Decentralized Federated Reinforcement Learning for User-Centric Dynamic TFDD Control

Ziyan Yin, *Student Member, IEEE*, Zhe Wang, *Member, IEEE*, Jun Li, *Senior Member, IEEE*, Ming Ding, *Senior Member, IEEE*, Wen Chen, *Senior Member, IEEE*, and Shi Jin, *Senior Member, IEEE*

***Abstract*—The explosive growth of dynamic and heterogeneous data traffic brings great challenges for 5G and beyond mobile networks. To enhance the network capacity and reliability, we propose a learning-based dynamic time-frequency division duplexing (D-TFDD) scheme that adaptively allocates the uplink and downlink time-frequency resources of base stations (BSs) to meet the asymmetric and heterogeneous traffic demands while alleviating the inter-cell interference. We formulate the problem as a decentralized partially observable Markov decision process (Dec-POMDP) that maximizes the long-term expected sum rate under the users' packet dropping ratio constraints. In order to jointly optimize the global resources in a decentralized manner, we propose a federated reinforcement learning (RL) algorithm named federated Wolpertinger deep deterministic policy gradient (FWDDPG) algorithm. The BSs decide their local time-frequency configurations through RL algorithms and achieve global training via exchanging local RL models with their neighbors under a decentralized federated learning framework. Specifically, to deal with the large-scale discrete action space of each BS, we adopt a DDPG-based algorithm to generate actions in a continuous space, and then utilize Wolpertinger policy to reduce the mapping errors from continuous action space back to discrete action space. Simulation results demonstrate the superiority of our proposed algorithm to the benchmark algorithms with respect to system sum rate.**

***Index Terms*—Dynamic TFDD, decentralized partially observable Markov decision process, federated learning, multi-agent reinforcement learning, resource allocation.**

Manuscript received 13 June 2022; revised 16 September 2022; accepted 27 October 2022. Date of publication 11 November 2022; date of current version 17 February 2023. This work was supported in part by the National Key Project under Grants 2020YFB1807700 and 2018YFB1801102, in part by National Natural Science Foundation of China under Grants 61872184, 62202232, and 62071296, in part by the Fundamental Research Funds for the Central Universities under Grant 30921013104, in part by the Future Network Grant of Provincial Education Board in Jiangsu, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20210331, in part by Shanghai Kewei under Grants 20JC1416502 and 22JC1404000, and in part by Pudong under Grant PKX2021-D02. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. Yonina C. Eldar. *(Corresponding authors: Zhe Wang; Jun Li.)*

Ziyan Yin and Jun Li are with the School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: ziyan.yin@njust.edu.cn; jun.li@njust.edu.cn).

Zhe Wang is with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: zwang@njust.edu.cn).

Ming Ding is with Data61, CSIRO, Sydney NSW 2015, Australia (e-mail: ming.ding@data61.csiro.au).

Wen Chen is with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: wenchen@sjtu.edu.cn).

Shi Jin is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: jinshi@seu.edu.cn).

Digital Object Identifier 10.1109/JSTSP.2022.3221671

I. Introduction

DRIVEN by the burgeoning demands of various services coming from smart cities and industries, 5th generation (5G) and beyond wireless communication systems are facing the challenges of diverse quality-of-service (QoS) requirements [1], [2], [3]. The conventional "one-size-fit-all" network infrastructure may not be able to simultaneously meet the heterogeneous service requirements. Network slicing has been proposed to "slice" the mobile infrastructure into multiple logical networks, which provides flexible network services in a cost-efficient way [4], [5]. The key problem for network slicing is to dynamically and efficiently allocate the computation and communication resources, e.g., computing frequencies [4], transmit power [6], [7], radio spectrum [8] and transmission time [9], to meet various and even conflicting QoS demands.

Time division duplexing (TDD), as a typical application of network slicing, is able to accommodate asymmetric traffic demands in the uplink (UL) and downlink (DL) by allowing the UL and DL traffic to operate in different subframes [10]. The TDD system can be mainly classified into two categories: static TDD (S-TDD) and dynamic TDD (D-TDD). For S-TDD [11], [12], [13], all base stations (BSs) adopt the same and synchronized UL and DL subframe configurations, which however may not be efficient if the traffic demands are dynamic and asymmetric across the cells. To improve the resource utilization efficiency, D-TDD is proposed, where BSs can adopt different subframe configurations. However, D-TDD suffers from additional inter-cell interference due to the asynchronous transmissions, i.e., the UL/DL transmissions in a cell may interfere with the DL/UL transmissions in its neighboring cells [14]. To alleviate the inter-cell interference, the BSs can be divided into different clusters [15], where the BSs within each cluster adopt the same subframe configuration. Another interference alleviation approach is to adjust the wireless signal transmission strategies, i.e., interference cancellation [12], [16], power control [17], [18] and beamforming [17], [19], [20], where the BSs cooperatively optimize their signal transmission strategies via convex optimization or heuristic algorithms. For this type of approach, the subframe configuration is usually selected from pre-defined candidates, e.g., the seven subframe configurations of 3GPP [21], without adapting to the real-time traffic demands.

The network traffic demands and channel states are highly dynamic and unpredicted in D-TDD systems, making it costly

1932-4553 © 2022 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: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

to design the adaptive subframe configurations by the conventional model-based optimization methods. Advanced model-free methods such as single-agent reinforcement learning (RL) [22], [23] and multi-agent reinforcement learning (MARL) [24], [25], [26] have been recently applied to solve the sequential resource allocation problems in complex and dynamic wireless networks, where the agents can learn the policy in a trial-and-error manner. There are two main types of MARL approaches for designing the subframe configurations in the D-TDD system: centralized MARL [24] and decentralized MARL [25], [26]. The subframe configuration in [24] depends on the coordination of a centralized controller. The BSs in [25], [26] independently make local subframe configuration decisions, while treating other BSs as part of the environment.

Based on the aforementioned literature, there are two challenges left unsolved. The first challenge is to design the D-TDD scheme to meet the heterogeneous QoS demands of different user equipment (UE) types. In the existing literature, most of the D-TDD subframe configurations are cell-centric, where each BS allocates the number of UL/DL subframes depending on the average UL/DL data traffic inside this cell without further differentiating the resource demands of the specific UEs. However, the data traffic patterns and the QoS requirements may vary significantly for different UE types in a heterogeneous network, which has been largely overlooked in the existing literature. To satisfy the user-centric heterogeneous QoS demands, we propose a learning-based dynamic time-frequency division duplexing (D-TFDD) framework. The second challenge is to jointly optimize the resources for local traffic adaptation and global interference alleviation without collecting the private states from each BS. In the existing literature, the centralized MARL subframe configuration requires the states of all BSs, which may not be easy to implement in practice due to the curse of dimensionality and privacy issues. Moreover, the decentralized MARL subframe configuration may not efficiently avoid inter-cell interference if the BSs' learning processes are independent. To tackle this challenge, inspired by the advantages of federated learning (FL) [27], [28], we propose a federated reinforcement learning algorithm to design the dynamic resource allocation, aiming to meet heterogeneous UE demands while coordinating the inter-cell interference in a decentralized and privacy-preserving manner.

In this work, we propose a user-centric learning-based resource allocation framework in a heterogeneous cellular network consisting of multiple BSs, ground UEs (GUEs) and unmanned aerial vehicles (UAVs), where the BSs adaptively allocate time-frequency resources to satisfy the heterogeneous QoS demands characterized by the packet dropping ratio constraints. We summarize the main contributions as follows.

- We propose a learning-based D-TFDD scheme in a heterogeneous cellular system with dynamic UL and DL packet queuing processes. The proposed scheme exploits the merits of both D-TDD and dynamic frequency division duplexing (D-FDD) by jointly allocating the time-frequency resources. We adopt D-TDD to adapt the BSs' subframe allocation to the asymmetric UL/DL traffic from a cell-centric perspective, and utilize D-FDD to cater the subchannel allocation to the heterogeneous QoS demands from a user-centric perspective.
- We formulate the dynamic resource allocation problem under the proposed D-TFDD scheme as a decentralized partially observable MDP (Dec-POMDP), where each BS only has partial observation of the network environment. The BSs adaptively decide the subframe and subchannel allocations to maximize the long-term expected sum rate of the network while satisfying the UEs' packet dropping ratio constraints.
- We propose a federated reinforcement learning algorithm named federated Wolpertinger deep deterministic policy gradient (FWDDPG) to solve the above optimization problem. The dimensionality of action space for D-TFDD control at each BS increases substantially as the number of UEs, subframes and subchannels increases. To deal with the large-scale discrete action space, we first adopt a DDPG-based policy at each BS to generate actions in a continuous space, and then discretize the actions based on Wolpertinger policy to reduce the mapping errors. For model aggregation across the BSs, we adopt a peer-to-peer FL architecture without a centralized server, where the BSs exchange their neural network parameters with their one-hop neighbors to avoid privacy leakage and single point failure.
- Simulation results show that our proposed D-TFDD scheme outperforms other benchmark TDD schemes, verifying the advantages of dynamically allocating multi-domain resources in serving heterogeneous UEs. Furthermore, the proposed algorithm outperforms independent DDPG (IDDPG) and it is even superior to the centralized multi-agent DDPG (MADDPG) by properly adjusting the Wolpertinger coefficient. The simulation reveals that, with sufficient system resources, the BSs prefer allocating more subchannels to the UEs with heavier traffic loads for throughput enhancement, and adopting similar subframe configurations across the cells for interference alleviation. Furthermore, in a resource-constrained regime, the BSs prioritize meeting local QoS constraints over avoiding interference.

The rest of this paper is organized as follows. In Section II, we present the system model of the proposed D-TFDD network. In Section III, we formulate the dynamic resource optimization problem as a Dec-POMDP. In Section IV, we propose the FWDDPG algorithm to obtain the optimal resource allocation policies. Section V discusses the simulation results. At last, Section VI concludes the paper.

II. System Model

We consider a heterogeneous multi-cell network which consists of a set $\mathcal{B} \triangleq \{1, 2, \ldots, B\}$ of BSs serving a set $\mathcal{U} \triangleq \{\mathcal{U}_{\mathrm{GUE}}, \mathcal{U}_{\mathrm{UAV}}\}$ of UEs, where $\mathcal{U}_{\mathrm{GUE}}$ and $\mathcal{U}_{\mathrm{UAV}}$ denote the set of GUEs and UAVs, respectively. We denote $\mathcal{U}^b$ as the set of UEs served by BS b inside cell b.

A time-framed D-TFDD framework is shown in Fig. 1, where the DL/UL subframe configurations are dynamic across cells and

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

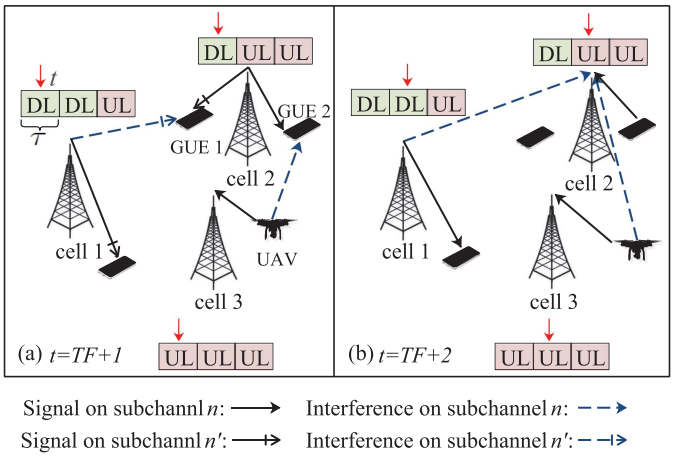


Fig. 1. An illustration of proposed D-TFDD framework in time frame T.

time frames. Each time frame is made up of F subframes, and the length of the subframe is τ. For cell b in time frame T, the first $f^b(T) \in \{0, 1, \ldots, F\}$ number of successive subframes are used for DL transmissions and the rest of $F - f^b(T)$ number of successive subframes are used for UL transmissions.

