IEEE ComSoc IEEE Communications Society | IEEE COMPUTER SOCIETY | IEEE Signal Processing Society | VTS | IEEE Transactions on Machine Learning in Communications and Networking

Received 14 February 2023; revised 11 July 2023 and 1 October 2023; accepted 13 October 2023.
Date of publication 17 October 2023; date of current version 25 October 2023.

The associate editor coordinating the review of this article and approving it for publication was S. Coleri.

Digital Object Identifier 10.1109/TMLCN.2023.3325299

Deep Reinforcement Learning for Multi-User Massive MIMO With Channel Aging

ZHENYUAN FENG [1] (Member, IEEE), AND BRUNO CLERCKX [1,2] (Fellow, IEEE)

[1]Department of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ London, U.K.
[2]Silicon Austria Labs (SAL), 8010 Graz, Austria

CORRESPONDING AUTHOR: Z. FENG (z.feng19@imperial.ac.uk)

ABSTRACT The design of beamforming for downlink multi-user massive multi-input multi-output (MIMO) relies on accurate downlink channel state information (CSI) at the transmitter (CSIT). In fact, it is difficult for the base station (BS) to obtain perfect CSIT due to user mobility, and latency/feedback delay (between downlink data transmission and CSI acquisition). Hence, robust beamforming under imperfect CSIT is needed. In this paper, considering multiple antennas at all nodes (base station and user terminals), we develop a multi-agent deep reinforcement learning (DRL) framework for massive MIMO under imperfect CSIT, where the transmit and receive beamforming are jointly designed to maximize the average information rate of all users. Leveraging this DRL-based framework, interference management is explored and three DRL-based schemes, namely the distributed-learning-distributed-processing scheme, partial-distributed-learning-distributed-processing, and central-learning-distributed-processing scheme, are proposed and analyzed. This paper 1) highlights the fact that the DRL-based strategies outperform the random action-chosen strategy and the delay-sensitive strategy named as sample-and-hold (SAH) approach, and achieved over 90% of the information rate of two selected benchmarks with lower complexity: the zero-forcing channel-inversion (ZF-CI) with perfect CSIT and the Greedy Beam Selection strategy, 2) demonstrates the inherent robustness of the proposed designs in the presence of channel aging. 3) conducts detailed convergence and scalability analysis on the proposed framework.

INDEX TERMS Deep learning, interference management, massive MIMO, reinforcement learning, wireless communication.

I. INTRODUCTION

DUE to the increasing demand for data and connectivity in fifth-generation (5G) [1] and sixth-generation (6G) [2], multi-antenna technologies have attracted great attention in academia and industry. The research on multi-antenna techniques has promoted the development of multi-input multi-output (MIMO) technology. MIMO nowadays plays an indispensable role in the physical layer, media access control (MAC) layer, and network layer in wireless communications and networking [3]. At the physical layer, multi-antenna beamforming strategies have attracted great interest due to their ability to achieve considerable antenna gains, multiplexing gains, and diversity gains [4], [5], and gradually evolved into a massive MIMO system, in which the number of antennas at the BS reaches tens or even hundreds, attracting a larger number of users. To enable a high throughput in the massive MIMO system, the base station (BS) relies on the huge demand for global and instantaneous channel state information (CSI) based on efficient channel estimation techniques [6], [7]. Nevertheless, the ground/air/space platforms such as high-speed trains/unmanned aerial vehicles (UAV)/satellites have a common characteristic of 3D mobility which leads to a stringent time constraint on CSI acquisition and even causes misalignment of narrow beams. Therefore, in future communication systems, how to maintain good connectivity and system capacity without perfect channel state information at the transmitter (CSIT) (so-called imperfect CSIT) is regarded as an important problem that yearns for prompt solutions.

The imperfect CSIT is usually caused by the drastic change of the propagation environment due to user mobility [8] and CSI feedback/acquisition delay between the base station (BS) and users [9]. The CSI delay due to user mobility or feedback or acquisition delay is the time gap between the time point when the downlink training happens and the BS

starts downlink data transmission with the estimated channel. Such delay can be in the level of milliseconds which causes the estimated channels to be outdated when actually downlink transmission happens. This delay becomes more catastrophic at high user mobility since rapid channel variation inevitably causes performance degradation in massive MIMO systems [10]. This phenomenon, which is known as *channel aging*, describes the mismatch between estimated and update-to-date channels. That is to say, it represents the divergence arising between the channel estimation happening in the BS/users and the actual channel through which the data transmission occurs.

A. RELATED WORKS

To address the issue above, many papers have studied massive MIMO systems with imperfect CSIT [8], [9], [10], [11], [12], [13], [14], [15], [16] since CSIT is pivotal to the performance of systems that account for a great number of antennas and users. In [8], the impact of channel aging due to mobility is partially overcome through finite impulse response Wiener predictor without considering hardware phase noise, which is further studied in [10]. To tackle the CSI feedback/acquisition delay, one strategy is to use space-time interference alignment to optimize the degree of freedom (DoF) with delayed CSIT [11], [12]. Another method investigates the channel prediction based on the channel correlation [13] and past CSI [14]. In addition, to maintain the multi-user connectivity and mitigate the degrading effect of user mobility, low complexity power allocation methods are derived in [15] for Space Division Multiple Access (SDMA) which is outperformed by Rate-Splitting Multiple Access (RSMA) in [16] in terms of ergodic sum-rate.

On the one hand, the channel prediction approaches in the papers cited above [8], [9], [10], [11], [12], [13], [14] demonstrate good performance but experience extremely high complexity in channel prediction algorithms due to the increasing dimension of the antenna arrays. On the other hand, power allocation strategies in [15] and [16] exhibit lower complexity but sacrifice performance for tractability. To maintain a better balance of performance and complexity, an alternative strategy with lower complexity and looser CSI requirement needs to be developed urgently.

Machine learning (ML) [17] has demonstrated great usefulness in wireless systems [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38]. To cope with complex problems in a large-dimensional MIMO system, deep learning (DL) has drawn research interest in not only beamforming design [20], [21] by feeding CSI to the neural network but also channel prediction [22], [23], [24], [25], [26] by treating the time-varying channel as a time series, thanks to the strong representation capability of the deep neural network (DNN). Nevertheless, under stringent time constraints in mobility scenarios, the excellent generalization performance of DNN can not be fully exploited due to an insufficient number of data samples. In view of it, by elaborately treating the time-varying channel problem as a Markov decision process (MDP), deep reinforcement learning (DRL) has been regarded as a useful technology to design wireless communication systems by leveraging fast convergence of DL frameworks as well as continuous improvement characteristic in reinforcement learning (RL) algorithms [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39]. Systematically, a comprehensive tutorial in [39] reveals the applications of DRL for 5G and beyond. DRL is used to solve the power allocation problem in time-varying channels in [27] for single transmit antenna scenarios, and is further studied in [28] for multi-antenna beamforming and in [35] for multi-user conditions. In [30], [31], [32], and [37], DRL is utilized to tackle the passive beamforming design problem in reconfigurable intelligent surfaces (RIS)-aided communications and help reduce the computations compared to alternative frameworks. In terms of active beamforming using DRL, several efforts have been made on designing low complexity algorithms based on deep Q-network (DQN) [28], [34], [35], [36] and partially observed MDP [38] frameworks.

B. MOTIVATION AND SPECIFIC CONTRIBUTIONS OF THE PAPER

Existing works [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37] assume that perfect CSIT or instantaneous channel gain via receiver feedback is known at the transmitter. Unfortunately, such an assumption is impractical in real-world systems with CSI feedback/acquisition delay and user mobility [8], [9]. In addition, beamforming is not limited to the transmitter and can also be used at the receiver to perform better interference management. To our best knowledge, predicting the beamformers of both transmitter and receiver with imperfect CSIT is never considered in DRL-based papers. Instead, all the existing work focuses on high-level multi-cell single-user (SU) single-input-single-output (SISO) [27], [29] (no transmit and receive beamforming) and multi-input-single-output (MISO) (only transmit beamforming) [28], [31], [34], [38] scenarios without considering multiple receive antenna cases, which motivates this work. In addition to DRL-based strategies, a traditional procedure of beamforming is to use the frequency-division duplexing (FDD) pilot-based channel estimation procedure and zero-forcing channel inversion (ZF-CI) scheme [4]. Compared with DRL-based approaches, a key disadvantage of this method is that the system performance is heavily dependent on how fast the channel is changing as well as the feedback delay.

Motivated by the above, we study the joint transmit precoder and receive combiner design in massive MIMO downlink transmission with channel aging. The contributions of this paper are summarized as follows.

- We construct an efficient multi-agent DRL-based framework for massive MIMO downlink transmission,[1] in light of which three DRL-based algorithms were derived

[1]The terminology massive MIMO in this paper implies multiple receive antennas.

based on stream-level, user-level, and system-level agent modeling. This is the first paper showing that the DRL-based framework can be used to address very high-dimension optimization problems and demonstrates 1) robustness on the degrading effect of channel aging, 2) stringent interference management especially the inter-stream interference and multi-user interference.

- To address the challenge of high-dimensional antenna beamforming problems, by utilizing the DRL-based framework, three DRL-based schemes, namely distributed-learning-distributed-processing DRL-based scheme (DDRL), partial-distributed-learning-distributed-processing DRL-based scheme (PDRL), and central-learning-distributed-processing DRL-based scheme (CDRL), are proposed, analyzed, and evaluated. For DDRL, each stream is modeled as an agent. All the agents save their experiences in a private experience pool for later training. In contrast, in CDRL, the whole system is modeled as a central agent. What's more, to bridge DDRL and CDRL, we demonstrate another algorithm, i.e., PDRL which offers a more flexible design by modeling each user as an agent to balance the performance and complexity. Note that the DDRL and CDRL are different from those in [27] and [28] since we are tackling the problem with 1) receive beamforming with multiple receive antennas, 2) transmit beamforming under imperfect CSIT, 3) multiple streams for each user, and 4) a large number of transmit antennas at BS compared with SISO in [27] and 4-antenna MISO in [28] and [35], respectively.
- Leveraging the DRL-based framework mentioned above, the precoders at BS and combiners at users are jointly designed by gradually maximizing the average information rate through the observed reward. In particular, the BS decides the transmit precoder and receive combiner for each stream with imperfect CSIT and perfect CSIR. The merits of this design are shown through extensive simulations by benchmarking our schemes against the conventional, sample-and-hold (SAH) approach [26], zero-forcing channel-inversion (ZF-CI) strategy [4], greedy beam selection and random action-chosen scheme.
- We demonstrate the advantages of DRL-based strategies over the benchmarks above. In particular, the proposed algorithms show 1) fast convergence to efficient beamforming policy, 2) the robustness on tracing the channel dynamic against channel uncertainty due to channel aging, and 3) lower complexity compared with traditional beamforming strategy. All of these properties are essential in practical wireless networks.
- By numerical results, we show that our proposed DRL-based schemes outperform the SAH approach and random action-chosen scheme. In particular, DDRL can achieve nearly 90% of the performance of the state-of-the-art ZF-CI method with perfect CSI (ZF-CI PCSI) and 95% of the performance of the Greedy Beam Selection method but incurs more hardware complexity and more uplink overhead in an FDD setup. By increasing the resolution of the codebook and hyper-parameter tuning on the reward function, the performance can be further improved.

Organizations: The whole Section II is devoted to the system model, channel model, and the formulated sum-rate problem. In Section III, the basics of DRL are introduced, and three practical multi-agent DRL-based approaches are proposed. The simulation results are demonstrated in Section IV and this paper is concluded in Section V.

