Adaptive Multi-objective Reinforcement Learning for Pareto Frontier Approximation: A Case Study of Resource Allocation Network in Massive MIMO

Ruiqing Chen[1,3,4], Fanglei Sun[1], Liang Chen[1,3,4], Kai Li[1], Liantao Wu[1], Jun Wang[2], Yang Yang[1],(Fellow, IEEE)
[1]ShanghaiTech University, Shanghai, China
[2]University College London, London, UK
[3]Shanghai Institute of Microsystem and Information Technology, Shanghai, China
[4]University of Chinese Academy of Sciences, Beijing, China

***Abstract*—Multi-Objective Optimization (MOO) has always been an important issue in the field of wireless communications. With the development of 5G networks, more objectives have been concerned to improve the user experience. The relationship between these multiple objectives is complex or even conflicting, which increases the difficulty of solving the MOO problems. Traditional multi-objective optimization algorithms (e.g., genetic algorithm) have higher computation complexity and require to store multiple models for the preference of different objectives. Therefore, in this paper, a multi-objective scheduling model based on the Actor-Critic framework is proposed, which can effectively solve the multi-user scheduling problem under Massive Multiple-Input Multiple-Output (MIMO), and utilize a single model to approximate the Pareto frontier. In the single-cell downlink scheduling scenario, the proposed model is applied to the two objective optimization, i.e., channel capacity and fairness. The simulation results show that the performance of our model is close to the theoretical optimal value in the single-objective case. The Pareto frontier can be uniformly approximated in the multi-objective case, and it has strong robustness to never-seen preference combinations.**

***Index Terms*—Massive MIMO, multi-objective reinforcement learning (MORL), Pareto frontier, single cell Multi-User (MU)-MIMO scheduling**

I. Introduction

Massive MIMO technology can make deep use of space resources, enabling users within the coverage of the Base Station (BS) to communicate with the BS on the same time-frequency resource, and has become one of the key technologies of 5G networks [1]. The MOO problem has been widely studied in the resource allocation of 5G networks and has become a hot topic for Massive MIMO. The optimization objectives include average user rates, average area rates, downlink power, energy efficiency, throughput, fairness, construction cost, packet drop probability, etc [2]–[8]. These studies are mainly based on traditional optimization methods or genetic algorithms. However, as the number of antennas increases in massive MIMO, the computational complexity of traditional methods and genetic algorithms increases drastically.

This paper is partially supported by the project of cooperation with Huawei Noah's Ark Lab.

To reduce complexity and improve performance, in the single-objective resource allocation scenario, researchers have proposed many solutions based on Deep Learning (DL). DL technology is applied in MU-MIMO to improve the performance of resource allocation in [9] and [10], but the exponentially increased number of scheduling combinations makes the scale of the network even larger. Deep Reinforcement Learning (DRL) regards the scheduling problem as a sequential decision-making process, combining the advantages of DL extraction features to obtain a scheduling policy that maximizes the objective. An algorithm is proposed in [11] for resource allocation based on DRL, whereas treating all user combinations as the action space is not suitable for large user numbers. A multi-agent RL framework constructed in [12] to solve the problem of dimensional disaster, nevertheless, multi-agents would consume longer decision time and larger storage space.

MORL is used to explore multi-objective optimization problems under the framework of reinforcement learning. Those algorithms can be mainly divided into two categories: 1) the single-policy algorithm is devoted to finding the optimal policy that meets a certain preference; 2) the multi-policy algorithm is dedicated to finding the optimal policy under different preferences to approximate the Pareto frontier. The task of an algorithm is to obtain a more accurate and uniformly distributed Pareto frontier approximation. Random mixing and gradient methods are applied to MORL in [13], which can maximize the conditional objectives simultaneously, but no policy gradient is utilized in it. Reference [14] firstly applied policy gradient to MORL, and proposed the Radial Algorithm (RA) and the Pareto Following Algorithm (PFA). However, both two algorithms needed to save corresponding models for different preferences, which costs longer training time and larger storage space in actual application. A single policy proposed by [15] to approximate the Pareto frontier, which solved the problem of saving multiple models. Nevertheless, this algorithm was based on Q-Learning framework [16] so that it cannot handle large-scale action spaces.