We adopt orthogonal frequency division multiple access (OFDMA) for multiple access inside each cell, where the set of orthogonal subchannels is denoted by $\mathcal{N} \triangleq \{1, 2, \ldots, N\}$ with the bandwidth W for each subchannel. Assume that the number of subchannels is not less than the number of UEs served by any BS, i.e., $N \geq |\mathcal{U}^b|, \forall b$. Let $\phi^n_{b,u}(T) \in \{0, 1\}$ denote whether or not subchannel n is allocated to UE u inside cell b for DL transmissions, where $\phi^n_{b,u}(T) = 1$ denotes the subchannel n is allocated to UE u for $f^b(T)$ number of successive DL subframes and $\phi^n_{b,u}(T) = 0$ means not. Similarly, for UL transmissions, the subchannel allocation is defined as $\phi^n_{u,b}(T) \in \{0, 1\}$. Assume each subchannel n can serve at most one receiver within a cell at a time, which can be represented as $\sum_{u \in \mathcal{U}^b} \phi^n_{b,u}(T) \leq 1$ and $\sum_{u \in \mathcal{U}^b} \phi^n_{u,b}(T) \leq 1$.

We consider quasi-static fading, where the channel state stays constant during each time frame for any given subchannel. Let $g^n_{\text{tx,rx}}(T)$ denote the channel fading gain from transmitter tx to receiver rx on subchannel n at time frame T, where tx and rx can be any UE $u \in \mathcal{U}$ or any BS $b \in \mathcal{B}$. The channel fading gain of $g^n_{\text{tx,rx}}(T)$ includes both large-scale and small-scale fading [29]. To compute the large-scale fading, the distance from transmitter tx to receiver rx is needed. We assume each UAV follows a pre-defined trajectory inside its associated cell (to fulfill specific tasks, e.g., surveillance), and the GUEs' and BSs' locations are static. For the ease of analysis, we discretize the flight trajectory of each UAV by a series of discrete locations, where we assume its location is static within a time frame T and can be different across time frames [30], [31]. Here, we adopt three-dimensional Cartesian coordinate and define the locations of transmitter tx and receiver rx at time frame T as $(X_{\text{tx}}(T), Y_{\text{tx}}(T), H_{\text{tx}}(T))$ and $(X_{\text{rx}}(T), Y_{\text{rx}}(T), H_{\text{rx}}(T))$, respectively. Then, the three-dimensional distance between transmitter tx and receiver rx is

$$\beta_{\text{tx,rx}}(T) = \|(X_{\text{tx}}(T), Y_{\text{tx}}(T), H_{\text{tx}}(T)) - (X_{\text{rx}}(T), Y_{\text{rx}}(T), H_{\text{rx}}(T))\|_2, \tag{1}$$

where $\|\cdot\|_2$ is Euclidean distance. We adopt a general path loss model $\xi(\beta_{\text{tx,rx}}(T))$ to consider both line-of-sight (LoS) and none-line-of-sight (NLoS) links. According to the well known International Telecommunication Union (ITU) model [32], [33], the probability of having a LoS link between transmitter tx and receiver rx is given by

$$\Pr^{\text{LoS}}(\beta_{\text{tx,rx}}(T)) = \prod_{j=0}^{c_4} \left[1 - \exp\left(- \frac{\left[H_{\text{tx}}(T) - \frac{(j+0.5)(H_{\text{tx}}(T) - H_{\text{rx}}(T))}{c_4+1} \right]^2}{\left(\sqrt{2}c_3\right)^2} \right) \right], \tag{2}$$

where $\{c_1, c_2, c_3\}$ are environment-dependent parameters and $c_4 = \lfloor \frac{\beta_{\text{tx,rx}}(T)\sqrt{c_1 c_2}}{1000} - 1 \rfloor$. The probability of having a NLoS link between transmitter tx and receiver rx is given by

$$\Pr^{\text{NLoS}}(\beta_{\text{tx,rx}}(T)) = 1 - \Pr^{\text{LoS}}(\beta_{\text{tx,rx}}(T)). \tag{3}$$

Then, the general path loss model $\xi(\beta_{\text{tx,rx}}(T))$ is given by

$$\xi(\beta_{\text{tx,rx}}(T)) = \begin{cases} A^{\text{LoS}} \beta_{\text{tx,rx}}(T)^{\alpha^{\text{LoS}}}, & \text{with prob. (2)}, \\ A^{\text{NLoS}} \beta_{\text{tx,rx}}(T)^{\alpha^{\text{NLoS}}}, & \text{with prob. (3)}. \end{cases} \tag{4}$$

Let A^{LoS} and A^{NLoS} denote the reference path loss for LoS and NLoS links, and α^{LoS} and α^{NLoS} denote the path loss exponent for LoS and NLoS links, respectively. Furthermore, Nakagami-m small-scale fading is adopted in our model. Let $h^n_{\text{tx,rx}}(T)$ denote the small-scale fading gain on subchannel n between transmitter tx and receiver rx at time frame T, and the cumulative distribution function of $h^n_{\text{tx,rx}}(T)$ can be obtained as

$$\mathcal{F}(x) \triangleq \Pr\left[h^n_{\text{tx,rx}}(T) \leq x\right] = 1 - \sum_{j=0}^{m_{\text{tx,rx}}} \frac{(m_{\text{tx,rx}} x)^j}{j!} \exp\left(-m_{\text{tx,rx}} x\right), \tag{5}$$

where $m_{\text{tx,rx}}$ is the fading parameter. Taking into account both the large-scale and small-scale fading, the channel fading gain is thus given by

$$g^n_{\text{tx,rx}}(T) = \left[\xi(\beta_{\text{tx,rx}}(T))\right]^{-1} h^n_{\text{tx,rx}}(T). \tag{6}$$

Consider that a typical UE u is associated with a typical BS b. Let P_b and P_u denote the transmit power of BS b and UE u, respectively. Consider a DL receiver UE u is receiving information from BS b on subchannel n. UE u may suffer from the BS-to-UE interference from the set of DL cells $\mathcal{B}^{n,\text{DL}}(t) \backslash b$ and UE-to-UE interference from the set of UL cells $\mathcal{B}^{n,\text{UL}}(t)$ in subframe t, where the total interference power received at UE u is given by

$$I^n_u(t) = \sum_{b' \in \mathcal{B}^{n,\text{DL}}(t) \backslash b} \sum_{u' \in \mathcal{U}^{b'}} \phi^n_{b,'u'}(T) P_{b'} g^n_{b,'u}(T) + \sum_{b' \in \mathcal{B}^{n,\text{UL}}(t)} \sum_{u' \in \mathcal{U}^{b'}} \phi^n_{u,'b'}(T) P_{u'} g^n_{u,'u}(T). \tag{7}$$

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

Note that, though we assume the channel fading gains remain unchanged during a time frame T, the set of interfering cells can be different across different subframes t due to the dynamic time and frequency allocation. Therefore, the signal to interference plus noise ratio (SINR) at the DL receiver UE u in subframe t on subchannel n is given by

$$\mathrm{SINR}_{b,u}^{n}(t)=\frac{P_b g_{b,u}^{n}(T)}{I_u^n(t)+N_0 W}, \tag{8}$$

where N_0 is the variance of white Gaussian noise. Consider that data transmission between any pair of transmitter tx and receiver rx is successful only if the received SINR is no less than a pre-defined threshold ς_{rx}. The DL achievable rate at UE u is expressed as

$$R_{b,u}^{n}(t)=\mathbb{1}\left(\mathrm{SINR}_{b,u}^{n}(t)\geq\varsigma_u\right)\tau W\log_2\left(1+\varsigma_u\right), \tag{9}$$

where $\mathbb{1}(\cdot)$ is the indicator function that takes the value of 1 if the event happens and the value of 0 if not. Here, the achievable rate is measured in bits per subframe. As the UE can operate on multiple subchannels, the total DL achievable rate $R_{b,u}(t)$ for UE u is given by

$$R_{b,u}(t)=\sum_{n\in\mathcal{N}}\phi_{b,u}^{n}(T)R_{b,u}^{n}(t). \tag{10}$$

Next, we discuss the UL achievable rate at a typical BS b. Consider a UL receiver BS b that is operating on subchannel n may receive the co-channel interference from adjacent DL and UL cells, i.e., the BS-to-BS interference from the set of DL cells $\mathcal{B}^{n,\mathrm{DL}}(t)$ and UE-to-BS interference from the set of UL cells $\mathcal{B}^{n,\mathrm{UL}}(t)\backslash b$ in subframe t, which is expressed as

$$\begin{aligned}I_b^n(t)=&\sum_{b'\in\mathcal{B}^{n,\mathrm{DL}}(t)}\sum_{u'\in\mathcal{U}^{b'}}\phi_{b,'u'}^{n}(T)P_{b'}g_{b,'b}^{n}(T)\\&+\sum_{b'\in\mathcal{B}^{n,\mathrm{UL}}(t)\backslash b}\sum_{u'\in\mathcal{U}^{b'}}\phi_{u,'b'}^{n}(T)P_{u'}g_{u,'b}^{n}(T).\end{aligned} \tag{11}$$

The SINR and achievable rate at the UL receiver BS b are respectively given by

$$\mathrm{SINR}_{u,b}^{n}(t)=\frac{P_u g_{u,b}^{n}(T)}{I_b^n(t)+N_0\,W}, \tag{12}$$

and

$$R_{u,b}^{n}(t)=\mathbb{1}\left(\mathrm{SINR}_{u,b}^{n}(t)\geq\varsigma_b\right)\tau W\log_2\left(1+\varsigma_b\right). \tag{13}$$

Therefore, for UE u, the total UL achievable rate $R_{u,b}(t)$ is given by

$$R_{u,b}(t)=\sum_{n\in\mathcal{N}}\phi_{u,b}^{n}(T)R_{u,b}^{n}(t). \tag{14}$$

Each UE u maintains a local UL queue and a DL queue at the BS side. At the beginning of time frame T (before the data transmission), let $\hat{Q}_u^{\mathrm{DL}}(T)$ and $\hat{Q}_u^{\mathrm{UL}}(T)$ respectively denote the DL and UL queue lengths of UE u, which are the sizes of the remaining packets in the DL and UL buffers.

For UE u in time frame T, the amount of DL received packets at UE u during $f^b(T)$ successive DL subframes is defined as

$$\psi_u^{\mathrm{DL}}(T)=\min\left\{\hat{Q}_u^{\mathrm{DL}}(T),\sum_{t=TF+1}^{TF+f^b(T)}R_{b,u}(t)\right\}. \tag{15}$$

where $\psi_u^{\mathrm{DL}}(T)$ cannot exceed the amount of packets in the current DL queue $\hat{Q}_u^{\mathrm{DL}}(T)$. Similarly, the amount of UL received packets at BS b from UE u during the remaining $F-f^b(T)$ subframes is given by

$$\psi_u^{\mathrm{UL}}(T)=\min\left\{\hat{Q}_u^{\mathrm{UL}}(T),\sum_{t=TF+1+f^b(T)}^{(T+1)F}R_{u,b}(t)\right\}, \tag{16}$$

where $\psi_u^{\mathrm{UL}}(T)$ cannot exceed the amount of packets in the current UL queue $\hat{Q}_u^{\mathrm{UL}}(T)$.

For any UE u, we consider that UL and DL packets arrive at the end of each time frame T. Consider that the sizes of UL packets $D_u^{\mathrm{UL}}(T)$ and DL packets $D_u^{\mathrm{DL}}(T)$ follow Poisson processes of $\mathcal{P}(\lambda_u^{\mathrm{UL}})$ and $\mathcal{P}(\lambda_u^{\mathrm{DL}})$, respectively, where λ_u^{UL} and λ_u^{DL} are the average UL and DL packet sizes of UE u, respectively.

