SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2024, Vol. 67, Iss. 1, 112302:1–112302:18
https://doi.org/10.1007/s11432-022-3697-x

Slicing capacity-centered mode selection and resource optimization for network-assisted full-duplex cell-free distributed massive MIMO systems

Jie WANG[1], Jiamin LI[1,2*], Pengcheng ZHU[1],
Dongming WANG[1,2], Hongbiao ZHANG[3], Yue HAO[3] & Bin SHENG[1,2]

[1]*National Mobile Communications Research Laboratory, Southeast University, Nanjing* 210096*, China;*
[2]*Purple Mountain Laboratories, Nanjing* 211111*, China;*
[3]*China Mobile Research Institute, Beijing* 100032*, China*

Received 8 September 2022/Revised 11 December 2022/Accepted 3 February 2023/Published online 27 December 2023

Abstract Network-assisted full-duplex (NAFD) cell-free distributed massive multiple-input multiple-output (MIMO) systems enable uplink (UL) and downlink (DL) communications within the same time-frequency resources, which potentially reduce latency by avoiding the overhead of switching UL/DL modes. However, how to choose UL/DL modes remains an important factor affecting system performance. With the dramatic increase in the number of users and access points (APs), massive access brings significant overhead in the mode selection. Additionally, the different quality of service (QoS) among users also makes the effective utilization of resources difficult. As one of the most promising technologies in sixth-generation (6G), network slicing enables the adaptive configuration of limited UL/DL resources through the resource isolation assisted NAFD technique. Therefore, we propose a slicing capacity-centered scheme. Under this scheme, APs are motivated by slicing requirements and associated slices to form different subsystems. Collaborative mode selection and resource allocation are performed within each subsystem to reduce overhead and improve resource utilization. To implement this scheme efficiently, a double-layer deep reinforcement learning (DRL) mechanism is used to realize the joint optimization of mode selection and resource allocation. Simulation results show that the slicing capacity-centered scheme can effectively improve resource utilization and reduce overhead.

Keywords network-assisted full-duplex, network slicing, mode selection, resource optimization, deep reinforcement learning

1 Introduction

With the continuous expansion of the scale of communication services, the diversity demand of users has further increased, imposing new requirements for the relevant theories and technologies of ultra-reliable and low-delay communication (URLLC) in sixth-generation (6G). To flexibly apply limited radio resources to improve throughput, flexible duplex technology has attracted great attention in 6G communication by adjusting the allocation of radio resources. Recently, co-frequency co-time full-duplex technology has enabled theoretically to improve spectral efficiency, but cross-link interference will limit system performance when applied to large-scale networks [1]. Thus, the cell-free with network-assisted full-duplex (NAFD) approach was proposed to reduce the cross-link interference in the spatial domain, in which the uplink (UL) and downlink (DL) wireless links are jointly served in the same time-frequency resource. Furthermore, NAFD technology under cell-free distributed massive multiple-input multiple-output (MIMO) systems was investigated in [2–4], and the delay of interference cancellation was reduced by jointly processing to achieve URLLC. Most of the previous studies analyzed system performance based on fixed working modes [5–7]. For example, Ref. [5] proposed an efficient power allocation scheme that only dependent on slowly varying large-scale fading from the perspective of multi-objective optimization. In [6], joint beamforming and power control were studied by maximizing aggregated spectral efficiency.

* Corresponding author (email: jiaminli@seu.edu.cn)

Ref. [7] focused on user scheduling and transceiver design. Ref. [8, 9] realized flexible mode selection to improve system performance, and flexible duplex transmission was assisted by selecting the UL/DL working mode of access points (APs) within the same time-frequency resource block. However, mode selection and resource optimization in the above research are aimed at global capacity. All APs cooperate dynamically to select the working mode and allocate radio resources. On the one hand, all APs cooperate with each other to exchange the information of users in the entire network, resulting in mass overhead and a lack of scalability. Although fairness between users is guaranteed to a certain extent, the customized services of users are neglected. Therefore, the mode selection and resource utilization mechanism for NAFD systems need to be further studied to develop a flexible way to balance the capacity requirement and signaling overhead.

As a key technology in 6G, network slicing abstracts a physical network into a virtual logical network suitable for different scenarios, offering a strong guarantee for differentiated quality of service (QoS) to users. Ref. [10] combined the self-management, self-optimization, and self-learning of multi-granularity network resources, realizing a slicing adaptive control strategy under unforeseen network conditions. The resource allocation solution of network slicing has been used in different systems such as multi-cell networks [11,12], cloud-radio access networks (C-RANs) [13–15], hybrid networks composed of traditional distributed small cell and C-RAN [16]. In addition, Ref. [17,18] focused on a layered slicing framework, which can allocate radio resources in different time scales according to the long-term trend and short-term trend of network conditions, so as to use radio resources more effectively. However, these studies only aimed at UL/DL scenarios and slicing is highly compatible with NAFD systems. To be specific, the interference between users is further reduced through resource isolation firstly. Secondly, users are clustered according to different service types, which provides a new idea for mode selection and capacity expansion in NAFD systems.

Notably, the cell-free massive MIMO is an innovative architecture to improve spectral efficiency [19,20]. Since the feature that all APs serve all users increases signaling overhead and computational complexity, which raises the issue of scalability for massive access [21]. To achieve practical deployment, cell-free massive MIMO systems need to develop efficient and flexible clustering methods to achieve wide coverage. Ref. [22] obtained user location information based on fingerprints, and designed a pilot assignment scheme with the help of user clustering to further suppress the interference caused by pilot reuse. Location-based user clustering was further extended in [23] into a user-centric framework where a selected cooperative AP service cluster is formed for each user to make the selection of AP reach the global optimum. However, the user-centric selected cooperative AP service clusters usually overlap, which complicates the transmissions as the clusters are coupled with each other by the per-AP power constraint, and a strict interference management mechanism is required between clusters. In addition, the AP-centric clustering method first divides APs into multiple clusters, and then each user joins the cluster to which its associated AP belongs. Various dynamic AP clustering methods were proposed by adopting heuristic methods [24–26]. However, with AP-centric clustering algorithms, since the clusters are formed from the AP side only, the edge problem could emerge. In contrast to the above studies, we propose a slicing capacity-centered clustering method to achieve massive access. This method divides the entire system into several nonoverlapping subsystems by using the information of APs and users. It not only avoids the interference problem in the user-centric scheme but also avoids the disadvantage of focusing only on the AP-side information in the AP-centric scheme.

Therefore, a slicing capacity-centered mode selection and resource allocation scheme is proposed for NAFD systems. This scheme is divided into two modules: the formation of AP-slice subsystems, mode selection and resource optimization inside each subsystem. In the former, APs are driven by slicing demand for clustering, and the association scheme between APs and slices is determined to form different AP-slice subsystems. The latter is mainly realized by isolating communication resources among subsystems, and different subsystems are allocated to different resource blocks. Then, specific resources are used to implement cooperative working between APs in each subsystem so that the signaling costs are further saved while the slicing capacity is guaranteed. In addition, considering the high reliability requirement of 6G system for mass services and the characteristics of cell-free distributed MIMO systems, multi-link selection and nonorthogonal multiple access (NOMA) can further support mass users access with limited spectrum resources, reducing retransmission delay and providing optimal throughput. Meanwhile, the complexity caused by network slicing prompts us to use a deep reinforcement learning (DRL) strategy [27–30]. The deep Q-network (DQN) algorithm can adjust the slicing configuration according to the satisfaction feedback. The multi-agent deep deterministic policy gradient (MADDPG) has

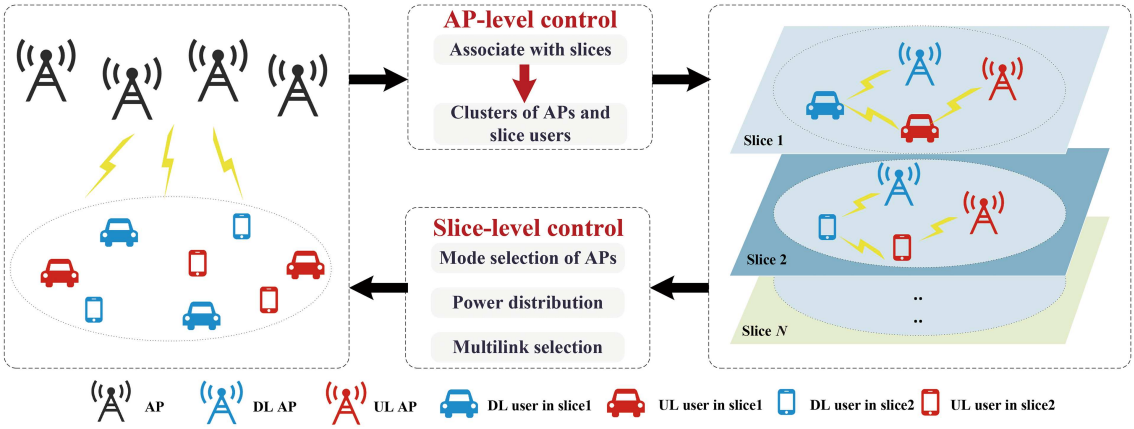

Figure 1 (Color online) Slicing capacity-centered NAFD cell-free distributed massive MIMO system.

better performance of multi-cooperation and competition, effectively realizing AP collaboration within slices and among slices.

Motivated by the above research, this paper proposes a slicing capacity-centered model selection and resource optimization scheme. This scheme uses an effective DRL algorithm to achieve resource isolation and clustering cooperation to achieve adaptive configuration. The main contributions of this paper are highlighted as follows:

• First, network slicing is introduced to assist the NAFD technique in reducing the pressure of contention for resources. Different slices allocate different time-frequency resources to isolate interference. Moreover, multi-connection technology allows users to share resources to improve communication reliability and access efficiency.

