Power Allocation in Multi-user Cellular Networks With Deep Q Learning Approach

Fan Meng
Southeast University
Information Science and Engineering
Nanjing 210096, China
mengxiaomaomao@outlook.com

Peng Chen
Southeast University
State Key Laboratory of Millimeter Waves
Nanjing 210096, China
chenpengseu@seu.edu.cn

Lenan Wu
Southeast University
Information Science and Engineering
Nanjing 210096, China
wuln@seu.edu.cn

***Abstract*—The model-driven power allocation (PA) algorithms in the wireless cellular networks with interfering multiple-access channel (IMAC) have been investigated for decades. Nowadays, the data-driven model-free machine learning-based approaches are rapidly developing in this field, and among them the deep reinforcement learning (DRL) is proved to be of great potential. Different from supervised learning, the DRL takes advantages of exploration and exploitation to maximize the objective function under certain constraints. In our paper, we propose a two-step training framework. First, with the off-line learning in simulated environment, a deep Q network (DQN) is trained with deep Q learning (DQL) algorithm, which is well-designed to be in consistent with this PA issue. Second, the DQN will be further fine-tuned with real data in on-line training procedure. The simulation results show that the proposed DQN achieves the highest averaged sum-rate, comparing to the ones with present standard DQL training. With different user densities, our DQN outperforms benchmark algorithms and thus a good generalization ability is verified.**

***Index Terms*—Deep reinforcement learning, deep Q learning, interfering multiple-access channel, power allocation.**

I. Introduction

Data transmitting in wireless communication networks has experienced explosive growth in recent decades and will keep rising in the future. The user density is greatly increasing, resulting in critical demand for more capacity and spectral efficiency. Therefore, both intra-cell and inter-cell interference managements are significant to improve the overall capacity of a cellular network system. The problem of maximizing a generic sum-rate is studied in this paper, and it is non-convex, NP-hard and cannot be solved efficiently.

Various model-driven algorithms have been proposed in the present papers for power allocation (PA) problems, such as fractional programming (FP) [1], weighted MMSE (WMMSE) [2] and some others [3], [4]. Excellent performance can be observed through theoretical analysis and numerical simulations, but serious obstacles are faced in practical deployments [5]. First, these techniques highly rely on tractable mathematical models, which are imperfect in real communication scenarios with the specific user distribution, geographical environment, etc. Second, the computational complexities of these algorithms are high.

In recent years, the machine learning (ML)-based approaches have been rapidly developed in wireless communications [6]. These algorithms are usually model-free, and are compliant with optimizations in practical communication scenarios. Additionally, with developments of graphic processing unit (GPU) or specialized chips, the executions can be both fast and energy-efficient, which brings in solid foundations for massive applications.

Two main branches of ML, supervised learning and reinforcement learning (RL) [7], are briefly introduced here. With supervised learning, a deep neural network (DNN) is trained to approximate some given optimal (or suboptimal) objective algorithms, and it has been realized in some applications [8]–[10]. However, the target algorithm is usually unavailable and the performance of DNN can be bounded by the supervisor. Therefore, the RL has received widespread attention, due to its nature of interacting with an unknown environment by exploration and exploitation. The Q learning method is the most well-studied RL algorithm, and it is exploited to cope with PA in [11]–[13], and some others [14]. The DNN trained with Q learning is called deep Q network (DQN), and it is proposed to address the curse of dimensionality and the lack of generalization. In [15], distributed dynamic downlink power allocation in single-user cellular networks with deep reinforcement learning (DRL) approach is studied.

In this paper, we extend the same system-level optimization problem [15] but with multiple users, and an interfering multiple-access channel (IMAC) scenario is considered. The design of our DQN model is discussed and introduced. Simulation results show that the DQN outperforms the present standard DQNs and the benchmark algorithms. The contributions of this work are summarized as follows:

- A model-free two-step training framework is proposed. The DQN is first off-line trained with DRL algorithm in simulated scenarios. Second, the learned DQN can be further dynamically optimized in real communication scenarios, with the aid of transfer learning.
- The PA problem using deep Q learning (DQL) is discussed, then a DQN enabled approach is proposed to be trained with current sum-rate as reward function, including no future rewards. The input features are well-designed to help the DQN get closer to the optimal solution.
- After centralized training, the proposed DQN is tested

978-1-5386-8088-9/19/$31.00 ©2019 IEEE

by distributed execution. The averaged rate-sum of DQN outperforms the model-driven algorithms, and also shows good generalization ability in a series of benchmark simulation tests.

