Dynamic Beam-Based Random Access Scheme for M2M Communications in Massive MIMO Systems

Kan Zheng, Senior Member, IEEE, Haojun Yang, Member, IEEE, Xiong Xiong, Jie Mei, Member, IEEE, Lu Hou, Member, IEEE, and Kuan Zhang, Member, IEEE

Abstract—Internet of things, supported by machine-to-machine (M2M) communications, is one of the most important applications for the 6th generation (6G) systems. A major challenge facing 6G is enabling a massive number of M2M devices to access networks in a timely manner. Therefore, this article exploits the spatial selectivity of massive multi-input multi-output (MIMO) to reduce the collision issue when massive M2M devices initiate random access simultaneously. In particular, a beam-based random access protocol is first proposed to make efficient use of the limited uplink resources for massive M2M devices. To address the non-uniform distribution of M2M devices in the space and time dimensions, an Markov decision process (MDP) problem with the objective of minimizing the average access delay is then formulated. Next, we present a dynamic beam-based access scheme based on the double deep Q network (DDQN) algorithm to solve the optimal policy. Finally, simulations are conducted to demonstrate the effectiveness of the proposed scheme including the model training and random access performance.

Index Terms—Internet of things (IoT), random access, machine-to-machine (M2M) communications, and reinforcement learning (RL).

I. INTRODUCTION

RECENTLY, various types of applications in Internet of things (IoT) have rapidly grown, especially for industrial field. A high number of end devices are deployed, and adopt IoT networks to transmit sensing data and control information in a timely manner. To well support IoT applications, the 3GPP has developed machine-type communications (MTCs) and ultra-massive machine-type communications (umMTCs) as typical use cases in 5th generation (5G) and 6th generation (6G) networks [1], [2]. Consequently, the cellular-enabled IoT networks are receiving considerable attention from the industry and academia for operating diverse IoT applications.

The current mobile cellular networks, however, may not be able to meet the requirements of either MTCs or umMTCs, because they were originally designed to support human-type communications. In contrast to human-to-human (H2H) communications, machine-to-machine (M2M) communications typically involve a massive number of devices with low-data rate transmitting small payloads sporadically [3]. Moreover, MTC devices usually are not uniformly deployed across geographical areas, due to various IoT applications existing in networks. Based on the unique features and location distribution of MTC traffic, the current cellular networks need to be greatly improved to efficiently handle M2M communications. In general, M2M devices are first required to establish air interface connections in cellular networks before data transmission. The corresponding access requests are transmitted in an uncoordinated manner over the random access channels (RACHs). Once an MTC device has been granted to access, it is scheduled to use specific radio resources over which data transmission takes place in a deterministic manner. However, when a massive number of MTC devices are deployed in the networks, the RACHs with the existing access schemes become seriously overloaded result in heavy congestion. To this end, one of key issues is how to guarantee the timely access requirements of massive M2M devices.

Some recent advances have been conducted on dealing with the previous challenge [4], [5]. First of all, the number of M2M devices that initiate random access at the same time can be limited by a pre-defined access probability in access class barring (ACB), thereby reducing the risk of collision [6], [7], [8], [9], [10]. Then, by delaying the access time of the M2M devices with delay-insensitive applications, all random access requests can be scheduled into different time slots, which can relieve physical random access channel (PRACH) congestion [11], [12], [13]. Meanwhile, the base station can dynamically allocate radio resources to M2M devices based on the conditions of PRACHs and network loads [14], [15]. Additionally, the base station can fully control random access by adjusting the paging timing of M2M devices [5]. A dedicated time slot of random access also can be assigned to reduce the interference on H2H communications [6], [16]. Finally, the code-expanded access is another good method [17]. For instance, in order to reduce preamble

Manuscript received 22 September 2022; revised 1 May 2023; accepted 29 May 2023. Date of publication 15 June 2023; date of current version 14 November 2023. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62001301, and in part by the Zhejiang Provincial Natural Science Foundation of China under Grant QZFH0100025. The review of this article was coordinated by Dr. Giuseppe Araniti. 

(Corresponding author: Kan Zheng.)

Kan Zheng and Jie Mei are with the College of Electrical Engineering and Computer Sciences, Ningbo University, Ningbo, Zhejiang 315211, China (e-mail: zhengkan@nbu.edu.cn; meijie@nbu.edu.cn).

Haojun Yang is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: haojun.yang@uwaterloo.ca).

Xiong Xiong and Lu Hou are with the Intelligent Computing and Communications (IC2) Lab, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: ajodfaj@bupt.edu.cn; houlu8674@bupt.edu.cn).

Kuan Zhang is with the Department of Electrical and Computer Engineering, University of Nebraska-Lincoln, Omaha, NE 68182 USA (e-mail: kuan.zhang@unl.edu).

Digital Object Identifier 10.1109/TVT.2023.3266660
proposes an improved We propose a beam-based random access protocol for Based on the Markov decision process (MDP), an opti-
In order to demonstrate the effectiveness of the proposed s = equal sectors. The M2M devices are uniformly beams generated by a base station proposes an number of M2M devices initiate the access simultaneously collision problem can be effectively alleviated, when a massive devices in the different beams. Owing to the spatial orthogonality preambles and radio resources can also be reused by the M2M To further improve the utilization efficiency, the same set of beams, mutually isolated spatial areas in specific directions is feasible for the cellular network, which is more beneficial to let a certain beam only serve the devices within its corresponding area.

To address the access problem of massive M2M devices located at different geographic locations, this article proposes a beam-based dynamic random access scheme based on massive MIMO technique. In particular, some M2M devices located in the coverage of a certain beam can be divided into the same group, and can use the same set of preambles and radio resources. To further improve the utilization efficiency, the same set of preambles and radio resources can also be reused by the M2M devices in the different beams. Owing to the spatial orthogonality of different beams, the mutual interference among the M2M devices in different groups is extremely small. Therefore, the collision problem can be effectively alleviated, when a massive number of M2M devices initiate the access simultaneously.

The main contributions of this article include:

- We propose a beam-based random access protocol for massive M2M devices. A group of beams with unequal beamwidth are applied to address the impact of the non-uniform distribution of M2M devices in space and time on random access process.
- Based on the Markov decision process (MDP), an optimization problem with the objective of minimizing the average access delay is formulated. To solve the optimal policy, we develop a dynamic beam-based random access scheme based on the double deep Q network (DDQN) algorithm.
  - In order to demonstrate the effectiveness of the proposed scheme, the model training performance and random access performance are compared with other reference random access schemes by the simulation results.