• Second, APs are motivated by the slicing requirement to associate different slices to form different subsystems. Given the AP-slice association scheme, each AP performs working mode selection and resource optimization with users in each subsystems, which further reduces the signaling cost and access latency.

• Third, the double DRL mechanism is proposed to jointly optimize mode selection and resource allocation. The upper-layer control policy uses DQN to form flexible duplex AP-slice subsystems. The lower-layer control policy uses MADDPG to select working modes cooperatively and perform radio resource allocation in each subsystem.

The remainder of this article is organized as follows. In Section 2, the channel model, signal transmission model, and slicing delay model are presented. Section 3 proposes the optimization problem and double-layer optimization mechanism for the proposed scheme. The DRL algorithm as a solution is introduced in detail in Section 4. The simulation results are presented in Section 5, and the entire paper is summarized in Section 6.

Notation. In this paper, scalars are represented by lowercase letters (e.g., j), matrixes are represented by uppercase bold letters (e.g., $\boldsymbol{G}$) and vectors are represented by lowercase bold letters (e.g., $\boldsymbol{g}$). $|\cdot|$ represents the absolute value of a complex scalar. $\boldsymbol{g}^{\mathrm{H}}$ and $\boldsymbol{g}^{\mathrm{T}}$ represent Hermitian transpose and transpose of $\boldsymbol{g}$, respectively. $\boldsymbol{I}_{m\times m}$ denotes an $m \times m$ identity matrix. $\mathcal{CN}(0, \Delta)$ represents the complex circularly symmetric Gaussian distribution with a 0-mean vector and covariance matrix Δ.

2 System model

This paper considers a slicing capacity-centered NAFD cell-free distributed massive MIMO system. As illustrated in Figure 1, the system is configured with J APs and K users, and each AP is equipped with N antennas. Within the service scope of APs, K single-antenna users are associated with different slices according to their service types, including L UL users and M DL users. I slices support I different services. The user set of the i-th slice includes L_i UL users and M_i DL users.

The proposed slicing capacity-centered scheme provides a new idea for mode selection and resource allocation in NAFD scenarios. In terms of mode selection, APs associate with slices to form different subsystems firstly, and the number of subsystems is equal to the number of slices. A binary variable

Table 1 Symbols setting

Parameter	Symbol	Parameter	Symbol
AP	j	UL/DL users in each subsystem i	l_i,m_i
The group of APs	$\mathcal{J}=\{1,2,\ldots,J\}$	The group of UL users in subsystem i	$\mathcal{L}_i=\{1,2,\ldots,L_i\}$
Slice (subsystem)	i	The group of DL users in subsystem i	$\mathcal{M}_i=\{1,2,\ldots,M_i\}$
The group of slices (subsystems)	$\mathcal{I}=\{1,2,\ldots,I\}$	UL/DL AP in subsystem i	u_i, d_i
User	k	The group of UL APs in subsystem i	$\mathcal{U}_i=\{1,2,\ldots,U_i\}$
The group of users	$\mathcal{K}=\{1,2,\ldots,K\}$	The group of DL APs in subsystem i	$\mathcal{D}_i=\{1,2,\ldots,D_i\}$
UL/DL user	l,m	PRB	f
The group of UL users	$\mathcal{L}=\{1,2,\ldots,L\}$	The group of PRBs	$\mathcal{F}=\{1,2,\ldots,F\}$
The group of DL users	$\mathcal{M}=\{1,2,\ldots,M\}$	The association factor of AP j and slice i	$V_j=i$
The DL(UL) mode factor of AP j	$Z_j=1(0)$	The user association factor	e_{u_i,f,l_i}, e_{d_i,f,m_i}

$V^{1\times J}$ represents an association between APs and slices. Only the AP subset C_i serving slice i collaborate to select the UL/DL working mode inside the subsystem i to ensure the capacity demand of slice i. A binary variable $Z^{1\times J}$ is used to model the mode selection of APs. In a coherent time, U_i APs perform UL reception and D_i APs perform DL transmission inside the subsystem i, where $U_i+D_i=C_i$. In terms of resource allocation, users in subsystems use F different physical resource blocks (PRBs), which are shared by all APs in different subsystems. The bandwidth allocated by each PRB is $B=W/F$, where W is the bandwidth of the system. NOMA can be used to allocate different power to each user so that each PRB can be reused by multiple users. The detailed parameters are shown in Table 1.

2.1 Channel model

Considering a block fading channel, the DL channel gain between the m_i-th DL user and the d_i-th DL AP is

$$\boldsymbol{g}_{d_i,m_i}=\sqrt{\lambda_{d_i,m_i}}\boldsymbol{h}_{d_i,m_i}, \tag{1}$$

where $\lambda_{d_i,m_i}\triangleq b_{d_i,m_i}^{-\varsigma}a_{d_i,m_i}$ represents the large-scale fading, b_{d_i,m_i} is the distance, ς is the path loss exponent, a_{d_i,m_i} is a log-normal shadow fading variable. In addition, $\boldsymbol{h}_{d_i,m_i}$ models small-scale fast fading, and each element follows standard Rayleigh distribution $\mathcal{CN}(0,1)$.

Similarly, the UL channel gain from the l_i-th UL user to the u_i-th UL AP, the interference channel from the l_i-th UL user to the m_i-th DL user and the interference channel from the d_i-th DL AP to the u_i-th UL AP can be modeled as

$$\boldsymbol{g}_{u_i,l_i}=\sqrt{\lambda_{u_i,l_i}}\boldsymbol{h}_{u_i,l_i}, \tag{2}$$

$$\boldsymbol{g}_{m_i,l_i}=\sqrt{\lambda_{m_i,l_i}}\boldsymbol{h}_{m_i,l_i}, \tag{3}$$

$$\boldsymbol{G}_{I,d_i,u_i}=\sqrt{\lambda_{d_i,u_i}}\boldsymbol{H}_{d_i,u_i}, \tag{4}$$

since each AP is equipped with N antennas, $\boldsymbol{H}_{d_i,u_i}$ is an $N\times N$ small-scale fast fading matrix.

2.2 Signal transmission model

For DL transmission in the slicing capacity-centered scheme, the DL APs inside subsystem i decode DL signals, and then jointly send signals to the DL users of subsystem i. Specifically, the signal received by the m_i-th DL user can be expressed as

$$\Upsilon_{m_i}=\sum_{m_i'\in\mathcal{M}_i}\boldsymbol{g}_{m_i'}^{\mathrm{H}}\boldsymbol{w}_{m_i'}\rho_{m_i'}+\sum_{l_i\in\mathcal{L}_i}\boldsymbol{g}_{m_i,l_i}\sqrt{p_{l_i}}\chi_{l_i}+\Lambda_{m_i}, \tag{5}$$

where $\boldsymbol{g}_{m_i'}=[\boldsymbol{g}_{1,m_i'}^{\mathrm{T}},\ldots,\boldsymbol{g}_{D_i,m_i'}^{\mathrm{T}}]^{\mathrm{T}}$ is channel vector between all DL APs inside subsystem i and m_i'-th DL user, $\boldsymbol{w}_{m_i'}=[\boldsymbol{w}_{1,m_i'}^{\mathrm{T}},\ldots,\boldsymbol{w}_{D_i,m_i'}^{\mathrm{T}}]^{\mathrm{T}}$ is the DL precoding vector, maximum ratio transmission precoding is considered in this paper, $\rho_{m_i'}$ represents the transmitted data signal with $\mathbb{E}[\rho_{m_i'}\rho_{m_i'}^{\mathrm{H}}]=1$, p_{l_i} indicates the transmission power of UL user l_i, χ_{l_i} denotes the transmitted data signal with $\mathbb{E}[\chi_{l_i}\chi_{l_i}^{\mathrm{H}}]=1$, $\boldsymbol{\Lambda}_{m_i}\sim\mathcal{CN}(0,\sigma_{m_i}^2)$ is the additive white Gaussian noise power.

In the slicing capacity-centered scheme, all users are clustered based on QoS requirements (such as transmission rate and delay) and associated with the same slice. PRBs as communication resources are

isolated among slices to guarantee capacity, and users share resources within the slices. The link selected by a DL(UL) AP and a DL(UL) user is multiplexed to a PRB. Thus interference between peer users is limited to the PRB level. Taking into account the cooperation of APs and interference caused by peer users sharing the same PRB, the transmission rate of the m_i-th DL user by DL AP d_i on PRB f can be modeled as

$$r_{d_i,f,m_i} = B\log_2\left(1+\beta_{d_i,f,m_i}\right), \tag{6}$$

where the signal to interference noise ratio (SINR) β_{d_i,f,m_i} is modeled as

$$\beta_{d_i,f,m_i} = \frac{p_{d_i,f,m_i}\left|\boldsymbol{g}_{d_i,m_i}^{\mathrm{H}}\boldsymbol{w}_{d_i,m_i}\right|^2}{\sum\limits_{d_i'\in\mathcal{D}_i,d_i'\neq d_i}\sum\limits_{i\in\mathcal{I}}\sum\limits_{m_i'\in\mathcal{M}_i,m_i'\neq m_i} p_{d_i',f,m_i'}\left|\boldsymbol{g}_{d_i,m_i}^{\mathrm{H}}\boldsymbol{w}_{d_i',m_i'}\right|^2+\sum\limits_{l_i\in\mathcal{L}_i}p_{l_i}|\boldsymbol{g}_{m_i,l_i}|^2+\sigma_{m_i}^2}, \tag{7}$$

where p_{d_i,f,m_i} represents the power allocated from DL AP m_i to the DL user m_i on PRB f.