The remainder of this paper is organized as follows. Section II outlines the PA problem in the wireless cellular network with IMAC. In Section III, our proposed DQN is introduced in detail. Then, this DQN is tested in distinct scenarios, along with benchmark algorithms, and the simulation results are analyzed in Section IV. Conclusions and discussion are given in Section V.

II. System Model

We consider power allocation in the cellular network with IMAC. In a communication system with N cells, at the center of each cell a base station (BS) simultaneously serves K users with a sharing frequency band. A simple network example is shown in Fig. 1. At time slot t, the independent channel gain between the n-th BS and the user k in cell j is denoted by $g^t_{n,j,k}$, and can be expressed as

$$g_{n,j,k} = |h^t_{n,j,k}|^2 \beta_{n,j,k}, \tag{1}$$

where $h^t_{n,j,k}$ is the small scale complex flat fading element; $\beta_{n,j,k}$ is the large scale fading component, taking account of both geometric attenuation and shadow fading. According to the Jakes' model [16], the small-scale flat fading is modeled as a first-order complex Gauss-Markov process

$$h^t_{n,j,k} = \rho h^{t-1}_{n,j,k} + n^t_{n,j,k} \tag{2}$$

where $h^1_{n,j,k} \sim \mathcal{CN}(0,1)$ and $n^t_{n,j,k} \sim \mathcal{CN}(0, 1-\rho^2)$. The correlation ρ is determined by $\rho = J_0(2\pi f_d T_s)$, where $J_0(\cdot)$ is the first kind zero-order Bessel function, f_d is the maximum Doppler frequency, and T_s is the time interval between adjacent instants. Therefore, the signal-to-interference-plus-noise ratio (SINR) of this link can be described by

$$\gamma^t_{n,k} = \frac{g^t_{n,n,k} p^t_{n,k}}{\sum_{k' \neq k} g^t_{n,n,k} p^t_{n,k'} + \sum_{n' \in D_n} g^t_{n',n,k} \sum_j p^t_{n',j} + \sigma^2}, \tag{3}$$

where D_n is the set of interference cells around the n-th cell, p is the emitting power of BS, and σ^2 denotes the additional noise power. With normalized bandwidth, the downlink rate of this link is given as

$$C^t_{n,k} = \log_2\left(1 + \gamma^t_{n,k}\right), \tag{4}$$

The optimization target is to maximize this generic sum-rate objective function under maximum power constraint, and it is formulated as

$$\begin{aligned} \max_{\mathbf{p}^t} \quad & \sum_n \sum_k C^t_{n,k} \\ \text{s.t.} \quad & 0 \le p^t_{n,k} \le P_{\max}, \ \forall n, k, \end{aligned} \tag{5}$$

where $\mathbf{p}^t = \{p^t_{n,k} | \forall n, k\}$, and $P_{\max}$ denotes the maximum emitting power. We also define sum-rate $C^t = \sum_n \sum_k C^t_{n,k}$, $\mathbf{C}^t = \{C^t_{n,k} | \forall n, k\}$, and channel state information (CSI) $\mathbf{g}^t = \{g^t_{n,j,k} | \forall n, j, k\}$. This problem is non-convex and NP-hard, so we propose a data-driven learning algorithm based on the DQN model in the following section.