Notations: Boldface lower- and upper-case letters $\mathbf{H}$, and $\mathbf{h}$, denote matrices and vectors, respectively. $\mathbb{E}\{\cdot\}$ represents statistical expectation. $(\cdot)^{-1}$, $(\cdot)^T$, $(\cdot)^*$, and $(\cdot)^H$ indicate inversion, transpose, conjugate, conjugate-transpose, respectively. $\mathcal{R}$ and $\mathcal{I}$ denote the real and imaginary parts of a complex number, respectively. $\mathbf{I}_M$ denotes an $M \times M$ identity matrix. $\mathbf{0}$ denotes an all-zero matrix. $||\mathbf{a}||$ denotes the norm of a vector $\mathbf{a}$. $|a|$ denotes the norm of a variable a.

II. SYSTEM MODEL

Consider the MIMO broadcast channel (BC) with one M-antenna BS and K N-antenna users indexed by $\mathcal{K} = \{1, \ldots, K\}$ [40]. The BS aims to deliver M_s streams in the time instant of interest. For simplicity, A number of KN_s streams are transmitted simultaneously from the M antennas of the BS. Each group of N_s streams indexed by $N_s \in \mathcal{N} = \{1, \ldots, N_s\}$ is targeted at one of the K users. Note that we consider a setting where $M \geq KN_s$ to ensure the spatial multiplexing gain. The transmit power P is uniformly allocated to all KN_s streams. We assume that the BS and all users operate in the same time-frequency resource and are synchronized. The transmitted signal, i.e., the precoded data vector, at time slot t can be written as

$$\mathbf{x}(t) = \sqrt{\frac{P}{KN_s}} \sum_{k=1}^{K} \sum_{n=1}^{N_s} \mathbf{p}_{k,n}(t) s_{k,n}(t) \tag{1}$$

where $s_{k,n}, \forall k \in \mathcal{K}, \forall n \in \mathcal{N}$, is the encoded message from message $W_{k,n}$ with zero mean and $\mathbb{E}(|s_{k,n}|^2) = 1$, and precoder $\mathbf{p}_{k,n}(t) \in \mathbb{C}^{M\times 1}$ is subject to $\|\mathbf{p}_{k,n}(t)\|^2 = 1$. The received signal at user k can be expressed as

$$\begin{aligned}\mathbf{y}_k(t) = &\sqrt{\frac{P}{KN_s}} \mathbf{H}_k(t) \sum\nolimits_{n=1}^{N_s} \mathbf{p}_{k,n}(t) s_{k,n}(t) \\ &+ \sqrt{\frac{P}{KN_s}} \mathbf{H}_k(t) \sum\nolimits_{j\neq k, j=1}^{K} \sum\nolimits_{i=1}^{N_s} \mathbf{p}_{j,i}(t) s_{j,i}(t) \\ &+ \mathbf{n}_k(t)\end{aligned} \tag{2}$$

where the noise vector $\mathbf{n}_k \in \mathbb{C}^{N\times 1}$ is assumed to follow a complex normal distribution, i.e., $\mathbf{n}_k \in \mathcal{CN}(\mathbf{0}, \sigma_n^2 \mathbf{I}_N)$. At the user side, the combiner vector for each stream is denoted as $\mathbf{w}_{k,n}(t) \in \mathbb{C}^{N\times 1}$, $\|\mathbf{w}_{k,n}(t)\|^2 = 1, \forall k \in \mathcal{K}, n \in \mathcal{N}$. Then, the achievable rate for user k and the average user rate at time

slot t can be written as

$$R_k(t) = \sum_{n=1}^{N_s} G_{k,n}(t), \quad \bar{R}(t) = \frac{\sum_{k=1}^{K} R_k(t)}{K}, \tag{3}$$

where $G_{k,n}$ is the achievable rate of stream n for user k. To indicate the downlink information rate in each stream, by adopting the Shannon capacity equation, $G_{k,n}$ is given as

$$G_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t))) = \log(1 + \gamma_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t))) \tag{4}$$

where, for consistency with the notation in the following sections, $\gamma_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t))$ denotes the Signal-to-Interference-plus-Noise Ratio (SINR) of stream n for user k as

$$\gamma_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t)) = \frac{\frac{P}{KN_s}|\mathbf{w}_{k,n}^H(t)\mathbf{H}_k(t)\mathbf{p}_{k,n}(t)|^2}{I_{k,n}(t) + I_{c,k}(t) + \|\mathbf{w}_{k,n}(t)\|^2\sigma_n^2(t)} \tag{5}$$

where $\mathbf{W}_k(t) = [\mathbf{w}_{k,1}(t), \ldots, \mathbf{w}_{k,N_s}(t)]$ denotes the combining matrix and $\mathbf{P}_k(t) = [\mathbf{p}_{k,1}(t), \ldots, \mathbf{p}_{k,N_s}(t)]$ denotes the precoding matrix. The inter-stream interference for stream n of user k and the multi-user interference for stream n of user k are shown as

$$I_{k,n}(t) = \sum_{i=1, i\neq n}^{N_s} \frac{P}{KN_s}|\mathbf{w}_{k,n}^H(t)\mathbf{H}_k(t)\mathbf{p}_{k,i}(t)|^2 \tag{6}$$

and

$$I_{c,k}(t) = \sum_{j\in\mathcal{K}, j\neq k} \sum_{i=1}^{N_s} \frac{P}{KN_s}|\mathbf{w}_{k,n}^H(t)\mathbf{H}_k(t)\mathbf{p}_{j,i}(t)|^2, \tag{7}$$

respectively.

A. CHANNEL MODEL

We assume an extended Saleh-Valenzuela geometric model [41]. The channel between BS and user k is modeled as a L-path channel as is shown below

$$\mathbf{H}_k(t) = \sqrt{\frac{\eta_k MN}{L}} \cdot \sum_{l=1}^{L} \alpha_{k,l}(t) \cdot \mathbf{u}_{k,l}(t)\mathbf{v}_{k,l}^H(t) \tag{8}$$

where η_k denotes the large-scale fading coefficient and complex gain $\alpha_{k,l}(\forall k \in \mathcal{K}, \forall l \in \{1, 2, \ldots, L\})$ is assumed to remain the same at each time slot and varies between adjacent time slots according to the first-order Gaussian-Markov process

$$\alpha_{k,l}(t) = \rho\alpha_{k,l}(t-1) + \sqrt{1-\rho^2}e_{k,l}(t) \tag{9}$$

where $e_{k,l}(t) \backsim \mathcal{CN}(0, 1)$ and ρ is the time correlation coefficient obeying Jakes' model [42].

$$\rho = J_o(2\pi f_d \Delta_t \cos\theta) \tag{10}$$

where f_d and Δ_t denote the Doppler frequency and the channel instantiation interval, respectively, and J_0 denotes the first kind 0^{th} Bessel function. Since the users are assumed to move forward to the BS or away, i.e., $\theta = 0$ and maximum Doppler frequency $f_d^{\max}$ is achieved which is written as

$$\rho = J_o(2\pi f_d^{\max}\Delta_t). \tag{11}$$

In the typical case of a uniform linear array (ULA) where the antennas are deployed at both ends of the transmission, the array steering vectors $\mathbf{u}_{k,l}$ and $\mathbf{v}_{k,l}$ corresponding to the angle of arrival (AoA) $\phi_{A,k,l}$ and the angle of departure (AoD) $\phi_{D,k,l}$ in the azimuth are written as

$$\mathbf{u}_{k,l} = \frac{1}{\sqrt{N}}[1, e^{j2\pi\frac{d}{\lambda}\cos\phi_{A,k,l}}, \ldots, e^{j2\pi\frac{d}{\lambda}(N-1)\cos\phi_{A,k,l}}]^T \tag{12}$$

and

$$\mathbf{v}_{k,l} = \frac{1}{\sqrt{M}}[1, e^{j2\pi\frac{d}{\lambda}\cos\phi_{D,k,l}}, \ldots, e^{j2\pi\frac{d}{\lambda}(M-1)\cos\phi_{D,k,l}}]^T, \tag{13}$$

respectively, where λ is the wavelength of the signal and d denotes the inter-antenna space, which is usually set as $d = \lambda/2$, $\phi_{A,k,l} \backsim \mathcal{U}(\theta_{A,k,l} - \frac{\delta_A}{2}, \theta_{A,k,l} + \frac{\delta_A}{2})$ and $\phi_{D,k,l} \backsim \mathcal{U}(\theta_{D,k,l} - \frac{\delta_D}{2}, \theta_{D,k,l} + \frac{\delta_D}{2})$ with $\{\theta_{A,k,l}, \theta_{D,k,l}\}$ referring to the elevation angles and $\{\delta_A, \delta_D\}$ denoting the angular spread for arrival and departure, respectively [43].

B. PROBLEM FORMULATION

As described above, the system performance heavily relies on precoding and combining vectors design. However, there is an inevitable feedback delay between the time point when the user estimates the channel and the BS starts transmitting data with the estimated channel fed back by the users. As can be seen in Section II-A, such delay becomes quite problematic in high mobility scenarios since the channel changes fast and the correlation coefficient ρ decreases dramatically. Therefore, it is necessary to develop strategies that are robust to feedback delay and user mobility, which, in this paper, is interpreted as maximizing the sum-rate of K users based on the knowledge of past channels. The problem can be formulated as follows

$$\max_{\mathbf{W}_k(t), \mathbf{P}_k(t)} \sum_{k=1}^{K}\sum_{n=1}^{N_s} G_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t)) \tag{14a}$$

$$\text{s.t } \|\mathbf{p}_{k,n}(t)\|^2 = 1, \forall k, n \tag{14b}$$

$$\|\mathbf{w}_{k,n}(t)\|^2 = 1, \forall k, n \tag{14c}$$

$$\mathcal{F}(\mathbf{H}_k(t')), \forall k \text{ until } t' = t-1 \text{ are available}, \tag{14d}$$

where $\mathcal{F}(\mathbf{H}_k(t'))$ is a function of $\mathbf{H}_k(t')$ which is listed in Section III. Problem (14) aims at optimizing the precoder and combiner to maximize the sum-rate for served users subject to constraints (14b)- (14d), which is a non-convex problem. To solve this problem, three efficient DRL-based strategies are proposed in Section and III.

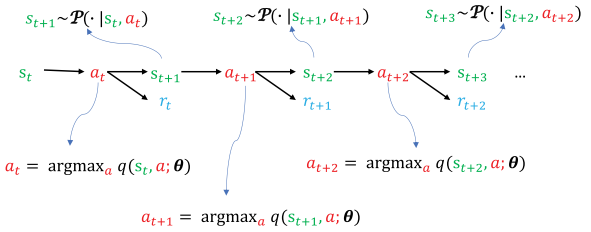

FIGURE 1. Markov decision process of Q-learning.

III. MULTI-AGENT DEEP REINFORCEMENT LEARNING FOR MULTI-USER MIMO DOWNLINK TRANSMISSION

To build up the foundation for the proposed DRL-based designs, an overview of DQN is illustrated first, followed by the description of the state, action, reward function, and three multi-agent DRL-based algorithms for the problem (14).

A. A BRIEF OVERVIEW OF DQN

In reinforcement learning (RL), an agent learns the optimal action policy to maximize the reward through trial-and-error interactions with the environment. RL is always formalized as an approach for Makov Decision Process (MDP) problems, which consists of $\mathcal{S}$, $\mathcal{A}$, $\mathcal{R}$, $\mathcal{P}$, and γ referring to a set of states, a set of actions, a reward function, a state transition function, and the discount factor. To be specific, at time t, an agent in state $s_t \in \mathcal{S}$ takes an action $a_t \in \mathcal{A}$ according to policy $\pi(a_t|s_t)$, obtains a reward $r_t = \mathcal{R}(a_t, s_t)$ and next state $s_{t+1} \in \mathcal{S}$ with probability $\mathcal{P}(s_t, a_t, s_{t+1})$ in return for the action taken. Formally, each transition (so-called experience of an agent in DQN) can be written as a tuple below

$$e_t = \langle s_t, a_t, r_t, s_{t+1} \rangle. \tag{15}$$

The optimal policy $\pi^*(a_t|s_t)$ is a mapping function between state and action to maximize the future accumulate reward

$$R_t = \sum_{\tau=0}^{\infty} \gamma^{\tau} \mathcal{R}(s_{t+\tau+1}, a_{t+\tau+1}) \tag{16}$$

where discount factor $\gamma \in [0, 1]$ balances the significance between immediate and future rewards. The optimal policy can be achieved by using dynamic programming (DP) methods that require detailed knowledge of the environment, i.e., $\mathcal{P}(s_t, a_t, s_{t+1})$, which is unavailable due to the variation of propagation channels.

