example. In particular, $rand()$ generates a random number in $[0, 1]$ according to a standard uniform distribution; $t_{\max}$ is the total number of the frames; $\bmod$ denotes the remainder operator.

Similar to existing DRL-based schemes, e.g., [36], the proposed DRL approach is an online algorithm, which can be deployed directly to interact with an unknown environment and learn from the interactions. After the algorithm achieves convergence, the DQN has already been well trained to learn the patterns of the environment and is able to handle the environmental states that did not appear before. Thus, in practical implementations, training will be turned off to reduce overheads after the convergence. Then, using the well trained DQN, each agent can make proper decisions to maintain high performance without further training.

D. Discussions About the Nonstationarity

Since each D2D pair takes actions autonomously and simultaneously, the developed DRL-based algorithm is a *multi-agent RL* (MARL) algorithm. Generally speaking, in MARL algorithms, the environmental dynamics of a single agent are nonstationary because they are affected by other agents [37]. For example, in **Algorithm 3**, the state transitions and received rewards of any D2D pair are determined by both its own action and the joint actions of all other D2D pairs. Since classical RL or DRL algorithms are designed for single-agent scenarios with a stationary environment, they may suffer instability and nonconvergence in multi-agent scenarios without special improvements. It is still an ongoing research topic to address this nonstationarity issue.

Conventionally, the analysis of the interactions among multiple agents falls into the study of game theory, which guarantees that the agents can converge to an equilibrium with a well-designed reward function under given rules, e.g., best response. However, the application of game theory typically requires a static environment, e.g., fixed CSI for the considered D2D scenario, and assumes the agents to take actions in turn based on the full knowledge of other agents. Because of these assumptions, game theory cannot be directly used to develop MARL algorithms, but it still provides some inspirations to address the nonstationarity issue. Firstly, the agents should be capable to know how other agents behave and affect their own environments. In other words, the agents need to have a full picture of their own environmental dynamics. This can be achieved by cooperation among the agents, in the form of centralized training [24], centralized decision-making [36], and shared observations [23], etc. Secondly, game theory provides the reward functions design with better convergence. For example, [23] and [24] design the reward function to include a profit term and a cost term within a game theoretical framework.

In this paper, we also address the nonstationarity issue from the two aforementioned aspects when developing the DRL-based algorithm. Firstly, we let adjacent D2D pairs share their historical observations and decisions, and include them in states. As such, the D2D pairs can fully understand and adapt to their own environmental dynamics. Secondly, we adopt a reward function similar to the one used in [24], and it has a game theoretical explanation that each D2D pair obtains its individual weighted rate as a profit and pays the cost of the resulted WSR reduction.

E. Advantages of the DRL Approach

Instead of aiming to maximize the instantaneous WSR, the DRL approach aims to improve the WSR from a long-term perspective by treating the problem as a sequential decision-making problem. As such, the correlations of the information can be exploited to reduce signalling overheads. To show the appealing advantages of the DRL approach, we compare the FP and DRL approaches in terms of the required computations and signallings.

- **Computations**: The FP-based algorithm is a centralized algorithm, of which the number of variables and

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

computation complexity grow rapidly with the network size. As for the DRL approach, each D2D pair uses the same amount of steps and data to train and make decisions, and thus the computation overheads at each D2D pair remain fixed for any network size.

- **Signallings**: The FP approach requires to collect instantaneous CSI from all D2D pairs and send back the channel selection and power control results during the preprocessing phase. Hence, the signallings to be exchanged for each D2D pair include MN channel gains and one result, which grow rapidly as the network size scales up. Moreover, the signallings need to be exchanged in a short time, which is challenging to realize. As for the DRL approach, from Section V-A, each D2D pair requires to receive at most $U(10+3N)$ messages, which remain the same with a fixed U for any network size. More importantly, the signallings exchanges can be delayed, and thus it is feasible for practical implementations.

Hence, compared with the FP approach, the DRL approach achieves more satisfactory scalability and higher feasibility.

VI. Simulation Results

In this section, we conduct simulations to examine the proposed schemes. In the simulations, D2D pairs are considered to be randomly located in a 500m×500m plain. For each D2D pair, the receiver is placed randomly around the corresponding transmitter between 10m and 50m. The maximum transmit power $\bar{P}$ and noise power σ are respectively set to be 20 dBm and -114 dBm. The path loss is determined by $-120.9-37.6\log_{10}(dist)$ dB, where $dist$ is the distance in km, and the shadowing is determined by an 8 dB log-normal distribution. The channel correlation time is 20 ms and the channel correlation factor ρ is 0.6. Except where noted, by default we set the number of channels N and that of D2D pairs M respectively to be 2 and 20, and we consider sum-rate maximization by letting $w_m = 1, \forall m \in \mathcal{M}$. Similar to [24], we consider the practical effects of finite-precision digital processing, which truncates the received SINR by a maximum value of 30 dB. Each simulation result is an average of the results obtained from 10 independent experiments with random initialization, including topology and channel gains.

A. Evaluation of the FP-Based Algorithm

As explained before, the studied joint channel selection and power control problem is an NP-hard problem, of which the globally optimal solution is difficult to obtain. As a result, we cannot compare the performance of the proposed FP-based algorithm with the globally optimal solution. Fortunately, we notice that the single-channel power control problem can be solved near-optimally by the FPPC algorithm. Hence, we evaluate the FP-based algorithm by considering two alternative suboptimal algorithms based on the FPPC algorithm. The first algorithm is the *exhaustive search* (ES)-FPPC algorithm, in which all possible channel selections are enumerated to find the one with which the FPPC algorithm maximizes the WSR. The second algorithm is the *random channel selection* (RCS)-FPPC algorithm, which executes the FPPC algorithm after allocating a random channel to each D2D pair. Note that the ES-FPPC algorithm is a quasi-optimal solution, but it requires the FPPC algorithm to be executed for N^M times, which results in prohibitively high computational complexity if either N or M is large. For this reason, we conduct small-scale simulations, where each algorithm is executed with M being $2 \sim 10$ respectively for 500 frames. The achieved average sum-rates are depicted in Fig. 3. From the figure, for a small M, i.e., $M = 2 \sim 4$, the FP-based algorithm reaches around $99 \sim 94\%$ of the sum-rates achieved by the ES-FPPC algorithm. As M further increases, this value slightly decreases but still remains over 91%. Meanwhile, the FP-based algorithm always outperforms the RCS-FPPC algorithm. Thus, the near-optimality of the FP-based algorithm can be demonstrated, and it can be used as a benchmark algorithm to evaluate the DRL-based algorithm.

