Multi-User MmWave Beam Tracking via Multi-Agent Deep Q-Learning

MENG Fan[1], HUANG Yongming[2], LU Zhaohua[3], XIAO Huahua[3]

(1. Purple Mountain Laboratories, Nanjing 211111, China;
2. School of Information Science and Engineering, Southeast University, Nanjing 210096, China;
3. State Key Laboratory of Mobile Network and Mobile Multimedia Technology, ZTE Corporation, Shenzhen 518057, China)

DOI: 10.12142/ZTECOM.202302008

https://kns.cnki.net/kcms/detail/34.1294.TN.20230508.1716.002.html, published online May 9, 2023

Manuscript received: 2022-11-08

Abstract: Beamforming is significant for millimeter wave multi-user massive multi-input multi-output systems. In the meanwhile, the overhead cost of channel state information and beam training is considerable, especially in dynamic environments. To reduce the overhead cost, we propose a multi-user beam tracking algorithm using a distributed deep Q-learning method. With online learning of users' moving trajectories, the proposed algorithm learns to scan a beam subspace to maximize the average effective sum rate. Considering practical implementation, we model the continuous beam tracking problem as a non-Markov decision process and thus develop a simplified training scheme of deep Q-learning to reduce the training complexity. Furthermore, we propose a scalable state-action-reward design for scenarios with different users and antenna numbers. Simulation results verify the effectiveness of the designed method.

Keywords: multi-agent deep Q-learning; centralized training and distributed execution; mmWave communication; beam tracking; scalability

Citation (Format 1): MENG F, HUANG Y M, LU Z H, et al. Multi-user mmWave beam tracking via multi-agent deep Q-learning [J]. ZTE Communications, 2023, 21(2): 53 - 60. DOI: 10.12142/ZTECOM.202302008
Citation (Format 2): F. Meng, Y. M. Huang, Z. H. Lu, et al., "Multi-user mmWave beam tracking via multi-agent deep Q-learning," *ZTE Communications*, vol. 21, no. 2, pp. 53 - 60, Jun. 2023. doi: 10.12142/ZTECOM.202302008.

1 Introduction

Millimeter wave (mmWave) communications have gained extensive attention due to vast bandwidth resources. The beamforming technique with large antenna arrays can improve the mmWave communication network coverage and make up for severe free-space path loss. MmWave signals are highly directional with beamforming, and thus beam tracking is needed to ensure the stability and quality of connected links in mobile scenarios. Currently, mmWave systems typically use hybrid analog-digital architectures to reduce the hardware cost and power consumption.

Traditional beam alignment exhaustively scans the whole beam space, and the introduced high overhead is unacceptable for mobile scenarios. The efficiency of beam training can be improved by the hierarchical searching method with a multi-resolution codebook. Refs. [1 - 3] have reduced the beam training overhead by exploiting prior knowledge of the mmWave channel such as the angle of departure (AoD) or the angle of arrival (AoA), and a low-resolution codebook is further considered in fast-varying scenarios[3]. As a heuristic solution, a deep learning based fast beamforming design method is introduced, without complex operations and iterations in conventional methods[4].

To better utilize implicit prior information embedded in the practical environments, data-driven approaches are feasible[5 - 7]. A fingerprint database is used in Ref. [8] to access historical training records according to the user's location. In Ref. [9], a data-driven data fusion module is developed to combine AoD and time of arrival (ToA) positioning, and positioning-based beam tracking methods are introduced for high-speed railway scenarios. In general, offline learning requires a large number of collected samples in advance, and recollection is needed once the environment changes, leading to difficulties in deployment. Meanwhile, reinforcement learning can realize online learning without offline data, and optimize the policy through interactions with the environment. To reduce beam training overhead, Ref. [10] proposes a multi-armed bandit (MAB) based approach where the training beams are selected by the upper confidence bound strategy. However, the simple MAB model has limited ability to learn from the surroundings, furthermore, a centralized deep Q-learning (DQL) method is proposed in Ref. [11], where the beam training problem is modeled as a Markov decision process (MDP). However, due to its multi-user single-agent model, the action space exponentially explodes with the growth of the user number and lacks scalability to different

user and antenna numbers.

In this paper, under the centralized training and distributed execution (CTDE) framework, we propose a beam tracking method with distributed DQL for the beam tracking problem in dynamical mmWave scenes. Specifically, a distributed beam tracking algorithm is designed to adapt to the changing environments, where each user is regarded as an agent. We also propose several enhancements on the vanilla DQL, including simplified deep Q-network (DQN) training and scalable state-action-reward designs. The main contributions are summarized as follows:

• We develop multi-agent DQL for simultaneous multi-user beam tracking, and the DQL method follows the CTDE framework, where all users share the same policy learned with collected data from all the users.

• We prove that the beam tracking problem is a quasi-static optimization problem instead of an MDP, and a simplified DQL training scheme is proposed to reduce the complexity.

• We propose scalable state-action-reward designs for the DQL which can work in scenarios with different BS antenna and user numbers. In comparison, the existing centralized DQL methods cannot be transferred to a different scenario due to a mismatch of input and output.

The rest of this paper is organized as follows. Section 2 presents the system model. Section 3 describes the beam tracking design with a distributed DQL method. Section 4 gives the simulation results. Section 5 draws the conclusions.

