Huang *et al. J Wireless Com Network* (2020) 2020:250
https://doi.org/10.1186/s13638-020-01872-5

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Reinforcement learning-based hybrid spectrum resource allocation scheme for the high load of URLLC services

Qian Huang[1,2*], Xianzhong Xie[1] and Mohamed Cheriet[2]

*Correspondence: huangq@stu.cqupt.edu.cn
[1] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
Full list of author information is available at the end of the article

Abstract

Ultra-reliable and low-latency communication (URLLC) in mobile networks is still one of the core solutions that require thorough research in 5G and beyond. With the vigorous development of various emerging URLLC technologies, resource shortages will soon occur even in mmWave cells with rich spectrum resources. As a result of the large radio resource space of mmWave, traditional real-time resource scheduling decisions can cause serious delays. Consequently, we investigate a delay minimization problem with the spectrum and power constraints in the mmWave hybrid access network. To reduce the delay caused by high load and radio resource shortage, a hybrid spectrum and power resource allocation scheme based on reinforcement learning (RL) is proposed. We compress the state space and the action space by temporarily dumping and decomposing the action. The multipath deep neural network and policy gradient method are used, respectively, as the approximater and update method of the parameterized policy. The experimental results reveal that the RL-based hybrid spectrum and the power resource allocation scheme eventually converged after a limited number of iterative learnings. Compared with other schemes, the RL-based scheme can effectively guarantee the URLLC delay constraint when the load does not exceed 130%.

Keywords: Ultra-reliable and low-latency communication, Radio resource allocation, mmWave, Hybrid spectrum, Reinforcement learning, Multipath deep neural network

1 Introduction

The rapid development of mobile cellular communication technology will completely reshape and change the world [1]. As one of the most important foreseen application scenarios of the fifth-generation cellular wireless network (5G) and beyond, ultra-reliable and low-latency communication (URLLC) has attracted increasing attention [2, 3]. Examples include the Industrial Internet of Things (IIoT), the Tactile Internet, the Internet of Vehicles (IoV), industrial automation, interactive telemedicine, emergency rescue, and so on [4]. The real-time and reliability of data transmission services in these applications are directly related to operational safety, production efficiency, and Quality of Service (QoS).

It is a challenge to implement URLLC as a result of its two mutually exclusive features of high reliability and low latency [5, 6]. Also, the actual service requirements in different scenarios have different constraints on latency and reliability. For example, it is mentioned in Release 16 of the 3rd-Generation Partnership Project (3GPP) that in an IoV scenario, for data packets with a length of 5220 bytes, a block error probability (BEP) of less than 10^{-5} and an air interface delay of up to 3 ms need to be guaranteed; in a factory automation scenarios with periodic, deterministic business, for data packets with a length of 32 bytes, a BEP of less than 10^{-6} and an air interface delay of up to 1 ms need to be guaranteed [7].

As a key technology of the physical layer in the access network, radio resource allocation is the focus of URLLC. Especially with a resources shortage (such as spectrum and power), efficient URLLC resource allocation methods are particularly important. Current technologies, such as power amplification and millimeter-wave (mmWave), can temporarily alleviate the dilemma caused by the shortage of radio resources [8–10]. But for the rapid growth of communication data, even mmWave with abundant spectrum resources still faces a shortage of spectrum.

Also, factors such as sporadic idle resources, irregular or sudden real-time communication requests, and unpredictable behavior of high-speed mobile users exist in wireless communication networks. These make the resource and channel requirements often change according to spatial or temporal factors. But the traditional wireless communication resource management typically depends on the determination allocation method, which cannot predict and deal well with these problems [11]. Compared to Long-Term Evolution (LTE) cells, the number of spectrum resource units and the amount of traffic processed simultaneously in mmWave cells have increased dramatically [12, 13]. If traditional resource allocation is used for mmWave resources, a huge resource mapping form will increase the mapping delay.

Reinforcement learning (RL) is a type of machine learning that explicitly considers the entire problem of the interaction between goal-oriented agents and uncertain environments. Resource allocation in wireless networks can be viewed as a collection of highly repetitive decisions, for which RL is considered appropriate. These repeated decision data provide RL with much training data. Therefore, it is possible to express a goal, which is difficult to optimize directly, as a reward function without precise modeling.

Most of the current research on RL methods in wireless resource allocation is based on the action-value function to approximate the optimal action selection [14–16]. In [14], the authors present a software-defined satellite-terrestrial network framework, which jointly considered the networking, caching, and computing resources in satellite-terrestrial networks. A deep Q-learning was used to approximate the optimal expected utility of each resource. The authors in [15] proposed a multiagent Q-learning resource management algorithm, which reduced the packet loss rate and power oscillation in the device-to-device (D2D) network. In [16], a Q-learning cooperative power allocation algorithm was proposed to increase the capacity of two-tier dense heterogeneous networks (HetNets).

A model-free deep-RL power allocation framework was proposed in [17] for URLLC in the downlink of an orthogonal frequency division multiple access (OFDMA) system. The authors formulate the URLLC optimization problem as minimizing power under

constraints of delay and reliability. The framework can support higher reliability and lower latency without the actual underlying model. The authors in [18] studied the channel allocation of URLLC services in multiuser multichannel wireless networks and proposed a risk-sensitive Q-learning algorithm. In the risk-sensitive Q-learning algorithm, the agent needs to consider the total expected reward and QoS requirement violation probability. In [19], a joint optimization method of spectrum and power based on efficient transfer actor-critic learning is proposed to ensure the URLLC constraints of IoV and maximizing network capacity. The authors of [20] studied the radio resource optimization of the age of information aware in IoV. To adapt to the high-speed mobility of IoV, the original Markov decision process (MDP) was decomposed into a series of vehicle user equipment-pair MDPs.

However, there have been few studies on radio resource allocation for high-load URLLC services. Motivated by radio resource optimization in the high load of URLLC services, we are concerned with an RL scheme of the hybrid spectrum and power resource allocation that uses policy parameterization and the policy gradient [21] to approximate the optimal allocation.

The policy gradient is a parameterized policy update method for RL. In the policy gradient, the action selection does not depend on the action-value function but on the updated direction of the policy parameters. And the desired parameterized form of the policy can be introduced into the RL system as a priori knowledge. Compared with the optimization based on the action-value function, the parameter change in the policy gradient is smoother. Therefore, the policy gradient has a stronger guarantee of convergence.