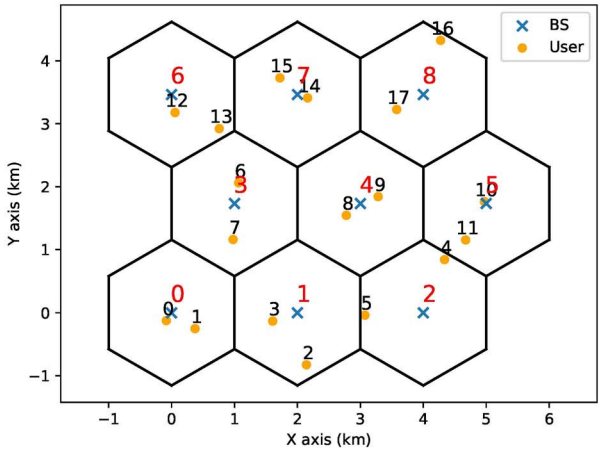

Fig. 1. An illustrative example of a multi-user cellular network with 9 cells. In each cell, a BS serves 2 users simultaneously.

III. Deep Q Network

A. Background

Q learning is one of the most popular RL algorithms aiming to deal with the Markov decision process (MDP) problems [17]. At time instant t, by observing state $s^t \in S$, the agent takes action $a^t \in A$ and interacts with the environment, then it gets the reward r^t and the next state s^{t+1} is obtained. The notations A and S are the action set and the state set, respectively. Since S can be continuous, the DQN is proposed to combine Q learning with a flexible DNN to settle infinite state space. The cumulative discounted reward function is given as

$$R^t = \sum_{\tau=0}^{\infty} \gamma^\tau r^{t+\tau+1}, \tag{6}$$

where $\gamma \in [0,1)$ is a discount factor that trades off the importance of immediate and future rewards, and r denotes the reward. Under a certain policy π, the Q function of the agent with an action a in state s is given as

$$Q_\pi(s, a; \boldsymbol{\theta}) = \mathbb{E}_\pi \left[R^t | s^t = s, a^t = a \right], \tag{7}$$

where $\boldsymbol{\theta}$ denotes the DQN parameters, and $\mathbb{E}[\cdot]$ is the expectation operator. Q learning concerns with how agents ought to interact with an unknown environment so as to maximize the Q function. The maximization of (7) is equivalent to the Bellman optimality equation [18], and it is describe as

$$y^t = r^t + \gamma \max_{a'} Q(s^{t+1}, a'; \boldsymbol{\theta}^t), \tag{8}$$

where y^t is the optimal Q value. The DQN is trained to approximate the Q function, and the standard Q learning update of the parameters $\boldsymbol{\theta}$ is described as

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t + \eta \left(y^t - Q(s^t, a^t; \boldsymbol{\theta}^t) \right) \nabla Q(s^t, a^t; \boldsymbol{\theta}^t), \tag{9}$$

where η is the learning rate. This update resembles stochastic gradient descent, gradually updating the current value $Q(s^t, a^t; \boldsymbol{\theta}^t)$ towards the target y^t. The experience data of the agent is loaded as $\left(s^t, a^t, r^t, s^{t+1}\right)$. The DQN is trained with recorded batch data randomly sampled from the experience replay memory, which is a first-in first-out queue.

B. Discussion on DRL

In many applications such as playing video games [17], where current strategy has long-term impact on cumulative reward, the DQN achieves remarkable results and beats humans. However, the discount factor is suggested to be zero in this static PA problem. The DQL aims to maximize the Q function. We let $\gamma = 0$, and maximization of (7) is given as

$$\max Q = \max_{a \in A} \mathbb{E}_\pi \left[r^t | s^t = s, a^t = a \right]. \tag{10}$$

In this PA problem, clearly that $s = \mathbf{g}^t$, $a = \mathbf{p}^t$. Then we let $r^t = C^t$ and get that

$$\max Q = \max_{\mathbf{0} \preceq \mathbf{p}^t \preceq \mathbf{p}_{\max}} \mathbb{E}_\pi \left[C^t | \mathbf{g}^t, \mathbf{p}^t \right]. \tag{11}$$

In the execution period the policy is deterministic, and thus (11) can be written as

$$\max Q = \max_{\mathbf{0} \preceq \mathbf{p}^t \preceq \mathbf{p}_{\max}} C^t \left(\mathbf{g}^t, \mathbf{p}^t \right), \tag{12}$$

which is a equivalent form of (5). In this inference process we assume that $\gamma = 0$ and $r^t = C^t$, indicating that the optimal solution to (5) is identical to that of (7), under these two conditions.