To tackle this issue, as illustrated in Fig. 1, model-free Q-learning algorithms are demonstrated to continuously improve the policy through interactions with the environment. To be specific, the state-action value (called Q-value) is denoted as an expected reward of (s, a) by policy π

$$\mathcal{Q}_{\pi}(s_t, a_t) = \mathbb{E}_{\pi}(R_t|s_t = s, a_t = a) \tag{17}$$

where the expectation is calculated over all the possible (s, a) pairs given by policy π, which can be iteratively computed from the Bellman equation

$$\begin{aligned}\mathcal{Q}_{\pi}(s_t, a_t) = \mathcal{R}(r_{t+1}|s_t = s, a_t = a) + \gamma \sum\nolimits_{s' \in \mathcal{S}} \\ \times \Big(\mathcal{P}(s_{t+1} = s', s_t = s, a_t = a) \\ \times \max\nolimits_{a' \in \mathcal{A}} \mathcal{Q}_{\pi}(s', a') \Big)\end{aligned} \tag{18}$$

where $\mathcal{P}(s_{t+1} = s', s_t = s, a_t = a)$ denotes the transition probability from state s to s' after taking action a. The optimal policy returns the maximum expected cumulative reward at each s, i.e., $\pi^* = \arg\max_{\pi} \mathcal{Q}^{\pi}(s, a)$. Then the Q-value function can be represented as

$$\begin{aligned}\mathcal{Q}_{\pi^*}(s_t, a_t) = r_{t+1}(s_t = s, a_t = a, \pi = \pi^*) \\ + \gamma \sum\nolimits_{s' \in \mathcal{S}} \mathcal{P}(s_{t+1} = s', s_t = s, a_t = a) \\ \times \max\nolimits_{a' \in \mathcal{A}} \mathcal{Q}_{\pi^*}(s', a').\end{aligned} \tag{19}$$

In classical Q-learning, a Q-value table $q(s, a)$, named as Q-table, is constructed to represent the Q-value function $\mathcal{Q}_{\pi}(s, a)$. This table consists of a discrete set of $|\mathcal{S}| \times |\mathcal{A}|$ which is randomly initialized. The agent then takes actions according to an ϵ-greedy policy, receives reward $r = \mathcal{R}(s, a)$ and transfers to the next state s_{t+1} to complete the experience e_t. The Q-table is updated as

$$q(s_t, a_t) \longleftarrow (1 - \alpha) q(s_t, a_t) + \alpha (r_{t+1} + \gamma \max_{a'} q(s_{t+1}, a')) \tag{20}$$

where $\alpha \in [0, 1)$ is the learning rate. However, it is challenging to directly obtain the optimal $\mathcal{Q}_{\pi^*}(s_t, a_t)$ due to the uncertain variation of the dynamic channel environment, i.e., an unlimited number of states. To address the problems with such an enormous state space, deep Q-network (DQN) is utilized here to approximate the Q-value function, which can be expressed as $q(s_t, a_t, \boldsymbol{\theta})$ with $\boldsymbol{\theta}$ denoting the weights of DQN. The optimal policy π^* can be represented by a group of weights of the DQN. In addition, two techniques are exploited to strengthen the stability of DRL: target network and experience replay. The target network $q(s_t, a_t, \bar{\boldsymbol{\theta}})$ is another network that is initialized with the same set of weights of trained DQN. The target DQN is used to generate the target Q-value which is exploited to formulate the loss function of trained DQN. The weights of target DQN are updated periodically for every fixed number of slots T_s by replicating the weights of trained DQN to stabilize the training of trained DQN. The experience replay is intrinsically a first-input-first-output (FIFO) queue that stores E_m historical experiences in each training slot. During training, E_b experiences are sampled from the experience pool $\mathcal{O}$ to train the trained DQN to minimize the prediction error between the trained DQN and the target DQN. The loss function is defined as

$$L(\boldsymbol{\theta}) = \frac{1}{2E_b} \sum_{\langle s,a,r,s' \rangle \in \mathcal{O}} (r' - q(s, a; \boldsymbol{\theta}))^2 \tag{21}$$

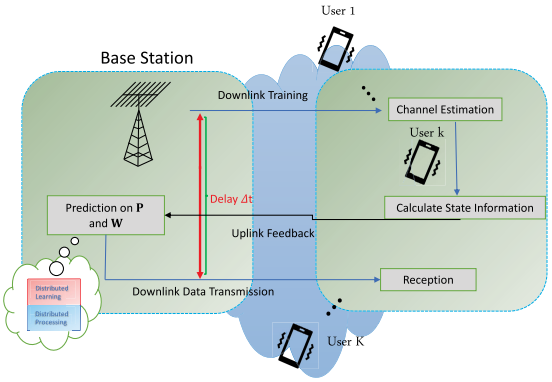

FIGURE 2. **The downlink training and uplink feedback of proposed DRL framework. The detail structure of the distributed-learning-distributed-processing framework is shown in Fig. 4.**

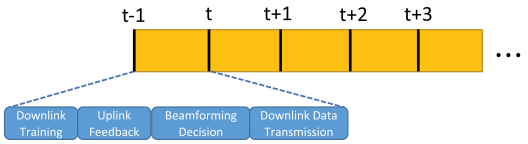

FIGURE 3. **Timing of time slot $t-1$.**

where $r' = r + \gamma \max_{a'} q(s', a'; \bar{\boldsymbol{\theta}})$, the weights of DQN $\boldsymbol{\theta}$ is updated by adopting a proper optimizer (e.g. RMSprop, Adam, and SGD). The specific gradient update is

$$\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \mathbb{E}_{s,a,r,s' \in \mathcal{O}}\Big[(r' - q(s, a; \boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} q(s, a; \boldsymbol{\theta})\Big]. \quad (22)$$

B. THE DISTRIBUTED-LEARNING-DISTRIBUTED-PROCESSING DRL-BASED ALGORITHM

In this section, we cast the problem (14) as a sequential decision-making process and tailor three multi-agent DRL algorithms to solve it. The DRL-based framework is elaborated first, followed by the derived algorithms. To our best knowledge, this is the first paper tackling the problem with 1) receive beamforming with multiple receive antennas, 2) transmit beamforming under imperfect CSIT, 3) multiple streams for each user, and 4) multiple users in a single cell compared with SISO in [27] and MISO in [28] with perfect CSIT, respectively. In addition, the PDRL is also firstly demonstrated in this paper to bridge DDRL and CDRL to balance the performance and complexity.

1) DOWNLINK TRAINING AND UPLINK FEEDBACK

As is shown in Fig. 2 and Fig. 3, at time slot $t-1$, the BS sends downlink pilots to users, based on which the downlink channels are perfectly estimated. User k can estimate the designed state information in Section III-B.4 and feed it back to the base station. With feedback from users, the BS can predict the indexes of precoders and combiners for time slot t and start downlink data transmission.

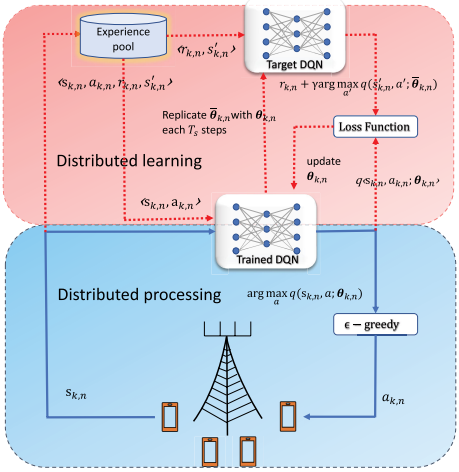

FIGURE 4. **The framework of distributed-learning-distributed-processing scheme.**

2) THE PROPOSED DRL-BASED ALGORITHM

To bring this insight to fruition, each stream is modeled as an agent, totally KN_s agents in our scheme. To be intuitive, we adopt a distributed-learning-distributed-processing framework as shown in Fig. 4 and demonstrated in Algorithm 1. At the initialization stage, all the KN_s pairs of DQNs are established at the BS. For instance, one pair of DQNs, namely trained DQN $q(s_{k,n}, a_{k,n}; \boldsymbol{\theta}_{k,n})$ and target DQN $q(s_{k,n}, a_{k,n}; \bar{\boldsymbol{\theta}}_{k,n})$ is possessed by agent (k, n). The input and output of trained DQN $q(s_{k,n}, a_{k,n}; \boldsymbol{\theta}_{k,n})$ are the local state $s_{k,n}$ and action $a_{k,n}$. In terms of the distributed learning procedure for agent (k, n), due to the feedback delay from users, only outdated CSI information is used to formulate the observations $s_{k,n}$ at the beginning of each time slot. Then, the DRL agent adopts an ϵ-greedy to balance exploitation and exploration by choosing actions, i.e, the precoder $\mathbf{p}_{k,n}$, and combiner $\mathbf{w}_{k,n}$ according to $s_{k,n}$, in which the agent executes an action with probability ϵ randomly, or executes the action $a_{k,n} = \max_a q(s_{k,n}, a; \boldsymbol{\theta}_{k,n})$ with probability $1-\epsilon$. Regarding the distributed learning process, the agent accumulates and stores the experience $e_{k,n} = \langle s_{k,n}, a_{k,n}, r_{k,n}, s'_{k,n} \rangle$ into experience pool and the historical experiences can be utilized to train the DQN with local state-action pairs together with the corresponding reward. Each agent has a profound view of the relationship between local state-action pairs and local long-term reward which, in return, leads the whole system to a distributed-learning-distributed-processing manner.

3) ACTIONS OF THE PROPOSED MULTI-AGENT DRL APPROACH FOR MASSIVE MIMO SCENARIO

As described in Section II, we aim to optimize the precoder $\mathbf{p}_{k,n}$ and combiner $\mathbf{w}_{k,n}, \forall k, n$. Then, the problem can be addressed by building two codebooks, i.e. $\mathcal{S}_{\mathrm{t}}$, $\mathcal{S}_{\mathrm{r}}$, which contain S_{t} and S_{r} beamforming vectors. In the decision-making stage, each agent chooses one precoder from $\mathcal{S}_{\mathrm{t}}$ and one

combiner from $\mathcal{S}_\mathrm{r}$. The action space can be represented as

$$\mathcal{A} = \{(\mathbf{c}_\mathrm{t}, \mathbf{c}_\mathrm{r}), \mathbf{c}_\mathrm{t} \in \mathcal{S}_\mathrm{t}, \mathbf{c}_\mathrm{r} \in \mathcal{S}_\mathrm{r}\} \tag{23}$$

where $\mathbf{c}_\mathrm{t}$ and $\mathbf{c}_\mathrm{r}$ denote the codewords of two codebooks and the cardinal number of action space $\mathcal{A}$ is $S_\mathrm{t} \times S_\mathrm{r}$. The design of codebooks comes from [44] which is also applied in [28], [34], and [35], and introduced here as a quantization of beam directions. To specify each element, we define matrix $\mathbf{C}_\mathrm{t} \in \mathbb{C}^{M \times S_\mathrm{t}}$ as

$$\mathbf{C}_\mathrm{t}[p, q] = \frac{\exp\left(j\frac{2\pi}{T}\lfloor\frac{M \bmod (q+\frac{S_\mathrm{t}}{2}, S_\mathrm{t})}{S_\mathrm{t}/T}\rfloor\right)}{\sqrt{M}} \tag{24}$$

where T is the number of available phase values and $\mathbf{C}_\mathrm{r} \in \mathbb{C}^{N \times S_\mathrm{r}}$ can be obtained by substituting the M and S_t with N and S_r accordingly. Each column of $\mathbf{C}_\mathrm{t}$ and $\mathbf{C}_\mathrm{r}$ corresponds to a specified codeword and the whole matrix forms a beamsteering-based beamformer codebook.