Based on the above analysis and our previous work on policy-based DRL scheduling algorithm [10], a MORL algo-

rithm based on the Actor-Critic framework [17] is introduced. This model can provide the corresponding optimal policy for the different preferences of objectives, which is convenient for practitioners to choose an appropriate strategy. The main contributions are summarized as follow:

- *A novel MORL solution for Massive MIMO resource allocation*: An adaptive MORL model is proposed to approximate Pareto frontier with a single model, learning the optimal policy under different preferences. Specifically, we find that the scheduling problem can be reduced to a combinatorial optimization problem, which is solved by introducing an Actor-Critic framework. This framework is designed to provide a scheduling sequence according to Channel State Information (CSI), which solves the dimensional disaster issue of the action space.
- A novel training policy is proposed, which utilizes uniformly distributed weight sequences to train the model. The adaptability of the model was improved and can provide the corresponding Pareto optimal solution for the never-seen weight sequences. The model utilizes offline training and online learning. In the actual scheduling process, there is no need to do complex matrix calculations in each scheduling period any more, which can directly predict the scheduling results by CSI.
- Simulation results illustrate that our model can utilize a single model to achieve better approximation of Pareto frontier performance than that of the state-of-the-art algorithms. Meanwhile, our model has strong robustness for the never-seen preference combinations.

II. Preliminaries

A. Multi-objective Markov Decision Processes

The Multi-Objective Markov Decision Processes (MOMDPs) is an extension of Markov Decision Process (MDP), and each objective in MOMDPs has a different reward function and discount factor. Formally, MOMDPs can be typically defined as a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \boldsymbol{R}, \boldsymbol{\gamma} \rangle$, where $\mathcal{S} \in \mathbb{R}^n$ is the state space, $\mathcal{A} \in \mathbb{R}^q$ signifies the action space, $\mathcal{P}(s'|s,a) : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ indicates the transition probability of taking action a at state s transfer to state s'. $\boldsymbol{R}$ represents a set of reward function $\{\mathcal{R}_m |\ \forall m \in \{1,2,...M\}\}$, where $\mathcal{R}_m(s,a) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ denotes the instant reward obtained after performing action a at state s. $\boldsymbol{\gamma}$ is a set of discount factors $\{\gamma_m |\ \forall m \in \{1,2,...M\}, \gamma_m \in [0,1)\}$. The target of an policy π in MOMDPs is to maximize a set of discounted accumulated reward $\mathcal{J}^\pi = \{J_m^\pi |\ \forall m \in \{1,2,...M\}\}$, where J_m^π can be illustrated as:

$$J_m^\pi = \mathop{\mathbb{E}}_{s\sim\mathcal{S}, a\sim\pi} \left[\sum_{t=0}^{\infty} \gamma_m^t \mathcal{R}_m(s_t, a_t) \right]. \tag{1}$$

B. Multi-objective Optimization

In a MOO problem, a set of Pareto optimal solution can be obtained, which represent the optimal solution under different trade-off of objectives. Pareto optimal solution have the following concepts:

Pareto dominance: For two policy π and π', policy π strongly dominates policy $\pi'(\pi \succ \pi')$ when satisfy:

$$\forall m,\ J_m^\pi \geq J_m^{\pi'} \cap \exists m,\ J_m^\pi > J_m^{\pi'},\ m \in \{1,2,...M\}. \tag{2}$$

Pareto frontier: If there is no policy π' satisfy $\pi' \succ \pi$, the policy π is a pareto optimal solution. The set of all Pareto optimal solutions can be expressed as: $\Pi = \{\pi | \nexists \pi', \pi' \succ \pi\}$, by the mapping of accumulated reward function J, we can obtain the Pareto frontier $\mathcal{J}^* = \{\mathcal{J}^\pi | \pi \in \Pi\}$.