As shown in Fig. 2, it is well-known that the optimal solution $\mathbf{p}^{t*}$ of (5) is only determined by current CSI $\mathbf{g}^t$, and the sum-rate $\mathbf{C}^t$ is calculated with $(\mathbf{g}^t, \mathbf{p}^t)$. Theoretically the optimal power $\mathbf{p}^{t*}$ can be obtained using a DQN with input being just $\mathbf{g}^t$. In fact, the performance of this designed DQN is poor, since it is non-convex and the optimal point is hard to find. Therefore, we propose to utilize two more auxiliary features: $\mathbf{C}^{t-1}$ and $\mathbf{p}^{t-1}$. Since the channel can be modeled as a first-order Markov process, the solution of last time period can help the DQN get closer to the optimum, and (12) can be rewritten as

$$\max Q = \max_{\mathbf{0} \preceq \mathbf{p}^t \preceq \mathbf{p}_{\max}} C^t \left(\mathbf{g}^t, \mathbf{p}^t, \mathbf{C}^{t-1}, \mathbf{p}^{t-1} \right). \tag{13}$$

Once $\gamma = 0$ and $r^t = C^t$, (8) is simplified to be $y^t = C^t$, and the replay memory is also reduced to be (s^t, a^t, r^t). The DQN works as an estimator to predict the current sum-rate of corresponding power levels with a certain CSI. These discussions provide guidance for the following DQN design.

C. DQN Design in Cellular Network

In our proposed model-free two-step training framework, the DQN is first off-line pre-trained with DRL algorithm in simulated wireless communication system. This procedure is to reduce the on-line training stress, due to the large data requirement of data-driven algorithm by nature. Second, with the aid of transfer learning, the learned DQN can be further dynamically fine-tuned in real scenarios. Since the practical wireless communication system is dynamic and influenced by unknown issues, the data-driven algorithm is regarded as a promising technique. We just discuss the two-step framework here, and the first training step is mainly focused in the following manuscript.

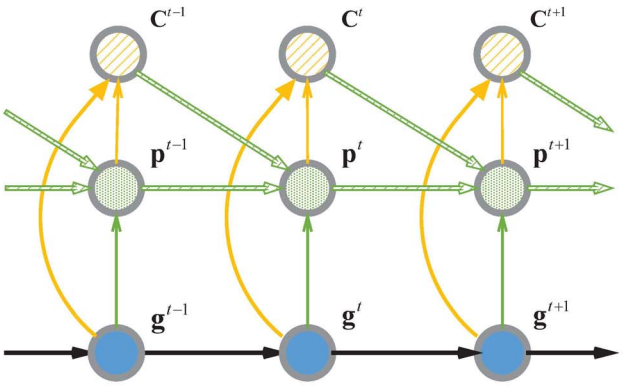

Fig. 2. The solution of DQN is determined by CSI $\mathbf{g}^t$, along with downlink rate $\mathbf{C}^{t-1}$ and transmitting power $\mathbf{p}^{t-1}$.

In a certain cellular network, each BS-user link is regarded as an agent and thus a multi-agent system is studied. However, multi-agent training is difficult since it needs much more learning data, training time and DNN parameters. Therefore, centralized training is considered, and only one agent is trained by using all agents' experience replay memory. Then, this agent's learned policy is shared in the distributed execution period. For our designed DQN, components of the replay memory are introduced as follows.