4) STATES OF THE PROPOSED DRL-BASED APPROACH FOR MASSIVE MIMO SCENARIOS

Under the mobility scenario, the receiver feedback is delayed at time slot t, and the state of agent (k, n) is constructed by the representative feature of observations from the last two successive time slots $t-1$ and $t-2$ without observations from time slot t. That is to say, at the beginning of time slot $t-u$, due to the delay of feedback, the BS is unable to instantaneously obtain the power of the received signal, i.e., $|\mathbf{w}_{k,n}^H(t)\mathbf{H}_k(t)\mathbf{p}_{k,n}(t)|^2$ and $|\mathbf{w}_{k,n}^H(t)\mathbf{H}_k(t-1)\mathbf{p}_{k,n}(t)|^2$. However, the historical feedback, i.e., $|\mathbf{w}_{k,n}^H(t-1)\mathbf{H}_k(t-1)\mathbf{p}_{k,n}(t-1)|^2$ and $|\mathbf{w}_{k,n}^H(t-1)\mathbf{H}_k(t-2)\mathbf{p}_{k,n}(t-1)|^2$ are usually available to the BS. Based on this assumption, the state $s_{k,n}(t)$ is designed as follows

- The "desired" information of the agent (k, n) which consists of 5 parameters, i.e., the channel gain $|\mathbf{w}_{k,n}^H(t-1)\mathbf{H}_k(t-1)\mathbf{p}_{k,n}(t-1)|^2$, the chosen index of precoder $U_{k,n}(t-1)$, the chosen index of combiner $V_{k,n}(t-1)$, the achievable rate of stream n for user k, i.e., $G_{k,n}(\mathbf{P}_k(t-1), \mathbf{W}_k(t-1))$, and the interference-plus-noise $I_{k,n}(t-1) + I_{\mathrm{c},k}(t-1) + \sigma_k^2$.
- Interference information of the agent (k, n) which is represented by 8 parameters, i.e., $\{\sum_{i=1,i\neq n}^{N_s} |\mathbf{w}_{k,n}^H(t-u)\mathbf{H}_k(t-u)\mathbf{p}_{k,i}(t-u)|^2$, $\sum_{i=1,i\neq n}^{N_s} |\mathbf{w}_{k,n}^H(t-1-u)\mathbf{H}_k(t-u)\mathbf{p}_{k,i}(t-1-u)|^2$, $\sum_{j\neq k}\sum_{i=1}^{N_s} |\mathbf{w}_{k,n}^H(t-u)\mathbf{H}_k(t-u)\mathbf{p}_{j,i}(t-u)|^2$, $\sum_{j\neq k}\sum_{i=1}^{N_s} |\mathbf{w}_{k,n}^H(t-1-u)\mathbf{H}_k(t-u)\mathbf{p}_{j,i}(t-1-u)|^2 | u \in \{1, 2\}\}$. It is worth noting here that in such a system, the interference information plays a key role in the maximization of its own information rate (the rate of stream n of user k), which, thus, should be included in state space.
- The information of agent (j, i), $(j, i) \neq (k, n)$, $\forall j, i$ consists of $10(KN-1)$ terms, i.e.,$\{U_{j,i}(t-u)$, $V_{j,i}(t-u)$, $G_{j,i}(\mathbf{M}_j(t-u)$, $\frac{P}{NK}|\mathbf{w}_{j,i}^H(t-u)\mathbf{H}_j(t-u)\mathbf{p}_{j,i}(t-u)|^2$, $\frac{P}{NK}|\mathbf{w}_{j,i}^H(t-u)\mathbf{H}_j(t-u)\mathbf{p}_{k,n}(t-u)|^2 | u \in \{1, 2\}\}$. The information of other agents plays an irreplaceable role for agent k to minimize the interference it causes to them, which, thus, should be included in state space.

To sum up, the cardinal number of state space is $10KN_s + 3$. Note that the adopted design is not guaranteed to be the optimal one but empirically achieves a good performance as demonstrated with evaluation results in Section III. The output size of the DQN is $S = S_\mathrm{t}S_\mathrm{r}$ which is equal to the number of available actions.

5) THE REWARD OF THE PROPOSED DRL-BASED APPROACH FOR THE MASSIVE MIMO SCENARIO

In this massive MIMO scenario, if agent (k, n) only tries to maximize the achievable rate of the stream (k, n) without taking the inter-stream and multi-user interference into consideration, a large interference will be delivered to other agents. Therefore, our proposed reward function $r_{k,n}$ consists of penalty coefficient λ and penalty term $P_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t))$ to quantify the adverse impact each agent causes to other agents. The penalty term $P_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t))$ is given as

$$\begin{aligned} P_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t)) = &\sum_{j=1,j\neq k}^{K}\sum_{i=1}^{N_s} \\ &\times \Bigg(\log_2\Big(1 + \frac{\frac{P}{KN_s}|\mathbf{w}_{j,i}^H(t)\mathbf{H}_j(t)\mathbf{p}_{j,i}(t)|^2}{\sigma^2 + \hat{I}_{k,n_1}(t) + \hat{I}_{\mathrm{c}_1,k}(t)}\Big) \\ &-G_{j,i}(\mathbf{W}_k(t), \mathbf{P}_k(t))\Bigg) + \sum_{i=1,i\neq n}^{N_s} \\ &\times \Bigg(\log_2\Big(1 + \frac{\frac{P}{KN_s}|\mathbf{w}_{k,i}^H(t)\mathbf{H}_k(t)\mathbf{p}_{k,i}(t)|^2}{\sigma^2 + \hat{I}_{k,n_2}(t) + \hat{I}_{\mathrm{c}_2,k}(t)}\Big) \\ &-G_{k,i}(\mathbf{W}_k(t), \mathbf{P}_k(t))\Bigg) \end{aligned} \tag{25}$$

where $\hat{I}_{k,n_1}(t)$, $\hat{I}_{k,n_2}(t)$, $\hat{I}_{\mathrm{c}_1,k}(t)$ and $\hat{I}_{\mathrm{c}_2,k}(t)$ are given by

$$\hat{I}_{k,n_1}(t) = \sum_{i=1,i\neq l}^{N_s} \frac{P}{KN_s}|\mathbf{w}_{j,i}^H(t)\mathbf{H}_j(t)\mathbf{p}_{k,i}(t)|^2, \tag{26}$$

$$\begin{aligned} \hat{I}_{\mathrm{c}_1,k}(t) = &\sum_{q\in\mathcal{K},q\neq k}\sum_{i=1}^{N_s} \frac{P}{KN_s}|\mathbf{w}_{j,i}^H(t)\mathbf{H}_j(t)\mathbf{p}_{q,i}(t)|^2 \\ &- \frac{P}{KN_s}|\mathbf{w}_{j,i}^H(t)\mathbf{H}_j(t)\mathbf{p}_{j,i}(t)|^2, \end{aligned} \tag{27}$$

$$\hat{I}_{k,n_2}(t) = \sum_{h=1,h\neq i,n}^{N_s} \frac{P}{KN_s}|\mathbf{w}_{k,i}^H(t)\mathbf{H}_k(t)\mathbf{p}_{k,h}(t)|^2, \tag{28}$$

and

$$\hat{I}_{\mathrm{c}_2,k}(t) = \sum_{q\in\mathcal{K},q\neq k}\sum_{h=1}^{N_s} \frac{P}{KN_s}|\mathbf{w}_{k,i}^H(t)\mathbf{H}_k(t)\mathbf{p}_{q,h}(t)|^2, \tag{29}$$

respectively. Note that $P_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t))$ is always a positive value due to the extraction of the interference from a specified stream. Then, the achievable rate for stream (k, n), i.e., $G_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t))$, is added into $r_{k,n}$ to highlight the contribution of agent (k, n) to the total information rate. Hence, $r_{k,n}$

at time slot t is given as

$$r_{k,n}(t) = G_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t)) - \lambda P_{k,n}(\mathbf{W}_k(t), \mathbf{P}_k(t)) \quad (30)$$

where penalty coefficient λ is used here as a weight parameter to manipulate the amount of negative effect in the reward function. In regard to the reward function, the rationale behind such a design is to maximize the achievable rate of improvement if the interference caused by stream (k, n) is totally eliminated. This design not only maximizes the achievable rate of stream (k, n), i.e., $G_{k,n}$ but also minimizes the negative effect it causes to other streams, i.e., $P_{k,n}$. Similar designs are comprehensively discussed in [27] and [28] which also confirm that a well-formulated reward function should act as a catalyst of the best decisions obtained by multiple agents.

6) DISCUSSION ON THE OVERHEAD AND COMPLEXITY OF THE PROPOSED FRAMEWORK

As is shown in Table 1, if the base station has to tell the users what combiner to use, then it can consume additional overhead on the downlink transmission. Fortunately, this overhead is negligible since only the indexes of combiners are delivered to users. Note that the precoders are also sent to terminals for the calculation of state information listed in the table. This reduces the computation burden on the base station for processing this state information.

In terms of the computational complexity of precoders and combiners in the demonstrated DRL-based approaches, the designed structure of target/trained DQNs includes four fully connected layers. Specifically, the input layer consists of $10KNs + 3$ neurons, followed by two hidden layers with L_1 and L_2 neurons and a specified activation function. The fourth layer serves as the output layer with S neurons. We employ two hidden layers in our design, as a two-layer feedforward neural network is sufficient to approximate any nonlinear continuous function based on the *universal approximation theorem* [46]. The computational complexity of fully connected DNN can be written as $\mathcal{O}((10KN_s+3)L_1+L_1L_2+L_2S)$ for each agent. This is much smaller than that of ZF-CI scheme due to the fact that ZF-CI involves matrix inversion which limits the scalability to a large number of transmit and receive antennas.

Remark 1: Note that different from [23] where the mobility estimation and channel prediction are needed, our work does not predict the channels sequentially. In this paper, we demonstrated a low complexity and efficient DRL-based framework and as this is the first work proposing DRL-based joint transmit and receiver beamforming for massive MIMO downlink transmission, we would like to keep the benchmarks as clear and simple as possible such that researchers can understand the fundamental benefits of the proposed strategies and carry on their studies in more practical scenarios in the future. The comparison with mobility estimation and channel prediction methods (such as VFK and MLP methods in [23]) could be addressed in future research, but not the scope of this paper.