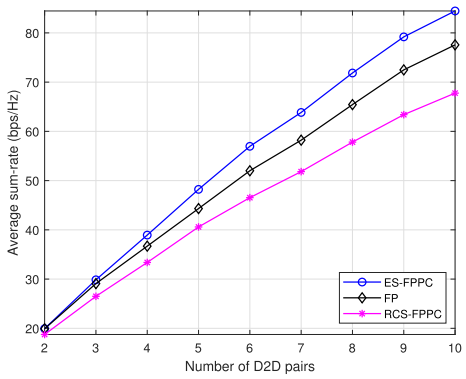


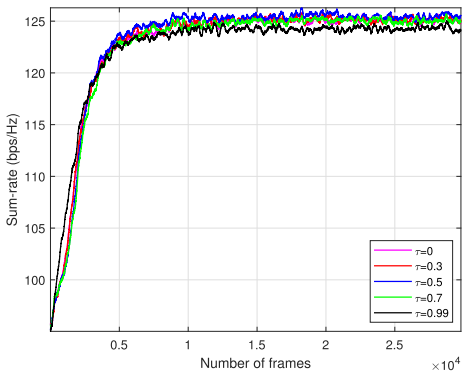
Fig. 3. The average sum-rates achieved by the ES-FPPC, FP-based, and RCS-FPPC algorithms.

In the sequel, we consider both the ideal and outdated cases of the FP-based algorithm, i.e., the ideal FP-based algorithm and the outdated FP-based algorithm. In particular, the ideal case is fed with instantaneous global CSI, while the outdated case is fed with the global CSI delayed by one frame. Note that the outdated FP-based algorithm is also unscalable due to the centralized execution, especially for the exponential growth of the required CSI as the network scales up. However, it is still a good way to show the potentials of the FP-based algorithm in an unideal situation.

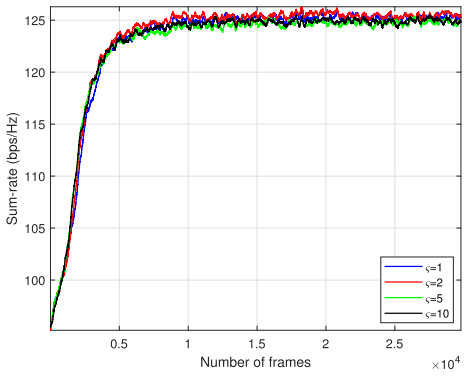
B. Parameter Selection for the DRL-Based Algorithm

In the proposed DRL-based algorithm, we design the DQN to consist of three *fully-connected* (FC) hidden layers, and they contain 300, 200, and 100 neurons respectively. We adopt "ReLu" [38] as the activation function of the first two hidden layers and adopt "tanh" as that of the third hidden layer. We discretize $\bar{P}$ into 10 levels, i.e., $L = 10$. The memory size R and the minibatch size b are set to be 2000 and 256 respectively. Adam optimizer is adopted to minimize the loss function (20) at the learning rate of 0.001. In the ε-greedy strategy, we initialize ε be 0.4 and multiply it with 0.9995 for each frame until ε reaches 0.001. The target DQN is synchronized with the trained DQN every 100 frames, i.e., $K = 100$. To achieve the best performance, we also need to

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

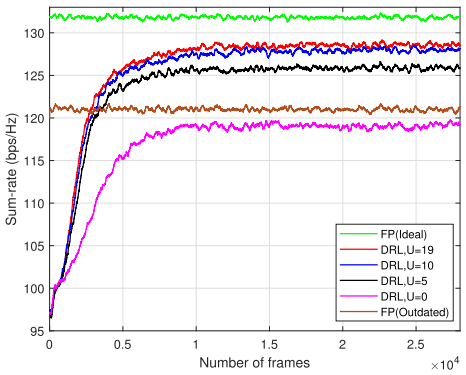


(a) $\tau = 0, 0.3, 0.5, 0.7, 0.99$.

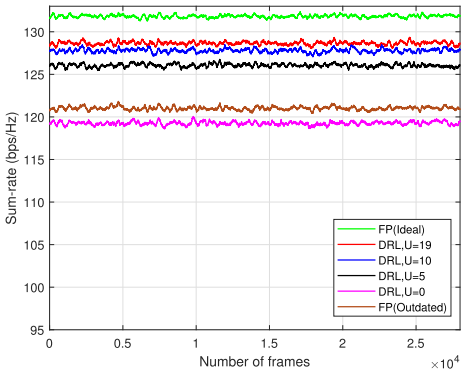


(b) $\varsigma = 1, 2, 5, 10$.

Fig. 4. Sum-rates achieved by DRL-based algorithm with different values of τ or ς. MA $= 200$.



(a) Training on.



(b) Training off.

Fig. 5. Sum-rates achieved by the DRL-based and FP-based algorithms. $U = 0, 5, 10, 19$. MA $= 200$.

select appropriate values for discount factor τ and trace-back depth ς.