2 System Model and Problem Formulation

2.1 System Model

We consider the downlink transmission in a link-level mmWave communication system composed of one base station (BS) and U mobile users (MU). The BS is equipped with M transmit antennas and N_{rf} radio frequency (RF) chains which are fully-connected, and each MU has a single receiving antenna. One data stream is simultaneously allocated to each user, and thus $U = N_{\mathrm{rf}}$. On the BS side, the hybrid analog-digital precoding is considered. The analog precoding matrix is denoted by $\boldsymbol{A} \in \mathbb{C}^{M \times N_{\mathrm{rf}}}$, where the u-th column, i.e., $\boldsymbol{A}[:,u]$, is the analog precoding vector of user u, and it is selected from the discrete Fourier transformation (DFT) codebook $\boldsymbol{F} \in \mathbb{C}^{M \times M}$. Similarly, the digital precoder is $\boldsymbol{V} = \left[\boldsymbol{v}_1,\cdots,\boldsymbol{v}_{N_{\mathrm{rf}}}\right]$, where the u-th column $\boldsymbol{H} = \left[\boldsymbol{h}_1,\cdots,\boldsymbol{h}_{N_{\mathrm{rf}}}\right]^H$ denotes the digital precoding vector, and $\boldsymbol{s}$ is the independent and identical distributed (i.i.d.) data stream. The received signal can be written as:

$$\boldsymbol{y} = \boldsymbol{HAVs} + \boldsymbol{w}, \tag{1}$$

where $\boldsymbol{w} \sim \mathcal{CN}\left(0,\sigma_n^2 \boldsymbol{I}_{N_{\mathrm{rf}}}\right)$ denotes zero mean additive white Gaussian noise (AWGN) with variance σ_n^2, and the channel matrix is denoted by $\boldsymbol{v}_u \in \mathbb{C}^{N_{\mathrm{rf}} \times 1}$, where $\boldsymbol{h}_u$ is the downlink channel vector from the BS to MU u.

Without sacrificing generality, the DFT codebook $\boldsymbol{F}$ is constructed by evenly sampling the beam space, and thus the i-th column is:

$$\boldsymbol{F}_i = \boldsymbol{a}(\phi_i)|_{\phi_i = \frac{\pi}{2}\left(\frac{2i}{M} - 1\right)}, \tag{2}$$

where the array response with azimuth being ϕ is:

$$\boldsymbol{a}(\phi) = \frac{1}{\sqrt{M}}\left[1,\cdots,e^{\mathrm{j}kd m \sin(\phi)},\cdots,e^{\mathrm{j}kd(M-1)\sin(\phi)}\right], \tag{3}$$

where $k = \frac{2\pi}{\lambda}$, and λ is the wavelength.

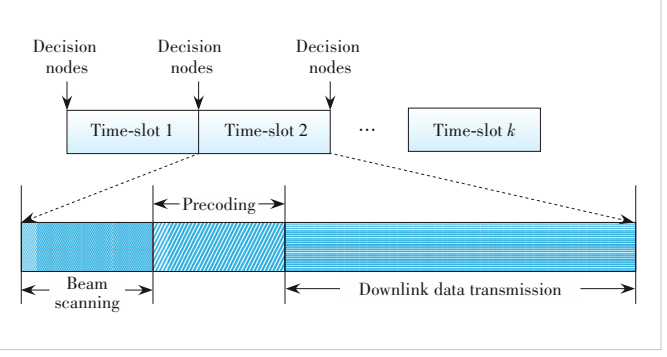
Instead of directly estimating high-dimensional channel state information (CSI) $\{\boldsymbol{h}_u\}$, we use low-dimensional equivalent CSI $\{\bar{\boldsymbol{h}}_u\}$ obtained by beam scanning. Specifically, the equivalent channel is a multiplication of the channel matrix $\boldsymbol{H}$ and the analog precoding matrix $\boldsymbol{A}$, and then the BS can be considered as a transmitter with N_{rf} ports. The equivalent channel vector between the BS and MU u is $\bar{\boldsymbol{h}}_u = \boldsymbol{A}^H \boldsymbol{h}_u$, which will be used for digital precoding.

2.2 Problem Formulation

As illustrated in Fig. 1, the precoded signal is transmitted within a correlation block which is divided into three phases: beam scanning, hybrid precoding, and data transmission. After the beam scanning in time slot t, we can obtain the equivalent channel vectors $\{\bar{\boldsymbol{h}}_u\}$. Then, the digital precoding problem is modeled as:

$$\begin{aligned} \max_{\{\boldsymbol{v}_u\}} \quad & \sum_{u \in \mathcal{U}} \log\left(1 + \frac{\left|\bar{\boldsymbol{h}}_u^{\mathrm{H}} \boldsymbol{v}_u\right|^2}{\sum_{v \neq u}\left|\bar{\boldsymbol{h}}_u^{\mathrm{H}} \boldsymbol{v}_v\right|^2 + \sigma_n^2}\right), \\ \text{s.t.} \quad & \sum_{u \in \mathcal{U}} \left(\boldsymbol{v}_u\right)_{\mathrm{F}}^2 \leqslant P_{\mathrm{m}}, \end{aligned} \tag{4}$$

where P_{m} denotes the maximal transmit power of the BS. Eq. (4) can be solved by minimum mean square error



▲Figure 1. Three phases of a time slot

(MMSE) precoding. We adopt a classical linear MMSE to derive the transmitter digital precoder as follows:

$$\boldsymbol{D} = \xi\bar{\boldsymbol{H}}\left(\bar{\boldsymbol{H}}^{H}\bar{\boldsymbol{H}} + \sigma_n^2\boldsymbol{I}_{N_{\mathrm{rf}}}\right)^{-1}, \tag{5}$$

where $\bar{\boldsymbol{H}} = \left[\bar{\boldsymbol{h}}_1,\cdots,\bar{\boldsymbol{h}}_{\mathrm{rf}}\right]^T$, and ξ is a factor to control the BS maximum transmit power.