Algorithm 1 DDRL Algorithm

Initialize: Establish a trained DQN and target DQN with random weights $\boldsymbol{\theta}_{k,n}$ and $\bar{\boldsymbol{\theta}}_{k,n}$, respectively, $\forall k \in \{1, 2, \ldots, K\}, \forall n \in \{1, 2, \ldots, N_s\}$, update the weights of $\bar{\boldsymbol{\theta}}_{k,n}$ with $\boldsymbol{\theta}_{k,n}$.
In the first E_s time slots, agent (k, n) randomly selects an action from action space $\mathcal{A}$, and stores the corresponding experience $\langle s_{k,n}, a_{k,n}, r_{k,n}, s'_{k,n}\rangle$ in its pool, $\forall k, n$.
for each time slot t **do**
  for each agent (k, n) **do**
    Obtain state $s_{k,n}$ from the observation of agent (k, n).
    Generate a random number ω.
    If $\omega < \epsilon$ **then**:
      Randomly select an action in action space $\mathcal{A}$.
    Else
      Choose the action $a_{k,n}$ according to the Q-function $q(s_{k,n}.a; \boldsymbol{\theta}_{k,n})$, $\forall k, n$
    End if .
    Agent (k, n) executes the $a_{k,n}$, immediately receives the reward $r_{k,n}$ and steps into next state $s'_{k,n}$, $\forall k, n$.
    Agent (k, n) puts experience $\langle s_{k,n}, a_{k,n}, r_{k,n}, s'_{k,n}\rangle$ into experience pool $\mathcal{O}_{k,n}$, randomly samples a minibatch with size E_b. Then, the weights of trained DQN $\boldsymbol{\theta}_{k,n}$ are updated using back propagation approach. The weights of target DQN $\bar{\boldsymbol{\theta}}_{k,n}$ is updated every T_s steps.
  end for
end for

C. THE LOW-COMPLEXITY CENTRALIZED-LEARNING-DISTRIBUTED-PROCESSING DRL-BASED ALGORITHM

In this section, we demonstrate an extra algorithm for the problem (14) for three reasons. *First*, a lower computation complexity is achieved in the centralized scheme by building and training on an extra pair of DQNs instead of distributedly training with KN_s agents. *Second*, a lower storage space is required with only a central experience pool during the learning process. *Third*, by saving and sampling the experiences from all distributed agents, the central agent can learn the common features from the channels of all users and intelligently guide the decision-making procedure of all distributed agents. Things need to be noted that CDRL is trained more efficiently using parameter sharing, which is based on homogeneous agents. This allows the policy to be trained with the experiences of all agents simultaneously. However, it still allows different actions between agents due to the fact that each agent receives different observations. This algorithm focuses on the decentralized parameter-sharing training scheme since we found it to be scalable if we continue to increase the number of users and streams.

It is worth noting that we don't need to design a new state, action, and reward function in CDRL since all the agents upload their experiences to a central pool for training the

TABLE 1. Comparison between strategies.

	FDD & ZF-CI [45]	TDD & ZF-CI [45]	FDD & MA-DRL
CSI Overhead Uplink	$MN + KN$ (MN channel coefficients and KN CSI pilot symbols)	KN sounding pilot symbols	$MN + KN + 11($ MN channel coefficients, KN CSI pilot symbols, and $\{\|\mathbf{w}_{k,n}^H(t-1)\mathbf{H}_k(t-1)\mathbf{p}_{k,n}(t-1)\|^2$, $G_{k,n}(\mathbf{P}_k(t-1), \mathbf{W}_k(t-1))$, $I_{k,n}(t-1) + I_{c,k}(t-1) + \sigma_k^2$, $\sum_{i=1,i\neq n}^{N_s} \|\mathbf{w}_{k,n}^H(t-v-u)\mathbf{H}_k(t-u)\mathbf{p}_{k,i}(t-v-u)\|^2$, $\sum_{j\neq k}\sum_{i=1}^{N_s} \|\mathbf{w}_{k,n}^H(t-v-u)\mathbf{H}_k(t-u)\mathbf{p}_{j,i}(t-v-u)\|^2$, $u \in \{1,2\}, v \in \{0,1\}\}$
CSI Overhead Downlink	MN CSI pilot symbols	0	$MN + 2N$ (MN channel coefficients and indexes of precoders and combiner $\mathbf{p}_{k,n}$ and $\mathbf{w}_{k,n}$)
Computing Complexity of Precoding and Combining Matrices	$\mathcal{O}((MN)^3)$	$\mathcal{O}((MN)^3)$	$\mathcal{O}((10KN+3)L_1 + L_1L_2 + L_2S)$

central DQN, and the central training DQN broadcasts its weights to all agents for later distributed execution.

The whole process is shown in Algorithm 2. At the initialization stage, only one pair of target and trained DQNs is built for the central agent. For each distributed agent, one trained DQN is established. In the first several time slots, each agent randomly selects an action and saves the experiences into the central experience pool. When the episode begins, the central agent adopts an ϵ-greedy strategy to balance exploitation and exploration so as to find the optimal policy. After learning from the sample experiences, the central agent broadcasts the updated weights of the central trained DQN to all other distributed agents for decision-making purposes.

D. BRIDGING THE DDRL AND CDRL: PARTIAL-DISTRIBUTED-LEARNING-DISTRIBUTED-PROCESSING SCHEME

In contrast with DDRL and CDRL, the partial-distributed-learning-distributed-processing DRL-based scheme (PDRL) offers a more flexible solution to the problem (14) by modeling each user as an agent. In the extreme case of $N_s = 1, K > 1$, PDRL boils down to DDRL by simply treating each stream as an agent. In the other extreme case of $K = 1, N_s > 1$, PDRL boils down to CDRL by forcing one central agent to do the training work. Compared with CDRL, PDRL demonstrates better performance-complexity balance by learning the representative features of the propagation environment for a specified user which is demonstrated in Fig. 12. The whole algorithm is illustrated in Algorithm 3.

IV. RESULT EVALUATION

This section demonstrates the performance of our proposed multi-agent DRL-based algorithm to maximize the average throughput of all the users. We first illustrate the simulation setup, followed by the simulation results in different scenarios.

A. SIMULATION SETUP

We consider a downlink transmission from one BS to multiple users. The BS serves $K = 4$ users in a single cell. The maximum transmit power P is fixed to 20 dBm and noise variance σ^2 at users is fixed to -114 dBm. The BS is equipped with $M = 32$ transmit antennas and the users are equipped with $N = 4$ receive antennas unless otherwise stated. Without loss of generality, the uniform linear array (ULA) is equipped in both transmitter and receiver sides with half-wavelength inter-antenna spacing. The large-scale channel fading is characterized by the log-distance path-loss model expressed below

$$\eta = L(d_0) + 10\omega \log_{10} \frac{d}{d_0}. \tag{31}$$

where $d = 10$ m is the BS-user distance. According to Table 3 of [47], the value of $L(d_0)$ for $d_0 = 1$ m is 68 dB and fading coefficient ω is 1.7. In terms of the shadowing model, the log-normal shadowing standard deviation β_k is set to 1.8 dB. The small-scale fading channel is generated according to the channel model introduced in Section II. Regarding the parameters of Jake's model with user speed 3.55 km/h, the maximum Doppler frequency $f_{\mathrm{d}}^{\max}$ and channel instantiation interval T_i are set as 800 Hz and 1×10^{-3} s, respectively [47]. The corresponding correlation coefficient ρ is $0.6514 \approx 0.65$.

As is illustrated in Fig. 4, the whole framework can be divided into 2 phases, the learning phase, and the processing phase. Before the learning phase, we randomly generate channels obeying Jake's model, randomly choose actions, observe the reward, and accumulate and store the corresponding experiences into the experience pool with size 1000 for the first 200 time slots, i.e., $E_m = 1000, E_s = 200$. In addition, the mini-batch size E_b is set as 32. Stepping into the learning stage, for the DNN, the number of neurons in two hidden layers, i.e, L_1, L_2, are both set as 256, followed by the *ReLu* activation function. The initial learning rate $\alpha(0)$ is $5e^{-3}$

Algorithm 2 CDRL Algorithm

Initialize: Establish a central trained DQN and central target DQN with random weights $\boldsymbol{\theta}_{\mathrm{c}}$ and $\bar{\boldsymbol{\theta}}_{\mathrm{c}}$ for the central agent, update the weights of $\bar{\boldsymbol{\theta}}_{\mathrm{c}}$ with $\boldsymbol{\theta}_{\mathrm{c}}$. Establish a trained DQN with random weight $\boldsymbol{\theta}_{k,n}$, $\forall k \in \{1, 2, \ldots, K\}, \forall n \in \{1, 2, \ldots, N_s\}$ for each distributed agent.
In the first E_s time slots, agent (k, n) randomly selects an action from action space $\mathcal{A}$, and stores the experience $\langle s_{k,n}, a_{k,n}, r_{k,n}, s'_{k,n}\rangle$, $\forall k, n$ in the experience pool of central agent $\mathcal{O}_{\mathrm{c}}$.
for each time slot t **do**
  for each agent (k, n) **do**
    Obtain state $s_{k,n}$ from the observation of agent (k, n).
    Generate a random number ω.
    If $\omega < \epsilon$ **then**:
      Randomly select an action in action space $\mathcal{A}$.
    Else
      Choose the action $a_{k,n}$ according to the Q-function $q(s_{k,n}.a; \boldsymbol{\theta}_{k,n})$, $\forall k, n$
    End if .
    Agent (k, n) executes the $a_{k,n}$, immediately receives the reward $r_{k,n}$ and steps into next state $s'_{k,n}$, $\forall k, n$.
    Agent (k, n) puts experience $\langle s_{k,n}, a_{k,n}, r_{k,n}, s'_{k,n}\rangle$ into central experience pool $\mathcal{O}_{\mathrm{c}}$.
  end for
  Central agent randomly samples a minibatch with size E_b. Then, the weights of central trained DQN $\boldsymbol{\theta}_{\mathrm{c}}$ are updated using the back propagation approach. The weights of target DQN $\bar{\bar{\theta}}_{\mathrm{c}}$ is updated every T_s steps. Then, central agent broadcasts the weights $\boldsymbol{\theta}_{\mathrm{c}}$ to all the distributed agents, i.e., $\boldsymbol{\theta}_{k,n} = \boldsymbol{\theta}_{\mathrm{c}}, \forall k, n$.
end for

and the decaying rate d_c is 10^{-4} such that the learning rate continues to decay with the number of time slots following $\alpha(t) = \alpha(t-1) * \frac{1}{1+d_c t}$. In terms of optimization, the adaptive moment estimation (*Adam*) is utilized to prevent the diminishing learning rate problem. To minimize the prediction error between trained DQN and target DQN, the weights of trained DQN are substituted into target DQN every 120 time slots, i.e., $T_s = 120$ with discount factor γ and penalty coefficient λ set as 0.1 and 1, respectively. During the processing phase, for ϵ-greedy strategy, we set the initial exploration coefficient ϵ as 0.7 which decays exponentially to 0.001 to strike a balance between exploration and exploitation during the training process. Note that the adopted parameters are not guaranteed to be optimal ones, which experimentally perform well in this setup. In the legend of simulation figures, DDRL and CDRL come from Algorithm 1 and Algorithm 2, respectively. The value of each point is a moving average over the previous 500 time slots unless otherwise stated.

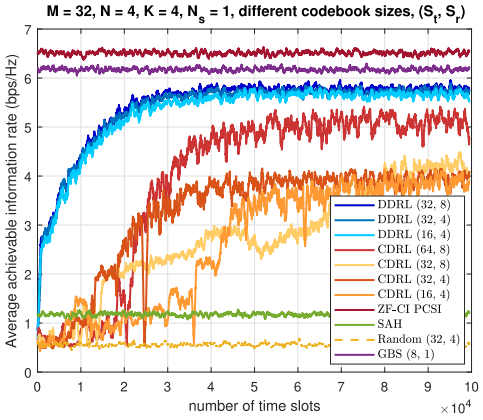

FIGURE 5. Average information rate versus the number of time slots with different codebook sizes (S_t, S_r).