C. Channel Capacity

Suppose the BS has M_t transmitting antennas, there are K users in a single cell, and each user's devices have single receiving antennas. The signal to interference plus noise ratio (SINR) can be computed with:

$$\text{SINR}_k = \frac{|\boldsymbol{h}_k \boldsymbol{u}_k|^2 P_k}{n_0 + \sum_{i \neq k} |\boldsymbol{h}_k \boldsymbol{u}_i|^2 P_i}, \tag{3}$$

where $\boldsymbol{h}_k \in \mathbb{C}^{1 \times M_t}$ denotes the channel matrix sent from the BS to user k, P_k indicates the power allocation of user k. $[\boldsymbol{u}_1, ..., \boldsymbol{u}_K]$ represents the precoding matrix and $\boldsymbol{u}_i$ satisfies $\|\boldsymbol{u}_i\| = 1$, $\boldsymbol{u}_i \in \mathbb{C}^{M_t \times 1}$. n_0 is the noise power of environment. According to Shannon's theorem, we can get the channel capacity of user k as:

$$\mathcal{C}_k = W \log_2(1 + \text{SINR}_k), \tag{4}$$

where W represents channel bandwidth. The average channel capacity is obtained by divide the number of user: $\mathcal{C} = \frac{1}{K} \sum_{i=1}^{K} \mathcal{C}_i$.

D. Fairness

According to the proportional fairness scheduling algorithm [18], fairness of user k at time slot t can be defined as:

$$\mathcal{F}_k[t] = \frac{\mathcal{C}_k[t]}{T_k[t]}, \tag{5}$$

where $\mathcal{C}_k[t]$ is the channel capacity of user k at timeslot t, $T_k[t]$ denotes the average throughputs of user k at timeslot t, which can be updated with:

$$T_k[t+1] = \begin{cases} (1-\lambda) T_k[t] + \lambda \mathcal{C}_k[t] & k = k^* \\ (1-\lambda) T_k[t] & k \neq k^* \end{cases}, \tag{6}$$

where $\lambda \in (0,1)$ is a weight factor, k^* indicates the scheduled user.

E. Problem Formulation

Based on the above concepts, considering maximizing channel capacity and fairness, the multi-objective optimization problem can be formulated illustrated as:

$$\begin{aligned} \text{maximize} \quad & w_1 \mathbb{E}(\mathcal{C}) + w_2 \mathbb{E}(\mathcal{F}), \\ \text{subject to} \quad & w_1 + w_2 = 1,\ w_1 \geq 0,\ w_2 \geq 0, \end{aligned} \tag{7}$$

where $\mathcal{C} = \frac{1}{T} \frac{1}{K} \sum_{t=0}^{T} \gamma_1^t \sum_{i=1}^{K} \mathcal{C}_i[t]$, $\mathcal{F} = \frac{1}{T} \frac{1}{K} \sum_{t=0}^{T} \gamma_1^t \sum_{i=1}^{K} \mathcal{F}_i[t]$, w_1, w_2 represent the preference between capacity and fairness, T denotes the length of transmission time interval (TTI), and K indicates the number of user.