1) State: The state design for a certain agent (n, k) is important, since the full environment information is redundant and irrelevant elements must be removed. The agent is assumed to have corresponding perfect instant CSI information in (3), and we define logarithmic normalized interferer set $\mathbf{\Gamma}_{n,k}^t$ as

$$\mathbf{\Gamma}_{n,k}^t := \frac{1}{g_{n,k,k}^t} \boldsymbol{g}_{n,k}^t \otimes \mathbf{1}_K, \tag{14}$$

where $\boldsymbol{g}_{n,k}^t = \{ g_{n',n,k}^t | n' \in \{n, D_n\}, \forall k \}$; $\otimes$ is the Kronecker product, and $\mathbf{1}_K$ is a vector filled with K ones. The channel amplitude of interferers are normalized by that of the needed link, and the logarithmic representation is preferred since the amplitudes of channel often vary by orders of magnitude. The cardinality of $\mathbf{\Gamma}_{n,k}^t$ is $(|D_n|+1)K$. To further decrease the input dimension and reduce the computational complexities, the elements in $\mathbf{\Gamma}_{n,k}^t$ are sorted in decrease turn and only the first I_c elements remain. As we discussed in III-B, these remained components' and this link's corresponding downlink rate $\mathbf{C}_{n,k}^{t-1}$ and transmitting power $\mathbf{p}_{n,k}^{t-1}$ at last time slot, are the additional two parts of the input to our DQN. Therefore, the state is composed of three features: $s_{n,k}^t = \{ \mathbf{\Gamma}_{n,k}^t, \mathbf{C}_{n,k}^{t-1}, \mathbf{p}_{n,k}^{t-1} \}$. The

cardinality of state, i.e., the input dimension for DQN is $|S| = 3I_c$.

2) Action: In (5) the downlink power is a continuous variable and is only constrained by maximum power constraint, but the action space of DQN must be finite. The possible positive emitting power is quantized exponentially in $|A|-1$ levels, along with a zero power level which represents no transmitting signals. Therefore, the allowed power set is given as

$$A = \left\{0, P_{\min}, P_{\min}\left(\frac{P_{\max}}{P_{\min}}\right)^{\frac{1}{|A|-2}}, \cdots, P_{\max}\right\}, \qquad (15)$$

where $P_{\min}$ is the positive minimum emitting power.

3) Reward: In some manuscripts the reward function is elaborately designed to improve the agent's transmitting rate and also mitigate the interference influence. However, most of these reward functions are suboptimal approaches to the target function of (5). In our paper, the C^t is directly used as the reward function, and it is shared by all agents. In the training simulations with small or medium scale cellular network, this simple method proves to be feasible.

IV. Simulation Results

A. Simulation Configuration

A cellular network with $N = 25$ cells is simulated. At center of each cell, a BS is deployed to synchronously serve $K = 4$ users which are located uniformly and randomly within the cell range $r \in [R_{\min}, R_{\min}]$, where $R_{\min} = 0.01$ km and $R_{\min} = 1$ km are the inner space and half cell-to-cell distance, respectively. The small-scale fading is simulated to be Rayleigh distributed, and the Jakes model is adopted with Doppler frequency $f_d = 10$ Hz and time period $T = 20$ ms. According to the LTE standard, the large-scale fading is modeled as $\beta = -120.9 - 37.6\log_{10}(d) + 10\log_{10}(z)$ dB, where z is a log-normal random variable with standard deviation being 8 dB, and d is the transmitter-to-receiver distance (km). The AWGN power σ^2 is -114 dBm, and the emitting power constraints $P_{\min}$ and $P_{\max}$ are 5 and 38 dBm, respectively.

A four-layer feed-forward neural network (FNN) is chosen as DQN, and the neuron numbers of two hidden layers are 128 and 64, respectively. The activation function of output layer is linear, and the rectified linear unit (ReLU) is adopted in the two hidden layers. The cardinality of adjacent cells is $|D_n| = 18, \forall n$, the first $I_c = 16$ interferers remain and the input dimension is 48. The power level number $|A| = 10$ and thus the output dimension is 10.

In the off-line training period, the DQN is first randomly initialized and then trained epoch by epoch. In the first 100 episodes, the agents only stochastically take actions, then they follow by adaptive ϵ-greedy learning strategy [18] to step in the following exploring period. In each episode, the large-scale fading is invariant, and thus the number of training episode must be large enough to overcome the generalization problem. There are 50 time slots per episode, and the DQN is trained with 256 random samples in the experience replay memory every 10 time slots. The Adam algorithm [19] is adopted as the optimizer in our paper, and the learning rate η exponentially decays from 10^{-3} to 10^{-4}. All training hyper-parameters are listed in Tab.I for better illustration. In the following simulations, these default hyper-parameters will be clarified once changed.