To demonstrate the effectiveness of our DRL-based approaches, four benchmark schemes are evaluated, which are as follows:

- ZF-CI PCSI: Each agent executes the action from the scheme in [40] with instantaneous and perfect CSI, i.e., $\mathbf{H}_k(t), \forall k$.
- SAH: This approach stores the most recent estimated channel, i.e., $\mathbf{H}_k(t-1), \forall k$ and this approach always sends the channel coefficients to the base station, which will be used for calculating the precoders using ZF-CI. This strategy essentially ignores the non-negligible delay between the channel estimation and the time point when the actual DL transmission happens [26]. When $\rho = 1$, SAH is the same as ZF-CI PCSI. SAH only captures delay but assumes perfect knowledge of CSI at $t-1$.
- Random: Each agent randomly chooses actions. The performance serves as a lower bound in the simulation.
- Greedy Beam Selection (GBS): Each agent exhaustively selects an action in a greedy manner, the actions with the highest sum information rate are chosen as the solution for each channel realization. The benchmark serves as the upper bound for DRL-based strategies. Note that the size of the beam selection set increases exponentially with the size of the codebooks ($(NK)^{S_t S_r}$). For instance, when $K = 4, N = 1, S_{\mathrm{t}} = 32, S_{\mathrm{r}} = 4$, the total number of action combination is 4^{32} which is quite large considering the hardware constraint. Thus, we consider (8, 1) in this benchmark.

B. DDRL VS CDRL

Fig. 5 depicts the average achievable information rate versus the number of time slots with different numbers of transmit and receive beamformer codebook size (S_{t}, S_{r}). A *first* observation is that the performance gaps between two DRL-based schemes and SAH are gradually increased with the number of S_{t} and S_{r} and DDRL roughly observes a gain of 380% over SAH when $S_{\mathrm{t}} = 32$ and $S_{\mathrm{r}} = 4$. The reason behind such

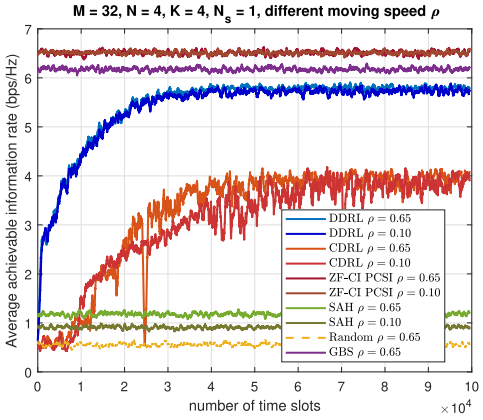

FIGURE 6. Average achievable information rate versus the number of time slots with different correlation coefficients.

a phenomenon is that, in DRL-based strategies, better interference management can be achieved by higher resolution in the codebook which significantly reduces the quantization error and effectively alleviates the interference from other streams. A *second* observation is that DDRL can achieve nearly 90% of the system capacity of the ZF-CI PCSI and 95% of the Beam Selection strategy, utilizing only a few pieces of information in the designed features from itself and other users. An interpretation is that the lack of instantaneous CSI and imperfect codebook design degrade the system performance and this 10% and 5%gap cannot be fully eliminated A *third* observation is that DDRL (32, 8) only demonstrates slightly better performance than DDRL (32, 4) and DDRL (16, 4) which shows of robustness on codebook size. A *fourth* observation is that the CDRL always demonstrates instability before convergence. An explanation is that the huge differences between the dynamic environment of different users make it extremely difficult to find the commonality among them. Then, each agent could be misled by the experiences of other agents, which, thus, results in fluctuations before convergence and degradation in system performance. Conversely, in DDRL, each agent selects a specific precoder and combiner for its intended stream which is relevant to its propagation environment and is considerably different among streams. This local adaptability greatly improves the performance of DDRL. After comprehensive considerations among computation complexity, system performance, and convergence speed, (32, 4) is chosen as a codebook baseline in the simulations of both DRL-based schemes. Compared with the Greedy Beam Selection method, the DRL-based strategies reveal extremely lower complexity on large action space but achieve roughly 95% of Greedy Beam Selection performance.2. Hence, CDRL is not suitable for practical systems where a large codebook is not available while DDRL demonstrates more robustness regarding codebook size but requires much more amount of memory and computing resources for training. We implemented the demonstrated algorithms with TensorFlow in a general computer, i.e., i7-8700 CPU, 3.20 GHz. The running time for different algorithms is listed in Table. 2.

Fig. 6 exhibits the average achievable information rate versus the number of time slots with different values of correlation coefficient ρ. The DDRL scheme with $\rho = 0.65$ and $\rho = 0.1$ can exceed the benchmark SAH with approximately 380% and 500%, respectively. This result greatly embodies the superiority of our DRL-based framework over the traditional massive MIMO optimizing scheme in mobility scenarios since a 20% performance degradation is caused by the fast-changing channels in SAH. In addition, it can be observed that DDRL with $\rho = 0.1$ demonstrates a slightly lower performance than $\rho = 0.65$ which is also shown in CDRL. An explanation is that the DDRL scheme is not sensitive to the dynamic and fast-changing wireless environment but CDRL needs more time steps to learn the representative features of the rapid-changing environment in high mobility scenarios, which results in a lower convergence. Note that the DRL-based methods have certain adaptability to environmental changes in user speed which can be interpreted as robustness on max Doppler frequency. Even though the correlation between adjacent channels is very small, the DRL-based frameworks still benefit from the exploration-exploitation strategy. Similar results are also observed in [34].

In Fig. 7, we assume that the users' rescheduling happens at the 50000th, 100000th, and 150000th-time slots. Instead of re-initializing the weights of all the DQNs in each agent, all of them continue the training process based on the designed information from newly scheduled users. *First*, a much higher start point and a comparable convergence time can be achieved in DDRL without witnessing a great performance collapse compared with the ZF-CI PCSI scheme. This can be interpreted by the fact that each agent tries to find the common features between the first scheduled and reschedule users which naturally makes a better decision based on these features and exhibits the ability for maintaining connectivity against user rescheduling in mobile networks. *Second*, with more rescheduling happening, a higher information rate and a faster convergence can be observed in CDRL. An interpretation is that the common features learned from the previously scheduled users boost the training in rescheduling. After learning and 'storing' more and more feature information into the weights of DQN, the central agent demonstrates universality to the channel uncertainty of rescheduled users and the neural network weights extracted from the previously trained DQN is a good candidate for the weight initialization of the current trained DQN in both schemes. However, if rescheduling happened frequently, the proposed DRL-based schemes can not converge to a set of good parameters if rescheduling happens before 25000-time slots but a jump-start happens when we implement a group of trained DQN to new users.

Fig. 8 investigates the average achievable information rate versus the number of time slots with $K = 4$ and 6, respectively. As opposed to the decrease of average user rate, the total cell throughput improves which suggests that both DRL-based approaches can benefit from multi-user diversity

TABLE 2. Running time for each network topology.

	DDRL (32, 4)	ZF PCSI	SAH	Beam Selection (8, 1)	Random Selection
Time	0.2s	0.8s	0.8s	10s	0.1s

TABLE 3. Total execution time for each channel realization with different (K, M, N, N_S) in DDRL($\epsilon = 0$).

	(1, 32, 4, 1)	(1, 32, 4, 2)	(2, 32, 4, 1)	(4, 32, 4, 1)	(4, 32, 4, 2)
Time	0.000992s	0.001993s	0.001996s	0.003786s	0.007270s

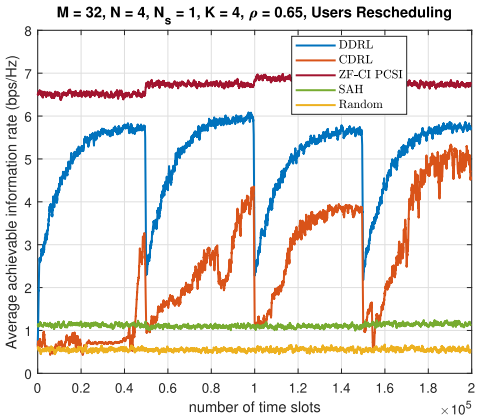

FIGURE 7. Average information rate versus number of time slots with users rescheduling at 5e4th, 1e5th and 15e4th time slot.

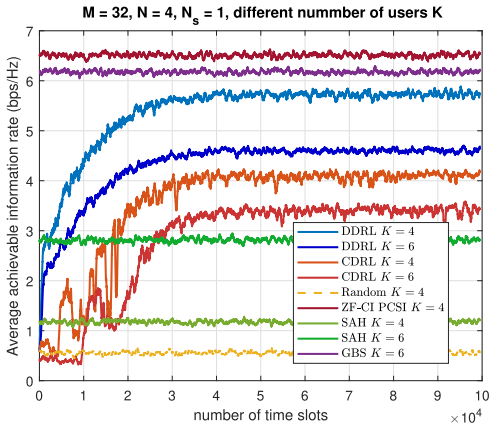

FIGURE 8. Average information rate versus number of time slots with different number of users K.

provided by time-varying channels across different users. In addition, although an 11.7% performance degradation can be observed in CDRL which is smaller than 19.8% in DDRL, CDRL achieves a much smaller performance gain over SAH in comparison to DDRL when $K = 6$. This suggests that DDRL is more robust in multi-user scenarios than CDRL.

To sum up in a nutshell, on the one hand, CDRL and DDRL are not equally suitable for massive MIMO systems with channel aging, namely, CDRL is less computationally complex but demonstrates instability and incurs performance loss. In contrast, DDRL offers a promising gain over CDRL by favoring an adaptive decision-making process and facilitating cooperation among all agents to mitigate interference but incurs a higher hardware complexity. Also, DDRL is more robust on rescheduling than CDRL considering the convergence speed and stability. On the other hand, both schemes demonstrate robustness on fading characteristics of the environment and changes on interference conditions.

C. MULTI-ANTENNA AND MULTI-STREAM

Fig. 9 illustrates the DRL-based scheme with 6 different numbers of transmit antennas M when $N = 1, N_s = 1, K = 1$. Without any penalty, i.e., inter-stream interference and multi-user interference, a near-optimal result can be observed by leveraging the proposed state, action, and reward design in an interference-free scenario with a stable increase of information rate, which thereby validates the effectiveness of the codebook design in Section III. The convergence time is proportional to M which limits its scalability. Intuitively, the reason for this effect on the convergence time is that it takes a longer time to learn the representative feature of high-dimension CSIT in sequential states. In contrast with Fig. 9, a serious multi-user and inter-stream interference is managed in Fig. 10 when $N = 4, N_s = 2, K = 4$. It can be observed that the transmit diversity and array gain cannot be fully achieved in the proposed DRL-based scheme if the rich interference is not properly suppressed due to the constraint of codebook precision and CSIT imperfections. Hence, the drawback of using DRL-based methods is that inter-stream interference can not be sufficiently alleviated if each agent fails to choose an action that causes small interference to all other agents during exploration and exploitation.

Fig. 11 characterizes the average achievable information rate versus the number of time slots with different numbers of streams for each user. First, DDRL has significantly higher performance compared to the conventional SAH scheme in different numbers of streams. Second, a 15.7% performance degradation can be observed from the DRL-based scheme between 1-stream and 2-stream scenarios which is smaller than that in SAH (around 28% between black and blue dotted lines). This reveals the privilege of the DDRL in inter-stream interference management.

An overview of the average information rate versus number of time slots with three DRL-based algorithms is

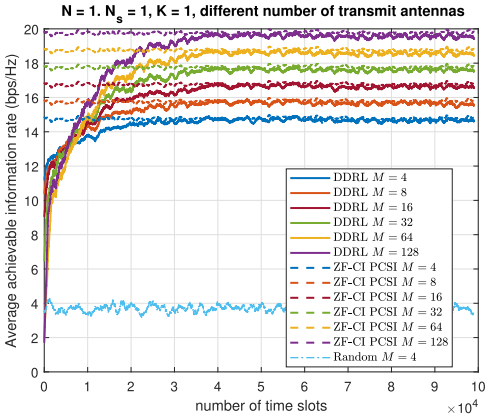

FIGURE 9. **Average information rate versus the number of time slots with different number of transmit antennas M when $N = 1$, $Ns = 1$, $K = 1$.**

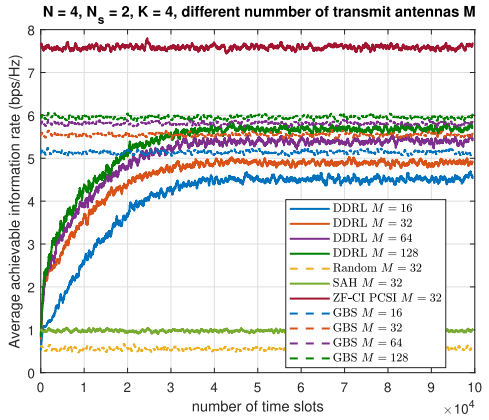

FIGURE 10. **Average information rate versus number of time slots with different number of transmit antennas M when $N = 4$, $Ns = 2$, $K = 4$.**

demonstrated in Fig. 12. Compared with DDRL and PDRL, a performance collapse is observed in CDRL due to the degrading effect of inter-stream interference. By flexibly modeling each user as an agent, PDRL greatly mitigates the inter-stream interference by learning the local observations from the target user's propagation channel.

