Downlink Power Control for Cell-Free Massive MIMO With Deep Reinforcement Learning

Lirui Luo, Jiayi Zhang, *Senior Member, IEEE*, Shuaifei Chen, *Graduate Student Member, IEEE*, Xiaodan Zhang, Bo Ai, *Fellow, IEEE*, and Derrick Wing Kwan Ng, *Fellow, IEEE*

***Abstract*—Recently, model-free power control approaches have been developed to achieve the near-optimal performance of cell-free (CF) massive multiple-input multiple-output (MIMO) with affordable computational complexity. In particular, deep reinforcement learning (DRL) is one of such promising techniques for realizing effective power control. In this paper, we propose a model-free method adopting the deep deterministic policy gradient algorithm (DDPG) with feedforward neural networks (NNs) to solve the downlink max-min power control problem in CF massive MIMO systems. Our result shows that compared with the conventional convex optimization algorithm, the proposed DDPG method can effectively strike a performance-complexity trade-off obtaining 1,000 times faster implementation speed and approximately the same achievable user rate as the optimal solution produced by conventional numerical convex optimization solvers, thereby offering effective power control implementations for large-scale systems. Finally, we extend the DDPG algorithm to both the max-sum and the max-product power control problems, while achieving better performance than that achieved by the conventional deep learning algorithm.**

***Index Terms*—Beyond 5-G network, cell-free massive MIMO, deep reinforcement learning, power control, DDPG, downlink.**

I. Introduction

Cell-free (CF) massive multiple-input multiple-output (MIMO) is a promising technique providing ubiquitous communications for the beyond fifth-generation (5 G) wireless systems, where all access points (APs) cooperate among each other to simultaneously serve all users exploiting the same time-frequency resources via time-division duplex (TDD) [1], [2]. As there are no cells or cell-boundaries in CF massive MIMO, this special feature offers the potential of providing uniformly good throughput for all user equipments (UEs) [3].

To unleash its potential, proper power control is necessary for optimizing the CF massive MIMO user rate and mitigating inter-user interference that requires the application of advanced optimization techniques [4]. Indeed, the conventional optimization-based power control method has been well-studied at the expense of high time complexity, which limits their practical implementation [5]. On the other hand, model-free machine learning-based approaches can significantly reduce the computational complexity while achieve almost the same performance as the optimization-based approaches [6]. However, most of the existing model-free machine learning-based studies focus on supervised learning [5], which are also impractical since the prior optimal output data is hard to obtain in reality.

Reinforcement learning (RL) is a disruptive method that does not require a training data set a priori such that it is suitable for dynamic wireless environments. In particular, RL is concerned with optimizing the policy of a goal-oriented agent that learns from interactions with its environment directly [7]. In recent years, RL-based solutions have been studied for various power control problems. For example, [8] proposed a deep Q-learning (DQN) power control method with imperfect channel state information. Also, the authors of [9] proposed a scalable distributive multi-agent DQN approach that can apply to large networks in real-world scenarios. Despite the promising performance brought by the application of the DQN algorithm, this approach is limited to discrete optimization problems only. As such, performing downlink continuous power control with directly DQN applying would undoubtedly decrease a considerable amount of achievable rate. As a remedy, in this paper, we propose a deep deterministic policy gradient (DDPG) algorithm framework that can be adopted for continuous-valued control for solving various types of downlink power control problems in CF massive MIMO systems [10]. Due to the shortcomings of overfitting in DRL [10], we introduce feedforward neural networks (NNs) attached to the output layer of the DDPG algorithm such that the proposed algorithm is adaptive to environments. The major contributions of this paper are two-fold:

- We first develop a DDPG framework with NNs for addressing the CF massive MIMO downlink max-min power control problem that approaches the performance produced by the convex optimization solver while greatly reducing the computational time complexity.
- We then extend the DDPG framework to the max-sum and max-product power control problems of CF massive MIMO downlink, which shows that the proposed algorithm has better performance than conventional deep learning algorithms.

Notations: Boldface letters denote column vectors. The superscripts ^{T}, *, and ^{H} denote transpose, conjugate, and conjugate transpose, respectively. $\mathbb{E}\{\cdot\}$ denotes the statistical expectation operators. The circularly symmetric complex Gaussian distribution and real-valued Gaussian distribution are denoted by $\mathcal{CN}(0,\sigma^2)$ and $\mathcal{N}(0,\sigma^2)$, respectively.

II. System Model

We consider a CF massive MIMO network with M single-antenna APs and K single-antenna UEs, where the APs are connected to a central processing unit (CPU) via the perfect fronthaul links. All K UEs are simultaneously served by all M APs within the coverage of the network.