Therefore, we can deduce that the DL queue length for UE u at the beginning of time frame $T+1$ evolves as

$$\hat{Q}_u^{\mathrm{DL}}\left(T+1\right)=\hat{Q}_u^{\mathrm{DL}}(T)-\psi_u^{\mathrm{DL}}(T)+D_u^{\mathrm{DL}}(T), \tag{17}$$

where DL buffer at the BS is assumed to be sufficiently large. Similarly, the UL queue length for UE u evolves as

$$\begin{aligned}&\hat{Q}_u^{\mathrm{UL}}\left(T+1\right)\\&=\min\left\{\hat{Q}_u^{\max},\hat{Q}_u^{\mathrm{UL}}(T)-\psi_u^{\mathrm{UL}}(T)+D_u^{\mathrm{UL}}(T)\right\},\end{aligned} \tag{18}$$

where $\hat{Q}_u^{\max}$ is UL data buffer size at UE u. Once the UL queue length exceeds the buffer size $\hat{Q}_u^{\max}$, the newly arrived packets will be dropped.

To characterize the reliability of UL transmission, we denote $d_u(T)$ as the dropping ratio of UE u estimated at the end of time frame T, which is the ratio of total dropped data to total arrived data over the most recent $T-\Gamma$ time frames, i.e.,

$$d_u(T)=1-\frac{\sum_{l=\Gamma+1}^{T}\psi_u^{\mathrm{UL}}(l)+\hat{Q}_u^{\mathrm{UL}}\left(T+1\right)-\hat{Q}_u^{\mathrm{UL}}\left(\Gamma+1\right)}{\sum_{l=\Gamma+1}^{T}D_u^{\mathrm{UL}}(l)}, \tag{19}$$

where $\Gamma=\max[0,T-\Lambda]$, and Λ is the window size that removes the effect of the earlier history.

We consider that each BS can offer E different types of slices, where each slice provides a customized service for the UEs with similar QoS requirements. Taking slice $e\in\{1,\ldots,E\}$ as an example, the set of UEs accessing slice e is defined as $\mathcal{U}^e$, and the maximum tolerable dropping ratio for each UE in this slice is $d_e^{\max}$. The packet dropping ratio constraint is given by

$$d_u(T)\leq d_e^{\max}. \tag{20}$$

Our target is to jointly optimize the subframe and subchannel allocation for maximizing the long-term sum rate under the UEs'

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

packet dropping ratio constraints, i.e.,

$$\max_{\left\{\phi_{b,u}^{n}(T),\phi_{u,b}^{n}(T),f^{b}(T),\forall b,\forall T\right\}} \sum_{T=0}^{\Psi}\sum_{u=1}^{U}\left[\psi_{u}^{\mathrm{DL}}(T)+\psi_{u}^{\mathrm{UL}}(T)\right],$$
$$\text{s.t.} \qquad d_{u}(T)\leq d_{e}^{\max},\forall u,\forall T, \tag{21}$$

where Ψ is the total number of time frames.

III. Decentralized Partially Observable MDP for D-TFDD Networks

All the BSs coordinate to control the inter-cell interference and serve UEs in a decentralized way. Each BS independently makes the resource allocation decisions based on its local observations, with the aim of maximizing the long-term expected sum rate while satisfying the local QoS requirements of its serving UEs. We model this cooperative multi-agent task as a Dec-POMDP.

State: Denote the joint state space of all BSs by $\mathcal{S}=\otimes\mathcal{S}^{b}$, $\forall b\in\mathcal{B}$, with $\otimes$ as the Cartesian product, where $\mathcal{S}^{b}$ is the set of states of BS b. Considering that each BS only has partial observations of the network due to privacy issues. We denote the state of BS b by

$$\mathbf{s}^{b}(T)=\left\{\left(\hat{Q}_{u}^{\mathrm{UL}}(T),\hat{Q}_{u}^{\mathrm{DL}}(T)\right)\middle|\, u\in\mathcal{U}^{b}\right\}, \tag{22}$$

which includes the current UL and DL queue lengths of all UEs served by this BS. The joint state of the network is denoted by $\mathbf{s}(T)=\otimes\mathbf{s}^{b}(T)\in\mathcal{S}$.

Action: Denote the action space of BS b by $\mathcal{A}^{b}$ and the joint action space of all BSs by $\mathcal{A}=\otimes\mathcal{A}^{b}$, $\forall b\in\mathcal{B}$. Let $\mathbf{a}^{b}(T)\in\mathcal{A}^{b}$ represent the action of BS b in time frame T. Each BS's action is to decide the number of DL subframes $f^{b}(T)$, the DL subchannel allocation $\phi_{b,u}^{n}(T)$ and UL subchannel allocation $\phi_{u,b}^{n}(T)$, i.e.,

$$\mathbf{a}^{b}(T)=\left\{\left(f^{b}(T),\left\{\phi_{b,u}^{n}(T)\right\}_{n\in\mathcal{N}},\right.\right.$$
$$\left.\left.\left\{\phi_{u,b}^{n}(T)\right\}_{n\in\mathcal{N}}\right)\middle|\, u\in\mathcal{U}^{b}\right\}. \tag{23}$$

Remark 1: We derive the size of action space for BS b as follows. Take the UL subchannel allocation for BS b as an example. Let J denote the number of UEs that are allocated with at least one UL subchannels in this cell and η_{j} denote the non-zero number of UL subchannels allocated to the j-th UE in this UE set. For each time frame, the UL subchannel allocation action has $\mathrm{Num}^{\mathrm{UL}}$ number of possible choices, i.e.,

$$\mathrm{Num}^{\mathrm{UL}}=\mathrm{Num}^{\mathrm{DL}}=\sum_{J=0}^{|\mathcal{U}^{b}|}\sum_{\eta_{1}=1}^{N-J+1}\sum_{\eta_{2}=1}^{N-J+2-\eta_{1}}\cdots$$
$$\sum_{\eta_{J}=1}^{N-\sum_{j=1}^{J-1}\eta_{j}} C_{|\mathcal{U}^{b}|}^{J}C_{N}^{\eta_{1}}C_{N-\eta_{1}}^{\eta_{2}}\cdots C_{N-\sum_{j=1}^{J-1}\eta_{j}}^{\eta_{J}}, \tag{24}$$

which is related to the total number of subchannels N and UEs $|\mathcal{U}^{b}|$ served by BS b. First, BS b selects $J\in\{0,\ldots,|\mathcal{U}^{b}|\}$ out of $|\mathcal{U}^{b}|$ UEs for subchannel assignment, which has $C_{|\mathcal{U}^{b}|}^{J}$ number of choices. Then, BS b sequentially assigns the subchannels to these J UEs, where the j-th UE can select from the remaining $N-\sum_{j'=1}^{j-1}\eta_{j'}$ subchannels and has $C_{N}^{\eta_{j}}-\sum_{j'=1}^{j-1}\eta_{j'}$ number of choices. We further denote the number of possible choices of DL subchannel allocation action by $\mathrm{Num}^{\mathrm{DL}}$ and can easily deduce that $\mathrm{Num}^{\mathrm{DL}}=\mathrm{Num}^{\mathrm{UL}}$. Moreover, for each time frame, since the BS allocates the first $f^{b}(T)$ successive subframes for DL transmission, the subframe configuration action has $F+1$ number of possible choices. Therefore, the size of the action space $|\mathcal{A}^{b}|$ is given by

$$\left|\mathcal{A}^{b}\right|=\mathrm{Num}^{\mathrm{UL}}\times\mathrm{Num}^{\mathrm{DL}}\times(F+1), \tag{25}$$

which increases rapidly with the number of subchannel N, the number of UE $|\mathcal{U}^{b}|$ and the number of subframe F.[1]

Define policy of BS b as a function mapping from the state space to action space, which is expressed as a conditional probability density function of

$$\pi^{b}\left(\mathbf{a}^{b}(T)\left|\mathbf{s}^{b}(T)\right.\right)$$
$$=\Pr\left(\mathbf{A}^{b}(T)=\mathbf{a}^{b}(T)\left|\mathbf{S}^{b}(T)=\mathbf{s}^{b}(T)\right.\right), \tag{26}$$

where $\mathbf{S}^{b}(T)$ and $\mathbf{A}^{b}(T)$ denote the state and action of BS b in time frame T that have not yet been observed or taken, and $\mathbf{s}^{b}(T)$ and $\mathbf{a}^{b}(T)$ represent the observed state and executed action, respectively. We denote the policy profile of the BSs by $\pi=[\pi^{1},\ldots,\pi^{B}]$.

Transition probability: The joint action $\mathbf{a}(T)=\otimes\mathbf{a}^{b}(T)\in\mathcal{A}$ causes the state transition of all BSs in time frame T. The transition probability ρ of the entire network environment that moves from state $\mathbf{s}(T)$ to state $\mathbf{s}(T+1)$ after taking joint action $\mathbf{a}(T)$ is assumed to be unknown by the BSs.

Reward: Each BS receives an immediate reward $r^{b}(T)$ when action $\mathbf{a}^{b}(T)$ is executed in state $\mathbf{s}^{b}(T)$, i.e.,

$$r^{b}(T)=\sum_{u\in\mathcal{U}^{b}}\left[\psi_{u}^{\mathrm{DL}}(T)+\psi_{u}^{\mathrm{UL}}(T)\ -\mathbb{1}\left(d_{u}(T)>d_{e}^{\max}\right)\varpi\right], \tag{27}$$

where $\psi_{u}^{\mathrm{DL}}(T)$ is the DL rate given in (15), $\psi_{u}^{\mathrm{UL}}(T)$ is the UL rate given in (16), and ϖ is a positive constant that penalizes the violation of the QoS requirements. We assume that each BS can only observe its own reward as the reward is private information.

Due to the correlated queue dynamics and inter-cell interference, action $\mathbf{a}^{b}(T)$ affects not only the achievable rate and dropping ratio of BS b, but also that of other BSs in the subsequent time frames. We characterize the long-term sum-reward of all BSs in the cooperative system by $V(T)$, i.e.,

$$V(T)=\sum_{l=T}^{\Psi}\sum_{b=1}^{B}\gamma^{l-T}r^{b}(l), \tag{28}$$

where $\gamma\in[0,1]$ is the discount factor that reflects the effect of future rewards.

Based on the above discussions, we define Dec-POMDP as a five-tuple of $(\{\mathcal{S}^{b}\}_{b\in\mathcal{B}},\{\mathcal{A}^{b}\}_{b\in\mathcal{B}},\rho,\{r^{b}\}_{b\in\mathcal{B}},\gamma)$. However, it is difficult to know the exact value of $V(T)$, due to the randomness of future states and actions. More specifically, the future states depend on the transition probability ρ, and the future actions

[1]For example, when $N=5$, $|\mathcal{U}^{b}|=3$ and $F=10$, we have $|\mathcal{A}^{b}|=11534336$.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

depend on the joint policy π. Given joint action $\mathbf{A}(T) = \mathbf{a}(T)$ is taken at joint state $\mathbf{S}(T) = \mathbf{s}(T)$, we define the conditional expectation of the long-term sum-reward of all BSs under joint policy π as

$$Q^{\pi}\left(\mathbf{s}(T), \mathbf{a}(T)\right) = \mathbb{E}_{\mathbf{S}(T+1), \mathbf{A}(T+1), \ldots}\left[V(T) \,|\mathbf{S}(T) = \mathbf{s}(T), \mathbf{A}(T) = \mathbf{a}(T)\right], \tag{29}$$

which is also defined as the state-action value function. The objective of the BSs is to find the optimal joint policy $\pi^* = [\pi^{1*}, \ldots, \pi^{B*}]$ that maximizes the state-action value function in (29), i.e.,

$$\pi^* = \arg\max_{\pi} Q^{\pi}\left(\mathbf{s}(T), \mathbf{a}(T)\right), \forall \mathbf{s}(T), \forall \mathbf{a}(T). \tag{30}$$

IV. Federated Reinforcement Learning Based Resource Allocation Algorithm

To solve the above resource allocation problem, there are two challenges to be addressed. The first challenge is to handle the large-scale discrete action space. As shown in *Remark 1*, the dimensionality of action space for BS b is high when the numbers of subchannels, subframes and UEs are large. The conventional value-based RL algorithms, e.g., deep Q network, may suffer from a long training convergence time and is even not tractable due to the curse of dimensionality. The policy-based algorithms, e.g., DDPG, can deal with continuous action space and achieve good convergence [34], [35]. To deal with the dimensionality of the large-scale discrete action space, we first adopt a DDPG-based algorithm to generate actions for each BS in the continuous action space, and then discretize the actions based on Wolpertinger policy [36] to reduce the action mapping errors. Moreover, the second challenge is to jointly optimize the sum-reward in a decentralized manner. The conventional MARL algorithms with centralized training, i.e., MADDPG [37], suffer from the threats of privacy leakage since each BS is required to upload its local private information (e.g., states, actions and rewards) to the centralized controller for joint model training. To jointly optimize training among BSs, we propose a federated reinforcement learning algorithm named FWDDPG, where each BS performs local model training in a decentralized manner and updates the local model parameters by aggregating the parameters received from its one-hop neighbors. The architecture of our proposed algorithm is shown in Fig. 2 and the details will be described in the following subsections.