For UL transmission in the slicing capacity-centered scheme, all UL APs inside subsystem i jointly receive signals from UL users of subsystem i. The received signal can be expressed as

$$\Upsilon^{u,i} = \sum_{l_i\in\mathcal{L}_i}\boldsymbol{g}_{l_i}^{\mathrm{H}}\sqrt{p_{l_i}}\chi_{l_i} + \sum_{m_i\in\mathcal{M}_i}\boldsymbol{G}_I\boldsymbol{w}_{m_i}\rho_{m_i}+\Lambda_{u,i}, \tag{8}$$

where $\boldsymbol{g}_{l_i} = [\boldsymbol{g}_{1,l_i}^{\mathrm{T}},\ldots,\boldsymbol{g}_{U_i,l_i}^{\mathrm{T}}]^{\mathrm{T}}$ is UL channel vector. $\boldsymbol{G}_I$ is the interference channel matrix between DL APs and UL APs inside subsystem i. $\Lambda_{u,i}\sim\mathcal{CN}\left(0,\sigma_{u,i}\boldsymbol{I}_{U\times N}\right)$ is the complex additive white Gaussian noise power.

Similar to the DL users, the isolation of resources by slicing limits the interference. The instantaneous rate available to the l_i-th UL user, associated with the UL AP u_i and multiplexed on the PRB f is modeled as

$$r_{u_i,f,l_i} = B\log_2\left(1+\beta_{u_i,f,l_i}\right), \tag{9}$$

where the SINR β_{u_i,f,l_i} is modeled as

$$\beta_{u_i,f,l_i} = \frac{p_{u_i,f,l_i}|\boldsymbol{v}_{u_i,l_i}^{\mathrm{H}}\boldsymbol{g}_{u_i,l_i}|^2}{\sum\limits_{u_i'\in\mathcal{U}_i,u_i'\neq u_i}\sum\limits_{i\in\mathcal{I}}\sum\limits_{l_i'\in\mathcal{L}_i,l_i'\neq l_i} p_{u_i',f,l_i'}|\boldsymbol{v}_{u_i,l_i}^{\mathrm{H}}\boldsymbol{g}_{u_i',l_i'}|^2+\sum\limits_{d_i\in\mathcal{D}_i}\sum\limits_{m_i\in\mathcal{M}_i}\delta_{d_i,u_i}|\boldsymbol{w}_{d_i,m_i}|^2|\boldsymbol{v}_{u_i,l_i}|^2+\mu_{l_i}}, \tag{10}$$

where p_{u_i,f,l_i} represents the power allocated from UL AP u_i to the UL user l_i in subsystem i on PRB f, and $\boldsymbol{v}_{u_i,l_i}$ is the receiver vector, δ_{d_i,u_i} indicates the interference power between UL and DL users in subsystem i, $\mu_{l_i}=\sigma_{u_i,l_i}^2|\boldsymbol{v}_{u_i,l_i}|^2$ is the noise of the corresponding receiver vector.

2.3 Slicing delay model

In this paper, both mode selection and resource allocation aim at the customization requirements of slices. Therefore, how to characterize different QoS of slices is the focus of this section. Generally, QoS performance mainly depends on the delay and reliability of communication. We propose the slice type with delay awareness as an index to distinguish the QoS requirements of different slices. Specifically, the delay-aware is to judge whether the service requirement is met by perceiving the total delay of the user from sending the request to receiving the service. The packet delay in the slice includes transmission delay, processing delay, and queuing delay [31, 32], it can be defined as

$$D_{i,\text{delay-aware}} = \vartheta_1 T_i + \vartheta_2 P_i + \vartheta_3 Q_i, \tag{11}$$

where ϑ_1,ϑ_2,ϑ_3 are weight factors of different delays.

Assuming that packet arrival and service process are independent within each slice, which follows the service time characteristics of Poisson arrival and exponential distribution. When the packet queue of a user is nonempty, it is considered to be an active user, so users can be considered active in the subsequent work.

2.3.1 *Transmission delay*

The transmission delay is the time required to transmit data packets on the links between APs and slices. Therefore, the transmission delay of slice i can be expressed as

$$T_i = \frac{\Omega_i}{r_i}, \tag{12}$$

where Ω_i represents the total packet length of slice i.

2.3.2 *Processing delay*

The processing delay refers to the time required by APs to process the data packets after receiving the data request from the corresponding users. Therefore, only the UL receiving process needs to be considered in the processing delay, and the processing delay of slice i can be expressed as

$$P_i = \frac{\Omega_{i,u}}{U_i R}, \tag{13}$$

where $\Omega_{i,u}$ represents the total packet length for UL reception of slice i, R represents the data processing rate of each AP.

2.3.3 *Queuing delay*

The queuing delay is decided by the scheduling policy [33]. Considering that UL reception and DL transmission are carried out simultaneously in NAFD scenarios, the queuing delay will be discussed from the following two aspects.

• In UL reception, NOMA enables multiple UL users to reuse the same PRB, and the arrival and service processes on each PRB can be expressed as an $M/M/1$ queuing system. The average number of UL users on each PRB in slice i is $n_i = L_i/F_i$, where F_i represents the number of PRBs allocated to slice i. Let τ_i represents the user service rate in slice i, which can be modeled as

$$\tau_i = \alpha_i \log_2 \left(1 + \gamma_i F_i\right), \tag{14}$$

where α_i and γ_i represent the scaling factors of service rate nonlinear increase with the number of PRBs. According to the queuing theory, the queuing delay (including waiting time and service time) of the arrival of packets in slice i is

$$Q_i = \frac{1}{\tau_i - n_i A_i}, \tag{15}$$

where A_i represents the packet arrival rate of slice i.

• In DL transmission, the arrival and service process of packets in slice i can be expressed as D_i independent $M/M/1$ queuing system. We express the service rate θ_i of each AP in slice i as

$$\theta_i = F_i \beta_i, \tag{16}$$

where β_i represents the service rate of each PRB. The average waiting time of slice i can be expressed as

$$Q_i = \frac{1}{\theta_i D_i - A_i}. \tag{17}$$

3 Problem formulation and double-layer optimization mechanism for the slicing capacity-centered scheme

To balance capacity requirement and signaling overhead, the mode selection and resource allocation paradigm centered on slicing capacity is proposed. In this scheme, the addition or deletion of network nodes (users or APs) in one subsystem will not affect the signaling overhead of other subsystems, which overcomes the unscalability of the traditional scheme to a certain extent. Moreover, the association between APs and slices is driven by delay-aware of slices. Under the premise of ensuring the global capacity, the slicing capacity-centered scheme guarantees slicing requirement and pursues minimization of system delay.

3.1 Problem formulation

The utility function of the system can be modeled as

$$U = \sum_{i \in \mathcal{I}} \lambda_i D_{i,\text{delay-aware}}, \tag{18}$$

where λ_i is positive weight factor, representing the priority of slice i, $D_{i,\text{delay-aware}}$ is the delay-aware of slice i.

Actually, whether the slicing requirement is satisfied is determined by the relationship between the total delay and maximum threshold, and the transmission delay is related to the transmission rate of all users. Therefore, it is necessary to analyze the system as a whole to determine the maximization of system utility. The slicing capacity-centered resource optimization and mode selection problem can be defined as the following optimization problem (19a).

Problem 1 (Global delay minimization)**.**

$$\min_{\pi=(\pi_U,\pi_L)} \quad U \tag{19a}$$

$$\text{s.t.} \quad \sum_{f \in \mathcal{F}} \sum_{m_i \in \mathcal{M}_i} e_{d_i,f,m_i} p_{d_i,f,m_i} \leqslant P_D^{\max}, \tag{19b}$$

$$\sum_{f \in \mathcal{F}} \sum_{l_i \in \mathcal{L}_i} e_{u_i,f,l_i} p_{u_i,f,l_i} \leqslant P_U^{\max}, \tag{19c}$$

$$\sum_{f \in \mathcal{F}} \sum_{c_i \in \mathcal{C}_i} \sum_{k_i \in \mathcal{K}_i} e_{c_i,f,k_i} r_{c_i,f,k_i} \geqslant R_i^{\min}, \tag{19d}$$

$$e_{j,f,k_i} e_{j,f',k_i} = 0, \ \forall f' \neq f \in \mathcal{F}, \tag{19e}$$

$$e_{c_i,f,k_i} e_{c_{i'},f,k_{i'}} = 0, \ \forall i \neq i' \in \mathcal{I}, \tag{19f}$$

$$\sum_{c_i \in \mathcal{C}_i} Z_{c_i} \neq 0, \ \sum_{c_i \in \mathcal{C}_i} Z_{c_i} \neq \mathcal{C}_i, \tag{19g}$$

$$\sum_{c_i \in \mathcal{C}_i} e_{c_i,k_i} \geqslant 0, \tag{19h}$$

$$D_{i,\text{delay-aware}} \leqslant D_{i,\max_0}, \tag{19i}$$

where $i \in \mathcal{I}$, $u_i \in \mathcal{U}_i$, $d_i \in \mathcal{D}_i$, $j \in \mathcal{J}$, $k_i \in \mathcal{K}_i$, $k_{i'} \in \mathcal{K}_{i'}$, $c_i \in \mathcal{C}_i$, $c_{i'} \in \mathcal{C}_{i'}$, $\mathcal{C}_i$ represents the number of AP inside subsystem i. e_{c_i,f,k_i} and e_{c_i,f,k_i} are the general forms of association factors.

Constraints (19b)–(19i) are the conditions that results in performance limitation when searching for the optimal policy. Constraints (19b) and (19c) indicate the transmission power limits for DL APs and UL users. Constraint (19d) represents the minimum data rate required to ensure each slice. Constraint (19e) indicates that an AP can allocate only one PRB to the same user. In this way, an AP can serve as many users as possible and further improve the reuse rate of a PRB. Constraint (19f) indicates the isolation of PRBs as communication resources among slices. After forming multiple subsystems, APs can only use communication resources inside subsystem to serve users. Resource isolation further reduces the interference between multiple links. Constraints (19e) and (19f) ensure the cooperation between APs and PRBs in NOMA and multi-link selection, and achieve the tradeoff between interference reduction and user reuse. Constraint (19g) is to meet the requirements of UL/DL data transmission, and the number of APs working in the two modes is limited. Constraint (19h) indicates that each active user connects to at least one AP and obtains communication resources. Constraint (19i) represents the upper-layer limit of delay that each slice can tolerate.