1.1 Contribution

In this paper, we focus on the radio resource allocation optimization problem of the mmWave access network with a high load of URLLC data. The main contributions of this work are as per the following.

- For the shortage of radio resources in mobile networks, a hybrid spectrum and power resource allocation scheme is proposed for variable resource profile URLLC data communication. The mmWave bands are divided into licensed and unlicensed spectrums. URLLC data preferentially occupy the unlicensed spectrum and share the licensed spectrum with ordinary data.
- To back this up, we employ a Greedy hybrid spectrum and power allocation scheme that takes into account the maximization of resource utilization and dynamic delay constraints. In addition, we present an RL-based hybrid spectrum and power allocation scheme, which can effectively guarantee the URLLC delay constraint in a high-load network environment.
- In the communication simulation of URLLC data with variable resource configuration, two extreme cases of preferential transmission of short or long data have occurred, both of which will increase the delay. For this problem, the design of the reward function takes into account the time overhead of all data received by the roadside base station (BS). This enables the RL-based scheme to get rid of work-conserving and to foresee, that is, to reserve resources for short data that may arrive

soon. The URLLC delay of the whole system is thereby optimized while ensuring resource utilization.

- A large amount of URLLC data makes the state space and the action space larger. Therefore, to reduce the state space, we limit the length of the URLLC cache queue and temporarily dump the overflowed URLLC data. To reduce the action space, we decouple the radio resource allocation process from the time step.
- To obtain a smoother policy update, we use the policy gradient method to update the parameterized policy. And to speed up the parameter learning, we use the deep neural network (DNN) to approximate the best policy. Because of the limited computing capability of the roadside BS, it is difficult for the RL agent deployed in it to support the URLLC delay constraint. A multipath DNN that can dynamically exit according to delay constraints is used to approximate the action preference function to reduce unnecessary computational load and training time.
- Finally, we analyzed and evaluated the performance of the RL-based scheme and the Greedy scheme and compared them with other resource allocation schemes. We analyzed mainly the convergence of the RL-based scheme and compared the delay and reliability performance of each scheme under different loads. The experimental results show that the delay performance of the RL-based scheme and the Greedy scheme is better than that of other schemes. Even under strict URLLC delay constraints, the RL-based scheme can still maintain reliability under a high load.

1.2 Organization

The remainder of this paper is organized as follows: Section 2 describes the system model of the hybrid spectrum and power allocation model for URLLC in mmWave cell. The two radio resource allocation schemes, Greedy and RL-based with time-variant resource state transition, are given in Sect. 3. Section 3 also presents the policy gradient method based on multipath DNN. Section 4 presents the experimental results and analysis, including the convergence, delay, and reliability of different resource allocation schemes at various loads. Finally, the conclusions are summarized in Sect. 5.

2 System model

In the complex road environment shown in Fig. 1, the dedicated resources originally in the roadside BS reserved for URLLC cannot carry such a large amount of critical data communication well. In this scenario, the delay of URLLC data will increase by insufficient communication resources, and even the loss rate will increase by timeout.

The hollow arrows in Fig. 2 indicate the processing of URLLC data in the BS. The URLLC data, such as a road safety state of emergency in Fig. 1, arrive in an independent and identically distributed (i.i.d.) manner to the roadside BS. The road safety state of emergency will be stored in the URLLC cache queue Q after it is received by the roadside BS.

Normally, because of the high priority of URLLC traffic, queue Q will not overflow. That is, the total amount of received URLLC data $NUM_U \leq |Q|$. But if a special scenario in Fig. 1 is encountered, $NUM_U > |Q|$ results from the surge in URLLC data, which is the scenario with which this paper is concerned. The overflowed part $NUM_U - |Q|$

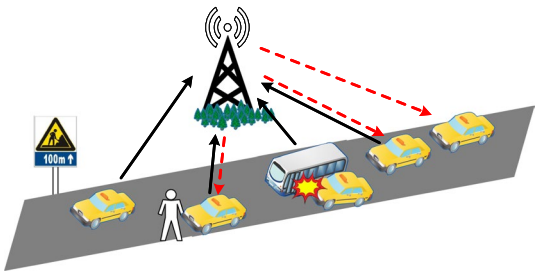

Fig. 1 A scenario example of complex road emergency safety. A roadside BS needs to handle multiple road emergency tasks at the same time