Note that the use of different channel models would not affect the development of our proposed algorithm. For simplicity, we assume Rayleigh fading. The channel coefficient between the m-th AP and the k-th UE is modeled as

$$g_{mk} = \beta_{mk}^{1/2} h_{mk}, \forall m \in \{1,\ldots,M\}, k \in \{1,\ldots,K\}, \tag{1}$$

Manuscript received October 12, 2021; revised January 17, 2022; accepted March 22, 2022. Date of publication March 31, 2022; date of current version June 24, 2022. This work was supported by the National Training Program of Innovation and Entrepreneurship for Undergraduates under Project 202210004006. The work of Derrick Wing Kwan Ng was supported in part by the UNSW Digital Grid Futures Institute, UNSW, Sydney, through cross-disciplinary fund scheme and in part by the Australian Research Council's Discovery Project under Grant DP210102169. The review of this article was coordinated by Prof. Yue Gao. *(Corresponding authors: Jiayi Zhang; Xiaodan Zhang.)*

Lirui Luo, Jiayi Zhang, and Shuaifei Chen are with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China, and also with the Frontiers Science Center for Smart High-speed Railway System, Beijing Jiaotong University, Beijing 100044, China (e-mail: 19211429@bjtu.edu.cn; jiayizhang@bjtu.edu.cn; 18111008@bjtu.edu.cn).

Xiaodan Zhang is with the School of Management, Shenzhen Institute of Information Technology, Shenzhen 518172, China (e-mail: zhangxd@sziit.edu.cn).

Bo Ai is with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China, and also with the Frontiers Science Center for Smart High-speed Railway System, Beijing Jiaotong University, Beijing 100044, China (e-mail: boai@bjtu.edu.cn).

Derrick Wing Kwan Ng is with the School of Electrical Engineering and Telecommunications, University of New South Wales, Kensington, NSW 2052, Australia (e-mail: w.k.ng@unsw.edu.au).

Digital Object Identifier 10.1109/TVT.2022.3162585

where $\beta_{mk} \in \mathbb{R}$ and $h_{mk} \in \mathbb{C}$ are the large- and the small-scale fading, respectively. We assume that h_{mk} are independent and identically distributed (i.i.d.) $\mathcal{CN}(0,1)$ random variables (RVs) [1].

A. Uplink Pilot Transmission

To estimate the channel, the UEs simultaneously send pilot sequences to the APs during the training phase. The pilot sequence $\boldsymbol{\varphi}_k$ with sample size τ_p used by the k-th UE is denoted as $\sqrt{\tau_p}\boldsymbol{\varphi}_k \in \mathbb{C}^{\tau_p \times 1}$, $||\boldsymbol{\varphi}_k||^2 = 1$. Thus the $\tau_p \times 1$ pilot signal vector received at the m-th AP yields

$$\mathbf{y}_{p,m} = \sqrt{\tau_p \rho_u} \sum_{k=1}^{K} g_{mk} \boldsymbol{\varphi}_k + \mathbf{w}_{p,m}, \tag{2}$$

where ρ_u denotes the uplink normalized transmission power for each pilot symbol and $\mathbf{w}_{p,m} \sim \mathcal{CN}(0,1)$ denotes the additive white Gaussian noise at the m-th AP. To obtain the best estimate channel from the AP to the k-th UE, we first project $\mathbf{y}_{p,m}$ onto $\boldsymbol{\varphi}_k^H$ and obtain:

$$\check{y}_{p,mk} = \sqrt{\tau_p \rho_u} g_{mk} + \sqrt{\tau_p \rho_u} \sum_{k' \neq k}^{K} g_{mk'} \boldsymbol{\varphi}_k^H \boldsymbol{\varphi}_{k'} + \boldsymbol{\varphi}_k^H \mathbf{w}_{p,m}. \tag{3}$$

Then, the typical MMSE estimate of g_{ul}^{mk} is denoted as

$$\begin{aligned} \hat{g}_{mk} &= \frac{\mathbb{E}\left\{\check{y}^*_{\mathrm{p},mk} g_{mk}\right\}}{\mathbb{E}\left\{\left|\check{y}_{\mathrm{p},mk}\right|^2\right\}} \check{y}_{p,mk} \\ &= \frac{\sqrt{\tau_p \rho_u} \beta_{mk}}{\tau_p \rho_u \sum_{k'=1}^{K} \beta_{mk'} \left|\boldsymbol{\varphi}_k^H \boldsymbol{\varphi}_{k'}\right|^2 + 1} \check{y}_{p,mk}. \end{aligned} \tag{4}$$