3.2 Double-layer optimization mechanism

To efficiently realize the mode selection and resource optimization scheme centered slicing capacity, a double-layer optimization mechanism is proposed, which can adapt to the long-term trend of slicing demand and track the dynamic changes of physical-layer state [34, 35]. The physical-layer state can be divided into local network state and traffic state, in which the traffic state includes the packet arrival rate of each slice and requirement of different slices (delay violation probability or transmission rate

threshold), while the local network state includes the channel state information (CSI) and the UL or DL working demands.

Therefore, the global policy centered on slicing capacity consists of upper-layer control policy π_U and lower-layer control policy π_L. The lower-layer control policy π_L determines the working mode of APs in the same subsystem, performs PRB multiplexing association and power allocation, and thus obtains the sum of transmission rates of all users in each subsystem based on the local network state. The upper-layer control policy π_U associates APs with slices based on the requirements of users to form different subsystems by observing the traffic state. The number of APs inside different subsystems is different and each AP can serve only one subsystem to ensure nonoverlapping between subsystems. For the upper-layer control policy, based on the sum rate of the feedback from the lower layer, the global total delay can be obtained.

The mode selection and resource allocation in the lower layer are constantly optimized under the association configuration of APs and slices in the upper layer, and the optimal performance of the lower layer is the reverse feedback of the upper layer. Therefore, when the physical-layer state of the network changes, that is, when the local network state and traffic state change, the network needs to be retrained. More specifically, when the packet arrival rate, requirement of different slices, CSI, and UL or DL working demands change, the double-layer optimization mechanism needs to be retrained to get the optimal mode selection and resource allocation scheme.

3.2.1 *Upper-layer control policy*

The upper-layer control policy is not involved in the mode selection of APs and radio resources allocation, but adjusts the association scheme between APs and slices to improve QoS. That is, subsystems of APs and users are divided globally. Based on the perception of traffic state and delay, the upper-layer control policy configures the number and location of APs inside different subsystems. Therefore, the upper-layer control policy π_U defines the dynamic change of the traffic state with the system performance feedback from the lower layer as the upper state $\boldsymbol{S}_U$ and translates it into the association scheme of APs and slices $\boldsymbol{N}_U$

$$\pi_U : \boldsymbol{S}_U \to \boldsymbol{N}_U, \tag{20}$$

$$\boldsymbol{S}_U = \{A_i, D_{i,\text{delay-aware}} | \forall i \in \mathcal{I}\}, \tag{21}$$

$$\boldsymbol{N}_U = \{C_i, O_i | \forall i \in \mathcal{I}\}, \tag{22}$$

where O_i is the location set of AP subset inside subsystem i. To ensure nonoverlapping of subsystems, different subsystems cannot share the same AP. Thus, we should satisfy

$$\sum_{i \in \mathcal{I}} C_i = J. \tag{23}$$

Compared with the traditional scheme, the proposed slicing capacity-centered scheme enables each slice to dynamically share APs according to environmental changes and service requirements, which significantly improves users satisfaction and resources utilization.

3.2.2 *Lower-layer control policy*

The lower-layer control policy is constrained by association scheme of APs and slices. The lower-layer controller selects the working modes of APs and allocates radio resources according to the dynamic of the physical layer and the association scheme of subsystems. Therefore, we define the lower-layer control policy as a mapping between the local network state $\boldsymbol{X}_L$ and the lower-layer scheme $\boldsymbol{E}_L$ under the control of the upper layer $\boldsymbol{N}_U$. In the overall wireless resource allocation scheme, link selection has been carried out along with the power allocation and mode selection.

$$\pi_L : (\boldsymbol{X}_L, \boldsymbol{N}_U) \to \boldsymbol{E}_L, \tag{24}$$

$$\boldsymbol{X}_L = \{Q_{k_i}, g_{k_i,c_i}, L_i, M_i | \forall i \in \mathcal{I}, c_i \in \mathcal{C}_i, k_i \in \mathcal{K}_i\}, \tag{25}$$

$$\boldsymbol{E}_L = \{Z_{c_i}, e_{u_i,f,l_i} p_{u_i,f,l_i}, e_{d_i,f,m_i} p_{d_i,f,m_i}\}, \tag{26}$$

where p_{u_i,f,k_i} and p_{d_i,f,k_i} indicate that the power allocated to the user k_i can be one of different power levels, and $c_i \in \mathcal{C}_i$, $f \in \mathcal{F}$, $i \in \mathcal{I}$, $u_i \in \mathcal{U}_i$, $d_i \in \mathcal{D}_i$, $l_i \in \mathcal{L}_i$, $m_i \in \mathcal{M}_i$ in (26), U_i and D_i are determined by binary selection vector $Z^{1\times J}$.

4 Solution for the slicing capacity-centered mode selection and resource optimization scheme based on double-layer mechanism

The slicing capacity-centered mode selection and resource optimization scheme uses a double-layer optimization mechanism to form subsystems including APs and users, select the working mode of APs based on service demand, and allocate radio resources. The problem is complex and highly coupled, and is susceptible to the influence of the global environment. Therefore, we adopt a scheduling scheme based on the DRL mechanism to solve the combinatorial optimization problem.

DRL algorithm usually contains five elements, including environment, agent, state, action, and reward. The agent has the ability to learn by interacting with the environment constantly and will act on the basis of the observed information combined with its own experience, which is also called policy. The state of the environment will be affected by the specific action taken by the agent. The agent will get a new state and reward from the changing environment. Therefore, the agent can perform new action according to the new observation and learn by using experience replay until the algorithm converges, that is, to find an optimal policy.

4.1 Double-layer DRL algorithm

In the proposed double-layer optimization mechanism, MADDPG is used in the lower-layer policy. APs act as agents to form a cooperative and competitive relationship when performing association of slices to form subsystems, working mode selection of APs, multi-link allocation, and power distribution. Based on the convergent lower-layer control policy, DQN is used to obtain the optimal upper-layer control policy. Thus, the communication agent has two layers, which include the DQN agent in the upper layer and MADDPG agents in the lower layer. The upper-layer action provides the premise scheme for the lower-layer action, and the lower-layer optimal reward updates the environment for the upper-layer decision. Every time the upper-layer action is executed, the lower layer algorithm needs to find the optimal policy again to feedback the upper layer to define its action value. Both of them are integrated and feed back to each other to realize the continuous optimization of system performance. The double-layer optimization algorithm is summarized as Algorithm 1.

Algorithm 1 Double-layer optimization algorithm

```
Initialize the neural networks with random parameters.
for training_episode_U = 1 to max do
    The upper-layer controller executes the DQN algorithm to obtain the AP-slice association scheme π_U and pass π_U to the lower-layer controller;
    for training_episode_L = 1 to max do
        The lower-layer controller executes the MADDPG algorithm to obtain the mode selection and resource allocation scheme π_L;
    end for
    The optimal lower-level policy π_L^* is obtained and feedback to the upper-layer controller;
end for
The optimal upper-level policy π_U^* is obtained.
```

The specific double-layer DRL mechanism is shown in Figure 2 on the next page. In the following part, we will describe the solution algorithms of the double-layer control policy, including the DQN algorithm used in the upper layer and the MADDPG algorithm used in the lower layer.

4.2 Solution for lower-layer control policy based on MADDPG

After APs are associated with slices to form multiple subsystems, the goal of the lower-layer controller is to find an optimal mode selection and resource allocation scheme, that is, the policy that obtains the maximum expected reward in all states can be represented by the following optimization problem.

Problem 2 (Lower-layer optimization)**.**

$$\begin{aligned} &\pi_L^* = \arg\max_{\pi_L} \left\{ \sum_{i\in\mathcal{I}} r_i | \pi_U \right\} \\ &\text{s.t.} \quad \text{(19b)–(19h)}. \end{aligned} \tag{27}$$