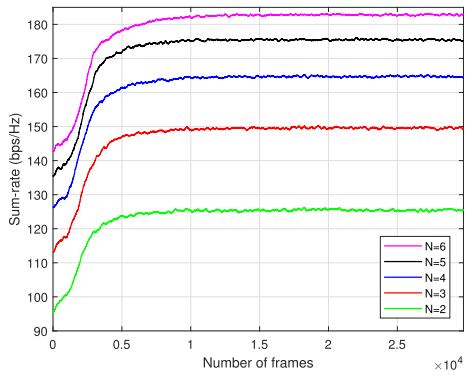
According to (9), DRL agents aim to maximize the long-term accumulative reward discounted by τ. On the one hand, with $\tau = 0$, each agent has the objective to maximize the immediate reward in each frame, which is almost equivalent to **Problem 1** that aims to maximize the instantaneous WSR. However, $\tau = 0$ fails to fully exploit the correlations of the information in successive frames and thus may reduce the WSR. On the other hand, a high value of τ is also undesirable. Due to the randomness of Rayleigh fading, the CSI in two frames is less correlated as their interval gets larger, and thus improper decisions could be made if too much importance is attached to future rewards. To better understand the impacts of τ on the WSR, by letting $\varsigma = 2$ and $U = 5$, we depict in Fig. 4a the sum-rates achieved by the proposed DRL-based algorithm with some typical values of τ. To obtain the figure, we execute the DRL-based algorithm for 30000 frames and adopt the *moving average* (MA) with a sliding window size of 200 for smoothness, which will be the same in the rest of the simulations. From the figure, a moderate τ of 0.5 can achieve the highest WSR. Hence, we fix τ to be 0.5.

Then, we proceed to the selection of trace-back depth ς. According to the channel model described given by (1), the CSI in frame t is only determined by that in frame $t-1$. With this property, D2D pairs are able to have enough information for the decision-making in frame t as long as states include all the information about frame $t-1$ and frame t. We notice that the states with $\varsigma = 2$ can exactly meet this requirement. For validation, by letting $U = 5$, we depict the sum-rates achieved by the DRL-based algorithm with different values of ς in Fig. 4b. From the figure, higher sum-rates can be achieved when ς is increased from 1 to 2. However, the resulted performance is degraded if ς is further increased to 5 or 10. It is thus implied that redundant information cannot help to improve the performance but instead may cause confusions and consequently result in performance loss. Therefore, we choose ς to be 2.

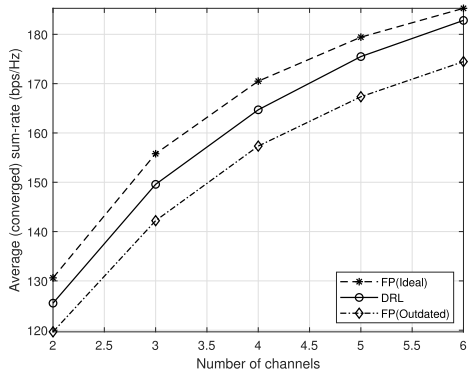
C. Evaluation of the DRL-Based Algorithm

We first evaluate the effects of the number of considered neighbors U on the performance of the DRL-based algorithm. Fig. 5 shows the sum-rates achieved by the DRL-based algorithm with different values of U, along with those by the FP-based algorithms. In particular, the DRL-based algorithm learns from scratch and keeps training on for 30000 frames in Fig. 5a, after which DRL turns off training and executes for another 30000 frames to obtain Fig. 5b. From Fig. 5a, for any value of U, the DRL-based algorithm can gradually achieve higher sum-rates until the convergence. With a higher value of U, faster convergence rate and higher converged sum-rates can be achieved. Particularly, the DRL-based algorithm can already achieve almost the sum-rates of the outdated FP-based algorithm with $U = 0$ and achieve around 98% of those of the

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

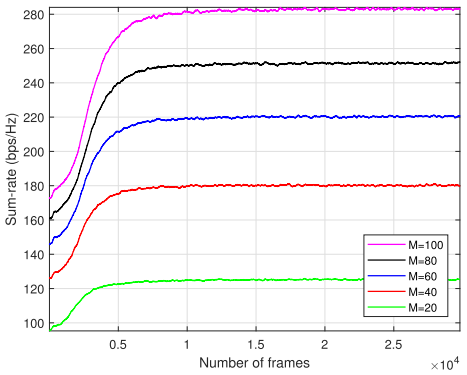


(a) Sum-rates achieved by DRL-based algorithm. MA= 200.

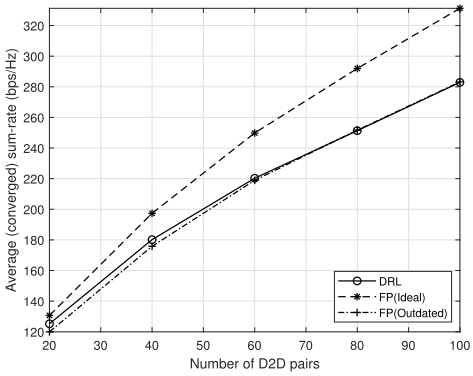


(b) Average (converged) sum-rates achieved by the DRL-based and FP-based algorithms.

Fig. 6. The effects of the number of channels. $N = 2, 3, 4, 5, 6$.



(a) Sum-rates achieved by DRL-based algorithm. MA= 200.



(b) Average (converged) sum-rates achieved by DRL-based algorithm and FP-based algorithm.

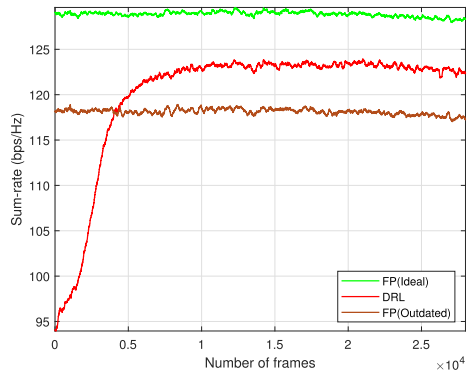
Fig. 7. The effects of the number of D2D pairs. $M = 20, 40, 60, 80, 100$.

ideal FP-based algorithm. Thus, considering more neighbors can bring more useful information and help D2D pairs learn the hidden patterns more easily. However, it can be observed that the performance improvement becomes trivial when U is large enough. Meanwhile, the required signallings exchanges keep increasing steadily with the growth of U. Therefore, in practice, a moderate value of U should be chosen to balance the performance and overheads. Since the DRL-based algorithm can already achieve around 96% of the performance of the ideal FP-based algorithm with $U = 5$, we choose U to be 5 in the rest of the simulations. From Fig. 5b, the DRL-based algorithm can maintain a high sum-rate even though training is turned off, which demonstrates that the trained DQN has the generalization capability to deal with new channel states without further training. Since the performance gain comes from training, we focus on the case with training on in the following due to space limit.