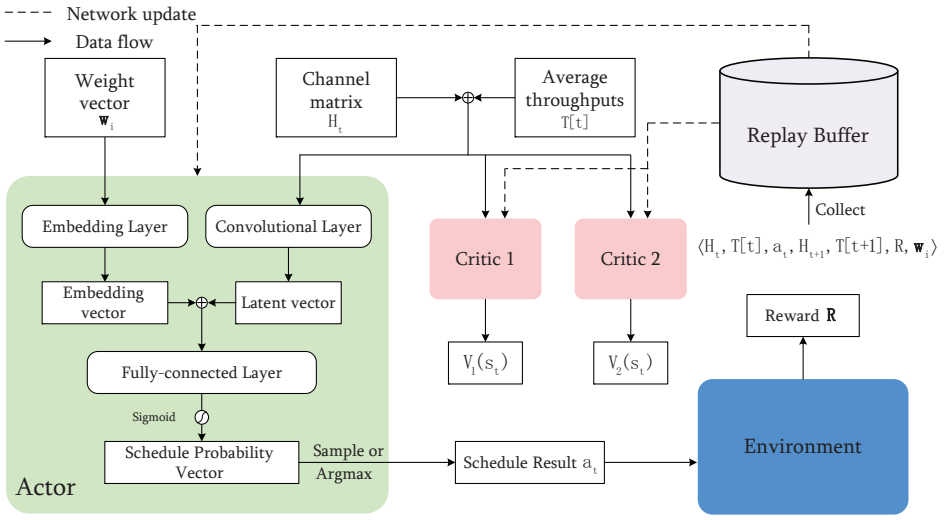

Fig. 1. The structure of adaptive MORL.

III. Adaptive MORL Algorithm

In this section, We begin with the definition of the MORL issue in massive MIMO, then introduce the network structure and the optimization targets of Actor and Critic. Finally, the detailed updating policy of the algorithm is given. The overall structure of our model is shown in Fig. 1.

A. MORL in Massive MIMO

Practically, the algorithm applied to the Massive MIMO scenario to find the Pareto frontier with conflicting objectives of capacity and fairness, making the scheduler adopts different scales of optimal policys according to practical demands. For single cell downlink MU-MIMO scheduling, we formulate it as the following:

- **State:** s_t consists of channel matrix H_t and average throughput $T[t]$, i.e., $s_t = (H_t, T[t])$. Although $P(H_{t+1}|H_t)$ depends on physical environment, whereas the action a_t will influence the average throughput $T[t+1]$, hence the state transition satisfy the Markov property.
- **Action:** a_t is the set of users scheduled by BS at time slot t, which belongs to discrete action space. Assuming that the number of users is K, there will be 2^K scheduling combinations, and $a_t \subseteq \{1, 2, ...K\}$.
- **Reward:** For objective of capacity and fairness, instant reward vector can be denoted as $\mathbf{r}_t = (\mathcal{C}_t, \mathcal{F}_t)$, which can be obtained by (4) and (5).

B. Multi-objective Critic

In the case of large action space, maintaining and updating the Q-value table is difficult. Therefore, the target of Critic is to predict the expected reward in the current state, i.e., $V(s_t)$, and introduce advantage function [19] to update the Actor. Suppose our system consists of M objectives, the advantage function of objective m is defined as:

$$\hat{A}_t^m = G_t - V(s_t) = \sum_{l=t}^{\infty} \gamma^{l-t} \delta_l, \ m \in \{1, 2, ...M\}, \tag{8}$$

where $\delta_t \approx r_{t+1} + \gamma V(s_{t+1}) - V(s_t)$ indicates temporal difference error, and $G_t = \sum_{l=0}^{\infty} \gamma^l r_{t+l}$ denotes accumulated reward. The optimization target for Critic networks are described as:

$$\mathcal{L}(\phi_m) = \frac{1}{T} \sum_{t=1}^{T} \left(V_{\phi_m}(s_t) - \sum_{l=0}^{\infty} \gamma^l r_{t+l} \right)^2, \tag{9}$$

where ϕ_m represents the parameter of Critic network for objective m, and each Critic updates independently.

In our model, multiple fully connected layers are used to build the Critic network structure, and the $tanh$ function is used to activate the output. The Critic network structure of different training objectives is the same.

C. Adaptive Actor

To obtain a complete Pareto frontier approximation, traditional multi-objective reinforcement learning algorithms need to adjust the weights and train the model repeatedly, which is time-consuming, labor-intensive and wastes storage space. We regard the multi-objective weight vector as the input to the Actor and adjust the updating policy so that a single Actor also can approximate the Pareto frontier.

Suppose the weight vector used for adjusting the importance of different objectives define as $\mathbf{w}_i = \{w_1, w_2, ..w_M\}$, hence the update target of Actor can be illustrated as:

$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{m=1}^{M} w_m \mathop{\mathbb{E}}_{s} \left[\hat{A}_m \left(s, \pi \left(s, \mathbf{w}_i | \theta \right) \right) \right], \tag{10}$$

where θ indicates the parameter of the Actor network, and w_m satisfy $w_m \in (0, 1)$, $\sum_{m=1}^{M} w_m = 1$, different i represents different proportion of weight vector.