We evaluate the system performance by an effective sum-rate. Let f_t be the optimal value of an objective function in Eq. (4). Considering the beam training overhead, the effective achievable sum rate during time slot t is defined as

$$R_t = \left(1 - \frac{|\mathcal{F}_t|t_S + t_P}{t_C}\right)f_t, \tag{6}$$

where $\mathcal{F}_t$ is a subset of the codebook $\mathcal{F}$ and its elements are the training beams to be scanned, $|\mathcal{F}_t|$ denotes the corresponding cardinal number, t_S is the duration of one training beam, t_{P} denotes the duration of precoding and online learning, and t_{C} denotes the duration of one time slot. With previous known experience, the investigated problem is to design a beam tracking algorithm to maximize time average of Eq. (6), where the digital precoding vectors $\{\boldsymbol{v}_u\}$ in Eq. (4) are derived from the beam scanning results.

3 Beam Tracking with Deep Q-Learning

3.1 Preliminary of Deep Q-Learning

Without loss of generality, single-agent DQL is developed for a problem modeled as a process of continuous interactions between an intelligent agent and the environment, i.e., MDP. In each interaction, the agent conducts an action a by a policy π with an observed state s, then receives a feedback reward r from the environment, and enters a new state s'. The goal is to learn a strategy for cumulative reward maximization. In a value-based algorithm, the action is selected by the values of state-action pairs, i.e., Q-values. The Q-value is defined as follows:

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty}\gamma^k r_{k+1}|_{s,a}\right], \tag{7}$$

where $\gamma \in [0,1)$ is the discount factor. The mapping from the state to the action values is realized by a learnable network DQN.

3.2 Centralized Deep Q-Learning

Intuitively, the investigated multi-user beam tracking problem can be modeled as an MDP, and the centralized DQL method is considered in Ref. [11]. Specifically, during time-slot t, the modulus of the channel vector of MU u in beam space is given as:

$$\boldsymbol{I}_u^t = \mathrm{abs}\left(\boldsymbol{F}^H\boldsymbol{h}_u^t\right). \tag{8}$$

Stacking $\{\boldsymbol{I}_u^t\}$ into a matrix $\boldsymbol{I}^t$, we have

$$\boldsymbol{I}^t = \left[\boldsymbol{I}_1^t,\cdots,\boldsymbol{I}_U^t\right] \in \mathbb{R}^{M\times U}, \tag{9}$$

where we can obtain an "image" $\boldsymbol{I}^t$ as the state s^t, which describes the distribution of effective paths or beam directions. Since mmWave channels are sparse in the beam domain and the training beam set is a subset of the DFT codebook, $\boldsymbol{I}^t$ is a sparse image and most elements of $\boldsymbol{I}^t$ are near zero.

To achieve the goal of sensing the environment, an action is defined based on the difference in the indices between two adjacent beams. An action for a single MU is defined by a pair of integers (μ,σ), where μ denotes the difference of the indices of the optimal beams in two adjacent time-slots, i.e.,

$$\mu^t = \left(b^t - b^{t-1}\right)\mathrm{mod}M, \tag{10}$$

where b denotes the beam index, "mod" denotes the modular arithmetic, and σ denotes the number of beams used to sweep the beam space, respectively. The action space corresponding to MU u is denoted by

$$\mathcal{A}_u = \begin{Bmatrix} \left(\mu_1 - \left\lfloor\frac{\sigma_1}{2}\right\rfloor,\mu_1 + \left\lfloor\frac{\sigma_1}{2}\right\rfloor\right),\cdots, \\ \left(\mu_L - \left\lfloor\frac{\sigma_L}{2}\right\rfloor,\mu_L + \left\lfloor\frac{\sigma_L}{2}\right\rfloor\right) \end{Bmatrix}\mathrm{mod}M, \tag{11}$$

where L is the size of the action space. The action space for all MUs is a product of $\{\mathcal{A}_u\}$, i.e., $\mathcal{A} = \prod_u\mathcal{A}_u$. Finally, the immediate reward in time-slot t is given in Eq. (6), i.e., $r^t = R_t$. The scanned beams are $\mathcal{F}_t = a^t \in \mathcal{A}$.

In DQL[12], a separate target network is introduced to stabilize DQN training, the weights of which change slowly compared with the primary network.

However, several shortcomings of the centralized framework must be observed. Firstly, as the user number U increases, the cardinality of DQN input space $|\mathcal{S}| = M \times U$ grows linearly, and cardinality of output space $|\mathcal{A}| = L^U$ grows exponentially. The training is difficult for such a DQN since the state-action space increases exponentially with user numbers. Additionally, exploration in high-dimensional space is inefficient, and thus the learning can be impractical. Secondly, the DQL lacks scalability in changing user number U and the BS antenna number M.