D. REWARD, PENALTY ANALYSIS AND STATISTICAL TEST

To reveal the significance of the neural network size (L_1, L_2), the learning rate α and the discount factor γ, Fig. 13 shows the sum reward versus the number of time slots with different (L_1, L_2), α and γ. The first observation is that a faster convergence is observed with larger α, this is intuitive since the gradient descent is sped up with a larger value of loss function. The second observation is that, compared with (256, 256), a reward degradation appears with (32, 32), which suggests that increasing the DNN size demonstrates a stronger representation capability of input features and boosts the performance of the DRL-based scheme. Due to the negligible performance improvement in (512, 512), (256,256) is chosen

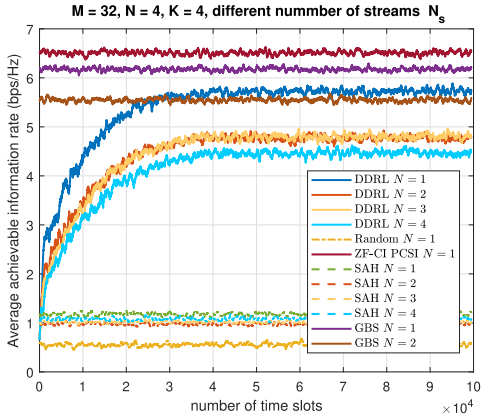

FIGURE 11. **Average information rate versus the number of time slots with different number of streams N.**

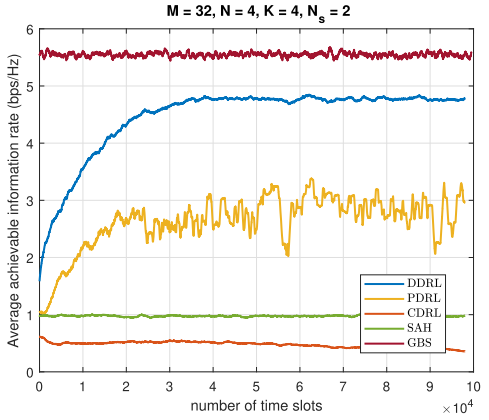

FIGURE 12. **Average information rate versus number of time slots with different DRL-based schemes.**

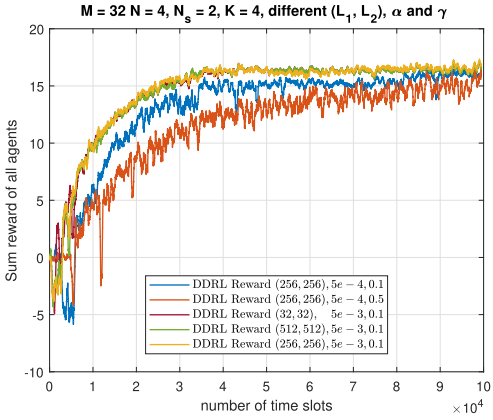

FIGURE 13. **Sum reward versus the number of time slots with different number of users K.**

as a baseline to maintain a balance between user connectivity and computational burden.

Fig. 14 offers an insight into the impact of different penalty values λ. This penalty term intrinsically represents an adjustment of reward function for each agent. Different from [27] and [28], it is demonstrated in Fig. 14 that the system capacity

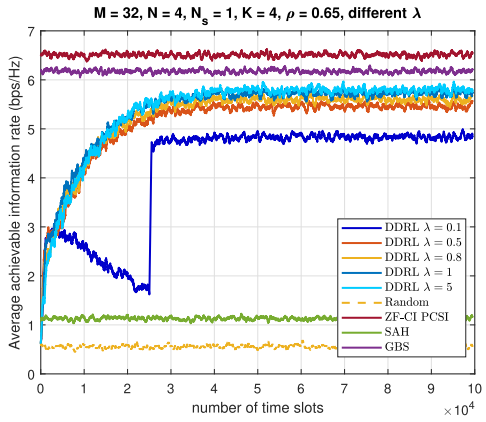

FIGURE 14. **Average information rate versus the number of time slots with different penalty λ.**

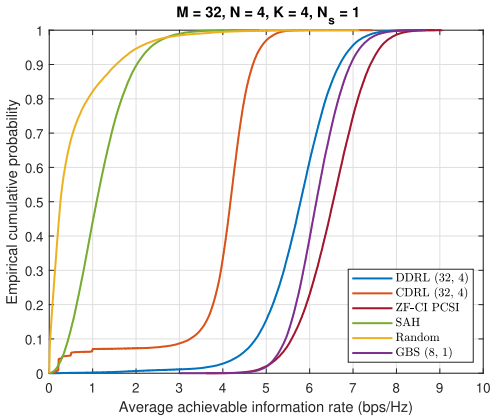

FIGURE 15. **Cumulative distribution function (CDF) of the average information rate over different DRL-based methods and benchmarks.**

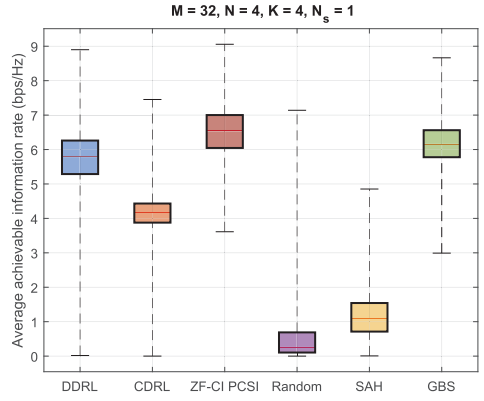

FIGURE 16. **Boxplot of the average information rate over different DRL-based methods and benchmarks. The bottom and top of each box are the 25th and 75th percentiles of the information rate values, respectively. The whisker length is set to infinity to ensure there are no outliers.**

is gradually increased with the penalty value from 0.1 to 5. An interpretation is that each agent causes high interference to other agents while still trying to maximize its information rate. Due to the uncertainty of the dynamic environment, the lack of perfect CSI introduces unpredictable interference for all the agents and an increase of penalty value can make a remedy for this by choosing an action that minimizes the interference to other agents instead of maximizing the received power of itself. This result also indicates that the decision-making process of all the agents is robust in unexpected high-interference scenarios.

The cumulative distribution function (CDF) over different DRL-based methods and benchmarks have been plotted in Fig. 15. It can be seen that the CDF curves confirm the discussion of the superiority of DDRL over other schemes. Also, the performance of DDRL is significantly limited by the codebook resolution which is confirmed in Fig. 16 where we show the boxplot of different methods.

V. CONCLUSION AND FUTURE RESEARCH

In this paper, we studied the beamforming optimization for massive MIMO downlink transmission with channel aging. An optimization framework in light of DRL was studied and three DRL-based algorithms were derived based on stream-level, user-level, and system-level agent modeling. Specifically, the transmit precoder at BS and receive combiner at user terminals were jointly optimized to maximize the average information rate. Furthermore, we analyzed the performance loss of DRL-based approaches as compared to the ideal case with continuous beamforming with different numbers of codebook sizes, users, antennas, streams, and user speeds. Interestingly, it was shown that even using a very low-resolution codebook in DDRL is still able to achieve 95% and 90% as in the case with GBS and ZF-CI, respectively. Simulation results showed that significant robustness on user mobility can be achieved by using some received power values of imperfect CSIT at the expense of more uplink overhead. Also, the convergence speed and scalability of the proposed algorithms are discussed. The convergence speed is linearly increased with the number of transmit antennas and performance degradation in the multiuser case is non-negligible due to the severe co-channel interference. In addition, CDRL consumes less computation complexity but demonstrates instability and incurs performance loss. In contrast, DDRL offers a promising gain over CDRL by favoring an adaptive decision-making process and facilitating cooperation among all agents to mitigate interference but incurs a higher hardware complexity and non-stationarity. Finally, the reward, penalty analysis, and statistical test confirm the fact that the performance of the proposed algorithms is greatly limited by the resolution of codebooks. Several important issues that are not addressed in our paper yet, some of which are listed as follows to motivate future research.

- *Multi-cell*: This paper considered single-cell multiuser conditions. However, when multi-cell is considered. The transmit power of the BS needs to be optimized, due to which the corresponding optimization problem is more challenging to solve, and thus is worthy of further investigation.

- *Extremely Large-scale MIMO*: To overcome the capacity constraints of conventional MIMO, extremely large-scale MIMO (XL-MIMO) are being proposed which can provide a much stronger beamforming gain to compensate for the severe path loss. As such, it is worth comparing the proposed massive MIMO with the XL-MIMO in future investigations.
- *Beamforming Codebook Design*: As is shown in this paper, the performance is greatly limited by the resolution of the designed codebook. A better codebook enables the system to handle larger and more complex channel conditions without compromising on performance.

APPENDIX
PDRL ALGORITHM

Algorithm 3 PDRL Algorithm

Initialize: Establish K pairs of trained/target DQNs with random weights $\boldsymbol{\theta}_k$ and $\bar{\boldsymbol{\theta}}_k, \forall k \in \{1, 2, 3, \ldots, K\}$ as user-specified agents, update the weights of $\bar{\boldsymbol{\theta}}_k$ with random $\boldsymbol{\theta}_k$. Build experience pool $\mathcal{O}_k, \forall k$. Establish a trained DQN with weight $\boldsymbol{\theta}_{k,n}, \forall k \in \{1, 2, \ldots, K\}, \forall n \in \{1, 2, \ldots, N\}$ for each distributed agent.
In the first E_s time slots, agent (k, n) randomly selects an action from action space $\mathcal{A}$, and stores the experience $\langle s_{k,n}, a_{k,n}, r_{k,n}, s'_{k,n}\rangle, \forall k, n$ in the experience pool of corresponding user-specified agent $\mathcal{O}_k$.
for each time slot t **do**
  for each agent (k, n) **do**
    Obtain state $s_{k,n}$ from the observation of agent (k, n).
    Generate a random number ω.
    If $\omega < \epsilon$ **then**:
      Randomly select an action in action space $\mathcal{A}$.
    Else
      Choose the action $a_{k,n}$ according to the Q-function $q(s_{k,n}.a; \boldsymbol{\theta}_{k,n}), \forall k, n$
    End if .
    Agent (k, n) executes the $a_{k,n}$, immediately receives the reward $r_{k,n}$ and steps into next state $s'_{k,n}, \forall k, n$.
    Agent (k, n) puts experience $\langle s_{k,n}, a_{k,n}, r_{k,n}, s'_{k,n}\rangle$ into central experience pool $\mathcal{O}_k$.
  end for
  User-specified agent k randomly samples a minibatch with size E_b. Then, the weights of its trained DQN $\boldsymbol{\theta}_k$ are updated using back propagation approach. The weights of its target DQN $\bar{\boldsymbol{\theta}}_k$ is updated every T_s steps. Then, the user-specified agent broadcasts the weights $\boldsymbol{\theta}_k$ to the corresponding distributed agents, i.e., $\boldsymbol{\theta}_{k,n} = \boldsymbol{\theta}_k, \forall n$.
end for

ACKNOWLEDGMENT

The authors would like to thank Hongyu Li, Yumeng Zhang, and Dr. Onur Dizdar for stimulating discussions.