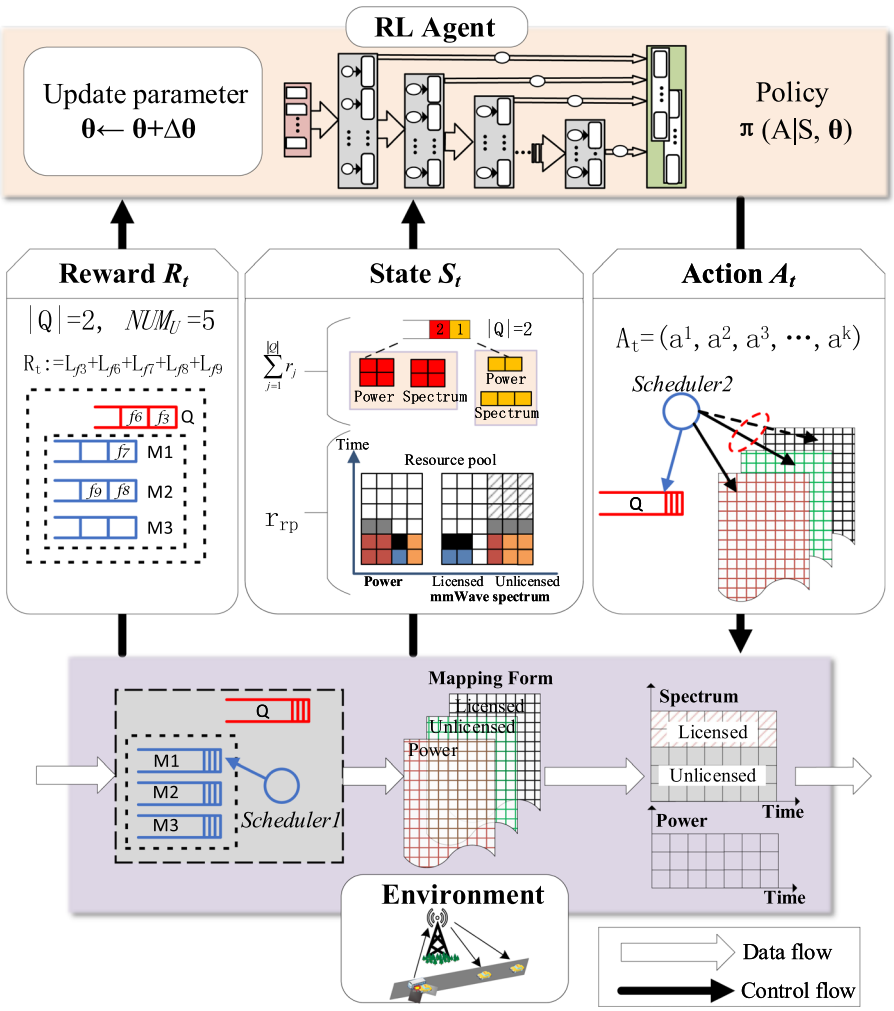

Fig. 2 The RL-based hybrid spectrum and power resource allocation model for URLLC. The overall framework of figure is an RL framework. There are control procedures (control flow) and data processing procedures (data flow) in this framework. The bottom of figure is the data flow in the external environment. Each step of the data flow is specifically completed according to the RL agent at the top of figure. The external environment processes data information into Status and Rewards and feeds back to the Agent. The Agent gets the Action after learning with the DNN and returns it to the environment

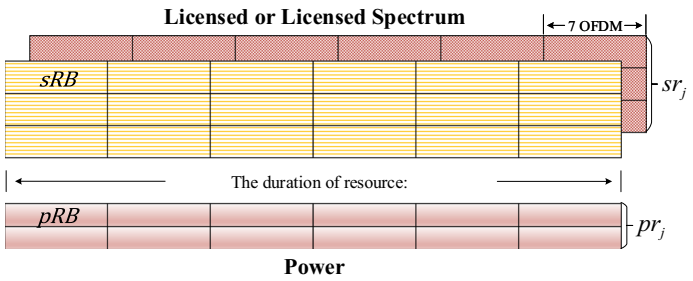

Fig. 3 The resource profile for URLLC data *j*. The size of the spectrum and power resources required for URLLC data *j*. The sRB and pRB represent a spectrum RB and a power RB, respectively. The length of an sRB or pRB is 7 OFDM symbols. The total spectrum resource length required for URLLC data is L_j, and the width is sr_j. The length of the total power resource required for URLLC data is L_j, and the width is pr_j

will be temporarily stored in the non-URLLC queue (M1, M2, and M3). When queue Q becomes non-full, the URLLC data temporarily stored in M1, M2, or M3 will be transferred to queue Q. The purpose of dumping the overflow part to the non-URLLC queue, instead of increasing $|Q|$, is to limit the size of the state space.

There are two schedulers in the BS: *Scheduler*1 and *Scheduler*2. *Scheduler*1 selects URLLC data from other queues to queue Q, and *Scheduler*2 allocates radio resources for the URLLC data in queue Q according to a certain policy.

Compared with the 15KHz of LTE, the subcarrier spacing supported by 5G New Radio (NR) is more diverse. And different subcarrier spacing can be used by different communication services. According to the definition of 3GPP 38.211, a minimum schedulable resource block (RB) includes 12 consecutive subcarriers in the frequency domain. Given the large bandwidth of mmWave, we use the 480 kHz subcarrier spacing configuration to design the RB in resource scheduling. The length of RB in the time domain is not defined by 3GPP 38.211. So, for compatibility with the LTE system, we define the length of RB as 7 OFDM symbols in the time domain.

We assume that the number of RBs required for each URLLC datum is known by the roadside BS. Without loss of generality, we consider spectrum and power radio resources. The resource profile of URLLC data j is given by

$$r_j = \{n_j \times sRB, m_j \times pRB\}, \tag{1}$$

where sRB and pRB represent the required spectrum RB and power RB, respectively, and n_j and m_j are the numbers of sRB and pRB.

To improve the spectrum utilization rate and reduce the action space, the mmWave band is divided into the licensed spectrum and the unlicensed spectrum. So, there are three resource mapping forms: power, licensed spectrum, and unlicensed spectrum. A more detailed resource profile for URLLC data j is shown in Fig. 3.

The resource length required for data j in the time, frequency, and power domains is L_j, sr_j, and pr_j, respectively. Therefore, another formula of Eq. (1) is

$$r_j = \{L_j, pr_j, sr_j, \delta_j\}, \tag{2}$$

where the binary variable δ_j is used to mark the licensed spectrum or the unlicensed spectrum.

$$\delta_j = \begin{cases} 0, \text{ unlicensed spectrum} \\ 1, \text{ licensed spectrum} \end{cases}. \tag{3}$$

This means that a URLLC datum does not allow simultaneous use of the licensed and unlicensed spectrums. Assume that the resource r_j allocated to the URLLC datum j is atomic. That is, once the radio resources are allocated to datum j until it is received, the allocated resources are not preemptible.

The RL agent in Fig. 2 resides in the roadside BS to avoid additional transmission delays to the core network. In this paper, the time interval for each resource allocation decision is defined as time step t. At each time step t, the RL agent selects one or more pending URLLC data to send according to the centralized view of the current radio resource state. Therefore, the roadside BS has to be a smart node with computational analysis capabilities.

It is assumed that all URLLC data have the same transmission weight. The resource size required for each URLLC datum varies. To avoid the resource allocation method favoring URLLC data with a large L_j, we use the L_j to normalize the total transmission delay $D_j = L_j + D_{j\_q}$ of data$j$, where D_{j_q} is the queueing delay. The optimization goal of this paper is to minimize the average delay $\mathbb{E}[D]$ of URLLC traffic in the system,

$$\mathbb{E}[D] = \mathbb{E}\left[\sum_{j=1}^{NUM_U} \frac{D_j}{L_j}\right] = \mathbb{E}\left[\sum_{j=1}^{NUM_U} \frac{L_j + D_{j\_q}}{L_j}\right]. \tag{4}$$

To make the resource allocation method as far-sighted as possible, we also considered the delay constraint of the overflow part $NUM_U - |Q|$. So, the average delay calculated in Eq. (4) applies to all NUM_U URLLC data and not just the first $|Q|$.