TABLE I
HYPER-PARAMETERS SETUP OF DQN TRAINING

Parameter	Value	Parameter	Value
Number of T per episode	50	Initial η	10^{-3}
Observe episode number	100	Final η	10^{-4}
Explore episode number	9900	Initial ϵ	0.2
Train interval	10	Final ϵ	10^{-4}
Memory size	50000	Batch size	256

The FP algorithm, WMMSE algorithm, maximum PA and random PA schemes are treated as benchmarks to evaluate our proposed DQN-based algorithm. The perfect CSI of current moment is assumed to be known for all schemes. The simulation code is available at https://github.com/mengxiaomao/PA_ICC.

B. Discount Factor

In this subsection, the sum-rate performance of different discount factor γ is studied[1]. We set $\gamma \in \{0.0, 0.1, 0.3, 0.7, 0.9\}$, and the average rate $\bar{C}$ over the training period is shown in Fig. 3. At the same time slot, obviously the values of $\bar{C}$ with higher $\gamma \in \{0.7, 0.9\}$ are lower than the rest with lower γ values. The trained DQNs are then tested in three cellular networks with different cell numbers N. As shown in Fig. 4, it shows that the DQN with $\gamma = 0.0$ achieves the highest $\bar{C}$ score, while the lowest average rate value is obtained by the one with the highest γ value. The simulation result proves that the non-zero γ has a negative influence on the sum-rate performance, which is consistent with the analysis in III-B. Therefore, a zero discount factor value is proposed for this static PA issue.

C. Algorithm Comparison

The DQN trained with zero γ is used, and the four benchmark algorithms stated before are tested as comparisons. In real cellular network, the user density is changing over time, and the DQN must have good generalization ability against this issue. In the training period the user number per cell K is 4, but in the testing period K is assumed to be in set $\{1, 2, 4, 6\}$. The averaged simulation results are obtained after 500 repeats. As shown in Fig. 5, the DQN achieves the highest $\bar{C}$ in all testing scenarios. Although the DQN is trained with $K = 4$, it still outperforms the other algorithms in the other cases. We also note that the gap between random/maximum PA schemes and the rest optimization algorithms is increased when K becomes larger. This can be mainly attributed that the intra-cell interference gets stronger with increased user

[1]The present standard DQL is the one with typical γ value in range $[0.7, 0.9]$ or even higher, and our proposed DQL has a zero γ value.

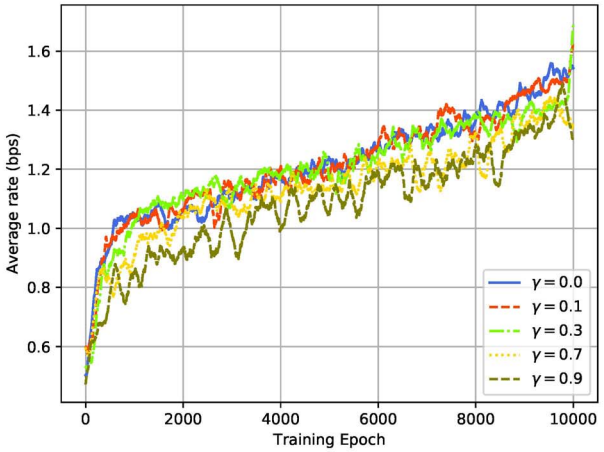

Fig. 3. With different γ values, the recorded average rate during training period (Curves are smoothed by averaged window).