A. Action Generation Based on Wolpertinger Policy

We adopt an actor-critic based algorithm with a deterministic policy, i.e., DDPG, to deal with the high dimensional action space, where the policy maps from state to a deterministic action instead of a probability distribution over the actions. However, the actions generated by the deterministic policy are continuous and may not be within the action space of $\mathcal{A}^b$. To solve this problem, we further discretize the output of the actor network and adopt the Wolpertinger policy for mapping error reduction,

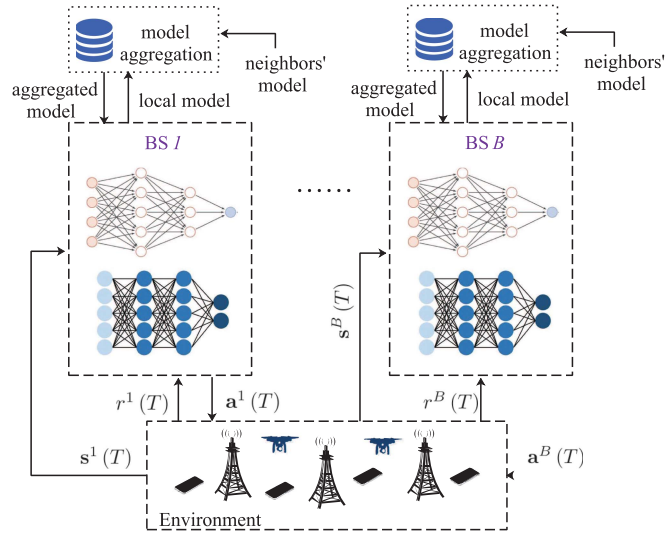


Fig. 2. The architecture of the proposed FWDDPG algorithm.

which can be divided into two phases: action generation and action refinement.

Action generation: Given state $\mathbf{s}^b(T)$, the actor network of BS b generates a proto-action $\hat{\mathbf{a}}^b(T)$ based on deterministic policy $\hat{\mu}$, i.e.,

$$\hat{\mathbf{a}}^b(T) = \hat{\mu}\left(\mathbf{s}^b(T); \theta^b(T)\right), \tag{31}$$

where $\theta^b(T)$ is the neural network parameter to approximate policy $\hat{\mu}$ of BS b. However, $\hat{\mathbf{a}}^b(T)$ is continuous and may not be a valid action in the discrete action set $\mathcal{A}^b$. Therefore, we map $\hat{\mathbf{a}}^b(T)$ to the elements of $\mathcal{A}^b$, i.e.,

$$\mathcal{A}_k^b(T) = \delta_k^b\left(\hat{\mathbf{a}}^b(T)\right) = \underset{\mathbf{a}^b(T) \in \mathcal{A}^b}{\overset{k}{\arg\min}} \left\|\mathbf{a}^b(T) - \hat{\mathbf{a}}^b(T)\right\|_2, \tag{32}$$

where $\delta_k^b(\hat{\mathbf{a}}^b(T))$ is the k-nearest-neighbor (k-NN) mapping function to return the k actions in $\mathcal{A}^b$ that are closest to $\hat{\mathbf{a}}^b(T)$ by Euclidean distance.

Action refinement: We select the best action out of k candidate actions generated by (32). The parameterized state-action value function of BS b is defined as $Q(\mathbf{s}^b(T), \mathbf{a}^b(T); \omega^b(T))$, where $\omega^b(T)$ is the critic neural network parameter. To avoid picking an action with a low Q-value, we adopt Wolpertinger policy, i.e.,

$$\begin{aligned} &\mu\left(\mathbf{s}^b(T); \theta^b(T), \omega^b(T)\right) \\ &= \underset{\mathbf{a}^b(T) \in \mathcal{A}_k^b(T)}{\arg\max} \; Q\left(\mathbf{s}^b(T), \mathbf{a}^b(T); \omega^b(T)\right) \\ &= \mathbf{a}^b(T), \end{aligned} \tag{33}$$

to refine the output of the critic network by selecting the action with the highest Q-value among the k-NN actions. The Wolpertinger policy's algorithm is given in Algorithm 1.

Remark 2: Note that the size k of the generated action set is task specific. There is a tradeoff between policy quality and computational cost. The policy quality can be evaluated by the difference between the highest Q-value achieved over all possible actions and the expected highest Q-value achieved by these k closest actions [36]. It can be deduced that the policy

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1: Wolpertinger Policy for BS b.

1: Observe state $\mathbf{s}^b(T)$ from environment.
2: Receive proto-action within continuous action space based on the actor network: $\hat{\mathbf{a}}^b(T) = \hat{\mu}(\mathbf{s}^b(T); \theta^b(T))$.
3: Retrieve a set of k approximately closest actions to $\hat{\mathbf{a}}^b(T)$: $\mathcal{A}_k^b(T) = \delta_k^b(\hat{\mathbf{a}}^b(T))$.
4: Compute the action with the highest Q-value: $\mathbf{a}^b(T) = \mu(\mathbf{s}^b(T); \theta^b(T), \omega^b(T))$.

quality increases with k. Moreover, the additional computational complexity for Wolpertinger policy grows linearly with k, where the details will be discussed in *Remark 3* in the next subsection.

B. The Local WDDPG Policy Training

In this subsection, we will discuss the training process of the actor and critic networks for WDDPG algorithm. We consider the model-free scenario with no prior distribution of the network environment, and adopt the conventional random strategies for initialization, i.e., randomly initialize critic and actor networks with parameters $\omega^b(0)$ and $\theta^b(0)$, $\forall b \in \mathcal{B}$, respectively.

In our proposed algorithm, we adopt off-policy, which involves two different policies of behavioral and target policies. We adopt Wolpertinger policy with Ornstein Uhlenbeck (OU) noise as the behavioral policy to encourage exploration, and use Wolpertinger policy without noise as the target policy. The learning data generated by behavioral policy is defined as a 4-element tuple $(\mathbf{s}^b(T), \mathbf{a}^b(T), r^b(T), \mathbf{s}^b(T+1))$ and is stored in the replay buffer (RB). The target policy uses the samples stored in the RB to update itself. With the experience replay and target networks, we next introduce the actor and critic network training processes.

Actor network training: We define the target function of the actor network as the expectation of the parameterized state-action value function, i.e., $J(\theta^b(T)) = \mathbb{E}_{\mathbf{S}^b(T)}[Q(\mathbf{S}^b(T), \mu(\mathbf{S}^b(T); \theta^b(T), \omega^b(T)); \omega^b(T))]$. The expectation is taken over all possible values of unobserved state $\mathbf{S}^b(T)$ in time frame T to remove the state randomness. To approximate the expectation over $\mathbf{S}^b(T)$, we take a minibatch of I transitions from RB, where the i-th transition is denoted by $(\mathbf{s}^b(i), \mathbf{a}^b(i), r^b(i), \mathbf{s}^b(i+1))$. We aim to find the optimal $\theta^b(T)$ that maximizes $J(\theta^b(T))$ by adopting a deterministic policy gradient method, where the gradient $\nabla_{\theta^b} J(\theta^b(T))$ can be derived in (34) shown at the bottom of this page. However, as the action $\mathbf{a}^b(i) = \mu(\mathbf{s}^b(i); \theta^b(i), \omega^b(T))$ executed by BS b is discrete, the parameter $\theta^b(T)$ of the actor network can not be directly updated via deterministic policy gradient method. Therefore, we use the continuous proto-action $\hat{\mathbf{a}}^b(i) = \hat{\mu}(\mathbf{s}^b(i); \theta^b(T))$ instead to derive the gradient of $\nabla_{\theta^b} J(\theta^b(T))$ as given by (35), shown at the bottom of this page. Accordingly, the parameter $\theta^b(T+1)$ is updated by

$$\theta^b(T+1) \leftarrow \theta^b(T) + \beta^b \nabla_{\theta^b} J\left(\theta^b(T)\right), \tag{36}$$

where β^b is the learning rate of actor network.

Critic network training: We adopt temporal-difference (TD) learning to update $\omega^b(T)$. With the transition $(\mathbf{s}^b(i), \mathbf{a}^b(i), r^b(i), \mathbf{s}^b(i+1))$ sampled from RB, the estimated value called TD target is given by $r^b(i) + \gamma Q(\mathbf{s}^b(i+1), \mu(\mathbf{s}^b(i+1); \theta^b(T), \omega^b(T)); \omega^b(T))$ and the output of the current critic network can be given by $Q(\mathbf{s}^b(i), \mathbf{a}^b(i); \omega^b(T))$. Note that bootstrapping occurs if we use the current critic network parameter $\omega^b(T)$ for both the TD calculation and updating, which may cause a non-uniform overestimation of the optimal state-action value function. To avoid the bootstrapping and reduce the overestimation, we introduce the target actor and critic networks that are copied from the original actor and critic networks. Accordingly, the parameterized state-action value function of the target critic network of BS b is denoted by $\tilde{Q}(\mathbf{s}^b(i+1), \tilde{\mu}(\mathbf{s}^b(i+1); \tilde{\theta}^b(T), \tilde{\omega}^b(T)); \tilde{\omega}^b(T))$, where $\tilde{\mu}(\mathbf{s}^b(i+1); \tilde{\theta}^b(T), \tilde{\omega}^b(T))$ is the target Wolpertinger policy, and $\tilde{\theta}^b(T)$ and $\tilde{\omega}^b(T)$ respectively denote the parameters of the target actor and critic networks. The TD target with respect to the target networks is given by $r^b(i) + \gamma \tilde{Q}(\mathbf{s}^b(i+1), \tilde{\mu}(\mathbf{s}^b(i+1); \tilde{\theta}^b(T), \tilde{\omega}^b(T)); \tilde{\omega}^b(T))$. The loss function and its gradient are given by (37) and (38), shown at the bottom of this page, respectively. The parameter

$$\nabla_{\theta^b} J\left(\theta^b(T)\right) = \frac{1}{I} \sum_i \nabla_{\theta^b} Q\left(\mathbf{s}^b(i), \mu\left(\mathbf{s}^b(i); \theta^b(T), \omega^b(T)\right); \omega^b(T)\right). \tag{34}$$

$$\nabla_{\theta^b} J\left(\theta^b(T)\right) \approx \frac{1}{I} \sum_i \left(\nabla_{\theta^b} \hat{\mu}\left(\mathbf{s}^b(i); \theta^b(T)\right) \cdot \nabla_{\hat{\mathbf{a}}^b} Q\left(\mathbf{s}^b(i), \hat{\mathbf{a}}^b(i); \omega^b(T) \middle| \hat{\mathbf{a}}^b(i) = \hat{\mu}\left(\mathbf{s}^b(i); \theta^b(T)\right)\right)\right). \tag{35}$$

$$\text{Loss}\left(\omega^b(T)\right) = \frac{1}{2I} \sum_i \left[Q\left(\mathbf{s}^b(i), \mathbf{a}^b(i); \omega^b(T)\right) - r^b(i) - \gamma \tilde{Q}\left(\mathbf{s}^b\left(i+1\right), \tilde{\mu}\left(\mathbf{s}^b\left(i+1\right); \tilde{\theta}^b(T), \tilde{\omega}^b(T)\right); \tilde{\omega}^b(T)\right)\right]^2. \tag{37}$$

$$\nabla_{\omega^b} \text{Loss}\left(\omega^b(T)\right) = \frac{1}{I} \sum_i \left[\left(Q\left(\mathbf{s}^b(i), \mathbf{a}^b(i); \omega^b(T)\right) - r^b(i) - \gamma \tilde{Q}\left(\mathbf{s}^b\left(i+1\right), \tilde{\mu}\left(\mathbf{s}^b\left(i+1\right); \tilde{\theta}^b(T), \tilde{\omega}^b(T)\right); \tilde{\omega}^b(T)\right)\right) \cdot \nabla_{\omega^b} Q\left(\mathbf{s}^b(i), \mathbf{a}^b(i); \omega^b(T)\right)\right] \tag{38}$$

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

$\omega^b(T+1)$ can be updated by

$$\omega^b\left(T+1\right) \leftarrow \omega^b(T) + \bar{\beta}^b \nabla_{\omega^b} \text{Loss}\left(\omega^b(T)\right), \tag{39}$$

where $\bar{\beta}^b$ is the learning rate of the critic network. Moreover, the target critic and actor networks are updated every step with a small step size to confirm soft updating, i.e.,

$$\tilde{\omega}^b\left(T+1\right) \leftarrow \kappa\omega^b\left(T+1\right) + \left(1-\kappa\right)\tilde{\omega}^b(T) \tag{40}$$

and

$$\tilde{\theta}^b\left(T+1\right) \leftarrow \kappa\theta^b\left(T+1\right) + \left(1-\kappa\right)\tilde{\theta}^b(T), \tag{41}$$

where κ is the update step size.