B. Downlink Payload Data Transmission

We assume that the APs treat the uplink channel estimations as downlink channels exploiting the channel reciprocity [11]. Besides, known for its low computational complexity and the ability to maximize the power of the desired signal [1], the conjugate beamforming is used. In the downlink, the APs transmit signals to the K UEs, which is given by

$$x_m = \sqrt{\rho_{\mathrm{d}}} \sum_{k=1}^{K} \eta_{mk}^{1/2} \hat{g}^*_{mk} q_k, \tag{5}$$

where $q_k \in \mathbb{C}$ satisfying $\mathbb{E}\{|q_k|^2\} = 1$ is the symbol intended for the k-th UE. Variable η_{mk} is the power control coefficients satisfying the following power allocation constraint:

$$\mathbb{E}\left\{|x_m|^2\right\} \leq \rho_{\mathrm{d}}. \tag{6}$$

Then, the power constraint can be formulated as

$$\sum_{k=1}^{K} \eta_{mk} \gamma_{mk} \leq 1, m = 1, \ldots, M, \tag{7}$$

where

$$\gamma_{mk} \triangleq \mathbb{E}\left\{|\hat{g}_{mk}|^2\right\} = \frac{\tau_p \rho_u \left(\beta_{mk}\right)^2}{\tau_p \rho_u \sum_{k'=1}^{K} \beta_{mk'} \left|\varphi_k^H \varphi_{k'}\right|^2 + 1}.$$

The corresponding downlink signal received at the k-th UE is given by

$$\begin{aligned} r_k = &\sqrt{\rho_d} \sum_{m=1}^{M} \eta_{mk}^{1/2} g_{mk} \hat{g}^*_{mk} q_k \\ &+ \sqrt{\rho_d} \sum_{m=1}^{M} \sum_{k' \neq k}^{K} \eta_{mk'}^{1/2} g_{mk} \hat{g}^*_{mk'} q_{k'} + w_k, \end{aligned} \tag{8}$$

where $w_k \sim \mathcal{CN}(0,1)$ denotes the noise at the k-th UE.

III. Downlink Power Control : Problem Definition

In this section, we formulate the downlink power control problem into three proportional fairness problems for the following evaluation. Exploiting the channel hardening in CF systems [1], the downlink signal-to-interference-plus-noise ratio (SINR) of the k-th UE is (9) shown at the bottom of this page. According to [1], the achievable rate of the k-th UE [10] is given by

$$\mathrm{R}_k = \log_2(1 + \mathrm{SINR}_k). \tag{10}$$

Then, the downlink power control optimization problem can be formulated as follows:

$$\begin{aligned} &\underset{\{\eta_{mk}\}}{\text{maximize}} \ U\{\mathrm{R}_1, \ldots, \mathrm{R}_K\} \\ &\text{s.t.} \sum_{k=1}^{K} \eta_{mk} \gamma_{mk} \leq 1, m = 1, \ldots, M, \\ &\qquad \eta_{mk} \geq 0, m = 1, \ldots, M, k = 1, \ldots, K, \end{aligned} \tag{11}$$

where the objective funtion is given by

$$\begin{aligned} &U\{\mathrm{R}_1, \ldots, \mathrm{R}_K\} \\ &= \begin{cases} \sum_{k=1}^{K} \mathrm{R}_k, \text{ for Max sum rate,} \\ \min\{\mathrm{R}_1, \ldots, \mathrm{R}_K\}, \text{ for Max-min fairness,} \\ \prod_{k=1}^{K} \mathrm{SINR}_k, \text{ for Max product SINR.} \end{cases} \end{aligned} \tag{12}$$

The max-min power control objective function in (12) is quasi-concave that can be addressed by a bisection method that involves solving a sequence of convex programs in [1] to obtain the globally optimal solution. Moreover, the max-sum and max-product problems are non-convex due to the non-convex objective functions [12], which complicate the solution development. The feasible supervised learning-based methods need to obtain prior data that is difficult to obtain in reality. Therefore, we will propose a DRL-based on the DRL model that overcomes the above shortcomings in the following section.