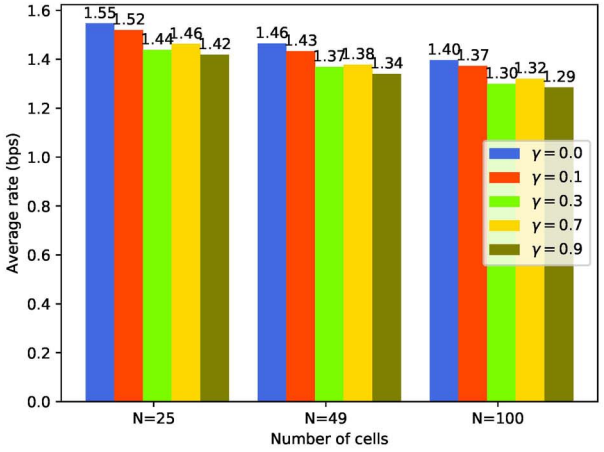

Fig. 4. The average rate $\bar{C}$ versus cellular network scalability for trained DQNs with different γ values.

density, which indicating that the optimization of PA is more significant in the cellular networks with denser users.

We also give an example result of one testing episode here ($K = 4$). In comparison with the averaged sum-rate values in Fig. 6, the performances of three PA algorithms (DQN, FP, WMMSE) are not stable, especially depending on the specific large-scale fading effects. Additionally, in some episodes the DQN can not be better than the other algorithms over the time (not shown in this paper), which means that there is still potential to improve the DQN performance.

In terms of computation complexity, the time cost of DQN keeps fixed as the total amount of users increases, due to the distributed execution. Meanwhile, both FP and WMMSE are centralized iterative algorithms, and thus the time cost is not constant, depending on the stopping criterion condition and

TABLE II
AVERAGE TIME COST PER EXECUTION T_c (sec).

	Algorithm		
	DQN	FP	WMMSE
CPU	$3.10e^{-4}$	4.80^{-3}	$1.84e^{-2}$

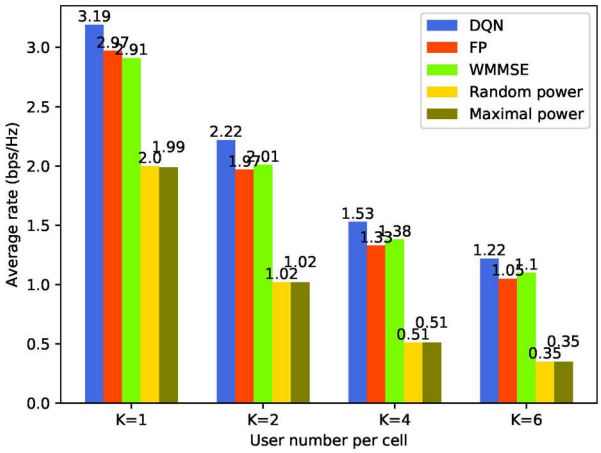

Fig. 5. The average rate $\bar{C}$ versus user number per cell. Five power allocation schemes are tested.

user number. The simulation platform is presented as: CPU Intel i7-6700. There are 100 users in the simulated cellular network, the time cost per execution T_c of our proposed DQL algorithm and the centralized model-based methods are listed in Tab. II. It can be seen that the time cost of DQN is about 15.5 and 61.0 times faster than FP and WMMSE, respectively.

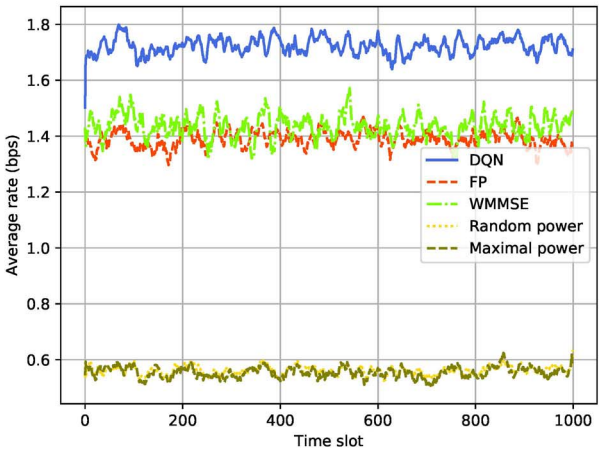

Fig. 6. Comparisons of all five power allocation schemes over 1000 time slots (Curves are smoothed by averaged window).