Remark 3: The computational complexity of WDDPG primarily depends on the actor and critic network architectures. Let $H_{\rm a}$ and $H_{\rm c}$ denote the total numbers of hidden layers of actor and critic networks. The h-th hidden layer for actor network and critic network involves $\zeta_{{\rm a},h}$ and $\zeta_{{\rm c},h}$ numbers of neurons, respectively. Recall that $|\mathcal{U}^b|$ denotes the number of UEs served by BS b. For the actor network, the number of neurons in the input layer depends on the dimension of the state, and the number of neurons in the output layer depends on the dimension of the action. Since the state of BS b is defined as the current UL and DL queue lengths of its serving UEs, there are $2|\mathcal{U}^b|$ neurons in the input layer. And there are 3 neurons in the output layer corresponding to the three types of actions i.e., the number of DL subframe, DL and UL subchannel allocations. Accordingly, the number of weights in the input layer, the h-th $(2 \le h \le H_{\rm a}-1)$ hidden layer and the last hidden layer can be computed as $2|\mathcal{U}^b|\zeta_{{\rm a},1}$, $\zeta_{{\rm a},h-1}\zeta_{{\rm a},h}$ and $3\zeta_{{\rm a},H_{\rm a}}$, respectively. For the critic network, the number of neurons in the input layer is the dimension of the state and action, i.e., $2|\mathcal{U}^b|+3$, and there is 1 neuron in the output layer. Then the numbers of weights in the input layer, the h-th $(2 \le h \le H_{\rm c}-1)$ hidden layer and the last hidden layer can be computed as $(2|\mathcal{U}^b|+3)\zeta_{{\rm c},1}$, $\zeta_{{\rm c},h-1}\zeta_{{\rm c},h}$ and $\zeta_{{\rm c},H_{\rm c}}$, respectively. The computational complexity of BS b in backward propagation training is given by $\mathcal{O}(\iota^{\rm BP}[2|\mathcal{U}^b|\zeta_{{\rm a},1} + \sum_2^{H_{\rm a}}\zeta_{{\rm a},h-1}\zeta_{{\rm a},h} + 3\zeta_{{\rm a},H_{\rm a}} + (2|\mathcal{U}^b|+3)\times\zeta_{{\rm c},1} + \sum_2^{H_{\rm c}}\zeta_{{\rm c},h-1}\zeta_{{\rm c},h} + \zeta_{{\rm c},H_{\rm c}}])$, where $\iota^{\rm BP}$ denotes the computational complexity for training a single weight in backward propagation. The computational complexity for training a single weight in forward propagation is similar to that in backward propagation. Here, we focus on the additional computational complexity caused by the Wolpertinger policy in forward propagation training, which is given by $\mathcal{O}(\iota^{\rm AP}k[(2|\mathcal{U}^b|+3)\zeta_{{\rm c},1} + \sum_2^{H_{\rm c}}\zeta_{{\rm c},h-1}\zeta_{{\rm c},h} + \zeta_{{\rm c},H_{\rm c}}])$, where $\iota^{\rm AP}$ is the computational complexity of training a single weight in forward propagation.

C. Global Policy Training With Federated Learning

Our objective is to find the optimal joint policy π^* that maximizes the global state-action value function in (30). The challenge is to maximize social welfare in a decentralized manner with local observations. If each BS independently adopts WDDPG algorithm, there is no communication overhead, but it suffers from low cooperation efficiency and can only adapt its resource allocation to the local traffic instead of the network. Due to the lack of global state information, it is difficult for the BSs to mitigate inter-cell interference among themselves. In order to alleviate inter-cell interference, it is necessary for the BSs to share local information with each other for joint model training. Although some conventional algorithms, e.g., MADDPG, can jointly train the critic networks at the centralized controller, each BS is required to upload its local states, actions, and rewards to the controller, which may cause privacy leakage issues and introduce high communication overhead. To protect privacy of the agents, we adopt a decentralized FL framework for joint model training among the BSs [38], [39], where each BS exchanges the local critic network parameters with its one-hop neighbors in every ℓ time frames. This enables the decentralized BSs to update their local critic network parameters to improve the global resource allocation efficiency with relatively low communication overhead. Note that our proposed scheme can indirectly exchange parameter information with multi-hop neighbors due to the propagation effect across multiple rounds of parameter update.

We consider the D-TFDD network topology as a undirected graph model $\mathcal{G} = (\mathcal{B}, \varrho)$, where $\mathcal{B}$ is the set of BS nodes and ϱ represents the set of edges. An edge $(b, b') \in \varrho$ means that BS b' is the one-hop neighbor of BS b. Let $\Upsilon^b = \{b \in \mathcal{B} : (b, b') \in \varrho\}$ be the set of one-hop neighbors of BS b, where $|\Upsilon^b|$ and $|\Upsilon^{b'}|$ are the numbers of neighbors of BS b and b', respectively. Due to the differences in training capabilities and network connections of each neighboring BS b', it is wise for BS b to weight the model parameters received from its one-hop neighbors differently according to their influences. We denote the weighting matrix by $Z = [z_{b,'b}]_{B\times B}$, where $z_{b,'b}$ weights the parameter sent from BS b' to BS b. By adopting Metropolis weights [40] in our model, we have

$$z_{b,'b} = \begin{cases} \frac{1}{1+\max\left\{|\Upsilon^b|,\left|\Upsilon^{b'}\right|\right\}}, & \forall\left(b,b'\right) \in \varrho, \\ 1-\sum_{b''\in\Upsilon^b(T)} z_{b,''b}, & b = b,' \forall b \in \mathcal{B}. \end{cases} \tag{42}$$

For every ℓ time frames, each BS exchanges parameter $\hat{\omega}^b(T)$ with its one-hop neighbors for global model training, where $\hat{\omega}^b(T) = \omega^b(T) + \bar{\beta}^b\nabla_{\omega^b}\text{Loss}(\omega^b(T))$. And then, each BS b aggregates the received parameters $\hat{\omega}^{b'}(T)$ from its one-hop neighbors based on the Metropolis weights and updates the parameter of the critic network in time frame $T+1$. For the rest of the time frames, BS b directly uses its local parameter $\hat{\omega}^b(T)$ to update its critic network. Therefore, the parameter update of the critic network can be expressed as

$$\begin{cases} \omega^b\left(T+1\right) \leftarrow \sum_{b'=1}^{B} z_{b,b'}\hat{\omega}^{b'}(T), & \text{if } T\%\ell = 0, \\ \omega^b\left(T+1\right) \leftarrow \hat{\omega}^b(T), & \text{otherwise.} \end{cases} \tag{43}$$

We summarize the proposed FWDDPG algorithm in Algorithm 2 and Fig. 3.

Remark 4: The computational complexity of the peer-to-peer FL architecture depends on the aggregation of critic network parameters from one-hop neighbors. For BS b, the critic network parameters of itself and its one-hop neighbors need to be multiplied by their respective weights and then summed up as the new critic network parameters for global training. We therefore can deduce that the computational complexity of the peer-to-peer FL architecture is $\mathcal{O}(\hbar[\sum_{b=1}^{B}(2|\Upsilon^b(T)|+1)])$, where $\hbar$ is

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

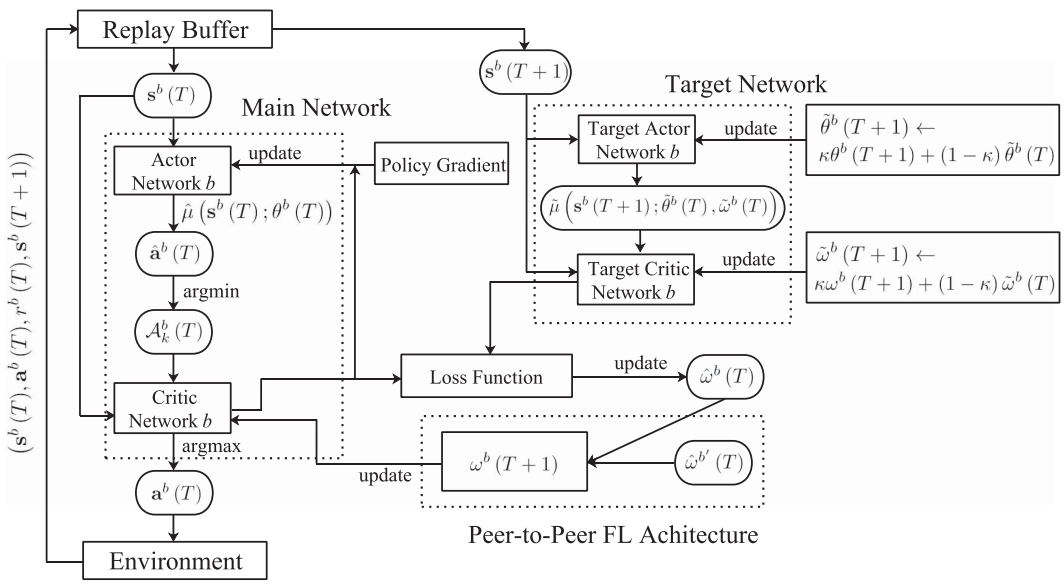


Fig. 3. The framework of the proposed FWDDPG algorithm.

the number of rounds for global training, and the number of additions and multiplications are $|\Upsilon^b(T)|$ and $|\Upsilon^b(T)|+1$ for BS b, respectively.

V. Simulation Results and Discussions

For simulations, we consider a D-TFDD network covers a square area of 3 km×3 km. Without loss of generality, we consider ten BSs with the height of 10 m serves 30 active UEs (including GUEs and UAVs), where each BS serves three UEs in its serving area with 5 subchannels and 10 subframes. The transmit power of BSs and UEs are 24 dBm and 23 dBm, respectively. The noise power at BSs, GUEs and UAVs are -91 dBm, -95 dBm and -99 dBm, respectively [21] [41]. As for the channel modeling, we set the ITU model factors $\{c_1, c_2, c_3\}$ as $\{0.3, 500, 20\}$ and the fading parameter $m_{\text{tx,rx}} = 1$ according to [32]. The parameters of the path loss model are listed in Table I according to [21] and [32]. The SINR threshold of UEs and BSs are set as 0 dB and -3 dB, respectively, and the bandwidth of each subchannel is 10 MHz. The duration of each subframe is 1 ms. In the following discussions, we assume GUEs and UAVs are with slice types 1 and 2, respectively. Unless otherwise specified, the slice model parameters are given as follows. The maximum dropping ratio for GUEs and UAVs are set as $d_1^{\max} = 0.3$ and $d_2^{\max} = 0.1$, respectively. The average UL and DL packet sizes for GUEs are $\lambda_1^{\text{UL}} = 150$ KB and $\lambda_1^{\text{DL}} = 200$ KB, and those for UAVs are $\lambda_2^{\text{UL}} = 50$ KB and $\lambda_2^{\text{DL}} = 80$ KB, respectively. The buffer sizes for GUEs and UAVs are $\hat{Q}_1^{\max} = 250$ KB and $\hat{Q}_2^{\max} = 150$ KB, respectively.