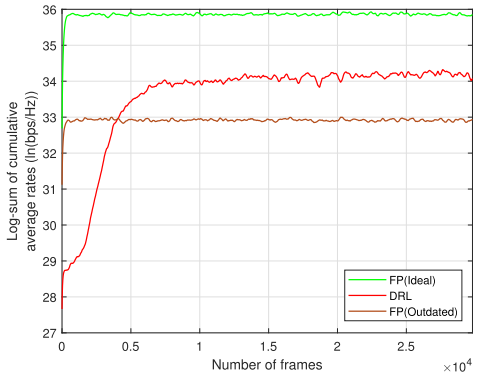
We then consider the effects of the number of channels N. Fig. 6a shows the sum-rates achieved by the DRL-based algorithm under different values of N. From Fig. 6a, the DRL-based algorithm needs around 10000 frames to converge with any value of N, and thus its convergence rate is not affected by the increase of N. We also depict in Fig. 6b the average converged sum-rates of the DRL-based algorithm and the average sum-rates achieved by the FP-based algorithms. Fig. 6b shows that the increase of N results in higher sum-rates for all algorithms, but the increments narrow down when N is large. The reasons are two-fold. Firstly, most of neighboring D2D pairs can already transmit on different channels with a large N, and thus the further increase of channels does not mitigate the mutual interference significantly. Secondly, a large N makes more D2D pairs achieve the maximum received SINR, which limits the room for further improving the sum-rates. Due to the second reason, the sum-rates achieved by the ideal FP-based algorithm increase more slowly than those by the other two algorithms. To this end, the gap between the DRL-based and ideal FP-based algorithms is getting smaller with the growth of N, as Fig. 6b shows. Particularly, the DRL-based algorithm can reach around 97.5% of the sum-rates of the ideal FP-based algorithm when $N = 6$. Moreover, the DRL approach can always have a performance gain of around 5% over the outdated FP-based algorithm. Thus, the DRL-based algorithm is effective under various numbers of channels.

Next, we vary the number of D2D pairs M and evaluate the performance of the DRL-based algorithm under different network densities. Fig. 7a depicts the sum-rates achieved by the DRL-based algorithm under different values of M. As shown in the figure, the DRL-based algorithm can still converge as the increase of M. Fig. 7b compares the average sum-rates achieved by the DRL-based algorithm after the convergence with those by the FP-based algorithms. From the figure, all the algorithms achieve higher sum-rates as more D2D pairs join the network. However, the mutual interference among D2D

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.



(a) Sum-rates achieved by the DRL-based and FP-based algorithms in the case with mobility.



(b) Log-sum of cumulative average rates achieved by the DRL-based and FP-based algorithms.

Fig. 8. Extensions to mobility and dynamic weights. MA = 200.

pairs becomes stronger and more complicated as the network gets denser, which reduces the improvements. Moreover, the increasing network density also makes D2D pairs have more neighbors with strong mutual interference. As Fig. 7b shows, when M increases, the gap between the sum-rates achieved by the ideal FP-based algorithm and those by the DRL-based algorithm gets larger. This is because the FP-based algorithms have full CSI, while the DRL-based algorithm only considers a fixed number of neighbors, e.g., $U = 5$ here. In particular, the FP-based algorithms require 16 and 25 times more CSI respectively when M increases from 20 to 80 and 100. Even so, when M is as large as 80 and 100, the DRL-based algorithm can still achieve over 85% of the sum-rates of the ideal FP-based algorithm and is comparable to the outdated FP-based algorithm. Thus, it can be demonstrated that the DRL-based algorithm is also effective for various numbers of D2D pairs.

D. Extensions

So far, we have presented the results with static topologies and static weights. However, as D2D pairs may move around in reality, it is important to consider the existence of some mobility. Moreover, static weights also cannot satisfy certain fairness requirements. To achieve wider application, we consider two extensions, including mobility and dynamic weights.

1) Mobility: With user mobility, the large-scale components of channels become time-varying. Thus, the neighbor sets of each D2D pair should be adjusted to track the dynamics. Considering the delay of fetching nonlocal information, we use the outdated channel gains to determine the neighbor sets. In particular, $\mathcal{M}_{T,U}^{m,(t)}$ denotes the transmitter-neighbor set of D2D pair m in frame t and it contains U D2D pairs who have the largest average channel gains from transmitter m to their receivers on all channels in frame $t-1$. Similarly, $\mathcal{M}_{R,U}^{m,(t)}$ can be also defined. Then, state and reward function can be modified using the new neighbor sets. With these modifications, we evaluate the performance of the DRL-based algorithm in a case of user mobility. For instance, receivers are moving around their corresponding transmitters at the speed of 5km/h. We depict in Fig. 8a the sum-rates achieved by the DRL-based and FP-based algorithms. From the figure, for all algorithms, the achieved sum-rates have large-scale fluctuations due to dynamic topologies. However, the DRL-based algorithm can still achieve over 95% of the performance of the ideal FP-based algorithm and outperform the outdated FP-based algorithm after the convergence. Thus, the DRL-based algorithm is still effective in the presence of user mobility.