IV. DDPG-Based Power Control Algorithm

In RL, a goal-oriented agent interacts with an environment and obtains feedback to develop the optimal policy. At each step, the agent takes an action based on the policy and receives the reward while moving to the next state. In each episode, the APs take numerous actions. In general, DRL is studied by means of Markov decision

$$\mathrm{SINR}_k = \frac{\rho_{\mathrm{d}} \left(\sum_{m=1}^{M} \eta_{mk}^{1/2} \gamma_{mk}\right)^2}{\rho_{\mathrm{d}} \sum_{k' \neq k}^{K} \left(\sum_{m=1}^{M} \eta_{mk'}^{1/2} \gamma_{mk'} \frac{\beta_{mk}}{\beta_{mk'}}\right)^2 \left|\boldsymbol{\varphi}_{k'}^H \boldsymbol{\varphi}_k\right|^2 + \rho_{\mathrm{d}} \sum_{k'=1}^{K} \sum_{m=1}^{M} \eta_{mk'} \gamma_{mk'} \beta_{mk} + 1}. \tag{9}$$

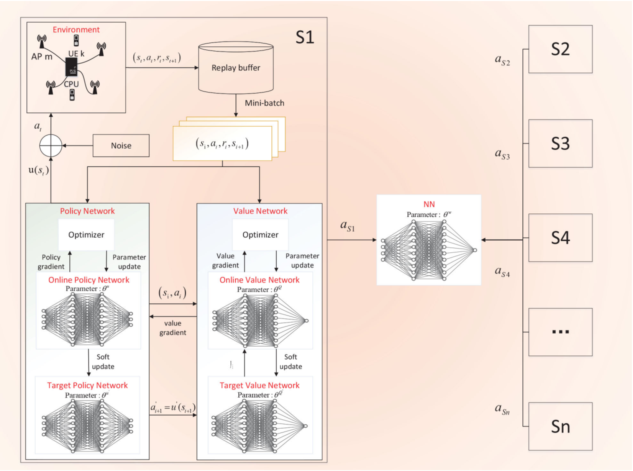

Fig. 1. The proposed DDPG-based power control design.

processes (MDPs) characterized by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \zeta)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state space and the action space of the agent, respectively. Variables $\mathcal{P}$ and $\mathcal{R}$ are the transition probability matrix and the expected reward, respectively. ζ denotes the reward's discount factor.

DRL algorithms can be divided into three implementation forms: value-based approaches, policy-based approaches, and actor-critic (AC) approaches.

(1) For value-based approaches, a Q table is constructed and maintained to find an optimal strategy to maximize the expected return of all subsequent actions starting from the current state, where each item $Q(s,a)$ in the Q table represents the expected return of taking action a at state s. Such algorithms have a high data utilization rate and have been proven to converge stably [13]. Since each (s,a) tuple needs to correspond to a Q value, it can only solve the problem that states and actions are countable and the number is small. In general, the value-based DQN algorithm maps the state-action to Q-values by the deep neutron networks, which makes it effective at continuous state space $\mathcal{S}$. However, since the optimal state-action value $Q(s,a)$ is unknown, DQN is not applicable to the continuous action space $\mathcal{A}$.

(2) For policy-based approaches, a policy function $\mu(s)$ is constructed, where a state s maps the corresponding action a or the probability distribution of action a directly. Since there is no need to estimate the maximum Q value, it can be used in continuous action space $\mathcal{A}$. However, the basic policy gradient algorithm requires a complete sequence of states and iteratively updates the policy function separately, which makes it difficult to converge.

(3) For AC approaches, the actor networks exploit the policy function, which is responsible for generating actions and interacting with the environment. Besides, critic networks exploit the value function, which is responsible for evaluating the performance of the actor and guiding the next following action of the actor. Since the critic networks fit the action-value function in the continuous space and the action networks do not require finding the optimal action-value function, the AC is suitable for the continuous optimization space.

Due to the continuous optimization space in the downlink power control problem, we design the DDPG algorithm under the category of AC. In the following, we will propose a DDPG-based approach to address the downlink power control problem in (11) and summarize them as shown in Fig. 1.

We choose the DDPG algorithm with a policy network $\mu(s \mid \theta^u)$ and a value network $Q(s, a \mid \theta^Q)$, where $Q(s,a)^u$ is the action-value function following a strategy that obeys the following Bellman equation [13]:

$$Q(s,a)^u = \mathbb{E}_{(s,a,r,s')\in\mathcal{B}}\left[r(s,a) + \zeta \max_{a'} Q^u\left(s', a'\right)\right], \tag{13}$$