The total number of training epochs is 1000 and the number of steps for each epoch is 300. We adopt two hidden layers for both actor and critic networks, where the first hidden layer has 60 neurons and the second hidden layer has 50 neurons. We train the neural networks by Adam optimizer, where we set the learning rates for the actor and critic networks as 0.0001 and 0.001, respectively. For each time frame, a mini-batch of

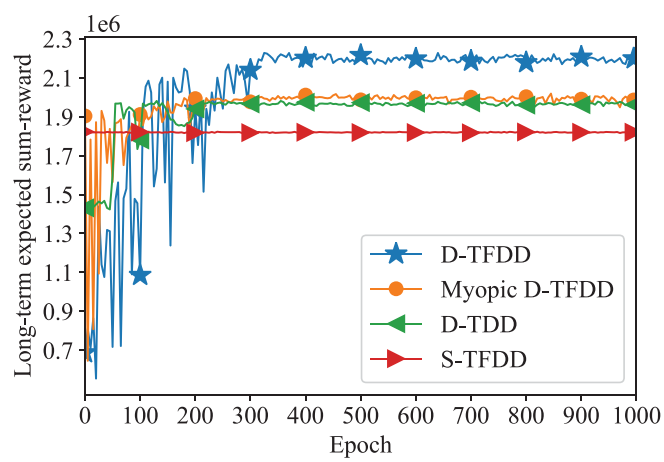


Fig. 4. Long-term expected sum-reward of all BSs for different types of TDD schemes.

300 experiences are randomly sampled every time from RB that is capable of storing 1000000 past experiences. We update the target critic or actor network by step size $\tau = 0.001$. We set the discount factor $\gamma = 0.99$. Unless otherwise specified, we adopt $k = 120$ as the default size of the actions generated by Wolpertinger policy.

In Fig. 4, we plot the long-term expected sum-reward of the BSs over 1000 training epochs. By adopting the proposed FWDDPG algorithm, we compare our D-TFDD scheme with other benchmark TDD schemes, i.e., S-TFDD, myopic D-TFDD and D-TDD. For our proposed D-TFDD scheme, both the subframe and subchannel allocations are adaptive to the UEs' dynamic traffic demands, aiming to maximize the long-term expected sum-reward of all BSs. For static-TFDD (S-TFDD) scheme, all the BSs adopt the same subframe and subchannel configurations, which are pre-defined and non-adaptive throughout time. For

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

Algorithm 2: FWDDPG Based Resource Allocation Algorithm.

Randomly initialize critic and actor networks with parameters $\omega^b(0)$ and $\theta^b(0)$, $\forall b \in \mathcal{B}$.
Initialize target critic and actor networks $\widetilde{\omega}^b(0) \leftarrow \omega^b(0)$, $\widetilde{\theta}^b(0) \leftarrow \theta^b(0)$, $\forall b \in \mathcal{B}$.
Initialize the k-NN mapping function δ_k^b using elements of $\mathcal{A}^b$, $\forall b \in \mathcal{B}$.
Initialize RB.
Initialize the number of subchannels N, the number of subframes F, the number of BSs B and the number of UEs $|\mathcal{U}^b|$ served by BS b.
for Epoch $= 1, 2, \ldots$ **do**
  Initialize the global state $\mathbf{s}(0)$.
  for $T = 0, 1, 2, \ldots$ **do**
    for $b = 1$ to B **do**
      Observe local state $\mathbf{s}^b(T)$.
      Generate local action based on the Wolpertinger policy: $\mathbf{a}^b(T) = \mu(\mathbf{s}^b(T); \theta^b(T), \omega^b(T))$.
    end for
    Execute joint action $\mathbf{a}(T) = (\mathbf{a}^1(T), \ldots, \mathbf{a}^B(T))$.
    for $b = 1$ to B **do**
      Observe reward $r^b(T)$ and new state $\mathbf{s}^b(T+1)$.
      Store transition $(\mathbf{s}^b(T), \mathbf{a}^b(T), r^b(T), \mathbf{s}^b(T+1))$ in RB.
      Randomly sample a minibatch of I transitions from RB.
      Update the critic by minimizing the loss in (37), then update $\hat{\omega}^b(T) \leftarrow \omega^b(T)$.
      Update the actor using the sampled gradient according to (35), then update $\theta^b(T+1) \leftarrow \theta^b(T)$.
      Update critic network according to (43).
      Update the target networks: $\widetilde{\omega}^b(T+1) \leftarrow \kappa\omega^b(T+1) + (1-\kappa)\widetilde{\omega}^b(T)$, $\widetilde{\theta}^b(T+1) \leftarrow \kappa\theta^b(T+1) + (1-\kappa)\widetilde{\theta}^b(T)$.
    end for
  end for
end for

myopic D-TFDD scheme, the subframe and subchannel configurations are adaptive to the UEs' dynamic demands in the current time frame only without considering the future rewards. For D-TDD scheme, only the subframe configuration is adaptive to the dynamic traffic demands, aiming at maximizing the long-term expected sum-reward of all BSs, while the subchannel allocation is pre-determined and does not change across time. In Fig. 4, the performance of S-TFDD scheme does not change much over time and is worse than the dynamic schemes since it is not adaptive to the dynamic UE demands. We notice that there are slight jitters along the curve, which is due to the randomness of the channel gains and packet arrivals, although these effects are almost averaged out over the long-term accumulation. We also see that our proposed D-TFDD scheme outperforms all other benchmark schemes. It has better performance than myopic D-TFDD scheme since it considers not only the short-term but also the long-term sum-reward. Furthermore, it takes into account both the dynamic subframe and subchannel allocations and is therefore better than D-TDD scheme.

TABLE I
THE PARAMETERS OF PATH LOSS MODEL

Parameters	Values
BS-to-GUE path loss factor	$A^{\mathrm{L}} = 34.02$ dB, $\alpha^{\mathrm{L}} = 2.2$, $A^{\mathrm{NL}} = 19.56$ dB, $\alpha^{\mathrm{NL}} = 3.9$.
BS-to-UAV path loss factor	$A^{\mathrm{L}} = 34.02$ dB, $A^{\mathrm{NL}} = 20.96$ dB, $\alpha^{\mathrm{L}} = 2.2$, $\alpha^{\mathrm{NL}} = 4.6 - 0.7\log_{10} H_u(T)$.
BS-to-BS path loss factor	$A^{\mathrm{L}} = 38.4$ dB, $\alpha^{\mathrm{L}} = 2$, $A^{\mathrm{NL}} = 49.36$ dB, $\alpha^{\mathrm{NL}} = 4$.
UAV-to-UAV path loss factor	$A^{\mathrm{L}} = 34.02$ dB, $A^{\mathrm{NL}} = 20.96$ dB, $\alpha^{\mathrm{L}} = 2.2$, $\alpha^{\mathrm{NL}} = 4.6 - 0.7\log_{10} H_u(T)$.
GUE-to-GUE path loss factor	$A^{\mathrm{L}} = 38.4$ dB, $\alpha^{\mathrm{L}} = 2$, $A^{\mathrm{NL}} = 49.36$ dB, $\alpha^{\mathrm{NL}} = 4$.

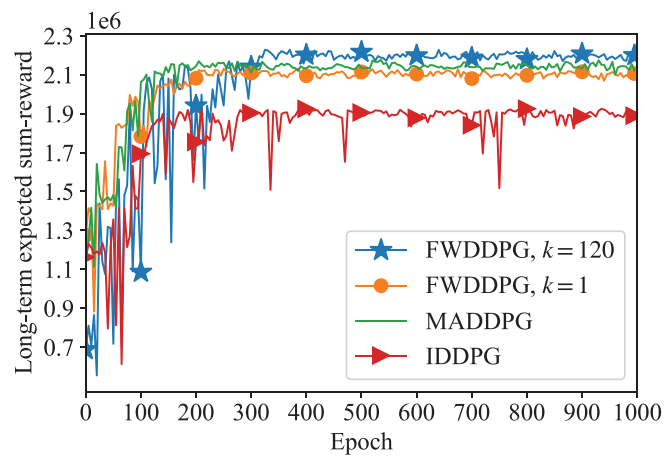


Fig. 5. Long-term expected sum-reward of all BSs for D-TFDD scheme under different MARL algorithms.

Fig. 5 depicts the long-term expected sum-reward of the proposed FWDDPG algorithm and compares it with two benchmark algorithms, i.e., MADDPG [37] and IDDPG. For MADDPG algorithm, the centralized training and decentralized execution framework is adopted, where the BSs upload the local states, actions and rewards to the centralized controller to jointly train the critic network to maximize the long-term expected sum-reward of all BSs in the network. For IDDPG algorithm, each BS trains its DDPG algorithm with local states in a non-cooperative manner, aiming to maximize its local long-term expected reward [25], [26]. In Fig. 5, we see that the sum-reward increases with the number of training epochs, which means all the algorithms can learn from interacting with the environment. Moreover, we see that IDDPG algorithm performs the worst among the three algorithms since the BSs are not cooperative. Next, we compare the performance of the proposed FWDDPG algorithm with MADDPG algorithm. First, we observe that MADDPG algorithm outperforms FWDDPG algorithm with $k = 1$. This is because MADDPG jointly trains the critic networks with the centralized controller, which is more efficient than the decentralized approaches. For $k = 1$, only the discrete action that is closest to the continuous action is selected for

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

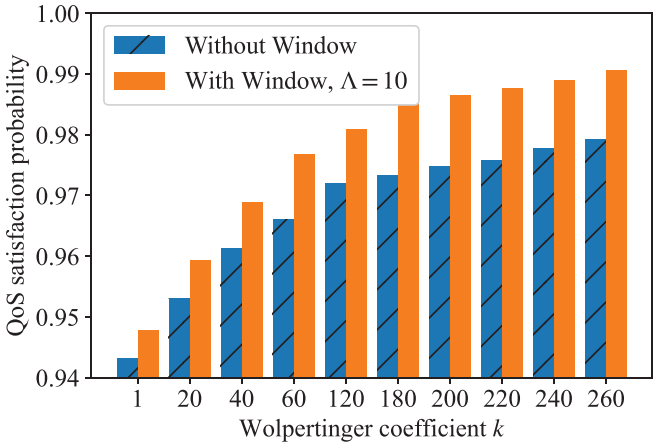


Fig. 6. QoS satisfaction probability in the D-TFDD network versus various Wolpertinger coefficients k.

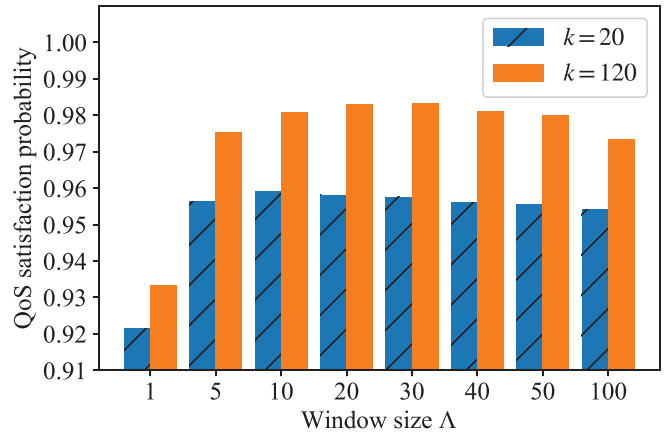


Fig. 7. QoS satisfaction probability versus various window sizes Λ.