The rest of this article is organized as follows. In Section II, a typical scenario and model are first presented. Then, a beam-based random access protocol for M2M devices is described in detail. Next, a dynamic beam-based random access problem is formulated as MDP model in Section III. Based on the deep deterministic policy gradient (DDPG), Section IV proposes an algorithm to find the optimal policy. Moreover, simulations are carried out to demonstrate the performance of the proposed dynamic scheme in Section V. Finally, the conclusions are presented in Section VI.

II. SYSTEM MODEL AND RANDOM ACCESS PROTOCOL

A. Scenario Description

As shown in Fig. 1, \( N_g \) beams generated by a base station with massive MIMO are adopted to communicate with M2M devices located in the different areas. Furthermore, a set of \( N_p \) preambles are used for random access within a single beam, but might be reused between different beams. In general, the spatial distribution of M2M devices is usually not uniform and is also highly dynamic. This results in different arrival rates of random access both in space dimension and time dimension. For purposes of analysis, the coverage area of a base station is divided into \( N_s \) equal sectors. The M2M devices are uniformly distributed in a single sector, and their random access requests follow a Poisson distribution, where the arrival rates of access can be denoted as \( \lambda_j, j \in \{1, 2, \ldots, N_s\} \). Generally, the number of sectors and beams can be the same or not. For the sake of illustration, they are set to be equal in this article, i.e., \( N_s = N_g \).

B. Description of Beam-Based Random Access Protocol

In this article, a beam-based random access protocol for massive M2M devices is proposed, and it mainly consists of the following four steps.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
1) **Step 1 – Preamble Transmission:** Similarly with the regular random access protocol in Long-Term Evolution (LTE) systems, in our proposed protocol, a M2M device randomly selects and sends a preamble to the base station on the first available time slot of random access.

2) **Step 2 – Random Access Response:** After detecting the preamble sent by the M2M device, the base station sends the corresponding random access response (RAR) in the downlink, including the index of the detected preamble, the indicator of uplink resource and other information. When more than one M2M device send the same preamble, the base station will send the same RAR to them. However, it cannot tell whether a collision happens now, which has to be addressed in the subsequent steps.

3) **Step 3 – Connection Request:** Based on the received RAR, the M2M device sends an radio resource control (RRC) message to the base station. When the base station is equipped with massive MIMO, it can distinguish various RRC messages from the M2M devices located at different areas by multiple directional beams. Specifically, since most of the useful signals radiated by the antennas are concentrated within a beam, the beam has the remarkable spatial selectivity. Thus, the M2M devices located in the different beam coverage can be served separately, thereby reducing mutual interference. That is to say, for those M2M devices that send the same preamble and receive the same RAR, the base station can also distinguish the RRC messages through exploiting the spatial selectivity of different beams. Therefore, the beam-based random access scheme can efficiently improve the successful probability of random access.

4) **Step 4 – Contention Resolution:** After the base station successfully demodulates the RRC messages, it sends a contention resolution message back to the corresponding M2M device. Only the M2M device whose RRC message has been successfully demodulated can confirm the contention resolution message and complete the random access procedure, while other M2M devices fail to access.

C. **Formulation of Beam-Based Random Access Protocol**

In this section, we elaborate the proposed protocol mathematically. As shown in Fig. 2, each beam can be characterized by the maximum gain direction of $\phi_i$ and beamwidth of $\Theta_i$. Generally, a beam is symmetric about its maximum gain direction, and most of its gain is concentrated within the beamwidth. In order to fully cover the service area of the base station, these beams need to be adjacent to each other with their beamwidth. In the other words, the maximum gain direction and beamwidth of each beam should meet the following constraints, i.e.,

\[
\begin{align*}
\text{C1: } & \sum_{i=1}^{N_g} \Theta_i = 2\pi, \\
\text{C2: } & \phi_{(i \mod N_g)+1} = [\phi_i + (\Theta_{(i \mod N_g)+1} + \Theta_i)/2] \mod 2\pi, \\
\text{C3: } & 0 \leq \phi_i, \Theta_i < 2\pi, i \in \{1, 2, \ldots, N_g\},
\end{align*}
\]

where $\mod(\cdot)$ represents the modulo operation.

Without loss of generality, let $(\theta_0, d_0)$ be the location of the interested M2M device in polar coordinate, where $\theta_0$ and $d_0$ represent the angle and distance between the device and the base station, respectively. It is regarded as having successfully connected to the network only after the base station demodulated the RRC message through the beam $i^*$. The serving beam $i^*$ here refers to the one that the polar coordinate $(\theta_0, d_0)$ of the interested M2M device is converged within its bandwidth range.
Assume that $s_0 \leq s < 2\pi \vee 0 \leq s_0 \leq \theta^H_i$, $\theta^L_i \leq s < \theta^H_i$, $\theta^L_i \leq s_0 \leq \theta^H_i$, $\theta^L_i \leq \theta^H_i$, and $\phi^H_i$ are the lower and upper boundaries of the serving beam $i^\ast$. Here we take the periodicity of the angle into account, i.e., $\theta^L_i = (\phi^L_i - \Theta_i/2) \mod 2\pi$, $\theta^H_i = (\phi^H_i + \Theta_i/2) \mod 2\pi$.

For the interested M2M device, the signals with RRC messages sent by the other $N_i$ M2M devices served by those beams different from the beam $i^\ast$ are regarded as interference, because the same uplink radio resource is used. The received power of the interference at the serving beam $i^\ast$ for the interested device can be expressed as (in dBm)

$$I_{i^\ast} = 10\log_{10} \left[ \sum_{q=1}^{N_i} 10^{R_i^q(\phi^L_i, a^q_i)/10} \right], q \in \{1, 2, \ldots, N_i\},$$

where $R_i^q(\phi^L_i, d)$ denotes the received power of the signal with the RRC message at the $i^\ast$-th beam of the base station. Then, it can be generally calculated in dBm by

$$R_i^q(\phi^L_i, d) = P_i + G_i + G_e + 20 \log_{10}(f_i(\theta)) - PL(d) + \chi, i \in \{1, 2, \ldots, N_g\},$$

where $P_i$ represents the transmit power of the M2M device, while $G_i$ and $G_e$ are the gains of transmit and receive antennas, respectively. $PL(d)$ is the path loss between the M2M device and base station, and $\chi$ is the shadow fading obeying a Gaussian distribution with zero mean and variance $\sigma^2$. Lastly, $f_i(\theta)$ represents the gain of the $i$-th beam in the direction $\theta$, and Appendix A gives the calculation of the gain.