where $\mathcal{B}$ is a set of the experience namely replay buffer and $Q^u(s', a')$ is the value at the next state and action. At each step, the transition (s_t, a_t, r_t, s_{t+1}) will be stored in the replay buffer while sampling mini-batch of transitions from $\mathcal{B}$ randomly to update the networks, which greatly improves the efficiency of data utilization. Note that in the considered case of limited computational resources, the DDPG algorithm with high data utilization is more suitable for CF systems in this paper than the algorithms that use additional computational resources for parallel computation acceleration, e.g., A3C. Due to the fact that $r(s,a) + \zeta \max_{a'} Q^u(s', a')$ would change as the parameters are updated, the parameters is hard to converge in general. As such, DDPG adopts a critic target network with parameters $\theta^{Q'} \in \mathbb{R}$ and a policy target network with parameters $\theta^{\mu'} \in \mathbb{R}$ to address this issue. Using the process as follows, the two networks update their parameters by $Poylak\ averaging$ [13]:

$$\begin{aligned} \theta^{Q'} &\leftarrow \sigma\theta^{Q} + (1-\sigma)\theta^{Q'}, \\ \theta^{\mu'} &\leftarrow \sigma\theta^{\mu} + (1-\sigma)\theta^{\mu'}, \end{aligned} \tag{14}$$

where $\sigma \in (0,1)$ is the soft coefficient that determines the change speed of the target network parameters.

The critic network needs to evaluate the outputted actions by the action network, thus the output of the network needs to be as close as possible to the true value of the action. In order to ensure the stability of the value function, the critic network regards the output value of the criticized target network as the true value and tries to reduce the distance between the output value of the critic network and the output value of the critic target network. By minimizing the following loss function, the critic network updates its parameters as

$$L = \frac{1}{N}\sum_i \left(r_i + \zeta Q'\left(s_{i+1}, \mu'\left(s_{i+1} \mid \theta^{\mu'}\right) \mid \theta^{Q'}\right) - Q\left(s_i, a_i\right)\right)^2,$$

where N is the batch size. The policy network needs to adjust the output policy to output the optimal action. It updates the parameters and finds the optimal policy through the output value of the policy network and the output action. To train the policy networks, DDPG maximizes the expected return as

$$J\left(\theta^{\mu}\right) = \mathbb{E}_u\left\{r_1 + \zeta r_2 + \zeta^2 r_3 + \cdots + \zeta^{n-1} r_n\right\} \tag{15}$$

and follows the gradient of (16) to update the weights θ^{μ}:

$$\nabla_{\theta_\mu} J(\theta) \approx \nabla_a Q(s,a) \nabla_{\theta^\mu}\mu(s). \tag{16}$$

where the term $\nabla_a Q(s,a)$ is obtained from the backpropagation of the critic network $Q(s, a \mid \theta^Q)$ w.r.t. the action input $\mu(s \mid \theta^\mu)$, which follows the deterministic policy gradient theorem [14]. The DDPG learning process is summarized in Algorithm 1.

Since it has been shown that DRL algorithms are prone to overfitting as it interacts with the same environment for a long time to obtain the data [9], we introduce a NN with parameters θ^ω after the output layer

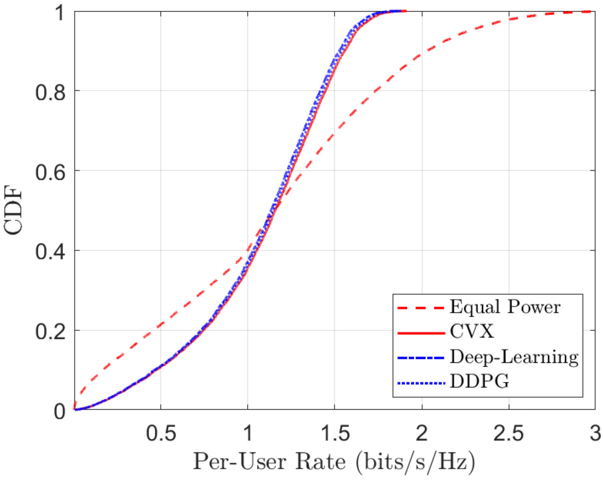

Fig. 2. CDF of the per-user rate for the max-min power control problems.

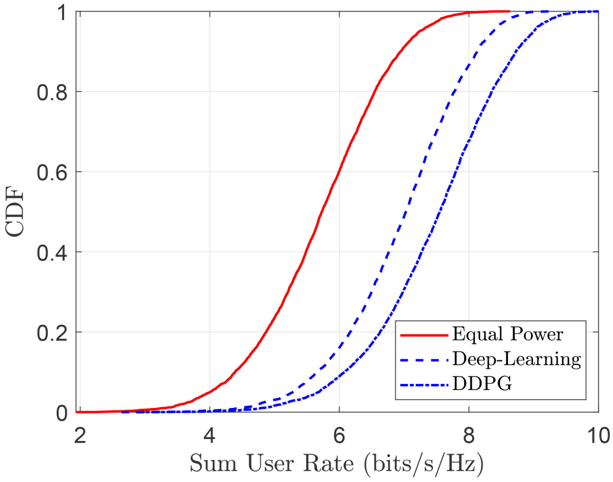

Fig. 3. CDF of the sum user rate.