2) Dynamic Weights: We take *proportional fairness* (PF) for example to extend the proposed scheme to dynamic weights. PF tries to maximize the log-sum of cumulative rates of all D2D pairs, i.e., $\max \sum_{m\in\mathcal{M}} \log \bar{C}_m^{(t)}$, where $\bar{C}_m^{(t)}$ is defined as the cumulative rate of D2D pair m in frame t and given by $\bar{C}_m^{(t)} = \iota C_m^{(t)} + (1-\iota)\bar{C}_m^{(t-1)}$. Term $\iota \in (0,1]$ determines the importance of historical cumulative rate. For this goal, D2D pairs should adjust their weights dynamically according to their cumulative rates [24]. Specifically, D2D pair m adjusts its weight in frame $t+1$ to be $w_m^{(t+1)} = 1/\bar{C}_m^{(t)}$. To enable the DRL-based algorithm to learn the changes of weights, the dynamic weights need to be included into states (17), for which we modify $\boldsymbol{\omega}_m$ to $\{\boldsymbol{\omega}_m^{(t)}, \ldots, \boldsymbol{\omega}_m^{(t-\varsigma+1)}\}$. Note that the weights of neighbors are nonlocal information, which is unavailable until the next frame. Hence, we let $\boldsymbol{\omega}_m^{(t)} = \{w_m^{(t)}, \{w_k^{(t-1)} | k \in \mathcal{M}_{R,U}^m\}, \{w_j^{(t-1)} | j \in \mathcal{M}_{T,U}^m\}\}$. Then, we conduct simulations to show the log-sum of the cumulative rates achieved by the DRL-based algorithm and those by the FP-based algorithms in Fig. 8b. In particular, we set $\iota = 0.01$ and initialize cumulative rates and weights by letting D2D pairs transmit at full power on a random channel in the first frame. From the figure, the DRL-based algorithm can achieve over 95% of the performance of the ideal FP-based algorithm and outperform the outdated FP-based algorithm, which demonstrates that the DRL approach is still applicable in the case of dynamic weights.

VII. Conclusion

In this paper, we have investigated the joint channel selection and power control problem for D2D networks to maximize the WSR. We have first developed an FP-based algorithm to solve the problem, but it requires instantaneous global network information and is not scalable. To overcome this issue, we have alternatively developed a distributed DRL-based scheme for each D2D pair to learn the correlation patterns among network information. With the learnt patterns, each D2D pair can exploit local information and outdated nonlocal

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

information to infer future network information and make the optimal decisions autonomously. Simulation results have demonstrated that even without instantaneous global CSI, the proposed DRL-based scheme can still achieve approximate performance of the FP-based algorithm.

APPENDIX
PROOF OF THEOREM 1

According to [39, *Theorem 3*], **Problem 2** is equivalent to

Problem 3:

$$\max_{\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{\lambda}} \quad \mathcal{L}(\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{\lambda}) \quad \text{s.t.} \quad \text{(P2.C1), (P2.C2), (P2.C3),} \tag{P3.Obj}$$

where $\mathcal{L}(\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{\lambda})$ is given by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{\lambda}) = &\sum_{m=1}^{M}\sum_{n=1}^{N} w_m \log_2(1+\lambda_{m,n}) \\ &- \sum_{m=1}^{M}\sum_{n=1}^{N} w_m \lambda_{m,n} \\ &+ \sum_{m=1}^{M}\sum_{n=1}^{N} \frac{w_m(1+\lambda_{m,n})A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})}{A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})+B_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})}. \end{aligned} \tag{22}$$

Similar to [39], **Problem 3** can be solved by iteratively optimizing $\boldsymbol{\alpha}, \boldsymbol{p}$, and $\boldsymbol{\lambda}$. We first consider to optimize $\boldsymbol{\lambda}$ for given $\boldsymbol{\alpha}$ and $\boldsymbol{p}$. It can be observed that the optimization of $\boldsymbol{\lambda}$ is independent from the constraints. Therefore, the optimal $\lambda^*_{m,n}$ can be obtained by simply letting $\partial\mathcal{L}(\boldsymbol{\alpha},\boldsymbol{p},\boldsymbol{\lambda})/\partial\lambda_{m,n} = 0$, i.e., $\lambda^*_{m,n} = A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})/B_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})$. Thus, we have (3).

With the optimized $\boldsymbol{\lambda}$, we then proceed to the optimization of $\boldsymbol{\alpha}$ and $\boldsymbol{p}$. If we define the part related to $\boldsymbol{\alpha}$ and $\boldsymbol{p}$ in (P3.Obj) as

$$\widetilde{\mathcal{L}}(\boldsymbol{\alpha},\boldsymbol{p}) = \sum_{m=1}^{M}\sum_{n=1}^{N} \frac{w_m(1+\lambda_{m,n})A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})}{A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})+B_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})}, \tag{23}$$

Problem 3 can be simplified as

Problem 4:

$$\max_{\boldsymbol{\alpha},\boldsymbol{p}} \quad \widetilde{\mathcal{L}}(\boldsymbol{\alpha},\boldsymbol{p}) \quad \text{s.t.} \quad \text{(P2.C1), (P2.C2), (P2.C3).} \tag{P4.Obj}$$

For a given $\boldsymbol{\alpha}$, we observe that (23) is in a fractional form in terms of $\boldsymbol{p}$, and thus we can adopt the quadratic transform [39] to further reformulate **Problem 4** as

Problem 5:

$$\max_{\boldsymbol{p},\boldsymbol{\vartheta}} \quad \widetilde{\mathcal{L}}^p(\boldsymbol{p},\boldsymbol{\vartheta}) \quad \text{s.t.} \quad \text{(P2.C1),} \tag{P5.Obj}$$

where $\boldsymbol{\vartheta}$ is the vector of the auxiliary variables introduced by the quadratic transform, and $\widetilde{\mathcal{L}}^p(\boldsymbol{p},\boldsymbol{\vartheta})$ is given by

$$\begin{aligned}\widetilde{\mathcal{L}}^p(\boldsymbol{p},\boldsymbol{\vartheta}) = &\sum_{m=1}^{M}\sum_{n=1}^{N} 2\vartheta_{m,n}\sqrt{w_m(1+\lambda_{m,n})A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})} \\ &- \sum_{m=1}^{M}\sum_{n=1}^{N} \vartheta^2_{m,n}[A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})+B_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})]. \end{aligned} \tag{24}$$