Finally, when the power difference in logarithmic value between the RRC message and the interference is larger than the required demodulation threshold $\Gamma$, the base station can successfully decode the RRC message of the interested M2M device, i.e.,

$$R_i^q(\phi^L_i, d_0) - I_{i^\ast} > \Gamma.$$  

To sum up, the dynamic beam-based random access protocol is proposed in Algorithm 1. In the next section, we mainly focus on the study of the dynamic random access part.

### III. Problem Formulation of Dynamic Beam-Based Random Access

According to the proposed beam-based random access process, the base station can dynamically adjust the direction and beamwidth of the beam based on the non-uniform distribution of M2M devices. Thus, the number of devices located in the coverage of different beams can be effectively balanced, thereby alleviating the collision in random access process. In this section, the dynamic random access problem is formulated as a MDP model, and its state, action, and reward are defined as follows.

### A. State

The state $s \in S$ includes the number of M2M devices that initiate random access in each sector area, i.e.,

$$S = \{s \mid s = (n_1, n_2, \ldots, n_{N_s})\},$$

where $n_j, j \in \{1, 2, \ldots, N_s\}$ represents the number of M2M devices accessing in the $j$-th sector.

### B. Action

The action $a \in A$ defines how the base station to adjust the direction and width of the beams, which can be expressed by

$$A = \{a \mid a = (\phi_1, \Theta_1, \phi_2, \Theta_2, \ldots, \phi_{N_e}, \Theta_{N_e})\},$$

where $\phi_i$ and $\Theta_i$, $i \in \{1, 2, \ldots, N_e\}$ are the maximum gain direction and beamwidth of the $i$-th beam, respectively. It is noted that the direction and beamwidth of each beam strictly follow the requirements in (1). For the simplicity of illustration, the action only includes several typical selections with the given number of beams, i.e.,

$$A = \{a_1, a_2, \ldots, a_{|A|}\},$$

where $|A|$ represents the size of the action space, and $a_1, a_2$ are pre-defined typical actions. Assume that the number of beams $N_g$ is equal to 6, and the action space $A$ contains 3 typical actions as shown in Table I, then the beam gain of each action is given in Fig. 3.

**Remark 1:** When one action is selected, e.g., $a_1$, the signal gains received by the base station from different directions may vary considerably, since multiple beams have different beamwidths. In other words, the base station have better signal gain but less coverage using one of beams, and vice versa. The beams used in actions $a_1$ and $a_2$ are obtained by rotating the beams of action $a_1$ counterclockwise by $\pi/3$ and $2\pi/3$, respectively. Therefore, the beams corresponding to different actions may better serve M2M devices located in various directions.

### C. State Transition

In this paper, the state transition is simply expressed as $s \xrightarrow{a} s'$, which means that the current state $s$ transfers to the new state $s'$ after executing action $a$. To better reflect the dynamic feature of random access process, the MDP model of this paper adopts episodic task based state transition process, including a series of relatively independent episodes [29]. Each episode contains the following state transition steps, i.e.,

1) **Phase 1:** At the beginning of each episode, the M2M devices in each sector initiate the random access process at the slot $t = 0$ with the given arrival rates $\lambda_j$, $j \in \{1, 2, \ldots, N_e\}$. Correspondingly, the initial state is represented by $s = S_0$.

2) **Phase 2:** Based on the number of devices, the base station adjusts the direction and beamwidth according to the policy $\pi$, namely $a = \pi(s)$. Then, the base station operates the proposed beam-based random access protocol as described in Section II.

---

1Here a typical group of beams with unequal beamwidth is presented only as an illustration of beam-based access schemes, since the beam design is beyond the scope of this article.
Additionally, the M2M devices that fail to access will continue to initiate an access request at the next slot $t'$. At this point, the state of the model is updated as $s' = S(t')$.

3) Phase 3: If most of the M2M devices initiating the random access process in the initial state $S(0)$ have successfully connected the network before updating the new state $S(t')$, the current episode terminates. This also means that only a few M2M devices fail to access the network, i.e., $\sum_{j=1}^{N_s} n'_j \leq n_T$, where $n_T$ is the performance threshold. Otherwise, update the slot $t \leftarrow t'$ and state $s \leftarrow s'$, and repeat the Phase 2 of the state transition.

Finally, when an episode is terminated, the model transfers to a new independent episode and performs Phase 1–3 as described above.

D. Reward

The reward $r(s, a)$ is a real-valued function of the state $s$ and the action $a$, reflecting the benefits and costs of the current instantaneous decision $a = \pi(s)$. Therefore, the reward definition has to be consistent with the optimization objective. Since the model of this article relies on instantaneous value rather than statistical one, the successful probability of access can not be directly used as the optimization objective. To this end, an alternative optimization is required. Generally, the policy $\pi$ can be regarded as working well, if the access requests of all M2M devices can be performed successfully and quickly. Meanwhile, it also implies that this policy $\pi$ can improve the successful probability of access. Therefore, we choose the access delay as the performance metric of random access rather than the probability. In conclusion, the goal of the model is to minimize the average delay of random access.