When $L_j + D_{j\_q} > D_{MAX}$, the data j will be dropped to ensure the low latency constraint of URLLC. D_{MAX} is the maximum transmission delay acceptable for URLLC. It is assumed that the mobile terminal can decode without errors. Transmission reliability can be expressed by loss probability

$$D_{loss} = \frac{N_{loss}}{N_{total}} \times 100\%, \tag{5}$$

where N_{loss} is the number of dropped URLLC data, and N_{total} is the total number of URLLC data received by the BS.

3 Design of hybrid spectrum and power allocation

Reinforcement learning (RL) is an MDP that includes the agent and environment. The RL agent is a learning and decision-making machine. All external things that interact with the agent are called the environment. The agent learns interactively with the environment and exchanges three types of control information: reward R, state S, and action A (the black solid arrows in Fig. 2). Therefore, Fig. 2 depicts in detail the process of data flow and control flow in the RL-based hybrid spectrum and power resource allocation method.

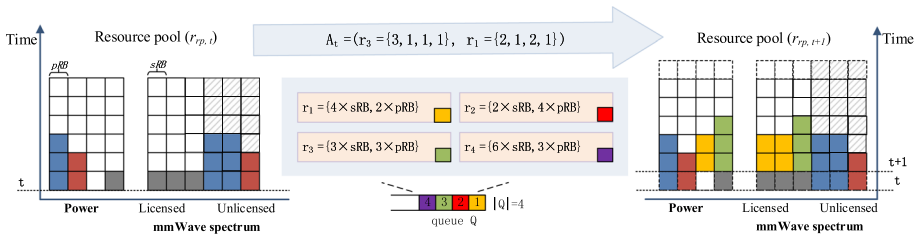

Fig. 4 An example of a state transition. The change of radio resource status from time step t to $t+1$. Each small square represents an RB. The two-dimensional coordinates on the left side of figure represent the state of the resource pool at time step t. The two-dimensional coordinates on the right side of figure represent the state of the resource pool at time step $t+1$. Some RBs have already been occupied by different data (the colored squares). For example, the blue squares indicate that a URLLC data communication still occupies 3 *pRBs* and 6 unlicensed *sRBs*. There are 4 URLLC data resource profiles recorded in queue Q. For example, the first data (yellow) in queue Q require 4 sRBs and 2 pRBs. At time step $t+1$, one or more data in queue Q will be allocated radio resources and sent to the mobile user. Since RBs before each time step have been used for data communication and released, the lowermost RBs (dotted squares) of the resource pool are released at time step $t+1$ and added to the top

More specifically, at each discrete time step t, the RL agent observes a certain state $S_t \in \mathcal{S}$ of the environment and obtains a state-action probability (policy π). Then, it makes an optimal action $A_t \subseteq \mathcal{A}(s)$ according to policy π. At the next time step, as a result of A_t, the agent receives a numerical reward, $R_{t+1} \in \mathcal{R} \subset \mathbb{R}$, and observes a new state S_{t+1}.

According to the MDP property, the state transition probabilities and rewards depend only on S_t and A_t. Thus, this paper gives two resource allocation schemes: a Greedy hybrid spectrum and power allocation and an RL-based hybrid spectrum and power allocation. Since the state space $\mathcal{S}$ and the action space $\mathcal{A}(s)$ are large, the multipath DNN optimal approximator is used to train policy π. The details are in part C of this section.

3.1 Time-variant resource state transition dynamics

Figure 4 is an example of a time-variant state transition of the spectrum and power resource pool. Each small square represents an RB. Ideally, the duration of allocable resources in the resource pool is unlimited. Nevertheless, to limit the length of an episode in RL, it is desirable to have a fixed state representation. Therefore, we consider the duration of the resource pool image as L_{rp}. To reduce the state space $\mathcal{S}$, only the first $|Q|$ URLLC data are considered in the state function. The current system state in time step t is expressed as

$$S_t = \{r_{rp\_t}, r_1, r_2, \ldots, r_{|Q|}\}, \tag{6}$$

where the matrix $r_{rp\_t} = [\cdot]_{L_{rp}\times 3}$ represents the current power and spectrum information of the resource pool, and r_j is given by Eq. (1). The three-column vectors in r_{rp_t} represent power, licensed spectrum, and unlicensed spectrum, respectively.

Figure 4 visually shows the change of radio resource status from time step t to $t+1$. As can be seen from the resource pool image r_{rp_t} (left side of Fig. 4), some RBs have already been occupied by different data (the colored squares). For example, the blue squares indicate that a URLLC data communication still occupies 3 *pRBs* and 6 unlicensed *sRBs*.

At this time, there are 4 URLLC data resource profiles recorded in queue Q, as shown in Fig. 4. For example, the first data (yellow) in queue Q require 4 *sRBs* and 2 *pRBs*, and the second data (red) require 2 *sRBs* and 4 *pRBs*.

Therefore, at time step $t+1$, one or more data in queue Q will be allocated radio resources and sent to the mobile user. Suppose the RL agent schedules only one datum in queue Q at each time step and determines whether to use the licensed or unlicensed spectrum. The size of the action space $|\mathcal{A}(s)|$ is up to 2^{Q+1}, which is a large learning space. $|\mathcal{A}(s)|$ can be reduced by decoupling the resource allocation process from the time step. Specifically, the agent performs the resource allocation of multiple data in a single time step. Therefore, the action space is defined as

$$\mathcal{A}(s) = \left\{a_1(\delta_1), a_2(\delta_2), \ldots, a_{|Q|}\left(\delta_{|Q|}\right)\right\}. \tag{7}$$

The $a_j\left(\delta_j\right)$ is the operation of allocating resources for data j, where δ_j indicates that the licensed or unlicensed spectrum is selected as in Eq. (3). Action A_t is a subset of $\mathcal{A}(s)$, which means that multiple URLLC data in queue Q are selected at time step t.

$$A_t = \left(r_{q_1}, r_{q_2}, \ldots, r_{q_k}\right), \tag{8}$$

where $q_i \in [1, |Q|]$ is an integer, $q_i \neq q_j$, and k $\leq |Q|$.