In our model, the inputs of the Actor are channel matrices, average throughputs and the weight vector. Convolutional neural networks are used to capture the correlation of channel states and average throughputs between different users. After

concatenating with the weight vector, passing the multiple fully-connected layer and activating with $sigmoid$ function, the schedule probability vector with a dimension of the number of users is obtained. Finally, executing sampling or argmax operation on schedule probability vector, the scheduling result for all users is generated.

The detailed updating policy of our model is described in Algorithm 1. First of all, we determine a series of uniform distributed weight vectors. Under a given weight vector, the Actor interacts with the environment and generates trajectories. Then the trajectory and the corresponding weight vector are pushed into the replay buffer. The Critic network of each objective is updated in turn, and finally the Actor network will be updated.

Algorithm 1 Adaptive Multi-objective RL

Initialize Actor parameter θ.
Initialize Critic parameters $\{\phi_1, \phi_2 ... \phi_m\}$.
Setting weight vector sequence $\{\mathbf{w}_1, \mathbf{w}_2, ... \mathbf{w}_N\}$.
for episode number in {1,2,...K} **do**
  for index i in {1,2,...N} **do**
    for step t in {1,2,...T} **do**
      select an action by $a_t = \pi(s_t, \mathbf{w}_i|\theta)$.
      perform action a_t and obtain a new state s_{t+1} and reward vector $\mathbf{r}_t = \{r_t^1, r_t^2, ... r_t^m\}$.
      push $\{s_t, a_t, \mathbf{r}_t, s_{t+1}, \mathbf{w}_i\}$ into replay buffer.
    end for
  end for
  for index m in {1,2,...M} **do**
    update Critic m with (9)
  end for
  update Actor with (10)
end for

IV. Experiments

A. Configuration

The channel matrices H are generated by our 5G wireless simulation platform based on the 3GPP 3D-UMa channel model [20] with ray-tracing data. A BS with 8 transmitting antennas and 10 users is configured. Each user has one receiving antennas, the transmit power of BS is $0.25w$ and noise power n_0 id configured as $2.84e^{-13}w$.

The model uses online learning, and the exploration rate is 0.5. The learning rate of our Critic and Actor is configured as $1e-3$, and the decay of learning rate is assigned as $1e-4$. The update coefficient of average throughput λ is set as 0.9. All discount factors are the same as 0.95. Using a replay buffer with a size of 1750 to store trajectories and setting the length of TTI as 50.

B. Analysis

1) Validity of single objective: To verify the effectiveness of the model, two single-objective networks are designed to maximize the capacity and fairness performance, respectively. Meanwhile, the greedy policy is set for comparison, which is obtained by following procedure: 1) Traverse scheduling combinations of each TTI and select the scheduling sequence that can maximize the objective in the current TTI; 2) Sum up the objective value of each TTI. The model convergence curve is shown in Fig. 2.

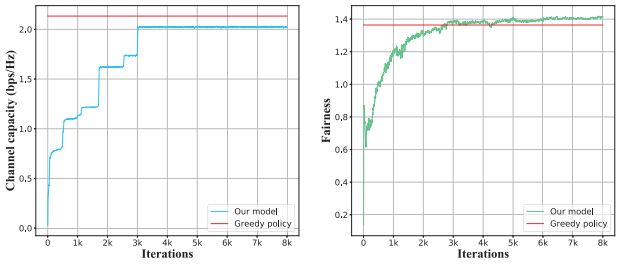

Fig. 2. Learning performance of capacity and fairness.

These training processes indicate that when taking capacity as objective, the performance of our model close to the value obtained by the greedy policy. When maximizing fairness reward, the model can obtain higher fairness than the value computed by the greedy policy, which means our model can capture the characteristics of the channel environment and give a scheduling result closer to the theoretical optimal fairness. In summary, in a single-objective environment, our model can provide an optimal schedule policy for different objectives.