According to the above state transition process, the model in a certain epoch transfers from the current state $s = (n_1, n_2, \ldots, n_{N_s})$ to a new state $s' = (n'_1, n'_2, \ldots, n'_{N_s})$ by taking the action $a$, where $n'_j, j \in \{1, 2, \ldots, N_s\}$, as defined in Formula (6), stands for the number of M2M devices that have not yet accessed the network in the $j$-th sector in the new state $s'$. Therefore, by adding up $n'_j, j \in \{1, 2, \ldots, N_s\}$ of the new state $s'$ can derive the total number of M2M devices that fail to access the network, i.e.,

$$\hat{r}(s, a) = \sum_{j=1}^{N_s} n'_j,$$  \hspace{1cm} (9)

Since these M2M devices may try to access again in the new state $s'$, each of them spends one more time slot compared with those M2M devices that have successfully accessed in the state $s$. Thus, $\hat{r}(s, a)$ can be the proxy of the “access delay” during the state transition $s \xrightarrow{a} s'$, it also is the number of time slots used for random access. By summing up $\hat{r}(s, a)$ of all states in an episode, the “total access delay” of M2M devices can be obtained by

$$\hat{G}_\tau = \sum_{t=0}^{T_\tau} \hat{r} \left( S(t), A(t) \right) = \sum_{m=1}^{T_\tau} \sum_{t=0}^{T_\tau} 1_{m \notin S(t)},$$  \hspace{1cm} (10)

where $T_\tau$ represents the termination moment of the episode $\tau$, while $S(t)$ and $A(t)$ represent the state and action at the time
slot \( t \) of random access, respectively. \( |S_{[0]}| \) is the number of M2M devices that initiate random access requests in the initial state, namely \( |S_{[0]}| = \sum_{j=1}^{N_j} n_j \). Let \( 1_{m\notin S_{[0]}} \) indicate whether the \( m \)-th M2M device in the initial state has successfully accessed the network at the time slot \( t \). If the M2M device has not yet connected to the network, i.e., \( 1_{m\notin S_{[0]}} = 1 \); otherwise, \( 1_{m\notin S_{[0]}} = 0 \). Thus, the access delay of the \( m \)-th M2M device can be calculated by \( \sum_{t=0}^{T} 1_{d \notin S_{[0]}} \). Next, the average access delay can be given by

\[
G_\tau = \hat{G}_\tau / (|S_{[0]}| - \hat{r}(S_{[T]}, A_{[T]})), \tag{11}
\]

where \( \hat{r}(S_{[T]}, A_{[T]}) < n_\tau \).

Based on the above definition, the reward finally can be expressed by

\[
r(s, a) = -\hat{r}(s, a). \tag{12}
\]

Here, a negative sign is adopted, this is because the reward definition of MDP must satisfy the following assumption, i.e., the larger the reward value, the better the action. Accordingly, the cumulative reward from time slot \( t \) to the termination moment of episode \( \tau \) can be given by

\[
G_{\tau(t)} = \sum_{k=t}^{T} \gamma^{k-t} r(S(k), A(k)) = -\sum_{k=t}^{T} \gamma^{k-t} \hat{r}(S(k), A(k)), \tag{13}
\]

where \( \gamma \) is the so-called discount factor. Since the formulated model in this section is based on the episodic task, then \( \gamma = 1 \). In conclusion, maximizing the the expected cumulative reward \( \mathbb{E}_{||G_{\tau(t)}||} \) is equivalent to minimizing the average “access delay” of M2M devices, which is consistent with our original optimization goal.

### IV. Dynamic Random Access Scheme With DDQN

To find the optimal policy \( \pi^* \) for maximizing the expected cumulative reward \( \mathbb{E}_{||G_{\tau(t)}||} \), we propose an dynamic random access scheme based on the DDQN algorithm to solve the MDP model. The DDQN algorithm is a value-based reinforcement learning technique that utilizes a neural network to approximate the value function \( Q_\pi(s, a) \), i.e.,

\[
Q(s, a; \theta) \approx Q_\pi(s, a) = \mathbb{E}_\pi \left[ \sum_{\tau=t}^{\infty} \gamma^{\tau-t} r(S(k), A(k)) | S(t) = s, A(t) = a \right], \tag{14}
\]

where \( \theta \) is the weight parameter of the neural network \( Q(s, a; \theta) \). \( Q(s, a; \theta) \) is also known as the prediction network, whose input is a \( N_c \)-dimension vector representing the current state \( s \) and output is a \( |A| \)-dimension vector representing the action value corresponding to each action \( a \). Then, by using the action value function, the corresponding policy can be directly obtained by

\[
\pi(s) = \arg\max_{a \in A} Q_\pi(s, a) = \arg\max_{a \in A} Q(s, a; \theta).
\]

#### Algorithm 2: Dynamic Random Access Scheme for Non-Uniform Grouping Case Based on DDQN

**Input:**

1. Initialize the prediction network \( Q(s, a; \theta) \) with random weights \( \theta \);
2. Initialize the target network \( Q(s, a; \theta^-) \) with weights \( \theta^- = \theta \);
3. Initialize the replay memory \( D \);
4. Initialize \( \epsilon = 1 \).

**Procedure:**

5. For \( \tau \leftarrow 1, \ldots, \infty \) do

   6. Initialize the state \( s = S_{[0]} \) according to the arrival rate \( \lambda_j, j \in \{1, 2, \ldots, N_j\} \);

   7. For \( t \leftarrow 1, \ldots, \infty \) do

      8. Generate a random number \( p, 0 < p \leq 1 \); /* \( \epsilon \)-greedy */

      9. \( a \leftarrow \begin{cases} \arg\max_{a \in A} Q(s, a; \theta) & 0 \leq p < \epsilon \\ \arg\max_{a \in A} Q(s, a; \theta^-) & \epsilon \leq p < 1 \end{cases} \)

      10. Take the action \( a \) and perform state transition \( s \leftarrow (s, a, r(s, a), s') \) into the replay memory \( D \);

     11. Random sample a batch of experiences \( \tilde{D} \) from \( D \);

     12. Calculate \( L(\theta) \); /* according to (15) */

      13. Update weights \( \theta \) by optimizer;

     14. If \( \epsilon > \epsilon_{\text{min}} \) then /* exponential decay of \( \epsilon \) */

      15. \( \epsilon \leftarrow \epsilon - \epsilon_{\text{decay}} \);

      16. End

     17. If \( t \mod n_\theta = 0 \) then /* every \( n_\theta \) steps */

      18. \( \theta^- \leftarrow \theta \);

     19. End

     20. If \( \hat{r}(s, a) \leq n_T \) then /* episode end */

     21. \( T_\tau = t \times / \)

     22. Break;

     23. End

     24. \( s \leftarrow s' \);

     25. End

   26. End

**Output:** Obtain the optimal policy \( \pi^* \) based on (16).

In order to estimate the action value function accurately, we have to use another target network, namely \( Q(s, a; \theta^-) \), which has the same structure with the prediction network [30]. Then, the mean squared error (MSE) function can be defined by

\[
L(\theta) = \frac{1}{|D|} \sum_{e \in D} \left[ (r(s, a) - Q(s, a; \theta) + \gamma Q(s', \arg\max_{a'} Q(s', a'; \theta^-))^2 \right], \tag{15}
\]
where \( e = (s, a, r(s, a), s') \) is an experience sample including the state \( s \), action \( a \), new state \( s' \) and the resulting reward \( r(s, a) \) in the state transition process of \( s \xrightarrow{a,r(s,a)} s' \). \( \tilde{D} \) is a set of \(|\tilde{D}| \) experiences, which are selected from the replay memory \( D \) with a first-in-first-out (FIFO) queue. The DDQN algorithm updates the weight parameters of the neural network iteratively with the objective of minimizing the loss function \( L(\theta) \) until the optimal action value function is estimated, i.e., \( Q = \arg\max_{s,a} \pi^*(s,a) \approx Q_\pi(s,a, \theta^*) \).