3.3 Simplicated DQN Training

3.3.1 Centralized Training and Distributed Execution Framework

Single-agent DQL for multi-user beam tracking can lead to action space explosion[13]. To address this issue, we propose

the multi-agent DQL with CTDE. Specifically, each MU is regarded as one agent. All agents are synchronized and distributed, and they share the same policy for online training and inference. The collected data from all agents are aggregated to form a centralized training set, and the shared policy is trained with the centralized training set. The shared policy is then executed by all agents. Thus, the training is centralized and the execution is distributed. The CTDE framework can solve the space explosion problem, and also improve network scalability and reduce training difficulty.

3.3.2 Non-MDP Problem

To adapt to the dynamic environment, low computational complexity is significant for online training, therefore we propose to simplify the vanilla DQN method, i.e., reducing the beam tracking problem as a static optimization problem and solving it in a greedy manner[13]. From the perspective of MDP, the following conclusion can be drawn.

Theorem 1. When the state transition function is independent of the current action and the reward is independent of the state to be transferred to, the maximized cumulative reward under the optimal policy is equivalent to the combination of single-step rewards.

The description of the assumed conditions can be mathematically formulated as:

$$P^{a}_{s\to s'} = P_{s\to s'}, \tag{12}$$

$$r^{a}_{s\to s'} = r^{a}_{s}, \tag{13}$$

where P denotes state transition probability. The proof of Theorem 1 is given in the Appendix.

In practice, the beam alignment success rate reaches a certain extent p_{thr} close to 100%. Once the misalignment occurs, the BS instantly realigns and a partial observation is obtained. This observation is very similar to the one observed when the beam is successfully tracked. Thus, the new state observed from the environment is mainly determined by the moving users and the fading channels, and is weakly related to the taken action. Additionally, when the reward is sum-rate R_t in Eq. (6), the current reward is irrelevant to the new state. Therefore, we can regard that the system satisfies Eqs. (12) and (13), and we set the discount factor γ as 0. Formally, the Q-value function in Eq. (7) can be simplified as follows:

$$Q\left(s_t,a_t\right) = r_t. \tag{14}$$

In summary, when Theorem 1 holds, we can replace the above Eq. (7) with Eq. (14) for DQN training, which has the following benefits:

1) With no need for target networks, the training complexity is reduced;

2) The variance of Q-value estimation is reduced, and thus the training is more efficient.

3.4 State, Action and Reward Design

To make the choice of action in each state logical, the design of the state must reflect the state of the user's interaction with the environment. Since the irregular movements of the user are the main cause of the dynamic changes in the environment, the state can be defined according to the movement of the user. We propose to use the index difference of optimal beams measured in successive time slots as the state. This state design reflects changes in the direction and rate of the user's motion over a period of time.

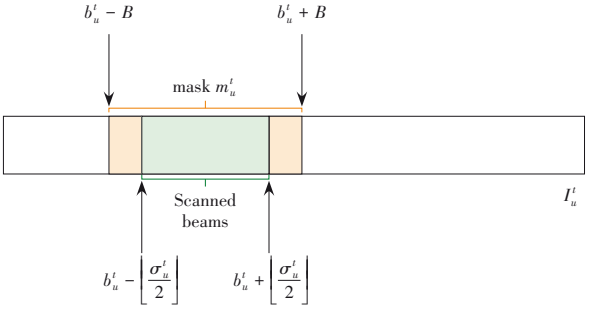
In the state design of centralized DQL, the state space grows linearly with the user number U. To achieve scalability against U, firstly we propose to decouple the centralized state $\boldsymbol{I}^t$ as a bunch of distributed states $\{\boldsymbol{I}^t_u\}$. Thus, once training is finished, the distributed state can be extended to the scenarios with any user number.

The mmWave channel is sparse in the beam domain, thus most of the element values in $\boldsymbol{I}^t_u$ are equal or close to zero. Besides, in the beam tracking period t for user u, only a small subset of training beams is scanned. Therefore, except for σ^t_u scanned beams, $\left(M-\sigma^t_u\right)$ elements in $\boldsymbol{I}^t_u$ are zero, indicating that we can retain the scanned beams as the distributed state and leave out the others. Secondly, as shown in Fig. 2, we propose to cap $\boldsymbol{I}^t_u$ with a mask $\boldsymbol{m}^t_u \in \mathbb{R}^{M\times 1}$, the center of which is b^t_u and the half width is $B = \max^{L}_{l=1}\sigma_l$, to achieve scalability against the BS antenna number M. Formally, the masked distributed state is:

$$s^t_u = \boldsymbol{I}^t_{u,i}\Big|_{i=\left(b^t_u-B\right)\bmod M}^{\left(b^t_u+B\right)\bmod M}. \tag{15}$$

The other elements in $\boldsymbol{I}^t_u$ are left out. Similarly, we decouple the centralized action $a\in\mathcal{A}$ as $\{a_u\in\mathcal{A}_u\}$. Thus, the state space is $2B$ and the action space is L which are fixed and irrelevant to the user number U. This indicates the proposed distributed design is scalable to changing user numbers and BS antenna numbers.