REFERENCES

[1] J. G. Andrews et al., "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.
[2] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini, "What should 6G be?" *Nature Electron.*, vol. 3, no. 1, pp. 20–29, Jan. 2020.
[3] Z. Xiao et al., "Antenna array enabled space/air/ground communications and networking for 6G," 2021, *arXiv:2110.12610*.
[4] V. Stankovic and M. Haardt, "Generalized design of multi-user MIMO precoding matrices," *IEEE Trans. Wireless Commun.*, vol. 7, no. 3, pp. 953–961, Mar. 2008.
[5] B. Clerckx and C. Oestges, *MIMO Wireless Networks: Channels, Techniques and Standards for Multi-Antenna, Multi-User and Multi-Cell Systems*. New York, NY, USA: Academic, 2013.
[6] W. Ding, F. Yang, C. Pan, L. Dai, and J. Song, "Compressive sensing based channel estimation for OFDM systems under long delay channels," *IEEE Trans. Broadcast.*, vol. 60, no. 2, pp. 313–321, Jun. 2014.
[7] J. Meng, W. Yin, Y. Li, N. T. Nguyen, and Z. Han, "Compressive sensing based high-resolution channel estimation for OFDM system," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 1, pp. 15–25, Feb. 2012.
[8] K. T. Truong and R. W. Heath, "Effects of channel aging in massive MIMO systems," *J. Commun. Netw.*, vol. 15, no. 4, pp. 338–351, Aug. 2013.
[9] T. R. Ramya and S. Bhashyam, "Using delayed feedback for antenna selection in MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 8, no. 12, pp. 6059–6067, Dec. 2009.
[10] A. K. Papazafeiropoulos, "Impact of general channel aging conditions on the downlink performance of massive MIMO," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 1428–1442, Feb. 2017.
[11] N. Lee and R. W. Heath, "Space-time interference alignment and degree-of-freedom regions for the MISO broadcast channel with periodic CSI feedback," *IEEE Trans. Inf. Theory*, vol. 60, no. 1, pp. 515–528, Jan. 2014.
[12] N. Lee, R. Tandon, and R. W. Heath, "Distributed space–time interference alignment with moderately delayed CSIT," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 1048–1059, Feb. 2015.
[13] H. Yin, H. Wang, Y. Liu, and D. Gesbert, "Addressing the curse of mobility in massive MIMO with Prony-based angular-delay domain channel predictions," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 12, pp. 2903–2917, Dec. 2020.
[14] A. Papazafeiropoulos and T. Ratnarajah, "Linear precoding for downlink massive MIMO with delayed CSIT and channel prediction," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2014, pp. 809–914.
[15] C. Kong, C. Zhong, A. K. Papazafeiropoulos, M. Matthaiou, and Z. Zhang, "Sum-rate and power scaling of massive MIMO systems with channel aging," *IEEE Trans. Commun.*, vol. 63, no. 12, pp. 4879–4893, Dec. 2015.
[16] O. Dizdar, Y. Mao, and B. Clerckx, "Rate-splitting multiple access to mitigate the curse of mobility in (Massive) MIMO networks," *IEEE Trans. Commun.*, vol. 69, no. 10, pp. 6765–6780, Oct. 2021.
[17] Y. Anzai, *Pattern Recognition and Machine Learning*. Amsterdam, The Netherlands: Elsevier, 2012.
[18] M. Nerini, V. Rizzello, M. Joham, W. Utschick, and B. Clerckx, "Machine learning-based CSI feedback with variable length in FDD massive MIMO," 2022, *arXiv:2204.04723*.
[19] N. Ma et al., "Reinforcement learning-based dynamic anti-jamming power control in UAV networks: An effective jamming signal strength based approach," *IEEE Commun. Lett.*, vol. 26, no. 10, pp. 2355–2359, Oct. 2022.
[20] J. Kim, H. Lee, and S.-H. Park, "Learning robust beamforming for MISO downlink systems," *IEEE Commun. Lett.*, vol. 25, no. 6, pp. 1916–1920, Jun. 2021.
[21] T. Lin and Y. Zhu, "Beamforming design for large-scale antenna arrays using deep learning," *IEEE Wireless Commun. Lett.*, vol. 9, no. 1, pp. 103–107, Jan. 2020.
[22] J. Yuan, H. Q. Ngo, and M. Matthaiou, "Machine learning-based channel prediction in massive MIMO with channel aging," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 2960–2973, May 2020.
[23] H. Kim, S. Kim, H. Lee, C. Jang, Y. Choi, and J. Choi, "Massive MIMO channel prediction: Kalman filtering Vs. machine learning," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 518–528, Jan. 2021.
[24] C. Wu, X. Yi, Y. Zhu, W. Wang, L. You, and X. Gao, "Channel prediction in high-mobility massive MIMO: From spatio-temporal autoregression to deep learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 1915–1930, Jul. 2021.

[25] Z. Qin, H. Yin, Y. Cao, W. Li, and D. Gesbert, "A partial reciprocity-based channel prediction framework for FDD massive MIMO with high mobility," 2022, *arXiv:2202.05564*.
[26] Y. Zhang, A. Alkhateeb, P. Madadi, J. Jeon, J. Cho, and C. Zhang, "Predicting future CSI feedback for highly-mobile massive MIMO systems," 2022, *arXiv:2202.02492*.
[27] Y. S. Nasir and D. Guo, "Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2239–2250, Oct. 2019.
[28] J. Ge, Y.-C. Liang, J. Joung, and S. Sun, "Deep reinforcement learning for distributed dynamic MISO downlink-beamforming coordination," *IEEE Trans. Commun.*, vol. 68, no. 10, pp. 6070–6085, Oct. 2020.
[29] L. Zhang and Y.-C. Liang, "Deep reinforcement learning for multi-agent power control in heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2551–2564, Apr. 2021.
[30] 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.
[31] H. Yang, Z. Xiong, J. Zhao, D. Niyato, L. Xiao, and Q. Wu, "Deep reinforcement learning-based intelligent reflecting surface for secure wireless communications," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 375–388, Jan. 2021.
[32] 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.
[33] W. Li, W. Ni, H. Tian, and M. Hua, "Deep reinforcement learning for energy-efficient beamforming design in cell-free networks," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Mar. 2021, pp. 1–6.
[34] R. Zhang, K. Xiong, Y. Lu, B. Gao, P. Fan, and K. B. Letaief, "Joint coordinated beamforming and power splitting ratio optimization in MU-MISO SWIPT-enabled HetNets: A multi-agent DDQN-based approach," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 2, pp. 677–693, Feb. 2022.
[35] H. Chen, Z. Zheng, X. Liang, Y. Liu, and Y. Zhao, "Beamforming in multi-user MISO cellular networks with deep reinforcement learning," in *Proc. IEEE 93rd Veh. Technol. Conf. (VTC-Spring)*, Apr. 2021, pp. 1–5.
[36] Q. Hu, Y. Liu, Y. Cai, G. Yu, and Z. Ding, "Joint deep reinforcement learning and unfolding: Beam selection and precoding for mmWave multiuser MIMO with lens arrays," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2289–2304, Aug. 2021.
[37] H. Ren, C. Pan, L. Wang, W. Liu, Z. Kou, and K. Wang, "Long-term CSI-based design for RIS-aided multiuser MISO systems exploiting deep reinforcement learning," *IEEE Commun. Lett.*, vol. 26, no. 3, pp. 567–571, Mar. 2022.
[38] M. Fozi, A. R. Sharafat, and M. Bennis, "Fast MIMO beamforming via deep reinforcement learning for high mobility mmWave connectivity," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 127–142, Jan. 2022.
[39] N. C. Luong et al., "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3133–3174, 4th Quart., 2019.
[40] H. Sung, S.-R. Lee, and I. Lee, "Generalized channel inversion methods for multiuser MIMO systems," *IEEE Trans. Commun.*, vol. 57, no. 11, pp. 3489–3499, Nov. 2009.
[41] V. Raghavan, J. Cezanne, S. Subramanian, A. Sampath, and O. Koymen, "Beamforming tradeoffs for initial UE discovery in millimeter-wave MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 543–559, Apr. 2016.
[42] T. Kim, D. J. Love, and B. Clerckx, "MIMO systems with limited rate differential feedback in slowly varying channels," *IEEE Trans. Commun.*, vol. 59, no. 4, pp. 1175–1189, Apr. 2011.
[43] Y.-C. Liang and F. P. S. Chin, "Downlink channel covariance matrix (DCCM) estimation and its applications in wireless DS-CDMA systems," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 2, pp. 222–232, Feb. 2001.
[44] W. Zou, Z. Cui, B. Li, Z. Zhou, and Y. Hu, "Beamforming codebook design and performance evaluation for 60 GHz wireless communication," in *Proc. 11th Int. Symp. Commun. Inf. Technol. (ISCIT)*, Oct. 2011, pp. 30–35.
[45] E. Björnson, E. G. Larsson, and T. L. Marzetta, "Massive MIMO: Ten myths and one critical question," *IEEE Commun. Mag.*, vol. 54, no. 2, pp. 114–123, Feb. 2016.
[46] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang, "The expressive power of neural networks: A view from the width," in *Advances in Neural Information Processing Systems*, vol. 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran, 2017, pp. 6231–6239.
[47] P. F. M. Smulders, "Statistical characterization of 60-GHz indoor radio channels," *IEEE Trans. Antennas Propag.*, vol. 57, no. 10, pp. 2820–2829, Oct. 2009.

ZHENYUAN FENG (Member, IEEE) received the B.Eng. degree in communication engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2016, and the M.S. degree in signal processing and communication from The University of Edinburgh, Edinburgh, U.K., in 2017. He is currently pursuing the Ph.D. degree with Imperial College London, U.K. He was a Data Scientist with the China Telecom Institution, Shanghai. His current research interests include wireless power transfer, deep learning, multi-agent reinforcement learning, and signal processing and communication.

BRUNO CLERCKX (Fellow, IEEE) received the M.S. and Ph.D. degrees in applied science from University Catholique de Louvain, Louvain-la-Neuve, Belgium, in 2000 and 2005, respectively. From 2006 to 2011, he was with Samsung Electronics, Suwon, South Korea, where he actively contributed to 4G (3GPP LTE/LTE-A and IEEE 802.16 m) and acted as the Rapporteur for the 3GPP Coordinated Multi-Point (CoMP) Study Item. He is currently a Full Professor, the Head of the Wireless Communications and Signal Processing Laboratory, and the Deputy Head of the Communications and Signal Processing Group, Electrical and Electronic Engineering Department, Imperial College London, London, U.K. He is also the CTO of Silicon Austria Labs (SAL). He has authored two books, more than 200 peer-reviewed international research articles, and 150 standards contributions. He is the inventor of 80 issued or pending patents among which 15 have been adopted in the specifications of 4G standards and are used by billions of devices worldwide. His current research interests include communication theory and signal processing for wireless networks. He is a TPC member and the symposium chair or the TPC chair of many symposia on communication theory, signal processing for communication, and wireless communication for several leading international IEEE conferences. He was an elected member of the IEEE Signal Processing Society SPCOM Technical Committee. He was an Editor of IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and IEEE TRANSACTIONS ON SIGNAL PROCESSING. He served as a Lead Guest Editor for the Special Issues of the *EURASIP Journal on Wireless Communications and Networking*, IEEE ACCESS, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, and IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING. He was an Editor of the 3GPP LTE-Advanced Standard Technical Report on CoMP. He is an IEEE Distinguished Lecturer of the IEEE Communications Society.