of the DDPG algorithm, which updates the parameters by minimizing the following loss:

$$\underset{\theta^\omega}{\text{minimize}} \frac{1}{N_s}\sum_{n=1}^{N_s} \ell\left(\hat{\boldsymbol{a}}_{(}n), \boldsymbol{a}^\star(n)\right), \tag{17}$$

where N_s is the number of samples input into the NN, $\ell(\cdot,\cdot)$ can be any measure method for distance, and $\hat{\boldsymbol{a}}, \boldsymbol{a}^\star$ are the output from DDPG algorithm and from the NN in each environment, respectively. Note that the different environments are the CF system with different UE positions. Since the reward would change significantly when the same DDPG network interacts with different environments, it is generally difficult for the network to converge. As a remedy, we run different DDPG networks in parallel in different environments. As NN only needs to reduce the distance of its outputs and the output of all the converged actor networks, it alleviates the convergence issues caused by the rapid changes in the reward and avoids the overfitting issues caused by a single DDPG network interacting with a stationary environment. In Table I, we summarize the specific design parameters of the DRL model in CF massive MIMO. κ is a hyperparameter that affects the exploration ability of the algorithm. Note that positive rewards encourage the agent to select the corresponding action that has been selected. Thus, we need to subtract the hyperparameter κ as the baseline from the reward to prevent the agent from being satisfied with doing so. In order to adapt to different environments, we let the algorithm set the value of κ automatically in two steps. Specifically, we first set κ to zero and train 1000 episodes for the model. Then, the obtained result is set as the value for κ in the next initialization. Such κ is suitable for environments of different characteristics. The typical value of σ is 0.001 to 0.01, thus we choose σ to be 0.005 to ensure that the network updates as fast as possible while ensuring stability.

Algorithm 1 DDPG-Based Power Control Design

```
Randomly **Initialize** $n_e$ environments.
**for** $n_e = 1$ to Max-number-environments **do**
    Randomly **initialize** the value network $Q(s,a \mid \theta^Q)$, the policy network $\mu(s \mid \theta^\mu)$ and the NN with weights $\theta^Q, \theta^\mu$, and $\theta^\omega$.
    **Initialize** the target value network $Q'$ the target policy network $\mu'$ with weights $\theta^{Q'} = \theta^Q$ and $\theta^{\mu'} = \theta^\mu$.
    Initialize replay buffer $\mathcal{B}$.
    **for** episode $= 1$ to Max-number-episodes **do**
        Randomly **Initialize** process $\mathcal{N}$ for action exploration.
        Set the initial state $s_1$.
        **for** $t = 1$ to Max-episode-steps **do**
            Take action $a_t = \mu(s_t) + \mathcal{N}_t$ and record the next state $s_{t+1}$ and the reward $r_t$.
            Store the transition $(s_t, a_t, r_t, s_{t+1})$ in $\mathcal{B}$.
            Sample mini-batch of $N$ transitions from $\mathcal{B}$ randomly.
            Minimizing the loss to update the critic network:

            $$\text{Loss} = \frac{1}{N}\sum_i \left(y_i - Q\left(s_i, a_i \mid \theta^Q\right)\right)^2, \tag{18}$$
            $$y_i = r_i + \zeta Q'\left(s_{i+1}, \mu'\left(s_{i+1} \mid \theta^{\mu'}\right) \mid \theta^{Q'}\right). \tag{19}$$
            Using sampled stochastic policy gradient ascent to update the actor policy:
            $$\nabla_{\theta^\mu} \approx \frac{1}{N}\sum_i \nabla_a Q\left(s, a \mid \theta^Q\right)\Big|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu}\mu\left(s \mid \theta^\mu\right)\Big|_{s=s_i}$$
            Update the target value network and the target policy network as

            $$\theta^{Q'} \leftarrow \sigma\theta^Q + (1-\sigma)\theta^{Q'}, \tag{20}$$
            $$\theta^{\mu'} \leftarrow \sigma\theta^\mu + (1-\sigma)\theta^{\mu'}, \tag{21}$$
        **end for**
    **end for**
**end for**
The NN receive the output from the actor networks in each environment and updates the $\theta^\omega$ by minimizing the cost using (17)
```