Since $\widetilde{\mathcal{L}}^p(\boldsymbol{p},\boldsymbol{\vartheta})$ is a concave function with respect to $\boldsymbol{p}$ and $\boldsymbol{\vartheta}$, we can obtain the optimal $\vartheta^*_{m,n}$ by simply letting $\partial\widetilde{\mathcal{L}}^p(\boldsymbol{p},\boldsymbol{\vartheta})/\partial\vartheta_{m,n} = 0$, i.e.,

$$\vartheta^*_{m,n} = \frac{\sqrt{w_m(1+\lambda_{m,n})h_{m,m,n}p_m\alpha_{m,n}}}{\sigma^2 + \sum_{k=1}^{M} h_{k,m,n}p_k\alpha_{k,n}}. \tag{25}$$

If we define $J_{m,n} = w_m(1+\lambda_{m,n})h_{m,m,n}$, (4) can be obtained from (25). Similarly, by letting $\partial\widetilde{\mathcal{L}}^p(\boldsymbol{p},\boldsymbol{\vartheta})/\partial p_m = 0$, we have

$$p'_m = \left[\frac{\sum_{n=1}^{N}\vartheta_{m,n}\sqrt{J_{m,n}\alpha_{m,n}}}{\sum_{n=1}^{N}\sum_{k=1}^{M} h_{m,k,n}\alpha_{m,n}\vartheta^2_{k,n}}\right]^2. \tag{26}$$

By combining (26) and (P2.C1), we can obtain the optimal $p^*_m = \min\{\bar{P}, p'_m\}$, i.e., (5).

Next, we focus on the optimization of $\boldsymbol{\alpha}$ for a given $\boldsymbol{p}$. Similar to the optimization procedures of $\boldsymbol{p}$, we first adopt the quadratic transform to reformulate the problem as

Problem 6:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\varphi}} \quad \widetilde{\mathcal{L}}^{\alpha}(\boldsymbol{\alpha},\boldsymbol{\varphi}) \quad \text{s.t.} \quad \text{(P2.C2), (P2.C3),} \tag{P6.Obj}$$

where $\boldsymbol{\varphi}$ is the vector of the auxiliary variables introduced by the quadratic transform, and $\widetilde{\mathcal{L}}^{\alpha}(\boldsymbol{\alpha},\boldsymbol{\varphi})$ is given by

$$\begin{aligned}\widetilde{\mathcal{L}}^{\alpha}(\boldsymbol{\alpha},\boldsymbol{\varphi}) = &\sum_{m=1}^{M}\sum_{n=1}^{N} 2\varphi_{m,n}\sqrt{w_m(1+\lambda_{m,n})A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})} \\ &- \sum_{m=1}^{M}\sum_{n=1}^{N} \varphi^2_{m,n}[A_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})+B_{m,n}(\boldsymbol{\alpha},\boldsymbol{p})]. \end{aligned} \tag{27}$$

Note that both the objective function and the constraints of **Problem 6** are concave with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\varphi}$. Thus, we can convert it into an unconstrained dual problem and solve it without duality gap. If we denote the vectors of the introduced Lagrangian dual variables as $\boldsymbol{\xi} \in \mathbb{R}^+_{1\times M}$ and $\boldsymbol{\nu} \in \mathbb{R}^+_{1\times M}$, the dual problem of **Problem 6** can be expressed as

Problem 7:

$$\min_{\boldsymbol{\xi},\boldsymbol{\nu}} \max_{\boldsymbol{\alpha},\boldsymbol{\varphi}} \quad \mathcal{L}_2(\boldsymbol{\alpha},\boldsymbol{\varphi},\boldsymbol{\xi},\boldsymbol{\nu}), \tag{P7.Obj}$$

where

$$\begin{aligned}\mathcal{L}_2(\boldsymbol{\alpha},\boldsymbol{\varphi},\boldsymbol{\xi},\boldsymbol{\nu}) = \mathcal{L}^{\alpha}(\boldsymbol{\alpha},\boldsymbol{\varphi}) &+ \sum_{m=1}^{M}\sum_{n=1}^{N}\xi_{m,n}\alpha_{m,n} \\ &+ \sum_{m=1}^{M}\nu_m\left(1-\sum_{n=1}^{N}\alpha_{m,n}\right). \end{aligned} \tag{28}$$

From KKT conditions, we have $\partial\mathcal{L}_2(\boldsymbol{\alpha},\boldsymbol{\varphi},\boldsymbol{\xi},\boldsymbol{\nu})/\partial\varphi_{m,n} = 0$, i.e., (6), $\partial\mathcal{L}_2(\boldsymbol{\alpha},\boldsymbol{\varphi},\boldsymbol{\xi},\boldsymbol{\nu})/\partial\alpha_{m,n} = 0$, i.e.,

$$\varphi^2_{m,n}h_{m,m,n}p_m - \varphi_{m,n}\sqrt{J_{m,n}p_m/\alpha_{m,n}} - \xi_{m,n} + \nu_m = 0, \quad \forall m \in \mathcal{M}, \tag{29}$$

$$\xi_{m,n}\alpha_{m,n} = 0, \quad \forall m \in \mathcal{M}, n \in \mathcal{N}, \tag{30}$$

and

$$\nu_m\left(1-\sum_{n=1}^{N}\alpha_{m,n}\right) = 0, \quad \forall m \in \mathcal{M}. \tag{31}$$

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

From (29) and (30), we can eliminate $\boldsymbol{\xi}$ and obtain

$$\alpha_{m,n} = \left(\frac{\varphi_{m,n}\sqrt{J_{m,n}p_m}}{\varphi_{m,n}^2 h_{m,m,n}p_m + \nu_m} \right)^2, \quad \forall m \in \mathcal{M},\ n \in \mathcal{N}, \tag{32}$$

If we insert (32) into (30), bisection search can be used to solve (7) over ν_m, with which $\alpha_{m,n}$ can be optimized as (8) shows.