For example, $A_t = (r_3, r_1)$ in Fig. 4 indicates that the RL agent selects the third and first data (green and yellow) of queue Q in sequence. Combining this with Eq. (2), we can see that the time domain, power domain, and frequency domain width of the third URLLC data (green) are (3, 1, 1), and the licensed spectrum is selected.

A_t is ended when the agent selects an inappropriate datum. As shown in Fig. 4, after allocating resources to the third datum, the second and fourth (red and purple) are both inappropriate data. There are not enough resources for them to start from $t+1$. Therefore, the condition for adding $a_j\left(\delta_j\right)$ to $A_t = \left(r_{q_1}, r_{q_2}, \ldots, r_{q_{k'}}\right)$ is

$$\begin{cases} j \notin [q_1, q_{k'}] \\ NUM_{sRB\_t}^{\delta_j=0or1} \geq sr_j \\ NUM_{pRB\_t} \geq pr_j \end{cases}, \tag{9}$$

where NUM_{pRB_t} and $NUM_{sRB\_t}^{\delta_j=0or1}$ are the available power and spectrum (licensed or unlicensed) of time step t, respectively.

Since RBs before each time step have been used for data communication and released, the resource pool will roll forward. More intuitively, as shown on the right side of Fig. 4, the lowermost RBs (dotted squares) of the resource pool are released at time step t + 1 and added to the top.

Algorithm 1: Greedy hybrid spectrum and power allocation

Initialize the resource pool image $r_{rp\_1} = [0]_{L_{rp}\times 3}$
Initialize the queue Q with the random resource profile
Initialize: $A_1 = (0)$, $N_{loss} \leftarrow 0$
For $t = 1, 2, 3, \cdots$
  Release NUM_{pRB_t-1}, $NUM^0_{sRB\_t-1}$, $NUM^1_{sRB\_t-1}$
  For each $j = 1, 2, \cdots, |Q|$
    If $L_j + D_{j\_q} > D_{MAX}$
      Discard the data that exceed the delay constraint, and update r_j with random resource profile in the cache queue M_i.
      $L_j + D_{j\_q} \leftarrow \text{New}(L_j + D_{j\_q})$
      Adjust the position of data j, so that $L_{j-1} + D_{j-1\_q} > L_j + D_{j\_q} > L_{j+1} + D_{j+1\_q}$
      $N_{loss} \leftarrow N_{loss} + 1$
    End if
  End for
  If power and spectrum resources are available: $NUM_{pRB\_t} \times \left(NUM^0_{sRB\_t} + NUM^1_{sRB\_t}\right) \neq 0$
    For j from 1 to $|Q|$, and $a_j(\delta_j) \notin A_t$
      If $NUM_{pRB\_t} \geq pr_j$, $NUM^0_{sRB\_t} \geq sr_j$
        Allocate unlicensed spectrum: $A_t \leftarrow a_j(0)$
        $NUM_{pRB\_t} \leftarrow NUM_{pRB\_t} - pr_j$
        $NUM^0_{sRB\_t} \leftarrow NUM^0_{sRB\_t} - sr_j$
      Else if $NUM_{pRB\_t} \geq pr_j$, $NUM^1_{sRB\_t} \geq sr_j$
        Allocate licensed spectrum:$A_t \leftarrow a_j(1)$
        $NUM_{pRB\_t} \leftarrow NUM_{pRB\_t} - pr_j$
        $NUM^1_{sRB\_t} \leftarrow NUM^1_{sRB\_t} - sr_j$
      End if
    End for
  End if
End for

In summary, according to the constraint Eq. (9) and the time-variant process in Fig. 4, we can get a Greedy radio resource allocation algorithm without RL (Algorithm 1). This algorithm greedily allocates radio resources while taking into account the dynamic delay constraints of each URLLC datum.

3.2 Policy gradient update

Algorithm 1 is a Greedy resource allocation method, expecting to transmit as many URLLC data as possible in a single time step. In the next task, we design a more far-sighted RL process to replace the main part of Algorithm 1. The purpose of RL is to maximize the expected discounted return

$$\mathbb{E}[G_t] = \mathbb{E}\left[\sum_{k=t+1}^{\mathrm{K}} \gamma^{k-t-1} R_k\right], \tag{10}$$

where $0 \leq \gamma \leq 1$ is a discount rate that determines the value of future rewards, K is the end time of an episode, and G_t is the cumulative discounted return.

Therefore, the successful application of RL depends heavily on how well the reward signal R_t meets the design goals. Although the frame scheduling process is decoupled from the time step, the minimum time unit of the reward is still a time step. For minimizing the average delay $\mathbb{E}[D]$ in Eq. (4) by maximizing the reward, the reward function at time step t is designated as

$$R_t = \sum_{j=1}^{NUM_U} \frac{L_j + D_{j\_q}}{-L_j}. \tag{11}$$

In the parameterized policy update method, the optimal action is selected according to the updated direction of the policy. The policy $\pi(A_t|S_t) = Pr\left\{arg\max_{A_t} R_t\right\}$ is the probability of selecting action A_t under state S_t. Even if we compress state space $\mathcal{S}$ and action space $\mathcal{A}(s)$, a multipath DNN with the parameter θ is used to approximate the optimal policy to speed up learning. So $\pi(A_t|S_t, \theta_t)$ is the probability of selecting action A_t through a multipath DNN with parameter θ_t under state S_t.

$$\pi(A_t|S_t, \theta_t) = \frac{e^{h(S_t, A_t, \theta_t)}}{\sum_{A_t'} e^{h\left(S_t, A_t', \theta_t\right)}}, \tag{12}$$

where $A_t' \in \mathcal{A}(s)$, and $h(S_t, A_t, \theta_t)$ is an action preference function that can be parameterized by multipath DNN.