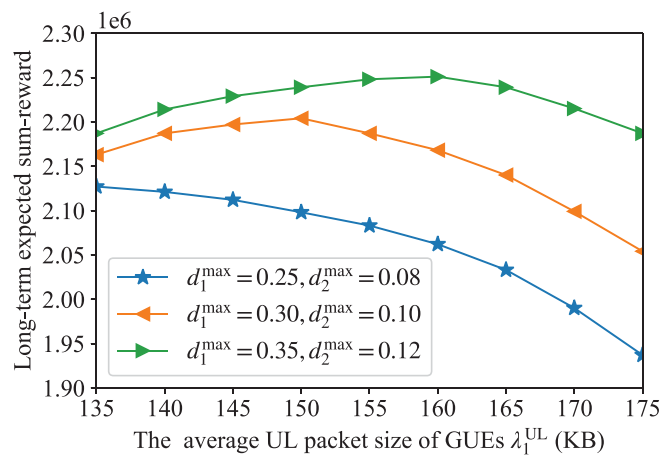


Fig. 8. Long-term expected sum-reward of all BSs versus various average UL packet sizes ($\lambda_1^{\mathrm{DL}} = \lambda_1^{\mathrm{UL}} + 50$ KB, $\lambda_2^{\mathrm{DL}} = \lambda_1^{\mathrm{UL}} - 70$ KB, $\lambda_2^{\mathrm{UL}} = \lambda_1^{\mathrm{UL}} - 100$ KB).

execution, where the proposed algorithm is equivalent to that without Wolpertinger policy. However, this disadvantage can be compensated by adjusting the coefficient k in the proposed FWDDPG algorithm. For example, for $k = 120$, we see that the performance of FWDDPG algorithm exceeds that of MADDPG algorithm. Intuitively, this is because a larger k can help include more candidates of valid actions, which increases the chance of selecting a better policy with a higher Q-value, though it may be at the cost of slower convergence speed.

In Fig. 6, we plot the QoS satisfaction probability (the probability that the packet dropping ratio constraint is satisfied) in the D-TFDD network against the Wolpertinger coefficient k. On the one hand, we can see that the QoS satisfaction probability increases with k. This is because the policy quality improves as k increases, which is consistent with *Remark 2*. On the other hand, the computational complexity of WDDPG algorithm increases linearly with k according to *Remark 3*. We therefore can deduce that there exists an optimal value of k that balances the policy quality and computational complexity. Furthermore, we observe that the QoS satisfaction probability is increased by introducing the sliding window in (19). If no sliding window is adopted, the premature experiences from the very first time frame will be included in the dropping ratio, which therefore reduces the QoS satisfaction probability. By using the sliding window, we can remove the effects of earlier history and thus improve the system performance.

Furthermore, Fig. 7 shows the influence of window size Λ on the QoS satisfaction probability. We can see that the QoS satisfaction probability first increases and then decreases with the window size. When the window size is small, it means that only the latest samples are taken into the estimation of dropping ratio. The small number of samples leads to inaccurate representation of rewards, resulting in a low QoS satisfaction probability. When the window size increases, the increasing number of samples enhances the estimation accuracy of the dropping ratio and thus improves the QoS satisfaction probability. As window size further increases, more samples from the early history are included, which reduces QoS satisfaction probability.

Fig. 8 plots the influence of the average packet size and QoS constraint (i.e., maximum tolerable dropping ratio) on the long-term expected sum-reward of all BSs in the D-TFDD network. When the QoS constraint is not tight (e.g., $d_1^{\max} = \{0.30, 0.35\}$, $d_2^{\max} = \{0.10, 0.12\}$), the sum-reward first increases and then decreases with the average packet size. As the average packet size increases, the sum-reward first increases owning to the improvement in the sum rate. However, with the further increase of the average packet size, the sum-reward decreases due to the violation of the QoS constraints. Furthermore, when the QoS constraint is tight, the sum-reward decreases directly with the average packet size because the QoS requirement is not met.

In Figs. 9 and 10, we discuss the optimal policy for subchannel and subframe allocations. Without loss of generality, we consider a network composed of two BSs as a special case, where each BS allocates 5 subframes and 4 subchannels between two UEs. In Fig. 9, we study the effect of maximum tolerable dropping ratio and average packet arrival rate on the UL subchannel allocation policy, where the results can be extended to DL subchannel allocation. Next, we increase $d_1^{\max}$ and λ_1^{UL} to see the impact on the subchannel allocation ratio. On the one

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

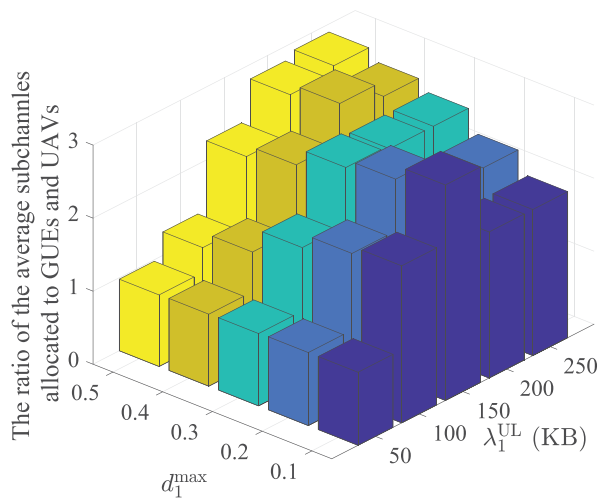


Fig. 9. The ratio of average subchannels allocated to GUEs and UAVs ($\lambda_2^{\mathrm{UL}} = 50$ KB, $d_2^{\max} = 0.1$).

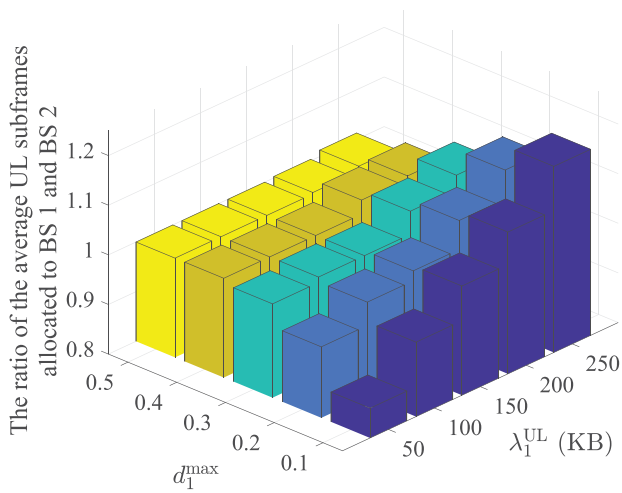


Fig. 10. The ratio of average UL subframes allocated to BS 1 and BS 2 ($\lambda_1^{\mathrm{DL}} = 200$ KB, $\lambda_2^{\mathrm{UL}} = 50$ KB, $\lambda_2^{\mathrm{DL}} = 80$ KB, $d_2^{\max} = 0.1$).

hand, when the average packet arrival rate λ_1^{UL} is relatively small (e.g., $\lambda_1^{\mathrm{UL}} \leq 150$ KB), the number of subchannels allocated to GUE increases with λ_1^{UL}. In this case, the QoS constraint of GUE is easily satisfied and thus the BS allocates more bandwidth resources to GUE to increase its rate. On the other hand, when $d_1^{\max}$ is relatively tight and λ_1^{UL} is relatively large (e.g., $d_1^{\max} = \{0.1, 0.2\}$, $\lambda_1^{\mathrm{UL}} > 150$ KB; $d_1^{\max} = 0.4$, $\lambda_1^{\mathrm{UL}} \geq 200$ KB), the number of subchannels allocated to GUE decreases with λ_1^{UL}. In this case, it is difficult to satisfy the QoS constraint of GUE with heavy traffic load, thus the BS allocates more subchannels to UAV that has lighter data traffic. From the above discussions, we can see that the subchannel allocation needs to balance throughput and QoS constraints. When the bandwidth resources are sufficient, the BS prefers to allocate more sunchannels to the UEs with heavier data traffic loads for throughput enhancement. Otherwise, it allocates fewer subchannels to those UEs whose QoS constraints are difficult to satisfy.

In Fig. 10, we further study the impact of the maximum tolerable dropping ratio and average packet arrival rate on the subframe allocation policy. To illustrate the asymmetric data traffic across the cells, we consider BS 1 serves two GUEs and BS 2 serves two UAVs, respectively. When the QoS constraint of $d_1^{\max}$ is relatively tight (e.g., $d_1^{\max} = \{0.1, 0.2\}$), the number of UL subframes for BS 1 rapidly increases with the average UL packet size λ_1^{UL}. For a large value of λ_1^{UL}, we can see that the subframe allocation is unbalanced between the two BSs in order to meet the heavier UL data traffic demands for GUEs. Moreover, when $d_1^{\max}$ is relatively large (e.g., $d_1^{\max} = \{0.4, 0.5\}$), two BSs have similar subframe configurations, which is to reduce inter-cell interference by controlling the number of unaligned subframes. From the above discussions, we can see that the subframe configuration needs to balance local traffic adaptation and inter-cell interference control. When the resources are sufficient, the BSs prefer to reduce the number of unaligned subframes for inter-cell interference control. In a resource-limited regime, each BS gives more priority to satisfying the local QoS constraints rather than interference avoidance.

VI. Conclusion

In this paper, we proposed a user-centric D-TFDD scheme that fully utilizes both the time-domain and frequency-domain resources to meet the heterogeneous UEs' dynamic traffic demands while alleviating inter-cell interference. Due to the limited observation space of the BSs, we formulated the D-TFDD control problem as a Dec-POMDP that maximizes the long-term expected sum rate of the network subject to the UEs' packet dropping ratio constraints. We proposed a federated reinforcement learning algorithm to solve this problem, where the BSs decide their local time-frequency configurations based on WDDPG algorithm and jointly update the global policy by exchanging the critic network parameters through FL architecture. Simulation results show that the proposed learning-based D-TFDD scheme is superior to other benchmark TDD schemes, and the proposed FWDDPG algorithm outperforms IDDPG and MADDPG algorithms by choosing the proper Wolpertinger coefficient. Our simulation results also reveal that, when the time-frequency resources are sufficient, the BS allocates more subchannels to the UEs with heavier traffic demands to improve the local data rate and adopts similar subframe configurations across the cells to mitigate inter-cell interference. In addition, in the resource-limited regime, the BS gives more priority to meeting local QoS constraints than to avoiding inter-cell interference.

References

[1] M. Setayesh, S. Bahrami, and V. W. Wong, "Resource slicing for eMBB and URLLC services in radio access network using hierarchical deep learning," *IEEE Trans. Wireless Commun.*, vol. 21, no. 11, pp. 8950–8966, Nov. 2022.

[2] S. Niu, H. Shao, Y. Hu, S. Zhou, and C. Wang, "Privacy-preserving mutual heterogeneous signcryption schemes based on 5G network slicing," *IEEE Internet Things J.*, vol. 9, no. 19, pp. 19 086–19 100, Oct. 2022.

[3] Y. Xu, G. Gui, H. Gacanin, and F. Adachi, "A survey on resource allocation for 5G heterogeneous networks: Current research, future trends, and challenges," *IEEE Commun. Surv. Tut.*, vol. 23, no. 2, pp. 668–695, Apr–Jun. 2021.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.

[4] Y. Li, A. Huang, Y. Xiao, X. Ge, S. Sun, and H.-C. Chao, "Federated orchestration for network slicing of bandwidth and computational resource," 2020. [Online]. Available: http://arxiv.org/abs/2002.02451

[5] C.-Y. Hsieh, T. Phung-Duc, Y. Ren, and J.-C. Chen, "Design and analysis of dynamic block-setup reservation algorithm for 5G network slicing," *IEEE Trans. Mobile Comput.*, early access, 2022, doi: 10.1109/TMC.2022.3169034.

[6] S. Messaoud, A. Bradai, O. B. Ahmed, P. T. A. Quang, M. Atri, and M. S. Hossain, "Deep federated q-learning-based network slicing for industrial IoT," *IEEE Trans. Ind. Inform.*, vol. 17, no. 8, pp. 5572–5582, Aug. 2021.