Afterwards, the optimal policy can be written as
\[
\pi^*(s) = \arg\max_{a \in A} Q_\pi^*(s,a) \\
\approx \arg\max_{a \in A} Q(s,a, \theta^*). \tag{16}
\]

Algorithm 2 proposes a dynamic random access scheme for non-uniform grouping case based on DDQN. First of all, the related model parameters including the so-called prediction network, target network, and replay memory are initialized. Then, the iteration of a episode is carried out. At the beginning of each episode, the initial state \( s = S(0) \) is generated according to the arrival rates of access in each sector, i.e., \( \lambda_j, j \in \{1, 2, \ldots, N_s\} \).

Subsequently, during each episode, the action \( a \) is selected based on the \( \varepsilon \)-greedy rule in the iteration of each access slot. In particular, the greedy policy is selected with the probability of \( 1 - \varepsilon \), while a random action is chosen with the probability \( \varepsilon \). The \( \varepsilon \) decays exponentially with the rate of \( \varepsilon_{\text{decay}} \) at each iteration until it decays to the minimum value of \( \varepsilon_{\text{min}} \). The \( \varepsilon \)-greedy rule can efficiently prevent the algorithm from falling into the local optimal solution. Next, the algorithm executes the action \( a \) and performs the state transition. Meanwhile, the generated experience \( e \) is saved in the replay memory \( D \). Finally, a batch of \( \tilde{D} \) experiences are randomly selected from the replay memory \( D \) to calculate the loss function \( L(\theta) \), and the weight parameter \( \theta \) of the prediction network is updated. It is noticed that only after \( n_\theta \) iterations, the weight parameter \( \theta^- \) of the target network is updated with the weight parameter \( \theta \) of the prediction network, namely \( \theta^- \leftarrow \theta \). When the algorithm converges, the optimal action value function \( Q_\pi = \arg\max_{s,a} Q(s,a, \pi^*(s,a)) \approx Q_\pi(s,a, \theta^*) \) is obtained as well as the optimal policy \( \pi^*(s) \approx \arg\max_{a \in A} Q(s,a, \theta^*) \).

The main notations used throughout the article have been summarized in Table II for ease of understanding.

### V. Simulation Results and Analysis

#### A. Simulation Configuration

In the simulations, a base station with massive MIMO is divided into \( N_s = 6 \) sectors. The sectors are bounded by the directions of \( \{j-1\pi/3 - \pi/6, (j-1)\pi/3 + \pi/6\}, j \in \{1, 2, \ldots, N_s\} \). The distance between the M2M devices and the base station varies randomly from 0 to 10 km. Moreover, the base station generates \( N_\theta = 6 \) beams, and adjusts the direction and width of the beams according to the action space \( A \) as shown in Table I. To reflect the non-uniform feature of device distribution, one of sectors is set as the high-density area with the high arrival rate of access, such as \( \lambda_1 \approx \lambda_{\text{high}} \), while other sectors are set with \( \lambda_j = \lambda_{\text{low}}, j = 2, 3, \ldots, N_s \). The total arrival rate is defined as \( \lambda = \sum_{j=1}^{N_s} \lambda_j \). Finally, we also define a distribution ratio between the high-density and low-density sectors as \( \rho = \frac{\lambda_{\text{high}}}{(N_s-1)\lambda_{\text{low}}} \) to evaluate the performance under various arrival rate. The main parameters of simulations are shown in Table III [27], [31].

In the DDQN algorithm, both the prediction and the target networks use a two-layer fully connected neural network, in which each fully connected layer contains 64 neurons with the rectified linear unit (ReLU) activation function. The size of

| Parameter | Description |
|-----------|-------------|
| \( N_p \) | Number of preambles |
| \( N_s \) | Number of sectors |
| \( N_\theta \) | Number of beams |
| \( \Theta \) | Beamwidth of the \( i \)-th beam |
| \( \phi \) | Direction of the \( i \)-th beam |
| \( R_i(\theta, d) \) | Received power at the \( i \)-th beam of the signal, which is sent from the M2M device that locates at \( (\theta, d) \) |
| \( P_i \) | Transmit power of M2M device |
| \( G_i \) | Transmit antenna gain of M2M device |
| \( G_s \) | Receive antenna gain of base station |
| \( \sigma \) | Standard variance of shadowing |
| \( \Gamma \) | Demodulation threshold |
| \( S \) | State space |
| \( s \) | State space \( \pi/s, \pi/2, \ldots, \pi/N_s \) |
| \( S_{(t)} \) | State at the decision epoch \( t \) |
| \( n_j \) | Number of M2M devices in the \( j \)-th sector |
| \( A \) | Action space \( A \) |
| \( a \) | Action |
| \( A(\pi) \) | Action at the decision epoch \( t \) |
| \( r(s,a) \) | Reward function of state \( s \) and action \( a \) |
| \( \pi(s) \) | Policy |
| \( \gamma \) | Discount factor |
| \( G_c(t) \) | Cumulative reward from time slot \( t \) to the end of episode \( \tau \) |
| * | A symbol with the superscript \( \ast \) denotes the symbol at the next decision epoch |

| Parameter | Description |
|-----------|-------------|
| \( Q(s,a;\theta) \) | Prediction network |
| \( Q(s,a;\theta^-) \) | Target network |
| \( \epsilon \) | Parameter of \( \varepsilon \)-greedy rule, which decays exponentially from \( \epsilon_1 \) to \( \epsilon_{\text{min}} \) with the rate \( \epsilon_{\text{decay}} \) |
| \( \alpha \) | Learning rate |
| \( L(\theta) \) | Loss function |
| \( \tau \) | Experience, also called sample |
| \( D \) | Replay memory \( D \) with the size \(|\tilde{D}|\) |
| \( \tilde{D} \) | Batch of samples for training with the size \(|\tilde{D}|\) |
| \( n_\theta \) | Weight update period of the target networks \( \theta^- \) |
| * | A symbol with the superscript \( \ast \) denotes the symbol with the optimal value |