Note that each update of the above variables maximizes the primal objective function (P2.Obj). In other words, if $\boldsymbol{p}$, $\boldsymbol{\alpha}$, and other auxiliary variables are updated repeatedly, the value of (P2.Obj) will be increased monotonically until it converges. Therefore, the converged $\boldsymbol{p}$ and $\boldsymbol{\alpha}$ constitute a locally optimal solution to **Problem 2**. From the above, **Theorem 1** can be proved.

REFERENCES

[1] J. Tan, L. Zhang, and Y.-C. Liang, "Deep reinforcement learning for channel selection and power control in D2D networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

[2] D. Feng, L. Lu, Y. Yuan-Wu, G. Li, S. Li, and G. Feng, "Device-to-device communications in cellular networks," *IEEE Commun. Mag.*, vol. 52, no. 4, pp. 49–55, Apr. 2014.

[3] J. Liu, N. Kato, J. Ma, and N. Kadowaki, "Device-to-device communication in LTE-advanced networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1923–1940, 4th Quart., 2015.

[4] L. Zhang, M. Xiao, G. Wu, M. Alam, Y.-C. Liang, and S. Li, "A survey of advanced techniques for spectrum sharing in 5G networks," *IEEE Wireless Commun.*, vol. 24, no. 5, pp. 44–51, Oct. 2017.

[5] 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, 4th Quart., 2014.

[6] Y. Jiang, Q. Liu, F. Zheng, X. Gao, and X. You, "Energy-efficient joint resource allocation and power control for D2D communications," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6119–6127, Aug. 2016.

[7] A. Asheralieva and Y. Miyanaga, "QoS-oriented mode, spectrum, and power allocation for D2D communication underlaying LTE—A network," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9787–9800, Dec. 2016.

[8] R. Yin, C. Zhong, G. Yu, Z. Zhang, K. K. Wong, and X. Chen, "Joint spectrum and power allocation for D2D communications underlaying cellular networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2182–2195, Apr. 2016.

[9] D. Feng *et al.*, "Mode switching for energy-efficient device-to-device communications in cellular networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 12, pp. 6993–7003, Dec. 2015.

[10] S. Xiao, X. Zhou, D. Feng, Y. Yuan-Wu, G. Y. Li, and W. Guo, "Energy-efficient mobile association in heterogeneous networks with device-to-device communications," *IEEE Trans. Wireless Commun.*, vol. 15, no. 8, pp. 5260–5271, Aug. 2016.

[11] G. Fodor *et al.*, "Design aspects of network assisted device-to-device communications," *IEEE Commun. Mag.*, vol. 50, no. 3, pp. 170–177, Mar. 2012.

[12] S. N. Swain, S. Mishra, and C. S. R. Murthy, "A novel spectrum reuse scheme for interference mitigation in a dense overlay D2D network," in *Proc. IEEE 26th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Aug. 2015, pp. 1201–1205.

[13] Z.-Y. Yang and Y.-W. Kuo, "Efficient resource allocation algorithm for overlay D2D communication," *Comput. Netw.*, vol. 124, pp. 61–71, Sep. 2017.

[14] A. Abrardo and M. Moretti, "Distributed power allocation for D2D communications underlaying/overlaying OFDMA cellular networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1466–1479, Mar. 2017.

[15] J. Lyu, Y. H. Chew, and W.-C. Wong, "A stackelberg game model for overlay D2D transmission with heterogeneous rate requirements," *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 8461–8475, Oct. 2016.

[16] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, vol. 1, no. 1. Cambridge, MA, USA: MIT Press, 1998.

[17] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[18] N. Cong Luong *et al.*, "Applications of deep reinforcement learning in communications and networking: A survey," 2018, *arXiv:1810.07862*. [Online]. Available: http://arxiv.org/abs/1810.07862

[19] L. Zhang, Y.-C. Liang, and D. Niyato, "6G visions: Mobile ultra-broadband, super Internet-of-Things, and artificial intelligence," *China Commun.*, vol. 16, no. 8, pp. 1–14, Aug. 2019.

[20] S. Wang, H. Liu, P. H. Gomes, and B. Krishnamachari, "Deep reinforcement learning for dynamic multichannel access in wireless networks," *IEEE Trans. Cognit. Commun. Netw.*, vol. 4, no. 2, pp. 257–265, Jun. 2018.

[21] O. Naparstek and K. Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 310–323, Jan. 2019.

[22] S. Liu, X. Hu, and W. Wang, "Deep reinforcement learning based dynamic channel allocation algorithm in multibeam satellite systems," *IEEE Access*, vol. 6, pp. 15733–15742, Feb. 2018.

[23] N. Zhao, Y.-C. Liang, D. Niyato, Y. Pei, M. Wu, and Y. Jiang, "Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5141–5152, Nov. 2019.

[24] Y. S. Nasir and D. Guo, "Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2239–2250, Oct. 2019.

[25] X. Li, J. Fang, W. Cheng, H. Duan, Z. Chen, and H. Li, "Intelligent power control for spectrum sharing in cognitive radios: A deep reinforcement learning approach," *IEEE Access*, vol. 6, pp. 25463–25473, Apr. 2018.

[26] L. Zhang, J. Tan, Y.-C. Liang, G. Feng, and D. Niyato, "Deep reinforcement learning-based modulation and coding scheme selection in cognitive heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 6, pp. 3281–3294, Jun. 2019.

[27] J. Tan, L. Zhang, Y.-C. Liang, and D. Niyato, "Intelligent sharing for LTE and WiFi systems in unlicensed bands: A deep reinforcement learning approach," *IEEE Trans. Commun.*, vol. 68, no. 5, pp. 2793–2808, May 2020.

[28] Y. He *et al.*, "Deep-reinforcement-learning-based optimization for cache-enabled opportunistic interference alignment wireless networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10433–10445, Nov. 2017.