[7] Y. Xu, H. Xie, Q. Wu, C. Huang, and C. Yuen, "Robust max-min energy efficiency for RIS-aided HetNets with distortion noises," *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 1457–1471, Feb. 2022.

[8] C. Qi, Y. Hua, R. Li, Z. Zhao, and H. Zhang, "Deep reinforcement learning with discrete normalized advantage functions for resource management in network slicing," *IEEE Commun. Lett.*, vol. 23, no. 8, pp. 1337–1341, Aug. 2019.

[9] R. Shrivastava, K. Samdanis, and A. Bakry, "On policy based RAN slicing for emerging 5G TDD networks," in *Proc. IEEE Glob. Commun. Conf.*, 2018, pp. 1–6.

[10] 3GPP, "Requirements for further advancements for E-UTRA (LTE-advanced)," TR 36.913, 2017.

[11] Z. Shen, A. Khoryaev, E. Eriksson, and X. Pan, "Dynamic uplink-downlink configuration and interference management in TD-LTE," *IEEE Commun. Mag.*, vol. 50, no. 11, pp. 51–59, Nov. 2012.

[12] H. Kim, J. Kim, and D. Hong, "Dynamic TDD systems for 5G and beyond: A survey of cross-link interference mitigation," *IEEE Commun. Surv. Tut.*, vol. 22, no. 4, pp. 2315–2348, Oct.–Dec. 2020.

[13] M. Song, H. Shan, H. H. Yang, and T. Q. S. Quek, "Joint optimization of fractional frequency reuse and cell clustering for dynamic TDD small cell networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 1, pp. 398–412, Jan. 2022.

[14] M. Ghermezcheshmeh, M. M. Razlighi, V. Shah-Mansouri, and N. Zlatanov, "Centralized dynamic-time division duplex utilizing interference alignment," *IEEE Trans. Wireless Commun.*, vol. 20, no. 10, pp. 6852–6866, Oct. 2021.

[15] J. Li, A. Huang, H. Shan, H. H. Yang, and T. Q. S. Quek, "Analysis of packet throughput in small cell networks under clustered dynamic TDD," *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 5729–5742, Sep. 2018.

[16] M. Ding, D. López-Pérez, R. Xue, A. V. Vasilakos, and W. Chen, "On dynamic time-division-Duplex transmissions for small-cell networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 11, pp. 8933–8951, Nov. 2016.

[17] C.-H. Lee, R. Y. Chang, S.-M. Cheng, C.-H. Lin, and C.-H. Hsiao, "Joint beamforming and power allocation for M2M/H2H co-existence in green dynamic TDD networks: Low-complexity optimal designs," *IEEE Internet Things J.*, vol. 9, no. 6, pp. 4799–4815, Mar. 2022.

[18] C. K. Sheemar, L. Badia, and S. Tomasin, "Game-theoretic mode scheduling for dynamic TDD in 5G systems," *IEEE Commun. Lett.*, vol. 25, no. 7, pp. 2425–2429, Jul. 2021.

[19] P. Jayasinghe, A. Tölli, J. Kaleva, and M. Latva-aho, "Bi-directional beamformer training for dynamic TDD networks," *IEEE Trans. Signal Process.*, vol. 66, no. 23, pp. 6252–6267, Dec. 2018.

[20] J. Rachad, R. Nasri, and L. Decreusefond, "3D beamforming based dynamic TDD interference mitigation scheme," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, Antwerp, Belgium, 2020, pp. 1–7.

[21] 3GPP, "Further enhancements to LTE Time Division Duplex (TDD) for Downlink Uplink (DL-UL) interference management and traffic adaptation(Release 11)," TR 36.828, Jun. 2012.

[22] 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. 10 433–10 445, Nov. 2017.

[23] Y. He, F. R. Yu, N. Zhao, V. C. M. Leung, and H. Yin, "Software-defined networks with mobile edge computing and caching for smart cities: A Big Data deep reinforcement learning approach," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 31–37, Dec. 2017.

[24] F. Tang, Y. Zhou, and N. Kato, "Deep reinforcement learning for dynamic uplink/downlink resource allocation in high mobility 5G HetNet," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 12, pp. 2773–2782, Dec. 2020.

[25] C. Tsai, K. Lin, H. Wei, and F. Yeh, "QoE-aware q-learning based approach to dynamic TDD uplink-downlink reconfiguration in indoor small cell networks," *Wireless Netw.*, vol. 25, no. 6, pp. 3467–3479, Jan. 2019.

[26] Y. Wang and M. Tao, "Dynamic uplink/downlink configuration using q-learning in femtocell networks," in *Proc. IEEE/CIC Int. Conf. Commun. China*, Shanghai, China, 2014, pp. 53–58.

[27] J. Li et al., "Blockchain assisted decentralized federated learning (BLADE-FL): Performance analysis and resource allocation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 10, pp. 2401–2415, Oct. 2022.

[28] K. Wei et al., "User-level privacy-preserving federated learning: Analysis and performance optimization," *IEEE Trans. Mob. Comput.*, vol. 21, no. 9, pp. 3388–3401, Sep. 2022.

[29] D. Lee and J. So, "Adaptive feedback bits and power allocation for dynamic TDD systems," *J. Commun. Netw.*, vol. 21, no. 2, pp. 113–124, Apr. 2019.

[30] J. Cui, Y. Liu, and A. Nallanathan, "Multi-agent reinforcement learning-based resource allocation for UAV networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 729–743, Feb. 2020.

[31] Z. Xiong et al., "UAV-assisted wireless energy and data transfer with deep reinforcement learning," *IEEE Trans. Cog. Commun. Netw.*, vol. 7, no. 1, pp. 85–99, Mar. 2021.

[32] M. M. Azari, G. Geraci, A. Garcia-Rodriguez, and S. Pollin, "UAV-to-UAV communications in cellular networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 9, pp. 6130–6144, Sep. 2020.

[33] ITU-R, "Propagation data and prediction methods required for the design of terrestrial broadband radio access systems operating in a frequency range from 3 to 60 GHz," Tech. Rep. P. 1410-5, Feb. 2012.

[34] C. Huang et al., "Multi-hop RIS-empowered terahertz communications: A DRL-based hybrid beamforming design," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 6, pp. 1663–1677, Jun. 2021.

[35] C. Huang, R. Mo, and C. Yuen, "Reconfigurable intelligent surface assisted multiuser MISO systems exploiting deep reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1839–1850, Aug. 2020.

[36] G. Dulac-Arnold, R. Evans, P. Sunehag, and B. Coppin, "Deep reinforcement learning in large discrete action spaces," *ArXiv:1512.07679v2*, 2016. [Online]. Available: http://arxiv.org/abs/1512.07679v2

[37] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," 2017. [Online]. Available: http://arxiv.org/abs/1706.02275

[38] Z. Xue et al., "A resource-constrained and privacy-preserving edge-computing-enabled clinical decision system: A federated reinforcement learning approach," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 9122–9138, Jun. 2021.

[39] J. Qi, Q. Zhou, L. Lei, and K. Zheng, "Federated reinforcement learning: Techniques, applications, and open challenges," 2021. [Online]. Available: https://arxiv.org/abs/2108.11887

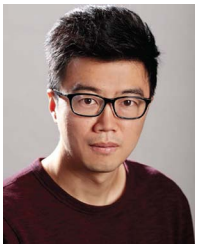
[40] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, Jul. 2008.

[41] 3GPP, "Technical specification group radio access network: Study on enhanced LTE support for aerial vehicles (Release 15)," Tech. Rep. TR 36.777, Dec. 2017.

Ziyan Yin (Student Member, IEEE) received the B.S. degree from the School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou, China, in 2017, and the M.Sc. degree from the School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing, China, in 2018, where she is currently working toward the Ph.D. degree. Her research interests include reinforcement learning, game theory, UAV communication, and anti-jamming.

Zhe Wang (Member, IEEE) received the Ph.D. degree in electrical engineering from The University of New South Wales, Sydney, NSW, Australia, in 2014. From 2014 to 2020, she was a Research Fellow with The University of Melbourne, Parkville, VIC, Australia, and Singapore University of Technology and Design, Singapore, respectively. She is currently a Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. Her research interests include applications of optimization, game theory, and machine learning to resource allocation in communications and networking. She is also an Editor of IEEE Open Journal of the Communications Society.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.



Jun Li (Senior Member, IEEE) received Ph.D. degree in electronic engineering from Shanghai Jiao Tong University, Shanghai, P. R. China, in 2009. From January 2009 to June 2009, he was with the Department of Research and Innovation, Alcatel Lucent Shanghai Bell, Shanghai, China, as a Research Scientist. From June 2009 to April 2012, he was a Postdoctoral Fellow with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, NSW, Australia. From April 2012 to June 2015, he was a Research Fellow with the School of Electrical Engineering, University of Sydney, Camperdown, NSW. Since June 2015, he has been a Professor with the School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing, China. From 2018 to 2019, he was also a Visiting Professor with Princeton University, Princeton, NJ, USA. He has coauthored more than 200 papers in IEEE journals and conferences, and holds one U.S. patents and more than ten Chinese patents in these areas. His research interests include network information theory, game theory, distributed intelligence, multiple agent reinforcement learning, and their applications in ultra-dense wireless networks, mobile edge computing, network privacy and security, and industrial Internet of things. He is an Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATION and TPC member for several flagship IEEE conferences.

Ming Ding (Senior Member, IEEE) received the B.S. and M.S. degrees (with first-class Hons.) in electronics engineering, and the Ph.D. degree in signal and information processing from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2004, 2007, and 2011, respectively. From April 2007 to September 2014, he was with the Sharp Laboratories of China, Shanghai, as a Researcher/Senior Researcher/Principal Researcher. He is currently a Senior Research Scientist with Data61, CSIRO, in Sydney, NSW, Australia. His research interests include information technology, data privacy and security, machine learning and AI, etc. He has authored more than 140 papers in IEEE journals and conferences, in recognized venues, and around 20 3GPP standardization contributions, and a Springer book *Multi-Point Cooperative Communication Systems: Theory and Applications*. Also, he holds 21 U.S. patents and co-invented more than 100 patents on 4G/5G technologies in CN, JP, KR, EU. He is an Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. Besides, he was the Guest Editor/Co-Chair/Co-Tutor/TPC Member for multiple IEEE top-tier journals/conferences. He was the recipient of several awards for his research work and professional services.

Wen Chen (Senior Member, IEEE) is currently a Tenured Professor with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China, where he is also the Director of Broadband Access Network Laboratory. He has authored or coauthored more than 120 papers in IEEE journals and more than 120 papers in IEEE Conferences, with citations more than 8000 in Google Scholar. His research interests include multiple access, wireless AI and meta-surface communications. He is the Shanghai Chapter Chair of IEEE Vehicular Technology Society, Editors of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE ACCESS, and IEEE OPEN JOURNAL OF VEHICULAR TECHNOLOGY. He is the Fellow of Chinese Institute of Electronics and the distinguished lecturers of IEEE Communications Society and IEEE Vehicular Technology Society.

Shi Jin (Senior Member, IEEE) received the B.S. degree in communications engineering from the Guilin University of Electronic Technology, Guilin, China, in 1996, the M.S. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2003, and the Ph.D. degree in information and communications engineering from Southeast University, Nanjing, in 2007. From June 2007 to October 2009, he was a Research Fellow with the Adastral Park Research Campus, University College London, London, U.K. He is currently with the Faculty of the National Mobile Communications Research Laboratory, Southeast University. His research interests include space time wireless communications, random matrix theory, and information theory. He and his co-authors were the recipient of the 2011 IEEE Communications Society Stephen O. Rice Prize Paper Award in the field of communication theory and the 2010 Young Author Best Paper Award by the IEEE Signal Processing Society. He is also an Associate Editor for IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE COMMUNICATIONS LETTERS, and *IET Communications*.

Authorized licensed use limited to: Shanghai Jiaotong University. Downloaded on March 27,2023 at 09:27:08 UTC from IEEE Xplore. Restrictions apply.