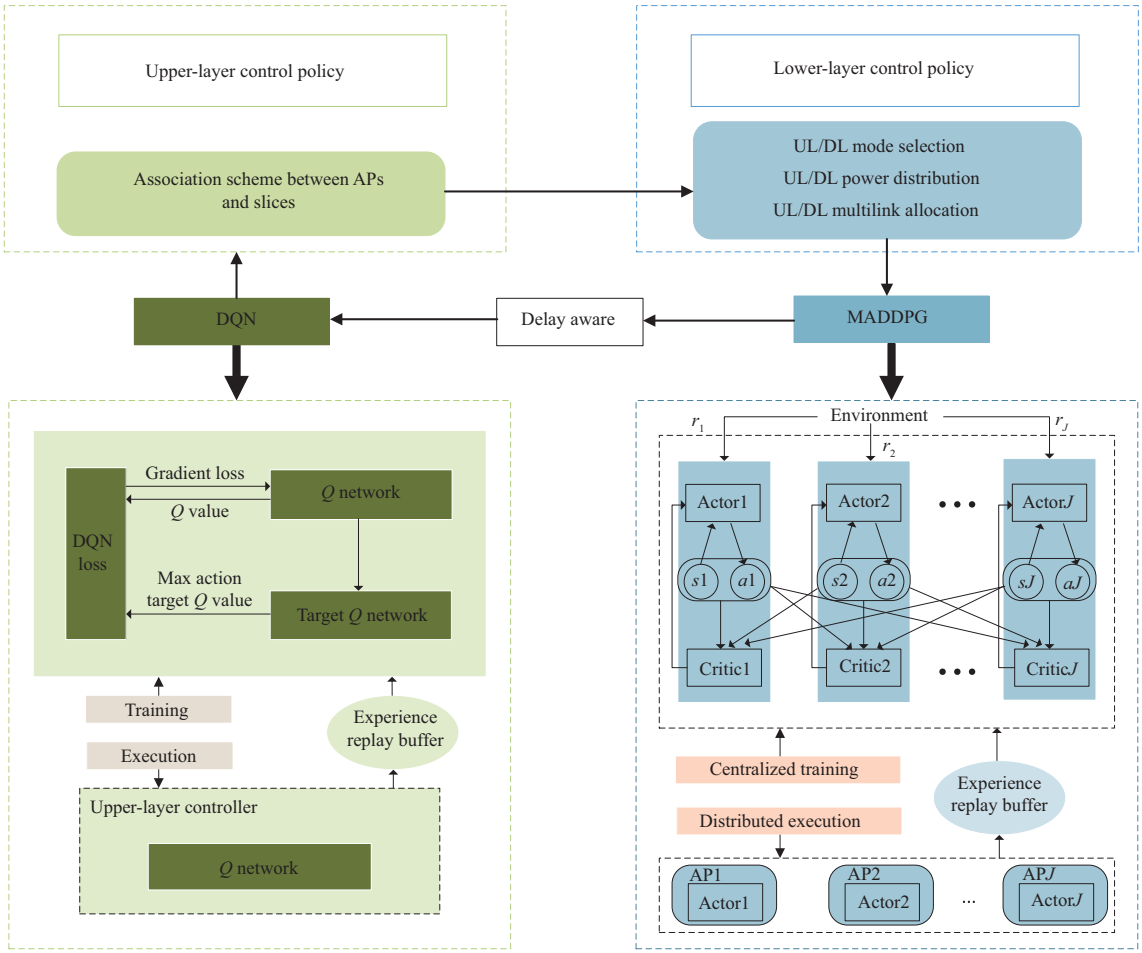

Figure 2 (Color online) Double-layer control policy for the slicing capacity-centered scheme.

MADDPG, as a typical DRL method, has the advantage of centralized training and distributed execution. Each agent trains a critic that requires global information and an actor that requires local information. Each agent is allowed to have its own reward function. Therefore, it can effectively deal with the relationship between agents. In the lower-layer MADDPG framework, APs act as agents and the communication network is the environment. To further simplify the process of solving the optimization problem, the variable N_U representing cluster formation in the upper-layer policy is separated. Only the number of APs C_i associated slices is determined in the upper layer, and the AP location information O_i of associated slices i is selected in the lower layer.

• State. The state $\boldsymbol{s}_j$ of agent j should be expressed as the number of UL/DL users in each slice, as well as CSI, which can be expressed as

$$\boldsymbol{s}_j = \{\boldsymbol{g}_{k,j}, L_i, M_i\}. \tag{28}$$

We set the APs to make random movement during each training, and the UL/DL work requirements of users remain invariant, so the CSI of the user will always change.

• Action. For agent j, the action space should include association scheme, mode selection scheme, and resource allocation scheme, specifically including the association of APs and slices, UL/DL selection, power distribution, and PRB allocation. Thus, the action space can be expressed as

$$\boldsymbol{a}_j = \{V_j, Z_{c_i}, e_{u_i,f,l_i} p_{u_i,f,l_i}, e_{d_i,f,m_i} p_{d_i,f,m_i}\}, \tag{29}$$

where $c_i \in \mathcal{C}_i$, $i \in \mathcal{I}$, $f \in \mathcal{F}$, $d_i \in \mathcal{D}_i$, $u_i \in \mathcal{U}_i$, $m_i \in \mathcal{M}_i$, $l_i \in \mathcal{L}_i$.

• Reward. Since the goal of optimization problem $\mathcal{P}1$ is to maximize the expected cumulative reward, which is directly determined by the sum rate of the system, the sum rate within the system is used as the reward value in the design. In addition, constraints should be considered in the selection of action,

and failure to meet constraints is considered a wrong assignment. Therefore, we set the reward function of agent j as the sum rate of all peer users after each AP performs resources allocation under constraints, otherwise, it is defined as negative feedback. The reward function of each agent can be expressed as

$$\boldsymbol{r}_j = \begin{cases} \sum_{f\in\mathcal{F}} e_{j,f,u_{V_j}} r_{j,f,u_{V_j}}, & Z_j = 0,\ \text{(19b)–(19h)}, \\ \sum_{f\in\mathcal{F}} e_{j,f,d_{V_j}} r_{j,f,d_{V_j}}, & Z_j = 1,\ \text{(19b)–(19h)}, \\ -r_{\text{reg}}, & \text{otherwise}, \end{cases} \tag{30}$$

where r_{reg} is a fixed value.

It can be observed that the reward of each agent depends on the actions of all agents and the current state of the environment. Inside each subsystem, the different choices of working modes and the interference caused by using the same PRB to serve different users mean that the actions taken by one agent will affect the performance of other agents. As a result, the environment observed by each agent becomes non-stationary and requires learning using the distributed policy of MADDPG.

There extends the actor-critic framework in the MADDPG. After receiving the state transition sequence $\left(\boldsymbol{s}_j, \boldsymbol{a}_j, \boldsymbol{r}_j, \boldsymbol{s}'_j\right)$ transmitted from all agents, the lower-layer controller stores it in the empirical replay buffer and learns the historical information in the experience replay buffer used by the critic network. If the critic network parameter of agent j is denoted as θ_j^C, and the behavior of agent j is evaluated from the global perspective, agents can be further guided to make a better choice. The network is then criticized for updating this parameter by minimizing loss

$$\boldsymbol{L}_j\left(\theta_j^C\right) = \mathbb{E}\left[\left(Q_j\left(\boldsymbol{s}_j, \boldsymbol{a}_j|\theta_j^C\right) - y_j\right)^2\right], \tag{31}$$

$$y_j = r_j + \zeta_L \max Q'_j\left(\boldsymbol{s}'_j, \boldsymbol{a}'_j|\theta_j^{C'}\right), \tag{32}$$

where $\theta_j^{C'}$ is parameter of the critic network. The Q function is used to estimate the performance of an agent in a given state and is updated using a deep neural network.

$$Q_j\left(\boldsymbol{s}\left(n\right), \boldsymbol{a}\left(n\right)\right) = \mathbb{E}\left[R_j\left(t\right)|\boldsymbol{s}_j\left(n\right), \boldsymbol{a}_j\left(n\right)\right] = \mathbb{E}\left[\sum_{k=0}^{\infty} \zeta_L^k r\left(n+k+1\right)|\boldsymbol{s}_j\left(n\right), \boldsymbol{a}_j\left(n\right)\right], \tag{33}$$

where n represents the n-th training of the algorithm. When the experience number is greater than the set minimum batch sampling size, the critic network parameter is updated through the following steps:

$$\theta_j^{C'} = \theta_j^C - \xi^C \nabla \boldsymbol{L}_j\left(\theta_j^C\right). \tag{34}$$

Meanwhile, according to the action value function calculated by the critic and its own observation information, the actor network updates the network parameter θ_j^C of agent j. The gradient direction for updating the actor network is

$$\nabla_{\theta_j^A} \boldsymbol{J}\left(\theta_j^A\right) \approx \mathbb{E}\left[\nabla_{\theta_j^A} Q_j\left(\boldsymbol{s}_j, \boldsymbol{a}_j|\theta_j^A\right) \nabla_{\theta_j^C} \pi_L^j\left(\boldsymbol{s}_j|\theta_j^C\right)\right]. \tag{35}$$

The actor network is updated by using policy gradient descent, that is

$$\theta_j^{A'} = \theta_j^A - \xi^A \nabla_{\theta_j^A} \boldsymbol{J}\left(\theta_j^A\right). \tag{36}$$

The MADDPG algorithm used by the lower-layer controller is summarized as Algorithm 2 on the next page.

4.3 Solution for upper-layer control policy based on DQN

Under the convergent lower-layer policy π_L^*, the optimal upper-layer policy π_U^* can be learned by solving the simple version of the target problem, that is

Algorithm 2 Lower-layer algorithm using MADDPG for mode selection and resource allocation

Initialize the critic and actor networks with random parameters.
for training_episode = 1 to $\max_1$ **do**
  All APs observe the initial environment state $\boldsymbol{s}$;
  for step = 1 to $\max_2$ **do**
    All APs take actions $\boldsymbol{a}$;
    All APs obtain the penalized reward $\boldsymbol{r}$;
    The environment evolves into next state $\boldsymbol{s}'$;
    Store sample data $(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{s}')$ in the experience replay buffer;
    while when step $\geqslant$ minimum batch sampling size **do**
      Update the critic networks by (34) and computes action gradients for all APs;
      Update the actor networks by (36) after all APs receive the action gradients;
    end while
    Update users location;
  end for
end for

Problem 3 (Upper-layer optimization)**.**

$$\begin{aligned} &\pi_U^* = \arg\min_{\pi_U} \{U|\pi_L^*\} \\ &\text{s.t.} \quad (19\text{i}). \end{aligned} \tag{37}$$

For the upper-layer controller, the AP subset inside each subsystem should be dynamically configured based on service traffic to maximize the overall utility of the system. Thus the state, action, and reward of the upper-layer controller are designed as follows.