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
the replay memory $|D|$ is set to 1200. During each iteration, the algorithm selects 64 experiences per batch from the replay memory to calculate the loss function $L(\theta)$. In addition, the Adam optimizer with a learning rate of $\alpha = 0.001$ is applied to update the weight parameters $\theta$ of the prediction network. Finally, the parameters of the $\varepsilon$-greedy rule are set as $\varepsilon_{\text{min}} = 0.01$ and $\varepsilon_{\text{decay}} = 0.99$, respectively. The weight update period of the target network $\theta^{-}$ is set as $n_{\theta} = 16$.

### B. Performance Analysis

1) **Training Performance**: Fig. 4 shows the training performance of the proposed DDQN-based scheme after iterating $2 \times 10^4$ episodes in terms of the loss function and average action value. All solid lines present the results after exponential moving average (EMA) processing with a weight of 0.99, while the original data is given in the shadow parts. The values of loss function under different arrival rates and distribution ratios are first presented in Fig. 4(a). Each curve shows a rapid downward trend during the first $10^3$ episodes, and then is stabilized within a certain range. Particularly, the loss function value in the case of $\lambda = 300$ ends up fluctuating between 60–80, while the loss function value under $\lambda = 150$ remains stable at about 30. After about the $2 \times 10^3$ episode, each loss function almost converges.

Subsequently, Fig. 4(b) gives the performance of average action value. Here the average action value refers to the average output of the prediction network in each episode, i.e.,

$$\frac{1}{T} \sum_{t=1}^{T} \sum_{a \in \mathcal{A}} Q(S(t), S(t); \theta).$$

Due to the different trajectories of state transition, the instantaneous action value is not comparable, and it cannot accurately reflect the convergence of the algorithm. Thus, instead of the instantaneous action value, the average one is chosen as the performance metric.

In the first $10^3$ episodes of Fig. 4(b), each curve starts at 0 and shows a rapid downward trend. This is because the weight parameters of the prediction network are initialized with the standard Glorot, resulting in a small output value at the beginning of the training. Then, as the weight parameters are updated rapidly during the training progress, the average action value of the prediction network changes dynamically. After the $10^3$ episodes, the curves under various $\lambda$ show different trends. Specifically, in the case of $\lambda = 150$, three curves first rise slowly and then fluctuate between $-250$ and $-400$. On the other hand, in the case of $\lambda = 300$, the curves with $\rho = 20$ or 5 still decline slowly, and gradually stabilize at around $-500$. Furthermore, when $\lambda = 300$ and $\rho = 2$, the corresponding curve eventually stabilizes between $-600$ and $-700$. Lastly, all of them almost reach a state of convergence after $10^4$ episodes.

It is noted that the average action value in the case of $\lambda = 300$ is much lower than that of $\lambda = 150$. This is because the larger arrival rate $\lambda$, the greater chance of collision, which leads to a larger access delay and a lower action value. According to Fig. 4, it is evident that the proposed scheme with the DDQN algorithm becomes convergent fast and steadily.

2) **Delay Performance**: The proposed scheme based on multiple beams with unequal beamwidth (hereinafter referred to as DDQN-BU) is compared with the following random access schemes, i.e.,

| Parameter | Description | Value |
|-----------|-------------|-------|
| $\lambda$ | Total arrival rate of M2M devices | 150, 300 |
| $\rho$ | Distribution ratio | 2, 5, 20 |
| $d$ | Distance between M2M devices and base station | 0 - 10 km |
| $N_p$ | Number of preambles | 48 |
| $N_s$ | Number of sectors | 6 |
| $N_b$ | Number of beams | 6 |
| $P_t$ | Transmit power of M2M device | 23 dBm |
| $G_t$ | Transmit antenna gain of M2M device | 0 dBi |
| $G_r$ | Receive antenna gain of base station | 18 dB |
| $\sigma$ | Standard variance of shadowing | 8 dB |
| $PL(d)$ | Path loss | $120.9 + 37.6 \log_{10}(d)$ |
| $\Gamma$ | Demodulation threshold | $-110$ dBm |
Fig. 4. Training performance of the proposed DDQN-based dynamic random access algorithm. (a) Loss function value $L(\theta)$ versus episode. (b) Average action value versus episode.

- **Static scheme based on multiple beams with equal beamwidth (Static-BE):** In this scheme, all beams have the equal beamwidth, i.e., $\Theta_i = 2\pi/N_g$. Thus, there is no need to make selection during the random access [27].
- **Random scheme based on multiple beams with unequal beamwidth (Random-BU):** This scheme randomly selects one of the beams in Fig. 3 to serve all M2M devices, regardless of their spatial distribution.

Table IV gives the average “access delay” when different random access schemes are applied under various arrival rates $\lambda$ and distribution ratios $\rho$. It can be seen that the DDQN-BU scheme has the lower average access delay than others in all cases. At the same time, its relative gain to Static-BE scheme increases with the higher arrival rate and the larger distribution ratio. For example, in the case of $\lambda = 300$ and $\rho = 20$, the DDQN-BU scheme has the lowest average access delay of 4.75 slots, while the average access delay of the Static-BE scheme and Random-BU scheme are 5.96 and 6.24 slots, respectively. Therefore, the DDQN-BU scheme allows M2M devices to access the network faster by dynamically adjusting the beam direction and beamwidth, and owns optimal access performance.

When the distribution ratio $\rho$ is first fixed, the average access delay of each scheme becomes larger with the increase of the arrival rate $\lambda$. We can further find that the gain of the DDQN-BU scheme compared with the Static-BE scheme also becomes more intuitive with the increase of the arrival rate $\lambda$. For example, at $\rho = 20$, the gain of DDQN-BU at $\lambda = 150$ and $\lambda = 300$ are 14.7% and 23.9%, respectively. This shows that the larger arrival rate $\lambda$ is, the more intuitive optimization effect of the DDQN-BU scheme on the access performance achieves.