2) Approximate Pareto frontier: Because the Pareto-Manifold Gradient Algorithm (PMGA) [21] cannot generate a million-level parameter space as the parameters of a DL network. Therefore, to verify the effect of the model approximation of the Pareto frontier, we compare our model with RA and PFA. The initial weight is generated by the following rule:

$$\mathbf{w}_i = \{\alpha_i, 1-\alpha_i\}, \alpha_i = \alpha_{i-1} + i \times d, \tag{11}$$

where $\alpha_i \in [0, 1]$. Set $\alpha_0 = 0,\ d = 0.1$ for our model, and $\alpha_0 = 0,\ d = 0.2$ for RA. PFA needs a two-step update rule. Firstly, move from the Pareto optimal solution with weight vector $\{1, 0\}$, then alternately training the model with weights vector $\{0, 1\}, \{0.5, 0.5\}$. Seven times of the two-step updates are performed. The experiment result is shown in Fig. 3.

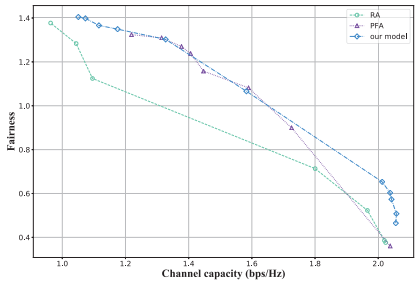

Fig. 3. Pareto frontier approximation.

The simulation results indicate that the performance of our model is similar to PFA and better than RA. However, the PFA corresponding model needs to be saved when converging to a different Pareto optimal solution. The practitioner does

not know the preference corresponding to this Pareto optimal solution. RA also has the problem of storing multiple models. Only one model needs to be stored in our algorithm and can provide optimal policy according to the preference vector, which is convenient for practitioners to choose an appropriate strategy.

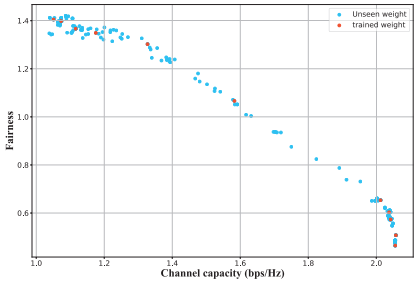

Fig. 4. Adaptiveness of model.

3) Adaptiveness: To verify the adaptability of our model, i.e. whether the model can give the correct Pareto optimal solution when encountering an unknown preference vector, we utilize (11) and configure $d = 0.005$ to generate a set of weights that the model has never seen in the training process. The approximate optimal Pareto solutions of the model are shown in Fig. 4.

The result illustrates that even if the model has never seen these weight vectors, it can still provide the optimal policy for the corresponding preference. In other words, the model can be trained with a limited number of weights to approximate the entire Pareto frontier. Of course, there have certain requirements for the selection of weights, which must be evenly distributed and include the weights are used to maximize the single objective. Therefore, our model has strong applicability and the ability to satisfy all preference requirements raised by practitioners.

V. Conclusion

To make it convenient for the practitioner to choose corresponding policies for different demands, a novel Actor-Critic framework of MORL is proposed. This framework is used to approximate the Pareto frontier with the optimization objectives of maximizing channel capacity and fairness in massive MIMO systems. The simulation results show that, for single-objective optimization, our model simplifies the scheduling problem into a combinatorial optimization problem, and solves the dimensional disasters problem in Massive MIMO technology as the number of users increases. For multi-objective optimization, we employ a single model to achieve better performance than the state-of-the-art multi-policy algorithm on Pareto frontier approximation. Furthermore, The model can still predict the corresponding optimal scheduling policy when encountering a never-seen preference. With the support of multi-agent technology, in the future, we will add other objectives and extend our model to solve multi-cell communication scheduling issues.

References

[1] C. Wang, F. Haider, X. Gao, X. You, Y. Yang, D. Yuan, H. M. Aggoune, H. Haas, S. Fletcher, and E. Hepsaydir, "Cellular architecture and key technologies for 5g wireless communication networks," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 122–130, 2014.