• State. The upper-layer controller observes the dynamic changes of the traffic state and dynamically allocates an AP subset inside each subsystem according to the observation results. The global state information includes the average arrival rate and average delay of each slice. Since the packet arrival rate of the slice is a set fixed value, the global state of the n-th training for the upper-layer controller is

$$\boldsymbol{s}_n = \{D_{i,\text{delay-aware}} | \forall i \in \mathcal{I}\}. \tag{38}$$

• Action. To simplify the solution process of the optimization problem and reduce the computational complexity, the upper-layer controller separates the action of the association scheme: C_i, O_i, and only determines the number of APs inside each subsystem, so the action space can be represented by

$$\boldsymbol{a}_n = \{C_i | \forall i \in \mathcal{I}\}. \tag{39}$$

• Reward. Under the given lower-layer optimal control policy, the convergence goal of the upper-layer control strategy is to maximize the overall utility function of the system. Therefore, we define the reward function as the system utility that satisfies the constraints, and the reward that does not meet the conditions is defined as negative feedback, which is expressed as

$$\boldsymbol{r}_n = \begin{cases} -U, \ (19\text{i}), \\ r_{\text{reg}}, \ \text{otherwise}. \end{cases} \tag{40}$$

During the learning process, the state transition sequence $(\boldsymbol{s}_n, \boldsymbol{a}_n, \boldsymbol{r}_n, \boldsymbol{s}'_n)$ is stored in the experience replay buffer, a small batch of samples are randomly selected, and the network parameter φ are updated by the random gradient descent algorithm. We use the ε-greed rule, in which the upper controller randomly selects the AP number of associated slices and the maximum action value function to minimize the loss function to explore.

$$\nabla_\varphi \boldsymbol{L}(\varphi) = \mathbb{E}[(r + \zeta_U \max_{\boldsymbol{a}'_n} Q(\boldsymbol{s}'_n, \boldsymbol{a}'_n|\varphi') - Q(\boldsymbol{s}_n, \boldsymbol{a}_n|\varphi)) \nabla_\varphi Q(\boldsymbol{s}_n, \boldsymbol{a}_n|\varphi)]. \tag{41}$$

After the optimization of the loss function $\boldsymbol{L}(\varphi)$, the previous iteration parameter φ' is fixed, and the gradient of the loss function is obtained.

$$\boldsymbol{L}(\varphi) = \mathbb{E}_s \left[\left(Q(\boldsymbol{s}_n, \boldsymbol{a}_n|\varphi) - y_U \right)^2 \right], \tag{42}$$

$$y_U = r + \zeta_U \max_{\boldsymbol{a}'_n} Q(\boldsymbol{s}'_n, \boldsymbol{a}'_n|\varphi'). \tag{43}$$

In the calculation, the loss function is optimized by random gradient descent, which decays exponentially with the decay factor $\varepsilon_{\text{decay}}$ in each learning step until the minimum value $\varepsilon_{\min}$ is reached. Algorithm 3 shows the learning process of the upper-layer control policy.

Algorithm 3 Upper-layer algorithm using DQN for association of APs and slices

```
Initialize the neural networks with random parameters.
for training_episode = 1 to max_3 do
    The upper-layer controller observes the initial environment state s;
    for step = 1 to max_4 do
        The upper-layer controller takes action a based on ε-greedy rule;
        The upper-layer controller obtains the reward r;
        The environment evolves into next state s';
        Store sample data (s, a, r, s') in the experience replay buffer;
        while when step ⩾ minimum batch sampling size do
            Update the weights φ of Q function by performing stochastic gradient descent to minimize the loss function L(φ) in (42);
        end while
    end for
end for
```

In summary, we have obtained a solution for mode selection and resource allocation based on the double-layer DRL framework. At the lower layer, the MADDPG algorithm based on the actor-critic architecture is adopted. According to the local network state, APs select the working mode, then get an allocation scheme of PRBs and power distribution for each peer user. The upper-layer controller uses the DQN algorithm to obtain the AP-slice clustering strategy according to the feedback from the lower layer. The performance of the system can be improved by iterative learning of the double-layer DRL algorithm.

4.4 Computational complexity analysis and signaling overhead analysis

4.4.1 *Computational complexity analysis*

For the upper-layer control policy, we use DQN considering the configuration of APs and slices. For the lower-layer control policy, we use MADDPG considering the mode selection and multi-connection control within slices. Therefore, the algorithm complexity of the two layers needs to be analyzed respectively based on the above analysis.

• Computational complexity analysis of the lower-layer algorithm. For the lower-layer using the MADDPG algorithm, according to the above description, the computational complexity of learning procedure for the lower-layer control policy should be

$$\mathcal{O}\left(T_L J\left(\sum_{l=0}^{L_{\text{actor}}} n_l^{\text{actor}} n_{l+1}^{\text{actor}} + \sum_{l=0}^{L_{\text{critic}}} n_l^{\text{critic}} n_{l+1}^{\text{critic}}\right)\right), \tag{44}$$

where T_L is the learning steps of lower-layer policy training, J is the total number of APs, i.e., the total number of agents, n_l^{actor} is the number of neurons in the l-th layer of the actor neural network, i.e., the lower-layer control policy π_L, n_l^{critic} is the number of neurons in the l-th layer of the critic neural network, L_{actor} and L_{critic} represent the number of the hidden layers in the actor and critic network respectively.

• Computational complexity analysis of the upper-layer algorithm. The upper-level controller as an agent uses the DQN algorithm of discrete action space, so the complexity of the learning procedure for the upper-level control policy (i.e., Algorithm 3) is given by

$$\mathcal{O}\left(T_U\left(\sum_{l=0}^{L_{\text{DQN}}} n_l^{\text{DQN}} n_{l+1}^{\text{DQN}}\right)\right), \tag{45}$$

where T_U is the learning steps of upper-level policy training, n_l^{DQN} is the number of neurons in the l-th layer of DQN, and L_{DQN} is the number of the hidden layers in the DQN.

4.4.2 *Signaling overhead analysis*

To show the superiority of the slicing capacity-centered scheme in terms of signaling overhead, we also introduce the global capacity-centered scheme as a comparison. The most significant difference between

Table 2 Comparison of signaling overhead

Scheme	Operation time of each training (s)	Number of information exchange
Slicing capacity-centered scheme	1.00–1.38	$\sum_{i=1} \mathcal{K}_i \times \mathcal{C}_i$
Global capacity-centered scheme	2.07–2.21	$\mathcal{K} \times \mathcal{J}$

Table 3 System simulation parameters

Parameter	Value	Parameter	Value
The number of APs	6	Data processing rate of per PRB	5 bit/s
The number of antennas for each AP	4	Average packet arrival rate	0.2, 1 packets/s
The number of slices	2	Minimum data rate for each slice	400, 800 bit/s
The number of PRBs	12	Maximum packet delay tolerance for each slice	4, 8 ms
The number of UL/DL users in each slice	3, 3	Weighting factors for each slice	10^{-3}, 1
Bandwidth of system	100 kHz	Weighting factors for delays	10^{-4}, 10^{-3}, 1
Pass loss exponent	3.6	learning rate	10^{-2}
Total transmit power of AP and user	15, 9 dBm	Discount factor ζ_L, ζ_U	0.95, 0, 9
User optional power level	0/2/3/4 dBm	batch size	1024
Total packet length in slices	100, 500 Byte	$\varepsilon_{\min}$	0.01
Average service rate of each AP	25 bit/s	$\varepsilon_{\text{decay}}$	0.99

the two schemes is that APs in the former select cooperatively working modes inside each subsystem, while APs in the latter select globally. As shown in Table 2, we use the operation time of simulation as an important indicator to measure the signaling overhead. Simulation results show that the time cost of the global capacity-centered scheme is almost twice that of the slicing capacity-centered scheme. In addition, if the cost of subsystem formation is not taken into account, the AP $\mathcal{C}_i$ in each subsystem i in the slicing capacity-centered scheme only needs to exchange the information of user $\mathcal{K}_i$ to make the optimal decision. However, the global capacity-centered scheme requires all APs $\mathcal{J}$ to exchange the information of all users $\mathcal{K}$ to make the optimal decision, where $\sum_{i=1} \mathcal{K}_i = \mathcal{K}, \sum_{i=1} \mathcal{C}_i = \mathcal{J}$. Therefore, based on these two indicators, we can believe that the slicing capacity-centered scheme saves signaling overhead while ensuring capacity.

5 Simulation results

In this section, we consider a slicing capacity-centered NAFD cell-free distributed massive MIMO system in a circular region with a radius of 100 m. We analyze the learning convergence of the slicing capacity-centered scheme from the lower and upper layers. Under the given system parameters settings and algorithm parameters settings, from the perspectives of mode selection and resource allocation, the performance of the proposed scheme is compared to those of the two schemes to verify the advantages of the proposed scheme.

5.1 System simulation and algorithm simulation setup

It is assumed that there are two types of slices with different requirements of delay in the coverage area, and the QoS requirements of users associated with each slice are consistent. Assume that each slice has the same number of active users, and set the same number of users to select the working mode of UL reception or DL transmission. Before performing mode selection and resource allocation within a slice, PRBs have been evenly allocated among slices as communication resources. Since two slices have different delay requirements, different priorities are expressed by setting different weights of delay. The slice with higher delay requirements is set to slice 1, and the other slice is slice 0. The detailed simulation parameters are shown in Table 3.