[29] U. Challita, W. Saad, and C. Bettstetter, "Interference management for cellular-connected UAVs: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2125–2140, Apr. 2019.

[30] C. H. Liu, Z. Chen, J. Tang, J. Xu, and C. Piao, "Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2059–2070, Sep. 2018.

[31] K. Shen and W. Yu, "Fractional programming for communication Systems—Part I: Power control and beamforming," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2616–2630, May 2018.

[32] T. Kim, D. J. Love, and B. Clerckx, "Does frequent low resolution feedback outperform infrequent high resolution feedback for multiple antenna beamforming systems?" *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1654–1669, Apr. 2011.

[33] S.-Y. Lien, C.-C. Chien, F.-M. Tseng, and T.-C. Ho, "3GPP device-to-device communications for beyond 4G cellular networks," *IEEE Commun. Mag.*, vol. 54, no. 3, pp. 29–35, Mar. 2016.

[34] R. Wang, J. Zhang, S. H. Song, and K. B. Letaief, "Optimal QoS-aware channel assignment in D2D communications with partial CSI," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7594–7609, Nov. 2016.

[35] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 57–73, Feb. 2008.

[36] Y. Yu, T. Wang, and S. C. Liew, "Deep-reinforcement learning multiple access for heterogeneous wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1277–1290, Jun. 2019.

[37] G. Papoudakis, F. Christianos, A. Rahman, and S. V. Albrecht, "Dealing with non-stationarity in multi-agent deep reinforcement learning," 2019, *arXiv:1906.04737*. [Online]. Available: http://arxiv.org/abs/1906.04737

[38] Y. Li and Y. Yuan, "Convergence analysis of two-layer neural networks with ReLu activation," in *Proc. NIPS*, 2017, pp. 597–607.

[39] K. Shen and W. Yu, "Fractional programming for communication systems—Part II: Uplink scheduling via matching," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2631–2644, May 2018.

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.

Junjie Tan (Student Member, IEEE) received the B.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2016, where he is currently pursuing the Ph.D. degree. His research interests include heterogeneous networks, machine learning, and intelligent wireless communications. He received the IEEE ICC 2019 Best Paper Award.

Ying-Chang Liang (Fellow, IEEE) was a Professor with The University of Sydney, Australia, a Principal Scientist and Technical Advisor with the Institute for Infocomm Research, Singapore, and a Visiting Scholar with Stanford University, Stanford, CA, USA. He is currently a Professor with the University of Electronic Science and Technology of China, China, where he leads the Center for Intelligent Networking and Communications, and serves as the Deputy Director of the Artificial Intelligence Research Institute. He has been recognized by Thomson Reuters (now Clarivate Analytics) as a Highly Cited Researcher since 2014. His research interests include wireless networking and communications, cognitive radio, symbiotic networks, dynamic spectrum access, the Internet-of-Things, artificial intelligence, and machine learning techniques.

He is a foreign member of Academia Europaea. He received the Prestigious Engineering Achievement Award from The Institution of Engineers, Singapore, in 2007, the Outstanding Contribution Appreciation Award from the IEEE Standards Association, in 2011, and the Recognition Award from the IEEE Communications Society Technical Committee on Cognitive Networks, in 2018. He is a recipient of numerous paper awards, including the IEEE Jack Neubauer Memorial Award, in 2014, and the IEEE Communications Society APB Outstanding Paper Award, in 2012. He was the Chair of the IEEE Communications Society Technical Committee on Cognitive Networks, and served as the TPC Chair and Executive Co-Chair of the IEEE Globecom'17. He is the Founding Editor-in-Chief of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS: COGNITIVE RADIO SERIES, as well as the Key Founder and currently the Editor-in-Chief of the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING. He is also serving as an Associate Editor-in-Chief for *China Communications*. He has served as a Guest/Associate Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE JOURNAL OF SELECTED AREAS IN COMMUNICATIONS, the *IEEE Signal Processing Magazine*, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, and the IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORK. He was also an Associate Editor-in-Chief of the *World Scientific Journal on Random Matrices: Theory and Applications*. He was a Distinguished Lecturer of the IEEE Communications Society and the IEEE Vehicular Technology Society.

Lin Zhang (Member, IEEE) received the bachelor's degree in communication engineering from Sichuan University, Chengdu, China, in June 2011, and the Ph.D. degree from the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China (UESTC), Chengdu, in June 2017. From October 2017 to June 2019, he was a Post-Doctoral Researcher with UESTC, where he is currently an Associate Professor. From October 2014 to April 2016, he worked as a Visiting Student with the School of Electrical Engineering, Royal Institute of Technology (KTH), Sweden. His research interests include cognitive radio, D2D communication, content caching, non-orthogonal multiple access, short packet communications, and intelligent wireless communications. He won the IEEE GLOBECOM 2012 Best Paper Award and Student Travel Grant Award; the IEEE Region 10 Distinguished Student Paper Award in 2016; and the IEEE ICC 2019 Best Paper Award.

Gang Feng (Senior Member, IEEE) received the B.Eng. and M.Eng. degrees in electronic engineering from the University of Electronic Science and Technology of China (UESTC), in 1986 and 1989, respectively, and the Ph.D. degree in information engineering from The Chinese University of Hong Kong in 1998. He joined the School of Electric and Electronic Engineering, Nanyang Technological University, in December 2000, as an Assistant Professor and became an Associate Professor in October 2005. He is currently a Professor with the National Laboratory of Communications, UESTC. He has extensive research experience and has published widely in wireless networking research. A number of his papers have been highly cited. His research interests include next-generation mobile networks, mobile cloud computing, and AI-enabled wireless networking. He has received the IEEE ComSoc TAOS Best Paper Award and ICC Best Paper Award in 2019.

Authorized licensed use limited to: ULAKBIM UASL - SABANCI UNIVERSITY. Downloaded on August 10,2024 at 18:02:26 UTC from IEEE Xplore. Restrictions apply.