The REINFORCE algorithm [22] is a policy gradient algorithm. It can learn state, action, and reward sequences sampled from the simulated environment, and it is as effective as the real environment. The policy gradient implemented through the REINFORCE algorithm is updated as follows

$$\theta_{t+1} = \theta_t + \alpha(G_t - b_t)\frac{\nabla\pi(A_t|S_t, \theta_t)}{\pi(A_t|S_t, \theta_t)} \tag{13}$$

where $\alpha > 0$ is the learning step size, b_t is the baseline, and $\frac{\nabla\pi(A_t|S_t, \theta_t)}{\pi(A_t|S_t, \theta_t)}$ is the direction of the policy update, which is called the trace vector. Combined with Eq. (12), $\nabla\pi(A_t|S_t, \theta_t)$ is expressed as follows:

$$\nabla\pi(A_t|S_t, \theta_t) = \pi(A_t|S_t, \theta_t)\left[h(S_t, A_t, \theta_t) - \sum_{A_t'} \pi\left(A_t'|S_t, \theta_t\right) h\left(S_t, A_t', \theta_t\right)\right]. \tag{14}$$

The role of b_t is to reduce the variance of the updated values and speed up learning. Similar to what happens with the multiarm bandits, the average cumulative discounted return is selected as the baseline

Algorithm 2: Policy gradient parameter training process

Initialize the environment R_0, S_0 randomly
Initialize: $N_{loss} \leftarrow 0$, $t \leftarrow 0$, $\theta_t \leftarrow 0$
Initialize: $\nabla\pi(A_0|S_0,\theta_0)$
For $k = 1, 2, \cdots,$ K in each *episode*
  $\{S_1, A_1, R_1, \cdots, S_K, A_K, R_K\} \sim \pi_{\theta_t}$
  $G_t = \sum_{k=t+1}^{K} \gamma^{k-t-1} R_k$
  For each time step $t = 1, 2, 3, \cdots$
    For each $j = 1, 2, \cdots, |Q|$
      If $L_j + D_{j\_q} > D_{MAX}$
        Discard the data that exceed the delay constraint, and update r_j with random resource profile in the cache queue M_i.
        $L_j + D_{j\_q} \leftarrow \text{New}(L_j + D_{j\_q})$
        $N_{loss} \leftarrow N_{loss} + 1$
      End if
    End for
    Update $\nabla\pi(A_t|S_t,\theta_t)$ with $h(S_t,A_t,\theta_t)$ parameterized by the multipath DNN.
    $b_t \leftarrow \frac{1}{t}\sum_{i=1}^{t} G_i$
    $\theta_{t+1} \leftarrow \theta_t + \alpha(G_t - b_t)\frac{\nabla\pi(A_t|S_t,\theta_t)}{\pi(A_t|S_t,\theta_t)}$
  End for
End for

$$b_t = \frac{1}{t}\sum_{i=1}^{t} G_i, \tag{15}$$

where G_i is given by Eq. (10).

From Eq. (13), it can be seen intuitively that the change of parameter θ is proportional to the discounted return G_t, is inversely proportional to the probability of the selection action $\pi(A_t|S_t,\theta_t)$, and changes along the direction of the trace vector. Algorithm 2 is the policy gradient parameter training process.

3.3 Multipath DNN training

In this part, the multipath DNN is used to parameterize the action preference function $h(S_t,A_t,\theta_t)$ to quickly approximate the optimal policy. Because of the poor parameter redundancy and heavy computing load, DNN is difficult to deploy in a roadside BS with limited computing resources. The proposed multipath DNN optimizes the training path to reduce unnecessary computing load and training time. The multipath DNN with N hidden layers shown in Fig. 5 is used to train policy $\pi(A_t|S_t,\theta_t)$. The multipath DNN based on a feedforward neural network is trained to predict radio resource allocation in the next time step.

As shown in Fig. 5, each hidden layer is composed of a fully connected (FC) sublayer and a rectified linear unit (ReLU) sublayer in series. The ReLU sublayer improves calculation speed and prediction accuracy by retaining the positive values and eliminating the

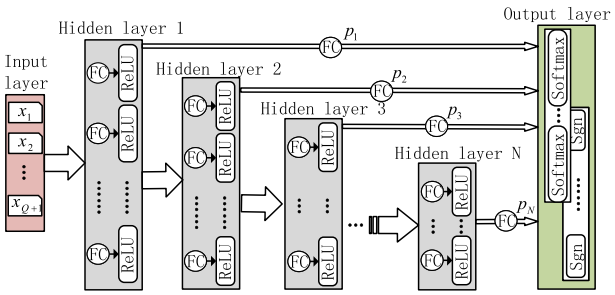

Fig. 5 The multipath DNN architecture. A multipath DNN architecture with *N* hidden layers. In figure, each hidden layer is composed of a FC sublayer and a ReLU sublayer in series. Each hidden layer has an exit, and the probability of exiting at this layer is p_i

negative values. Each hidden layer has an exit, and the probability of exiting at this layer is p_i.

The appropriate exit is selected by transmission delay constraint D_{MAX} and path cost. Thus, while meeting the requirements of delay-sensitive applications, the BS computing resources are fully utilized. Let parameter vector θ in multipath DNN be

$$\theta = \left\{p_i, w_i, \theta_i^{th}\right\}, \tag{16}$$

where $i \leq N$ is the number of layers, and w_i and θ_i^{th} are the weight vector and the threshold vector of the ith hidden layer, respectively. The output of the ith hidden layer is

$$y_i = \max\left(0, w_i x_i - \theta_i^{th}\right), \tag{17}$$

where x_i is the input vector of the ith hidden layer.

The cumulative path cost of the ith layer is

$$T_{i\_COST} = \sum_{n=1}^{i+1} (1 - p_{n-1}) t_n, \tag{18}$$

where t_n is the calculation time of the nth layer.

As mentioned earlier, data j will be allocated resources only if $L_j + D_{j\_q} < D_{MAX}$ is satisfied. However, considering the delay caused by DNN in the resource allocation process, the delay constraint needs to be further tightened.

The design goal of multipath DNN is to ensure reasonable accuracy while satisfying the low delay constraint of URLLC. Therefore, it is required that if T_{i_COST} exceeds the delay constraint $D_{MAX} - \max_j \left(L_j + T_{j\_q}\right)$, the output signal $\boldsymbol{y}_{i-1}$ is directly passed to the output layer. The normalized exponential function $f_{softmax}(x, \theta)$ is selected as the activation function of the exit layer neurons. Hence, the exit function can be expressed as

$$EXIT\left(y_i\right) = \frac{e^{wy_i}}{\sum_{k=1}^{K} e^{wy_i^k}}. \tag{19}$$

Algorithm 3 is used to select the appropriate exit path, which is a variant of the forward propagation algorithm.