5.2 Simulation results

5.2.1 *Simulation results of the slicing capacity-centered scheme*

In the slicing capacity-centered scheme, APs are associated with slices to serve only specific types of users, which not only reduces the collaboration overhead for mode selection but also further reduces interference. Figure 3 shows the rewards obtained by mode selection and resource optimization in the

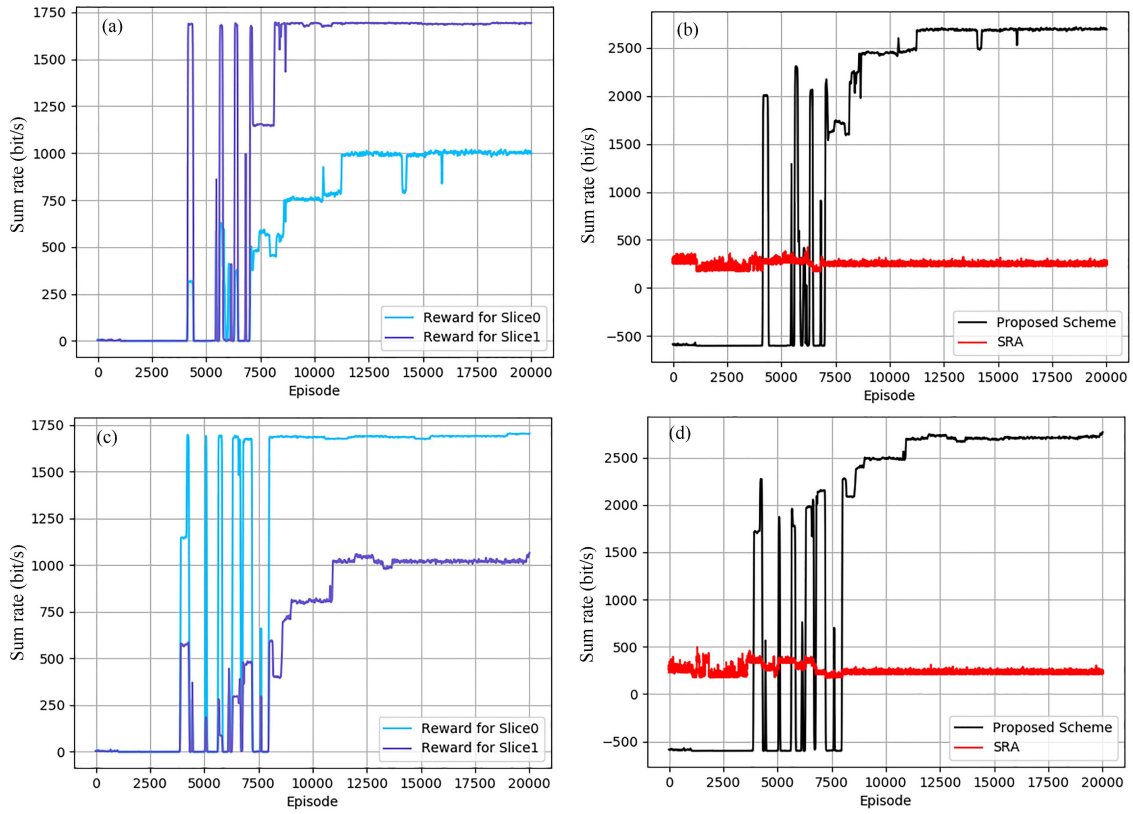

Figure 3 (Color online) Results of the lower-layer control policy under different association schemes between APs and slices in the upper-layer control. (a) Sum rate per slice (APs associated with slices 2:4); (b) sum rate of system (APs associated with slices 2:4); (c) sum rate per slice (APs associated with slices 4:2); (d) sum rate of system (APs associated with slices 4:2).

lower layer under different methods of clustering of APs and slices in the slicing capacity-centered scheme. The optimization problem shows that only associating one AP with a slice is forbidden. Figures 3(a) and (c) represent the rate of each slice when the number of APs associated with slice 0 and slice 1 is configured as 2:4 and 4:2, respectively, which show the impact of different association schemes between APs and slices on the transmission rate, so as to feed back to the upper layer to adjust the association scheme to improve the adaptability of utility and demand for slices. Figures 3(b) and (d), respectively, represent the global rate of the two association schemes. Meanwhile, a static resource allocation (SRA) scheme under the same working mode is also introduced. The SRA algorithm means that without hierarchical resource allocation and the association of APs, in other words, without slicing distinguishment, equal resources are allocated to each user and APs randomly under the fixed mode of APs.

Figure 3 shows that from the perspective of slicing, mode selection, and resource allocation considering the QoS of users, and on-demand allocation further improves the satisfaction of users. In the learning process of the proposed scheme, the AP-slice association method not only ensures the global capacity but also considers the customized service requirements of different slices. Therefore, the performance differences between slices in the system are considered when different association schemes between APs and slices are given in the upper layer. The mode selection and resource allocation inside each subsystem are independent of subsystems, and the sum rate differences between slices are only related to the association between the APs and slices controlled by the upper layer. The reward obtained by APs in the different association methods among two slices also proves that the proposed scheme concentrated on slicing capacity fully considers the different demands between slices and uses MADDPG to learn the optimal allocation policy, which not only obtains better performance but also provides a new paradigm for the mode selection of APs in NAFD systems to save the cost of collaboration among APs.

Figures 4(a) and (b) describe the learning results of the association between APs and slices by the upper-layer controller through the DQN algorithm. In the coverage area, regarding the slicing capacity-centered scheme, we consider two schemes of association between APs and slices, so there are two action spaces in total, corresponding to the two returns in Figure 3. Figure 4(a) shows that with the improvement

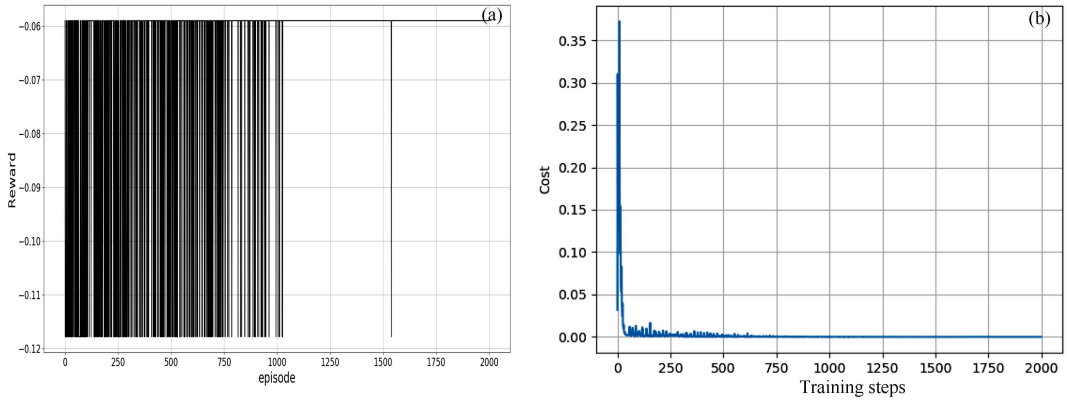

Figure 4 (Color online) Simulation results of the upper-layer control policy. (a) Result of AP-slice association in the upper layer; (b) cost of the upper-level DQN algorithm.

in learning progress, the DQN algorithm converges to the action with the highest reward. This result shows that the AP-slice association scheme determined by the upper-layer controller is driven by the priority of slices and converges the scheme that minimizes global delay. Figure 4(b) further proves that the DQN algorithm tends to converge with increasing training times. Since we set ε to increase as the learning progresses, the agent tends to take more randomly selected actions for more attempts at the beginning and tends to choose the optimal action later.

Comprehensively considering the delay of each slice, this paper selects the association scheme of APs and slices to maximize the system utility according to the weight setting. Therefore, the proposed DQN algorithm can solve the optimal association scheme of APs and slices under different network conditions. That is, the higher the delay requirement of a slice is, the greater the weight.

5.2.2 *Comparison of simulation results between the slicing capacity-centered scheme and the global capacity-centered scheme*

We refer to mode selection globally as a global capacity-centered scheme. To solve the problem of great overhead and the insufficient utilization of resources caused by global collaboration, we proposed the slicing capacity-centered scheme. Compared with the global capacity-centered scheme, the difference lies in that mode selection is in each subsystem formed by AP-slice association, while the similarity is that resources are isolated between slices, and different slices use different resources.

Figure 5 shows the reward using MADDPG for global mode selection with slicing assistance, which is the global rate of all users. Meanwhile, the SRA algorithm is used as the comparison algorithm.

The training trend in Figure 5 shows the superiority of the slicing-assisted dynamic resource allocation method in the global capacity-centered scheme. In addition, Figure 5 and Figures 3(b) and (d) show that when the MADDPG algorithm is used for learning, the optimal performance can be learned in both schemes with global capacity as the center and slicing capacity as the center and the optimization performance goal is improved manyfold compared with SRA. In the slicing capacity-centered scheme, PRBs are communication resources, APs are equivalent to computing resources, and the two resources are isolated between subsystems. From a global perspective, compared with the global capacity-centered scheme, the step-by-step isolation of the two resources can realize the gradual improvement in the achievable system capacity. Simulation results show that the time cost of the global capacity-centered scheme is almost twice that of the slicing capacity-centered scheme, and the advantage of the slicing capacity-centered scheme in saving signaling cost is also verified.

5.2.3 *Comparison of simulation results for the slicing capacity-centered scheme based on different algorithms*

Figure 6 compares the coverage and performance of different double-layer algorithms and the utility of the entire system in the training stage. The comparison schemes are as follows: (1) Comparative scheme: we use the K-means algorithm instead of the DRL algorithm to form flexible duplex subsystems, and the double-layer K-means-MADDPG algorithm is used to solve the slicing capacity-centered optimization scheme in this paper as a comparative scheme; (2) SRA.

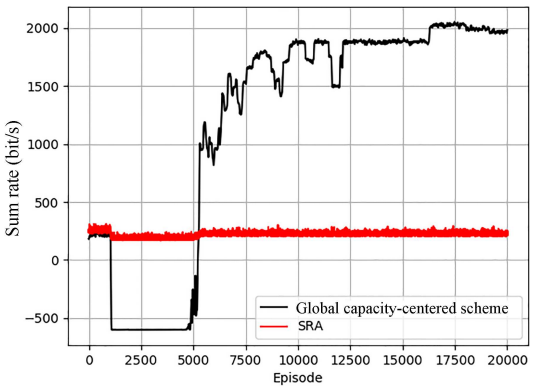

Figure 5 (Color online) Sum rate of the global capacity-centered scheme.

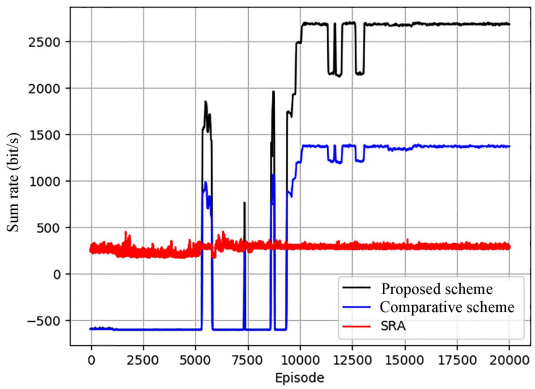

Figure 6 (Color online) System convergence and performance comparison of different schemes.