V. Numerical Results

A. Large-Scale Fading Model and Simulation Setup

In numerical analysis, we consider a network with $M = 15$ and $K = 5$, where UEs and APs are uniformly distributed at random in a square of size 1×1 km^2 at a carrier frequency 1.9 GHz. We follow

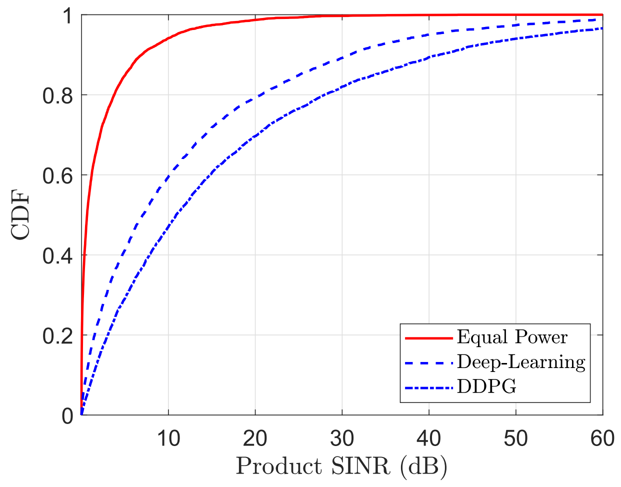

Fig. 4. CDF of the product user SINR.

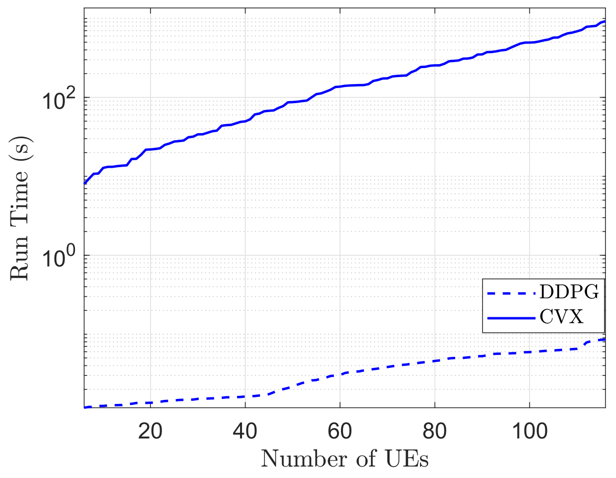

Fig. 5. Run time comparison.

the three-slope model in [1], to model the large-scale fading coefficient β_{mk}. The important parameters are summarized in Table II.

To verify the performance of the proposed DDPG algorithm, we will compare it with the solution obtained by the convex optimization solver and the conventional deep learning algorithm. For the max-sum and max-product power control problems, since the supervision data is unable to be obtained due to the non-convex objective function, we compare the proposed DDPG algorithm with the conventional deep learning algorithm that has been shown to perform well in the existing literature [15]. Such an unsupervised learning algorithm maximizes the objective function by some gradient ascent algorithm, i.e., reverse gradient descent. We trained 2000 episodes for the proposed model, each with 1000 steps. The networks have fully-connected layers with size 400 × 300. The $Relu$ activation function is used for each layer. The NNs were trained based on a dataset of 90000 samples of independent realizations of the UEs' positions and power allocations, obtained by the outputs from the DDPG networks in different environments by solving (12). Particularly, 90% percent of the samples were used for training and 10% for validation. Other 10000 samples formed the test dataset, which is independent of the training dataset.

TABLE I
Parameters in DRL Model

Parameters	Designs
State $s:\{s_1,\ldots,s_k\}$	SINR: $\{\mathrm{SINR}_1,\ldots,\mathrm{SINR}_k\}$
Reward r for max-min algorithm	$\mathrm{R}_k - \kappa$
Reward r for max-sum algorithm	$\sum_{k=1}^{K}\mathrm{R}_k - \kappa$
Reward r for max-prod algorithm	$\prod_{k=1}^{K}\mathrm{SINR}_k - \kappa$
Action a	η_{mk} for all m and k

TABLE II
System Parameters

Parameters	Values
Centre carrier frequency	1.9 GHz
M, K	15, 5
UL pilot length	20
Bandwidth	20 MHz
Noise figure	9 dB
AP/UE antenna height	15, 1.65 m
$\rho_{\mathrm{d}}, \rho_{\mathrm{p}}$	200, 100mW
Discount factor ζ	0.99
Learning rate ν	10^{-3}
Poylak averaging parameter σ	5×10^{-3}
Size of experience replay buffer $\mathcal{B}$	10^6

B. Results and Discussions

As shown in Fig. 2, the DDPG algorithm can approximate the optimal solution returned by the convex optimization solver, while the performance loss is negligible. Notice that traditional deep learning algorithms can also approximate the optimal solution well, since performing the gradient descent search on the quasi-concave objective function would not fall into numerous local optimal solutions.