[2] E. Bjornson, E. A. Jorswieck, M. Debbah, and B. Ottersten, "Multiobjective signal processing optimization: The way to balance conflicting metrics in 5g systems," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 14–23, 2014.

[3] R. Devarajan, S. C. Jha, U. Phuyal, and V. K. Bhargava, "Energy-aware resource allocation for cooperative cellular network using multi-objective optimization approach," *IEEE Transactions on Wireless Communications*, vol. 11, no. 5, pp. 1797–1807, 2012.

[4] M. Tala't, L. Shen, C. Yu, and K. Feng, "Optimal transmission policy for maximizing green energy utilization in small cell networks," in *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, 2019, pp. 1–6.

[5] C. Yu, M. Tala't, L. Shen, and K. Feng, "A multi-objective model checking for transmission policy optimization in hybrid powered small cell networks," *IEEE Access*, vol. 8, pp. 71339–71352, 2020.

[6] H. R. Chi and A. Radwan, "Multi-objective optimization of green small cell allocation for iot applications in smart city," *IEEE Access*, vol. 8, pp. 101903–101914, 2020.

[7] Fan-Hsun Tseng, "Multi-objective optimisation for heterogeneous cellular network planning," *IET Communications*, vol. 13, pp. 322–330, 2019.

[8] Wei-Yu Chen, Po-Ya Hsieh, and Bor-Sen Chen, "Multi-objective power minimization design for energy efficiency in multicell multiuser mimo beamforming system," *IEEE Transactions on Green Communications and Networking*, vol. 4, no. 1, pp. 31–45, 2020.

[9] J. Shi, W. Wang, X. Yi, J. Wang, X. Gao, Q. Liu, and G. Y. Li, "Learning to compute ergodic rate for multi-cell scheduling in massive mimo," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, pp. 785–797, 2021.

[10] Y. Yang, Y. Li, K. Li, S. Zhao, R. Chen, J. Wang, and S. Ci, "Decco: Deep-learning enabled coverage and capacity optimization for massive mimo systems," *IEEE Access*, vol. 6, pp. 23361–23371, 2018.

[11] G. Bu and J. Jiang, "Reinforcement learning-based user scheduling and resource allocation for massive mu-mimo system," in *2019 IEEE/CIC International Conference on Communications in China (ICCC)*, 2019, pp. 641–646.

[12] N. Zhao, Y. Liang, D. Niyato, Y. Pei, M. Wu, and Y. Jiang, "Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5141–5152, 2019.

[13] Christian Robert Shelton, "Importance sampling for reinforcement learning with multiple objectives," Tech. Rep., 2001.

[14] S. Parisi, M. Pirotta, N. Smacchia, L. Bascetta, and M. Restelli, "Policy gradient approaches for multi-objective sequential decision making," in *2014 International Joint Conference on Neural Networks (IJCNN)*, 2014, pp. 2323–2330.

[15] Runzhe Yang, Xingyuan Sun, and Karthik Narasimhan, "A generalized algorithm for multi-objective reinforcement learning and policy adaptation," *CoRR*, vol. abs/1908.08342, 2019.

[16] Christopher JCH Watkins and Peter Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[17] Vijay R Konda and John N Tsitsiklis, "Actor-critic algorithms," in *Advances in neural information processing systems*. Citeseer, 2000, pp. 1008–1014.

[18] David Tse and Pramod Viswanath, *Fundamentals of wireless communication*, Cambridge university press, 2005.

[19] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel, "High-dimensional continuous control using generalized advantage estimation," *arXiv preprint arXiv:1506.02438*, 2015.

[20] Yu Wenxin, Li Kai, Zhou Mingtuo, Li Jian, and Yang Yang, "Research on adaptive neural network for 3d channel amplitude prediction based on ray-tracing data," *Journal of University of Chinese Academy of Sciences*, in press, 2020. (in Chinese).

[21] Simone Parisi, Matteo Pirotta, and Marcello Restelli, "Multi-objective reinforcement learning through continuous pareto manifold approximation," *Journal of Artificial Intelligence Research*, vol. 57, pp. 187–227, 2016.