Algorithm 3: Exit path selection process

Initialize $\max_j(L_j + T_{j\_q}) \leftarrow 0, T_{i\_COST} \leftarrow 0$

$\{p_i, w_i, \theta_i^{th}\} \sim \theta$

For each $j = 1, 2, \cdots, |Q|$

Get the worst delay constraint of the current state.

If $\max_j(L_j + T_{j\_q}) < L_j + T_{j\_q}$

$\max_j(L_j + T_{j\_q}) \leftarrow L_j + T_{j\_q}$

End if

End for

For $i = 1,2, \cdots, N$

$y_i = \max(0, w_i x_i - \theta_i^{th})$

$T_{i\_COST} = \sum_{n=1}^{i+1}(1 - p_{n-1})\, t_n$

If the delay constraint will be broken after adding the path cost of the i-th layer, then exit from the previous layer.

If $T_{i\_COST} > D_{MAX} - \max_j(L_j + T_{j\_q})$

$EXIT(y_{i-1}) = \frac{e^{wy_{i-1}}}{\sum_{k=1}^{K} e^{wy_{i-1}^k}}$

End if

End for

4 Results and discussion

4.1 Experimental environment and parameters

In this section, the performance of the hybrid spectrum and power resource management scheme is verified in the downlink transmission of the mmWave cell simulation platform.

We consider a roadside BS and 4 URLLC data receiving users. Assuming the total transmission power of BS is 30w, and the licensed and unlicensed spectrums in mmWave are 26.5–29.5 GHz and 37–40 GHz[1] respectively; therefore, the maximum capacity of the resource pool in the BS is as follows: $\max NUM_{pRB} = 30w$, $\max NUM_{sRB}^0 = 3$ GHz, $\max NUM_{sRB}^1 = 3$ GHz. URLLC data arrive at the BS in an i.i.d. manner with an arrival rate λ. The URLLC cache load in the BS can be varied by adjusting λ during training.

As mentioned in Sect. 2, an RB consists of 7 OFDM symbols. To facilitate implementation, the time step t is also set as the length of the RB, and the L_{rp} is set as 10t. 3GPP defines the end-to-end delay of URLLC as 1 ms. To observe more abundant experimental results, D_{MAX} is set to 1 ms and 3 ms. The length L_j of data j is less than D_{MAX}.

Considering the processing delay and queueing delay, we specify $L_j \in [2 \times RB, 40 \times RB]$. Moreover, the URLLC data are mostly short control or emergency information. Consequently, it is specified that 80% of L_j are randomly distributed in $[2 \times RB, 20 \times RB]$, and the rest are randomly distributed in $[20 \times RB, 40 \times RB]$.

[1] In Table 5.2–1 of the 3GPP 38.101–2 protocol, three TDD millimeter-wave bands are defined for the 5G NR FR2 band: they are N257 (26.5 GHz ~ 29.5 GHz), N258 (24.25–27.5 GHz), and N260 (37–40 GHz), respectively.

Table 1 Partial parameter settings

Parameter	Value	Description
t	0.015625 ms	The time interval for each resource allocation decision
L_{rp}	0.15625 ms	The duration of the resource pool
$\|sRB\|$	5760 KHz	The length of an RB in the frequency domain
$\|pRB\|$	9 dBm	The length of an RB in the power domain
D_{MAX}	[3 ms, 1 ms]	Maximum delay constraint
$\|Q\|$	100	The length of the URLLC cache queue Q
NUM_U	100	The total amount of received URLLC data
α	0.001	Learning step size of policy gradient
N	4	The number of hidden layers
p_i	1/4	The initial probability of exit at the ith layer

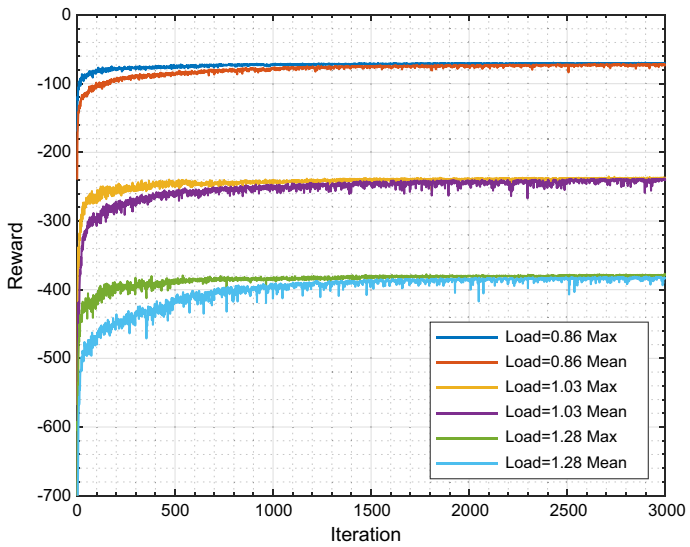

Fig. 6 The reward curves at various loads. The average and maximum total reward curves of the RL-based hybrid spectrum and power resource allocation scheme with loads of 0.86, 1.03, and 1.28. The x-axis is the number of iterative trainings, and the y-axis is the total reward in Eq. (11)

In addition, the spectrum demand sr_j of each datum is randomly distributed at $[2 \times |sRB|, 10 \times |sRB|]$, and the power demand pr_j is randomly distributed at $[10 \times |pRB|, 40 \times |pRB|]$. Parts of the parameter configuration are shown in Table 1.

A multipath DNN with four hidden layers (20 neurons per layer) is used to approximate the best policy, wherein the initial exit probability of each layer is p_i.