V. Conclusions

The distributed PA problem in the cellular network with IMAC has been investigated, and the data-driven model-free DQL has been applied to solve this issue. To be in consistent with the PA optimization target, the current sum-rate is used as reward function, including no future rewards. With our proposed discount factor $\gamma = 0$, and the DQN simply works as an estimator to predict the current sum-rate under all power levels with a certain CSI. Simulation results show that the DQN trained with zero γ achieves higher average sum-rate than the standard DQL with positive γ values. Then in a series of distinct scenarios, the proposed DQN outperforms the benchmark algorithms, also indicating that our designed DQN has good generalization abilities. In the two-step training framework, we have realized the off-line centralized learning with simulated communication networks, and the learned DQN is tested by distributed executions. In our future work, the on-line learning will be further studied to accommodate the real scenarios with specific user distributions and geographical environments.

VI. Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61801112, 61471117, 61601281), the Natural Science Foundation of Jiangsu Province (Grant No. BK20180357), the Open Program of State Key Laboratory of Millimeter Waves (Southeast University, Grant No. Z201804).

References

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

[2] Q. Shi, M. Razaviyayn, Z. Q. Luo, and C. He, "An iteratively weighted mmse approach to distributed sum-utility maximization for a mimo interfering broadcast channel," in *IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2011, pp. 4331–4340.

[3] M. Chiang, P. Hande, T. Lan, and C. W. Tan, "Power control in wireless cellular networks," *Found. Trends Netw.*, vol. 2, no. 4, pp. 381–533, 2008.

[4] H. Zhang, L. Venturino, N. Prasad, P. Li, S. Rangarajan, and X. Wang, "Weighted sum-rate maximization in multi-cell networks via coordinated scheduling and discrete power control," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 6, pp. 1214–1224, Jun. 2011.

[5] Z. Qin, H. Ye, G. Y. Li, and B. F. Juang, "Deep learning in physical layer communications," *CoRR*, vol. abs/1807.11713, 2018. [Online]. Available: http://arxiv.org/abs/1807.11713

[6] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.

[7] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning." *Nature*, vol. 521, no. 7553, p. 436, May 2015.

[8] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for interference management," *IEEE Trans. Signal Process.*, vol. 66, no. 20, pp. 5438–5453, Oct. 2018.

[9] F. Meng, P. Chen, L. Wu, and X. Wang, "Automatic modulation classification: A deep learning enabled approach," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10 760–10 772, Nov. 2018.

[10] H. Ye, G. Y. Li, and B. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[11] R. Amiri, H. Mehrpouyan, L. Fridman, R. K. Mallik, A. Nallanathan, and D. Matolak, "A machine learning approach for power allocation in hetnets considering qos," *CoRR*, vol. abs/1803.06760, 2018. [Online]. Available: http://arxiv.org/abs/1803.06760

[12] E. Ghadimi, F. D. Calabrese, G. Peters, and P. Soldati, "A reinforcement learning approach to power control and rate adaptation in cellular networks," in *2017 IEEE Int. Conf. Commun (ICC)*, May 2017, pp. 1–7.

[13] F. D. Calabrese, L. Wang, E. Ghadimi, G. Peters, L. Hanzo, and P. Soldati, "Learning radio resource management in RANs: Framework, opportunities, and challenges," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 138–145, Sep. 2018.

[14] L. Xiao, D. Jiang, D. Xu, H. Zhu, Y. Zhang, and V. Poor, "Two-dimensional anti-jamming mobile communication based on reinforcement learning," *IEEE Trans. Veh. Technol.*, pp. 1–1, 2018.

[15] Y. S. Nasir and D. Guo, "Deep reinforcement learning for distributed dynamic power allocation in wireless networks," *CoRR*, vol. abs/1808.00490, 2018. [Online]. Available: http://arxiv.org/abs/1808.00490

[16] Bottomley, G.E, and Croft, "Jakes fading model revisited," *Electron. Lett.*, vol. 29, no. 13, pp. 1162–1163, Jun. 1993.

[17] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, and G. Ostrovski, "Human-level control through deep reinforcement learning." *Nature*, vol. 518, no. 7540, p. 529, Feb. 2015.

[18] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980