Reward acquisition requires completion of analog and digital precoding. At time-slot t, the effective achievable rate of user u in Eq. (6) is defined as the reward



Note: The modular arithmetic is neglected.

▲Figure 2. State of user u after masking

$$r_u^t = \left(1 - \frac{|\mathcal{F}_t| t_S + t_P}{t_C}\right) \log\left(1 + \frac{\left|\bar{\boldsymbol{h}}_u^H \boldsymbol{v}_u\right|^2}{\sum_{v \neq u} \left|\bar{\boldsymbol{h}}_u^H \boldsymbol{v}_v\right|^2 + \sigma_n^2}\right). \quad (16)$$

The scanned beams is $\mathcal{F}_t = \bigcup_{u=1}^{U} a_u^t$.

In summary, compared with the centralized DQL introduced in Section 3.2, the proposed distributed DQL has the following benefits:

1) The input and output of the DQN are greatly reduced, and thus the DQN is simplified.

2) The DQN is scalable to the changing user number U and the BS antenna number M.

3) The sample number is U times higher than that of the centralized DQL.

We give two instances of DQNs in Table 1, and they both have a three-layer neural network (NN). The activation function $f(\cdot)$ and the neuron number of each DNN layer are listed on the left and the right sides, respectively. The activation functions are rectified linear unit (ReLU): $f(\boldsymbol{x}) = \max(0, \boldsymbol{x})$, and linear: $f(\boldsymbol{x}) = \boldsymbol{x}$.

3.5 Distributed Beam Tracking Algorithm Procedure

For clarity, the flow of the distributed beam tracking algorithm is summarized in Algorithm 1. At the beginning of each episode, the entire codebook is scanned to obtain the initial state s_u^t for every single user. Then, the agent selects action a^t by the ε-greedy strategy, and ε for the ε-greedy strategy varies as

$$\varepsilon = \frac{n_{\text{to}} - n_{\text{cur}}}{n_{\text{to}}}, \quad (17)$$

where n_{to} is a fixed value. We set another fixed value n_{thr} that is less than n_{to}, and n_{cur} varies as:

$$n_{\text{cur}} = \begin{cases} n, & n < n_{\text{thr}} \\ n_{\text{thr}} & n \geqslant n_{\text{thr}} \end{cases}. \quad (18)$$

In Algorithm 1, by performing steps (1), r^t and digital precoding vectors are obtained. Then downlink data transmission is executed in step (2). In step (3), the parameters of DQN are updated.

The ε-greedy strategy is used to explore the environment, the existence of which can lead to the failure to find the optimal beam at each time-slot, i.e., misalignment. Five consecutive moments of mis-alignment are defined as an incident. Once an incident occurs, the optimal beam initialization process starts immediately from this moment, which is called "calibration" and is achieved via exhaustive search.

▼**Table 1. Deep Q-network (DQN) setting**

DQL type	Centralized DQL	Distributed DQL (proposed)
Output layer	linear,L^U	linear,L
Hidden layer	ReLU,32	ReLU,32
Input layer	linear,$M \times U$	linear,$2B$

DQL: deep Q-learning ReLU: rectified linear unit

Algorithm 1: Distributed beam tracking algorithm

Initialize: 1) DFT codebook $\boldsymbol{F}$; 2) DQN with random weights θ; 3) replay memory D;
for each episode do
  scan optimal beam in codebook to obtain U initial states
  while $t \geqslant k$ and $t \leqslant$ snapshot **do**
    (1) **obtain** analog and digital precoding
      a) choose action a^t according to the ε-greedy strategy
      b) execute action a^t and observe the next state s^{t+1}
      c) compute reward r^t and obtain (s^t, a^t, r^t)
      4) obtain precoding vectors
    (2) **transmit** data during the remaining of time-slot t
    (3) **update** parameters θ of DQN
      a) store transition (s^t, a^t, s^{t+1}, r^t) in D
      b) sample batch of transitions from D
      c) update θ with the gradient descent optimizer
    let $t \leftarrow t + 1$
  end while
end for

4 Simulation Results

In this section, we evaluate the performance of the proposed beam training algorithm via numerical results. The MUs are assumed to move along a circle and the BS is located at the center. To reflect dynamical changes of the distances between the BS and the MUs, the time-varying path-loss of the MUs is incorporated into the mmWave channel model. The movement velocity of the MUs is assumed to be stochastic, and obeys a known probability law. Accordingly, switching to another beam in the next time-slot is also stochastic and obeys some probability law.

For each MU, the probability that the optimal beam of the MU switches to the i-th beam of the next S beams is denoted by $p_{S,i}(i = 0, \cdots, S)$, where $p_{S,0}$ is the probability that the optimal beam of the MU in the next time-slot is still the current beam. For example, two probability distributions are considered, where $p_{S,i}$ is given by:

$$p_{S,i} = e^{-\eta i} \left(\sum_{k=0}^{S} e^{-\eta k}\right)^{-1}. \quad (19)$$

The parameter $\eta > 0$ defines the "decay" rate. Specifically,

we consider $S = 4$ and $\eta = 1.0$. For each MU u, the action space $\mathcal{A}_u$ is given by