In Fig. 3 and Fig. 4, the DDPG algorithm is used to address both the max-sum and max-product problems. It is easy to observe that the DDPG algorithm is better than the traditional deep learning algorithm in both cases and is far better than equal power control. In particular, the DDPG algorithm brings 10.3% and 30.2% improvement at 95%-likely compared with the deep learning algorithm for the max-sum and max-product power control, respectively. In fact, the update rule (16) guarantees that the performance fluctuation at the initial training stage increases the possibility of exploring a better performance solution that reduces the chances of having the poorer performance solution following the deterministic policy gradient theorem [14]. In contrast, the traditional deep learning algorithm is based on gradient descent, which can not avoid getting trapped in some local optimal solutions.

C. Complexity Analysis

Note that the convex optimization solver (i.e., CVX) can only be used in the max-min power control problem. In this section, we compare the running time on the ordinary computing platform in solving the max-min power control problem as a proxy for the complexity in Fig. 5. It is easy to observe that the DDPG algorithm is around three orders of magnitude faster than that of the convex optimization solver. Note that since retraining is only done occasionally when the environment or the system changes significantly, training time is not an issue in the considered system.

VI. Conclusion

In this correspondence, we studied the optimization problem of downlink power control in CF massive MIMO networks. We proposed a DDPG-based approach to solve the downlink power control problem. Three power control strategies were considered, namely max-min, max-sum, and max-product. The simulation results showed that the DDPG algorithm can approximate the global optimal solution for the case of the max-min power control while being three orders of magnitude faster. Besides, the proposed scheme is better than the conventional deep learning algorithm in terms of increasing user data rate in the three studied application scenarios. In particular, the DDPG algorithm can achieve a more effective solution more easily than traditional deep learning when the objective function is non-convex.

References

[1] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO versus small cells," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1834–1850, Mar. 2017.

[2] X. Chen, D. W. K. Ng, W. Yu, E. G. Larsson, N. Al-Dhahir, and R. Schober, "Massive access for 5-G and beyond," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 3, pp. 615–637, Mar. 2021.

[3] J. Zhang, E. Björnson, M. Matthaiou, D. W. K. Ng, H. Yang, and D. J. Love, "Prospective multiple antenna technologies for beyond 5-G," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1637–1660, Aug. 2020.

[4] F. Meng, P. Chen, L. Wu, and J. Cheng, "Power allocation in multi-user cellular networks: Deep reinforcement learning approaches," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6255–6267, Oct. 2020.

[5] Y. Zhao, I. G. Niemegeers, and S. H. De Groot, "Power allocation in cell-free massive MIMO: A deep learning method," *IEEE Access*, vol. 8, pp. 87185–87200, May 2020.

[6] L. Sanguinetti, A. Zappone, and M. Debbah, "Deep learning power allocation in massive MIMO," in *Proc. IEEE 52nd Asilomar Conf. Signals, Syst., Comput.*, 2018, pp. 1257–1261.

[7] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger, "Deep reinforcement learning that matters," in *Proc. AAAI Conf. Artif. Intell.*, 2018, pp. 1–2, vol. 32, no. 1.

[8] J. Jang and H. J. Yang, "Deep reinforcement learning-based resource allocation and power control in small cells with limited information exchange," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13 768–13 783, Nov. 2020.

[9] 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.

[10] Y. Al-Eryani, M. Akrout, and E. Hossain, "Multiple access in cell-free networks: Outage performance, dynamic clustering, and deep reinforcement learning-based design," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 4, pp. 1028–1042, Apr. 2021.

[11] E. Björnson, J. Hoydis, and L. Sanguinetti, "Massive MIMO networks: Spectral, energy, and hardware efficiency," in *Proc. Found. Trends Signal Process.*, vol. 11, no. 3-4, 2017, pp. 154–655.

[12] S. Buzzi, C. D'Andrea, A. Zappone, and C. D'Elia, "User-centric 5-G cellular networks: Resource allocation and comparison with the cell-free massive MIMO approach," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 1250–1264, Feb. 2020.

[13] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, Nov. 2018.

[14] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 387–395.

[15] R. Nikbakht, A. Jonsson, and A. Lozano, "Unsupervised learning for cellular power control," *IEEE Commun. Lett.*, vol. 25, no. 3, pp. 682–686, Mar. 2021.