Next, we fix the arrival rate $\lambda$ but increase the distribution ratio $\rho$. The average access delay of each scheme increases when the non-uniform distribution of devices becomes stronger, namely the larger $\rho$. This phenomenon is expected because M2M devices collide more frequently as the number of devices increases within a given sector, and thus the average access delay eventually rises. More importantly, the gain of the DDQN-BU scheme also becomes larger with the increase of the distribution ratio $\rho$. For example, at $\rho = 20$, the gain improvements are 4.1%, 10.4% and 14.7% for $\rho = 2$, $\rho = 5$ and $\rho = 20$, respectively. This shows that the stronger non-uniform distribution, the more intuitive optimization effect of the DDQN-BU scheme on access performance. Compared with the reference schemes, the DDQN-BU scheme can better handle the access problem in the case of non-uniform distribution of M2M devices.

Finally, we further analyze the cumulative distribution function (CDF) performance of access delay. The CDF curves of access delay have the same trend under different arrival rates $\lambda$ and distribution ratios $\rho$, thus here only the case of $\lambda = 300$ and $\rho = 20$ is taken as an example for the sake of analysis. As shown in Fig. 5, when the access delay is small,
all curves increase rapidly, such an increase becomes slower as the access delay becomes larger. It means that most M2M devices have a low access delay, and only a few devices have a high access delay. For example, when the DDQN-BU scheme is adopted, about 66.4%, 91.5% and 99.2% of M2M devices access the network successfully within 5, 10 and 15 time slots, respectively. Furthermore, the CDF curve of the DDQN-BU scheme rises the fastest among all ones. This shows that the DDQN-BU scheme dynamically adjusts the beam direction and beamwidth according to the distribution of devices, which can efficiently reduce the access delay and improve the overall access performance.

VI. CONCLUSION

Many IoT applications have the non-uniform distribution of devices in the space dimension, and the dynamic service arrival rate in the time dimension. As a result, this article proposed a beam-based dynamic random access scheme in massive MIMO systems. Specifically, the base station can dynamically adjust a group of beams with different bandwidths based on the distribution of devices, which can efficiently balance the number of M2M devices served by different beams, reducing collisions during the access process. We also formulated this dynamic problem as an MDP model, and proposed a dynamic beam-based random access scheme based on the DDQN algorithm to obtain the optimal solution. Lastly, the performance of the proposed scheme was validated by simulations. The proposed scheme were proved to converge rapidly and stably. After the model is converged, the optimal policy can be determined by using the trained prediction neural network. The simulation results show that the proposed scheme performs much better than the other two reference schemes in terms of the average access delay. In the next steps, we plan to conduct further study on beam-based dynamic random access schemes in massive MIMO systems. For instance, one of the interesting topics will be how to design the optimal beam pattern and the corresponding action for the DRL-based scheme. Moreover, the implementation complexity of DRL-based scheme has to be reduced without much sacrifice of performances. More importantly, there are a few issues such as inter-cell interference and device mobility that could have an impact on how well M2M access techniques in massive MIMO systems perform in real deployment. Therefore, it is important to conduct in-depth research on deployment-related issues with the proposed DRL-based scheme.

APPENDIX A

CALCULATION OF BEAM GAIN

For the sake of analysis, all beams are generated by the uniform linear array (ULA), and the beamwidth is set as the half-power beamwidth (HPBW) [32], i.e.,

\[ \Theta_i = \pi - \cos^{-1} \left( \frac{1.391\lambda_a}{\pi d_a} \right), \quad i \in \{1, 2, \ldots, N_g\}, \quad (17) \]

where \( \lambda_a \) represents the wavelength, \( d_a \) represents the spacing of array elements, \( N_a \) represents the number of array elements, and \( d_a/\lambda_a = 1/4 \). Then, the number of array elements required to generate the beam with the beamwidth in (17) can be calculated by

\[ N_a = \left\lceil \frac{1.391\lambda_a}{\pi d_a \cos \left( \frac{\pi}{2} - \frac{\Theta_i}{2} \right)} \right\rceil. \quad (18) \]

Finally, the gain of the beam is given by

\[ f_i(\theta) = \frac{\sin \left( \frac{N_a}{k} |kd_a \cos(\theta - \phi_i) + \beta| \right)}{N_a \sin \left( \frac{1}{2} |kd_a \cos(\theta - \phi_i) + \beta| \right)}, \quad (19) \]

where \( k \) is equal to 2\( \pi/\lambda_a \), and \( \beta \) represents the phase difference between array elements. The maximum gain of the beam is set to be 1 for the sake of normalization, namely \( \max f_i(\theta) = 1 \).

REFERENCES

[1] N. Varsier, L.-A. Dufrene, M. Dumay, Q. Lampin, and J. Schwoerer, “A 5G new radio for balanced and mixed IoT use cases: Challenges and key enablers in FR1 band,” IEEE Commun. Mag., vol. 59, no. 4, pp. 82–87, Apr. 2021.
[2] H. Tataria, M. Shafi, M. Dohler, and S. Sun, “Six critical challenges for 6G wireless systems: A summary and some solutions,” IEEE Veh. Technol. Mag., vol. 17, no. 1, pp. 16–26, Mar. 2022.
[3] M. El-Tanab and W. Hamouda, “An overview of uplink access techniques in machine-type communications,” IEEE Netw., vol. 35, no. 3, pp. 246–251, May/Jun. 2022.
[4] A. Laya, L. Alonso, and J. Alonso-Zarate, “Is the random access channel of LTE and LTE-A suitable for M2M communications? A survey of alternatives,” IEEE Commun. Surv. Tut., vol. 16, no. 1, pp. 4–16, Jan.–Mar. 2014.
[5] F. Ghavimi and H.-H. Chen, “M2M communications in 3GPP LTE/LTE-A networks: Architectures, service requirements, challenges, and applications,” IEEE Commun. Surv. Tut., vol. 17, no. 2, pp. 525–549, Apr.–Jun. 2015.

[6] 5th Generation Partnership Project (3GPP), “Study on RAN improvements for machine-type communications,” 3GPP, Sophia Antipolis, France, Tech. Rep. 37.868 v11.0.0, Sep. 2011.