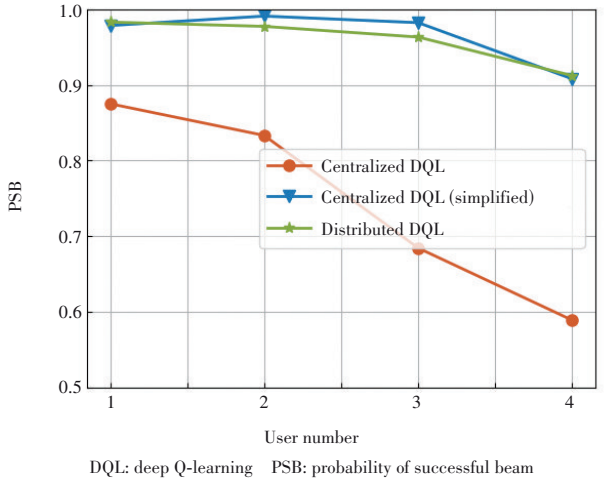
$$\mathcal{A}_u = \{(a,b)|a = \{0,1,2,3\};\ b = \{1,3,5\}\}. \quad (20)$$

Next, we evaluate the performance of the designed DQL algorithm. The simulation results of the centralized DQL in Ref. [11] , the proposed simplified DQL in Section 3.3.2 named centralized DQL (simplified) and the proposed distributed DQL are provided for comparison. Besides, the exhaustive search beam tracking, the bandit learning based beam tracking, Q-learning based beam tracking and the centralized DQL algorithm are studied in Ref. [11], and the simulation results show that the centralized DQL algorithm is the best. We use the average effective sum-rate (AESR) and the probability of successful beam (PSB) alignment as the two metrics for performance evaluation. The simulation platform is presented as Python 3.9, Tensorflow 2.9.0, CPU Intel i7-9700K and GPU Nvidia GTX-1070Ti.

The PSB for different beam tracking algorithms with $M = 32$, $U = 2$ is shown in Fig. 3. We have noticed that the proposed centralized DQL (simplified) has the fastest convergence speed and the highest PSB performance. Meanwhile, the centralized DQL converges slowly. The proposed distributed DQL converges fastly when the epoch number is up to 200, but it cannot work well with small epoch numbers lower than 100.

The PSB for different beam tracking algorithms with $M = 64$ is shown in Fig. 4, and the user number $U \in \{1,2,3,4\}$. We have observed that the proposed centralized DQL (simplified) and distributed DQL have similar PSB performance, with varying user numbers. However, the centralized DQL cannot work well and has a poor PSB performance. Besides, the training time costs are listed in Table 2. The time cost increases significantly for centralized methods and remains fixed for the proposed distributed method (the cost time rises due to interactions with the environment). As U increases, the action space grows exponentially for the centralized methods, and the training is very difficult for $U > 4$. This indicates the proposed distributed method is computationally efficient.

The AESR of the proposed distributed DQL for different beam tracking algorithms with $U = 2$ is shown in Fig. 5, and the user number $M \in \{32, 64, 128, 256\}$. Similar to the case with different user numbers, the proposed centralized DQL (simplified) and distributed DQL have similar AESR perfor-

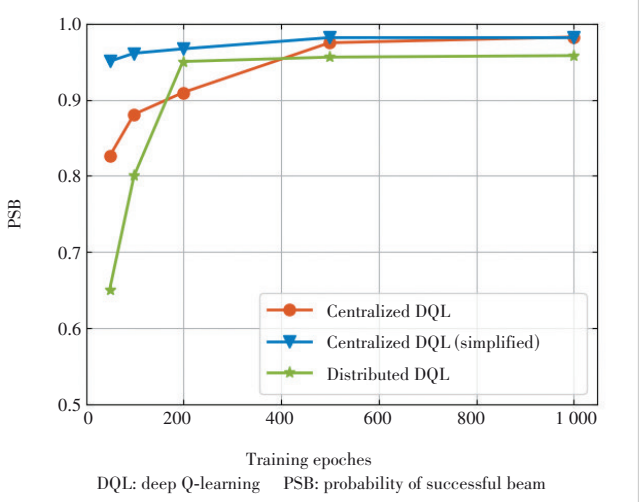


▲Figure 4. PSB performance versus user numbers

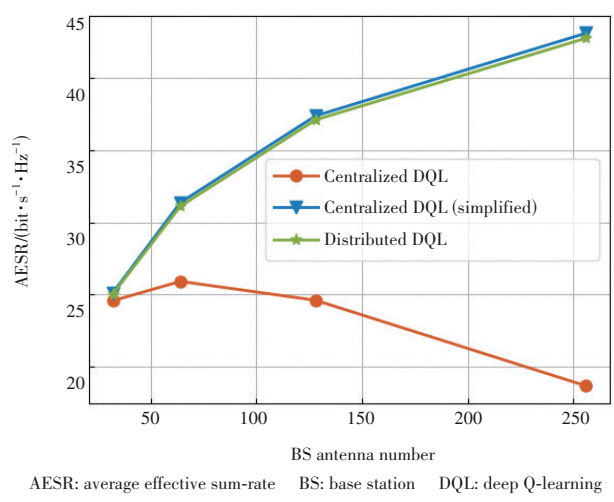
▼Table 2. Training time cost

User Number	Centralized DQL/s	Centralized DQL (simplified)/s	Distributed DQL (proposed)/s
$U = 1$	10.61	8.82	10.09
$U = 2$	11.91	10.18	11.51
$U = 3$	16.32	14.10	12.82
$U = 4$	87.55	61.97	14.40

DQL: deep Q-learning



▲Figure 3. PSB performance versus training epochs



▲Figure 5. AESR performance versus BS antenna numbers

mance, with varying BS antenna numbers, and the centralized DQL has a poor AESR performance.