The double-layer K-means-MADDPG scheme is also divided into upper and lower layers, in which the upper-layer uses the K-means algorithm for AP-slice association to obtain different subsystems, while the lower-layer uses MADDPG for mode selection and resource allocation. The difference between the proposed algorithm and the comparative algorithm lies in the different processing methods of the upper-layer. The K-means algorithm only relies on geographical location for AP clustering, and APs lack cooperation in the process of forming flexible duplex subsystems.

Therefore, Figure 6 shows that it is difficult to reach the global performance obtained by the double-layer DRL algorithm through cooperation. This difficulty is encountered because the independence of APs fails to properly divide flexible duplex subsystems using the K-means algorithm. However, as the MADDPG is still used for resource allocation and mode selection, the convergence speed is identical for the two algorithms. In addition, comparing Figures 5 and 6 shows that because of the lack of cooperation between APs about clustering, the total rate achieved by the slicing capacity-centered scheme using the K-means-MADDPG is considerably lower than that achieved by the global capacity-centered scheme using MADDPG.

Without offline learning, once the system is connected, the SRA scheme can immediately outperform the DRL scheme, which is simple and fast. However, after the training of the DRL algorithm, the much lower efficiency of the SRA scheme is obvious.

6 Conclusion

In this paper, we proposed a slicing capacity-centered scheme in NAFD distributed massive MIMO systems, which not only improves resource utilization but also provides a new paradigm for mode selection of APs and further saves signaling overhead and time cost. Moreover, the addition of network slicing can satisfy diversified vertical communication requirements in NAFD scenarios. To efficiently solve the optimization problem centered on slicing capacity, a double-layer DRL mechanism was proposed to transform the mode selection and resource optimization problem into the system utility optimization problem with constraints. The upper-layer control policy used the DQN algorithm to determine different subsystems formed by APs and slices, and the lower-layer control policy used the MADDPG algorithm to select the working mode and allocate radio resources inside each subsystem. The results show that the slicing capacity-centered scheme effectively balances flexibility and cost under the premise of ensuring the global capacity. Furthermore, this scheme considers the customization needs of massive users with the support of the double-layer optimization mechanism.

Acknowledgements This work was supported in part by National Key Research and Development Program (Grant No. 2021YFB2900300), National Natural Science Foundation of China (Grants Nos. 61971127, 61871122), Southeast University-China Mobile Research Institute Joint Innovation Center, and Major Key Project of PCL (Grant No. PCL2021A01-2).

References

1 Razlighi M M, Zlatanov N. Buffer-aided relaying for the two-hop full-duplex relay channel with self-interference. IEEE Trans Wirel Commun, 2018, 17: 477–491
2 Wang D M, Wang M H, Zhu P C, et al. Performance of network-assisted full-duplex for cell-free massive MIMO. IEEE Trans Commun, 2020, 68: 1464–1478

3 Wang D M, Zhao Z L, Huang Y Q, et al. Large-scale multi-user distributed antenna system for 5G wireless communications. In: Proceedings of the 81st Vehicular Technology Conference (VTC Spring), Glasgow, 2015. 1–5
4 Wan Z Q, Pan Q J, Li J M, et al. Performance analysis of full-duplex densely distributed MIMO with wireless backhaul. Sci China Inf Sci, 2023, 66: 162303
5 Li J M, Lv Q, Zhu P C, et al. Network-assisted full-duplex distributed massive MIMO systems with beamforming training based CSI estimation. IEEE Trans Wirel Commun, 2021, 20: 2190–2204
6 Xia X J, Zhu P C, Li J M, et al. Joint sparse beamforming and power control for a large-scale DAS with network-assisted full duplex. IEEE Trans Veh Technol, 2020, 69: 7569–7582
7 Xia X J, Zhu P C, Li J M, et al. Joint user selection and transceiver design for cell-free with network-assisted full duplexing. IEEE Trans Wirel Commun, 2021, 20: 7856–7870
8 Zhu Y, Li J M, Zhu P C, et al. Optimization of duplex mode selection for network-assisted full-duplex cell-free massive MIMO systems. IEEE Commun Lett, 2021, 25: 3649–3653
9 Zhu Y, Li J M, Zhu P C, et al. Load-aware dynamic mode selection for network-assisted full-duplex cell-free large-scale distributed MIMO systems. IEEE Access, 2021, 10: 22301–22310
10 Mei J, Wang X, Zheng K. An intelligent self-sustained RAN slicing framework for diverse service provisioning in 5G-beyond and 6G networks. Intell Converged Netw, 2020, 1: 281–294
11 Parsaeefard S, Dawadi R, Derakhshani M, et al. Joint user-association and resource-allocation in virtualized wireless networks. IEEE Access, 2016, 4: 2738–2750
12 Ye Q, Zhuang W H, Zhang S, et al. Dynamic radio resource slicing for a two-tier heterogeneous wireless network. IEEE Trans Veh Technol, 2018, 67: 9896–9910
13 Parsaeefard S, Dawadi R, Derakhshani M, et al. Dynamic resource allocation for virtualized wireless networks in massive-MIMO-aided and fronthaul-limited C-RAN. IEEE Trans Veh Technol, 2017, 66: 9512–9520
14 Luong P, Gagnon F, Despins C, et al. Joint virtual computing and radio resource allocation in limited fronthaul green C-RANs. IEEE Trans Wirel Commun, 2018, 17: 2602–2617
15 Tseliou G, Adelantado F, Verikoukis C. NetSliC: base station agnostic framework for network slicing. IEEE Trans Veh Technol, 2019, 68: 3820–3832
16 Liu Y N, Wang X B, Boudreau G, et al. Deep learning based hotspot prediction and beam management for adaptive virtual small cell in 5G networks. IEEE Trans Emerg Top Comput Intell, 2020, 4: 83–94
17 Li J L, Shi W S, Yang P, et al. A hierarchical soft RAN slicing framework for differentiated service provisioning. IEEE Wireless Commun, 2020, 27: 90–97
18 Wu S C, Liu L Y, Zhang W B, et al. Revenue-maximizing resource allocation for multitenant cell-free massive MIMO networks. IEEE Syst J, 2022, 16: 3410–3421
19 Wang D M, You X H, Huang Y M, et al. Full-spectrum cell-free RAN for 6G systems: system design and experimental results. Sci China Inf Sci, 2023, 66: 130305
20 Wang H, Sun C, Li J M, et al. Joint optimization of spectral efficiency and energy efficiency with low-precision ADCs in cell-free massive MIMO systems. Sci China Inf Sci, 2022, 65: 152301
21 Chen X M, Ng D W K, Yu W, et al. Massive access for 5G and beyond. IEEE J Sel Areas Commun, 2021, 39: 615–637
22 Wei C, Xu K, Shen Z X, et al. Fingerprint-based localization and channel estimation integration for cell-free massive MIMO IoT systems. IEEE Int Things J, 2022, 9: 25237–25252
23 Wei C, Xu K, Xia X C, et al. User-centric access point selection in cell-free massive MIMO systems: a game-theoretic approach. IEEE Commun Lett, 2022, 26: 2225–2229
24 Lee N, Morales-Jimenez D, Lozano A, et al. Spectral efficiency of dynamic coordinated beamforming: a stochastic geometry approach. IEEE Trans Wireless Commun, 2015, 14: 230–241
25 Karlsson M, Bjornson E, Larsson E G. Techniques for system information broadcast in cell-free massive MIMO. IEEE Trans Commun, 2019, 67: 244–257
26 Yemini M, Goldsmith A J. Virtual cell clustering with optimal resource allocation to maximize cellular system capacity. In: Proceedings of IEEE Global Communications Conference (GLOBECOM), Waikoloa, 2019. 1–7
27 Zappone A, Di Renzo M, Debbah M. Wireless networks design in the era of deep learning: model-based, AI-based, or both? IEEE Trans Commun, 2019, 67: 7331–7376
28 Sun G, Gebrekidan Z T, Boateng G O, et al. Dynamic reservation and deep reinforcement learning based autonomous resource slicing for virtualized radio access networks. IEEE Access, 2019, 7: 45758–45772
29 Ren Y, Guo A H, Song C L, et al. Dynamic resource allocation scheme and deep deterministic policy gradient-based mobile edge computing slices system. IEEE Access, 2021, 9: 86062–86073
30 Kandath H, Senthilnath J, Suresh S. Mutli-agent consensus under communication failure using actor-critic reinforcement learning. In: Proceedings of IEEE Symposium Series on Computational Intelligence (SSCI), Bangalore, 2018. 1461–1465
31 Liu X, Ding H, Hu S. Uplink resource allocation for NOMA-based hybrid spectrum access in 6G-enabled cognitive Internet of Things. IEEE Int Things J, 2021, 8: 15049–15058
32 Tong Z, Zhang T K, Zhu Y T, et al. Communication and computation resource allocation for end-to-end slicing in mobile networks. In: Proceedings of IEEE/CIC International Conference on Communications in China (ICCC), Chongqing, 2020. 1286–1291
33 Guo S S, Wu D L, Zhang H X, et al. Queueing network model and average delay analysis for mobile edge computing. In: Proceedings of International Conference on Computing, Networking and Communications (ICNC), Maui, 2018. 172–176
34 Han Y, Tao X F, Zhang X F, et al. Hierarchical resource allocation in multi-service wireless networks with wireless network virtualization. IEEE Trans Veh Technol, 2020, 69: 11811–11827
35 Shen X M, Gao J, Wu W, et al. AI-assisted network-slicing based next-generation wireless networks. IEEE Open J Veh Technol, 2020, 1: 45–66