[7] S.-Y. Lien, T.-H. Liu, C.-Y. Kao, and K.-C. Chen, “Cooperative access class barring for machine-to-machine communications,” IEEE Trans. Wireless Commun., vol. 11, no. 1, pp. 27–32, Jan. 2012.

[8] Z. Wang and W. V. S. Wong, “Optimal access class barring for stationary machine-type communication devices with timing advance information,” IEEE Trans. Wireless Commun., vol. 14, no. 10, pp. 5374–5387, Oct. 2015.

[9] S. Duan, V. Shah-Mansouri, Z. Wang, and W. V. S. Wong, “D-ACB: Adaptive congestion control algorithm for bursty M2M traffic in LTE networks,” IEEE Trans. Veh. Technol., vol. 65, no. 12, pp. 9874–9876, Dec. 2016.

[10] H. Jin, W. T. Toor, B. C. Jung, and J.-B. Seo, “Recursive pseudo-Bayesian access class barring for M2M communications in LTE systems,” IEEE Trans. Veh. Technol., vol. 66, no. 9, pp. 8595–8599, Sep. 2017.

[11] X. Yang, A. Fapojuwo, and E. Egbogah, “Performance analysis and parameter optimization of random access backoff algorithm in LTE,” in Proc. IEEE Veh. Technol. Conf., 2012, pp. 1–5.

[12] J. Chen, R.-G. Cheng, O. Agbodike, and Y.-S. Lyu, “A dynamic backoff window scheme for machine-type communications in cyber-physical systems,” IEEE Access, vol. 8, pp. 31045–31056, 2020.

[13] H. D. Althumali, M. Othman, N. K. Noordin, and Z. M. Hanapi, “Dynamic backoff collision resolution for massive M2M random access in cellular IoT networks,” IEEE Access, vol. 8, pp. 201345–201359, 2020.

[14] W. Li, Q. Du, L. Liu, P. Ren, Y. Wang, and L. Sun, “Dynamic allocation of each resource for clustered M2M communications in LTE networks,” in Proc. IEEE Int. Conf. Identification Inf. Knowl. Internet Things, 2015, pp. 140–145.

[15] H.-Y. Hwang, S.-M. Oh, C. Lee, J. H. Kim, and J. Shin, “Dynamic RACH preamble allocation scheme,” in Proc. IEEE Int. Conf. Inf. Commun. Technol. Convergence, 2015, pp. 770–772.

[16] N. Shahin, R. Ali, and Y.-T. Kim, “Hybrid slotted-CSMA/CA-TDMA for efficient massive registration of IoT devices,” IEEE Access, vol. 6, pp. 18366–18382, 2018.

[17] N. K. Pratas, H. Thomsen, C. Stefanovic, and P. Popovski, “Code-expanded random access for machine-type communications,” in Proc. IEEE Globecom Workshops, 2012, pp. 1681–1686.

[18] J. K. Kim, S. Lee, and M. Y. Chung, “Efficient random-access scheme for massive connectivity in 3GPP low-cost machine-type communications,” IEEE Trans. Veh. Technol., vol. 66, no. 7, pp. 6280–6290, Jul. 2017.

[19] L. Liu and W. Yu, “Massive connectivity with massive MIMO—Part I: Device activity detection and channel estimation,” IEEE Trans. Signal Process., vol. 66, no. 11, pp. 2933–2946, Jun. 2018.

[20] H. Han, Y. Li, W. Zhai, and L. Qian, “A grant-free random access scheme for M2M communication in massive MIMO systems,” IEEE Internet Things J., vol. 7, no. 4, pp. 3602–3613, Apr. 2020.

[21] F. Rusek et al., “Scaling up MIMO: Opportunities and challenges with very large arrays,” IEEE Signal Process. Mag., vol. 30, no. 1, pp. 40–60, Jan. 2013.

[22] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, “Massive MIMO for next generation wireless systems,” IEEE Commun. Mag., vol. 52, no. 2, pp. 186–195, Feb. 2014.

[23] Y. Zhang, M. Alrabeiah, and A. Alkhateeb, “Reinforcement learning of beam codebooks in millimeter wave and terahertz MIMO systems,” IEEE Trans. Commun., vol. 70, no. 2, pp. 908–919, Feb. 2022.

[24] Q. Wang, K. Feng, X. Li, and S. Jin, “PrecoderNet: Hybrid beamforming for millimeter wave systems with deep reinforcement learning,” IEEE Wireless Commun. Lett., vol. 9, no. 10, pp. 1677–1681, Oct. 2020.

[25] I. Ahmed, M. K. Shahid, and T. Faisal, “Deep reinforcement learning based beam selection for hybrid beamforming and user grouping in massive mimo-noma system,” IEEE Access, vol. 10, pp. 89519–89533, 2022.

[26] M. Forzi, A. R. Sharafat, and M. Benni, “Fast MIMO beamforming via deep reinforcement learning for high mobility mmWave connectivity,” IEEE J. Sel. Areas Commun., vol. 40, no. 1, pp. 127–142, Jan. 2022.

[27] X. Xiong, L. Hou, and L. Zhao, “A group-based massive multiple access scheme in cellular M2M networks,” Elsevier Comput. Commun., vol. 121, pp. 44–49, May 2018.

[28] E. D. Carvalho, E. Bjornson, J. H. Sorensen, P. Popovski, and E. G. Larsson, “Random access protocols for massive MIMO,” IEEE Commun. Mag., vol. 55, no. 5, pp. 216–222, May 2017.
Lu Hou (Member, IEEE) received the B.S. and Ph.D. degrees from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2014, and 2019, respectively. He is currently an Assistant Professor with Intelligent Computing and Communications (IC²) Lab, BUPT. His research interests include Internet of Things (IoT), vehicular networks, and artificial intelligence.

Kuan Zhang (Member, IEEE) received the bachelor degree in communication engineering, and master degree in computer applied technology from Northeastern University, Shenyang, China, in 2009 and 2011, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2016. June 2016 to 2017, he was a Postdoctoral Fellow with BBCR. Since 2017, he has been an Assistant Professor with the Department of Electrical and Computer Engineering, University of Nebraska-Lincoln, Lincoln, NE, USA. He was the recipient of the Best Paper Awards of IEEE WCNC 2013, Securecomm 2016, and IEEE ICC 2020.