The scalability is studied in Fig. 6. The distributed DQL indicates the training data and the test data are independent and identically distributed (i.i.d). The distributed DQL (generalized) indicates the training data has a fixed user number/BS antenna number, meanwhile the test data have changing user numbers/BS antenna numbers. The results show that the learned DQN in changing scenarios has the same AESR performance as those with fixed scenarios, and the scalability and generalization ability is verified. The centralized methods cannot work in this test due to mismatched input/output.

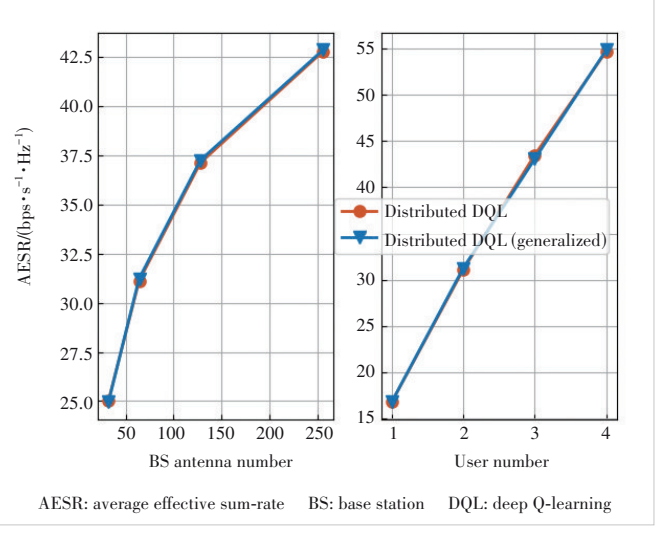
5 Conclusions

In this paper, we investigate multi-user beam tracking in dynamical mmWave scenes, and a multi-agent DQL method under centralized training and distributed execution framework is proposed for online learning. The vanilla DQL is improved in many aspects, such as distributed architecture, rational simplification of training, and state-action-reward designs. Moreover, the proposed method is adaptable to the environment, and is scalable for different BS antenna numbers and user numbers. Simulation results demonstrate the effectiveness of the proposed algorithm.

Appendix

Proof. With the policy π and the initial state s_1, the T-step cumulative reward is defined as:

$$V_\pi^T(s_1) = \mathbb{E}_\pi\left[\frac{1}{T}\sum_{t=1}^{T} r_t | s_1\right] = \sum_{a_1 \in A} \pi(a_1|s_1) \sum_{s_2 \in S} P_{s_1 \to s_2}^{a_1} \times \left(\frac{1}{T} r_{s_1 \to s_2}^{a_1} + \frac{T-1}{T} V_\pi^{T-1}(s_2)\right). \quad (19)$$



AESR: average effective sum-rate BS: base station DQL: deep Q-learning

▲**Figure 6. Scalability of proposed distributed DQL**

According to Eqs. (12) and (13), the state value function in Eq. (19) can be rewritten as:

$$V_\pi^T(s_1) = \sum_{a_1 \in A} \pi(a_1|s_1) \sum_{s_2 \in S} P_{s_1 \to s_2} \times \left(\frac{1}{T} r_{s_2}^{a_1} + \frac{T-1}{T} V_\pi^{T-1}(s_2)\right) = \frac{1}{T}\sum_{a_1 \in A} \pi(a_1|s_1) r_{s_1}^{a_1} + \frac{T-1}{T} \sum_{s_2 \in S} P_{s_1 \to s_2} V_\pi^{T-1}(s_2). \quad (20)$$

The full unrolling of Eq. (20) is given as:

$$V_\pi^T(s_1) = \frac{1}{T}\sum_{a_1 \in A} \pi(a_1|s_1) r_{s_1}^{a_1} + \frac{1}{T}\sum_{t=2}^{T}\sum_{a_t \in A} \pi(a_t|s_t) \sum_{s_t \in S}\prod_{t'=1}^{t-1} P_{s_{t'} \to s_{t'+1}} r_{s_{t'}}^{a_{t'}}. \quad (21)$$

As the state transfer is independent of the action and the state can be independently represented as $s =< s_1, \cdots, s_{T+1} >$. Therefore, the maximization of Eq. (21) with respect to a_t, $\forall t$ can be decomposed into the subproblem:

$$\max_{a_t} V_\pi^T(s) \Leftrightarrow \max_{a_t} r_{s_t}^{a_t}. \quad (22)$$

$$\max_{a_t} V_\pi^T(s) = \max_{a_t} \frac{1}{T}\sum_{t'=1}^{T}\sum_{a_{t'} \in A} \pi(a_{t'}|s_{t'}) r_{s_{t'}}^{a_{t'}} \Leftrightarrow \max_{a_t} \sum_{a_t \in A} \pi(a_t|s_t) r_{s_t}^{a_t} = \max_{a_t} r_{s_t}^{a_t}. \quad (23)$$

In summary, it can be proved that the maximization of Eq. (21) with respect to $\{a_t|\forall t\}$ can be decomposed into T subproblems:

$$\max_{\{a_t|\forall t\}} V_\pi^T(s) \Leftrightarrow \left\{\max_{a_t} r_{s_t}^{a_t} | \forall t\right\}. \quad (24)$$

The equivalence proof of γ-discounted cumulative reward is similar.

References

[1] KIM Y J, CHO Y S. Beam-tracking technique for millimeter-wave cellular systems using subarray structures [J]. IEEE transactions on vehicular technology, 2018, 67(8): 7806 – 7810. DOI: 10.1109/TVT.2018.2834346
[2] DUAN Q Y, KIM T, HUANG H, et al. AoD and AoA tracking with directional sounding beam design for millimeter wave MIMO systems [C]//The 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC). IEEE, 2015: 2271 – 2276. DOI: 10.1109/PIMRC.2015.7343676
[3] ZHANG D Y, LI A, SHIRVANIMOGHADDAM M, et al. Codebook-based train-

ing beam sequence design for millimeter-wave tracking systems [J]. IEEE transactions on wireless communications, 2019, 18(11): 5333 – 5349. DOI: 10.1109/twc.2019.2935731
[4] HUANG H, PENG Y, YANG J, et al. Fast beamforming design via deep learning [J]. IEEE transactions on vehicular technology, 2020, 69(1): 1065 – 1069. DOI: 10.1109/tvt.2019.2949122
[5] HE H T, JIN S, WEN C K, et al. Model-driven deep learning for physical layer communications [J]. IEEE wireless communications, 2019, 26(5): 77 – 83. DOI: 10.1109/mwc.2019.1800447
[6] HE W L, ZHANG C, HUANG Y M, et al. Intelligent optimization of base station array orientations via scenario-specific modeling [J]. IEEE transactions on communications, 2022, 70(3): 2117 – 2130. DOI: 10.1109/TCOMM.2021.3135532
[7] SU J Y, MENG F, LIU S H, et al. Learning to predict and optimize imperfect MIMO system performance: Framework and application [C]//The IEEE Global Communications Conference. IEEE, 2023: 335 – 340. DOI: 10.1109/GLOBECOM48099.2022.10001369
[8] VA V, CHOI J, SHIMIZU T, et al. Inverse multipath fingerprinting for millimeter wave V2I beam alignment [J]. IEEE transactions on vehicular technology, 2018, 67(5): 4042 – 4058. DOI: 10.1109/TVT.2017.2787627
[9] MENG F, LIU S H, HUANG Y M, et al. Learning-aided beam prediction in mmWave MU-MIMO systems for high-speed railway [J]. IEEE transactions on communications, 2022, 70(1): 693 – 706. DOI: 10.1109/TCOMM.2021.3124963
[10] ZHANG J J, HUANG Y M, ZHOU Y, et al. Beam alignment and tracking for millimeter wave communications via bandit learning [J]. IEEE transactions on communications, 2020, 68(9): 5519 – 5533. DOI: 10.1109/TCOMM.2020.2988256
[11] ZHANG J J, HUANG Y M, WANG J H, et al. Intelligent interactive beam training for millimeter wave communications [J]. IEEE transactions on wireless communications, 2021, 20(3): 2034 – 2048. DOI: 10.1109/TWC.2020.3038787
[12] SUTTON R S, BARTO A G. Reinforcement learning: an introduction [M]. Cambridge, USA: MIT Press. 2018
[13] MENG F, CHEN P, WU L N, et al. Power allocation in multi-user cellular networks: Deep reinforcement learning approaches [J]. IEEE transactions on wireless communications, 2020, 19(10): 6255 – 6267. DOI: 10.1109/TWC.2020.3001736

Biographies

MENG Fan (mengfan@pmlabs. com. cn) received his BS degree from the School of Electronic Engineering, University of Electronic Science and Technology of China in 2015 and the PhD degree with the School of Information Science and Engineering, Southeast University, China in 2020. He is now working in the Purple Mountain Laboratories. His research mainly focuses on applying machine learning techniques in the wireless communication systems. His research interests include machine learning in physical layer, model- and data-driven design, beamforming, beam alignment and tracking, and AI-enhanced positioning.

HUANG Yongming received his BS and MS degrees from Nanjing University, China in 2000 and 2003, respectively, and PhD degree in electrical engineering from Southeast University, China in 2007. Since March 2007, he has been with the School of Information Science and Engineering, Southeast University, where he is currently a full professor. During 2008 – 2009, he visited the Signal Processing Lab, Royal Institute of Technology, Sweden. He has authored or coauthored more than 200 peer-reviewed papers, and holds more than 80 invention patents. He has submitted 20 technical contributions to IEEE standards. His research interests include intelligent 5G/6G mobile communications and millimeter wave wireless communications.

LU Zhaohua received his PhD degree from Tianjin University, China in 2006. He is currently a senior wireless communication system research expert at ZTE Corporation and has long been engaged in the field of wireless communication system design and the key technologies of the physical layer. He has many technical contributions, papers, and patents in interference mitigation in the MIMO field.

XIAO Huahua received his MS degree in computer software and theories from Sun Yat-Sen University, China. He is currently a senior engineer in the field of antenna algorithm pre-research with ZTE Corporation. He has applied for more than 150 patents in the multi-antenna field home and abroad.