4.2 Experimental results and discussion

Figure 6 shows the average and maximum total reward curves of the RL-based hybrid spectrum and power resource allocation scheme with different loads. The abscissa is the number of iterative trainings, and the ordinate is the total reward in Eq. (11). Because we focus on scenarios in which radio resources are in short supply, the

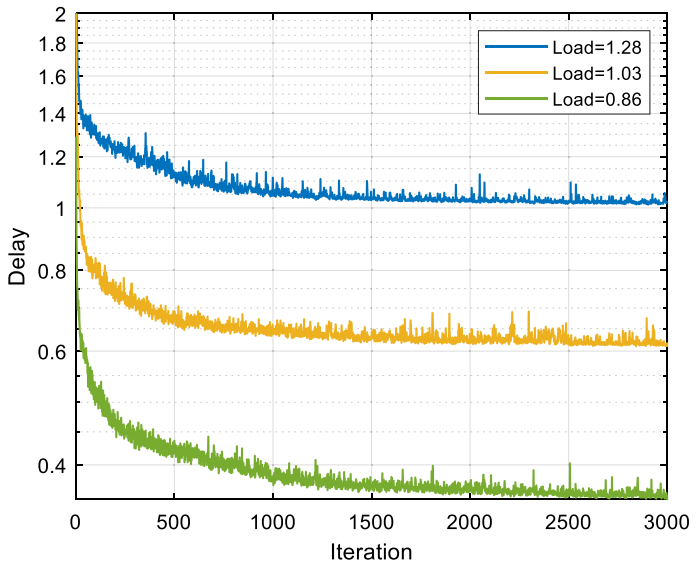

Fig. 7 The delay curves at various loads. The delay performance curves of the RL-based resource allocation scheme with loads of 0.86, 1.03, and 1.28. The *x*-axis is the number of iterative trainings, and the *y*-axis is the average delay. The total delay of URLLC data at various loads is decreased by the number of training increases and converges to a certain value after limited training

rewards with loads of 0.86, 1.03, and 1.28 are shown in Fig. 6. For such a high-load situation, the total reward can still be raised by increasing the number of trainings.

Additionally, when a training reaches 1000 times, the increase tends to become flat, and the gap between the average reward and the maximum reward become smaller. For example, the average and maximum reward curves at various loads almost overlap at 1500 iterations. This shows that the RL-based hybrid spectrum and power resource allocation scheme will eventually converge after a limited number of iterative learnings. The smaller the load, the faster the convergence speed.

Figure 7 shows the delay performance curves of the proposed RL-based resource allocation scheme at various loads. The total delay of URLLC data at various loads is decreased by the number of training increases and converges to a certain value after limited training. In Fig. 7, when iterating over 2000 times, the delay curve of the 1.28 load becomes relatively stable close to 1 ms. This indicates that when the overload does not exceed 30%, the delay constraint of URLLC can still be satisfied by the proposed RL-based resource allocation scheme through fast learning.

The delay comparison of the RL-based, Greedy, Random, shortest frame first (SFF), and Tetris [23] schemes at varying loads is shown in Fig. 8. Among them, the RL-based scheme selected 500 times, 1000 times, and 10,000 times training data, respectively. We record the delay performance of each scheme with 9 sampling points from load 0.11 to 1.28. It can be seen in Fig. 8 that the increase in load causes a scarcity of radio resources, thereby increasing the queueing delay of URLLC data.

It is noteworthy that as the load increases in Fig. 8, especially over 65%, the latency advantage of the RL-based scheme becomes more noticeable. This is because the Greedy, SFF, Random, and Tetris schemes send as much URLLC data as possible at the current time step. These schemes focus on instantaneous delay but ignore total delay.

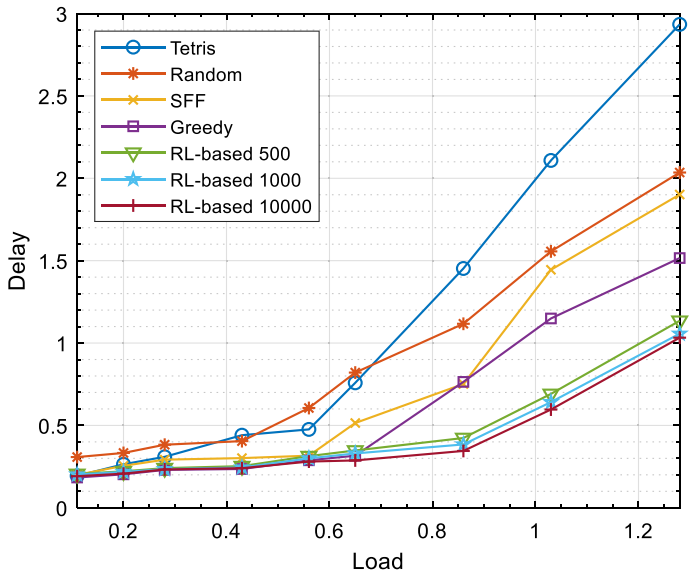

Fig. 8 Delay performance comparison of different schemes. The delay comparison of the RL-based, Greedy, Random, SFF, and Tetris schemes at varying loads is shown in figure. Among them, the RL-based scheme selected 500 times, 1000 times, and 10,000 times training data, respectively. The *x*-axis is the loads, and the *y*-axis is the average delay. We record the delay performance of each scheme with 9 sampling points from load 0.11 to 1.28

Compared with the other schemes, the RL-based radio resource allocation becomes visionary through continuous learning. URLLC data are usually quite small. The reward function in Eq. (11) takes into account all NUM_U URLLC data. Therefore, when allocating resources to the data in queue Q, some radio resources are reserved for the upcoming small data so that the small data can be quickly scheduled.

In addition, in the other four schemes, when the load exceeds 60%, the Greedy radio resource allocation scheme has the best delay performance. This is because the Greedy scheme is a greedy resource allocation under the premise of ensuring the delay constraints of each URLLC datum as much as possible. Preprocessing the data according to its remaining available time makes the delay performance of Greedy better than that of the other three schemes.

It is obvious from Fig. 9 that blindly increasing the number of trainings does not continuously optimize the delay of the RL-based hybrid spectrum and power resource allocation scheme. Take the image of delay at 1.28 load in Fig. 9 as an example. Compared to 1000 iterations, the delay under 10,000 iterations is only reduced by about 24 μs. Compared to 500 iterations, the delay under 1000 iterations is reduced by about 78 μs. Therefore, when implementing the RL-based radio resource allocation scheme in real scenarios, different iterations should be set for various loads to save computing resources while reducing delays.

Figures 10 and 11 show the loss probabilities of the different schemes with $D_{MAX} = 3$ ms and $D_{MAX} = 1$ ms, respectively. The abscissa is the load, and the ordinate is the loss probability in Eq. (5). We appropriately relax D_{MAX} to 3 ms in Fig. 10 so that we can observe the loss probability of all schemes within 130% of the load. In this case, the RL-based scheme and Greedy have a lower loss probability than the

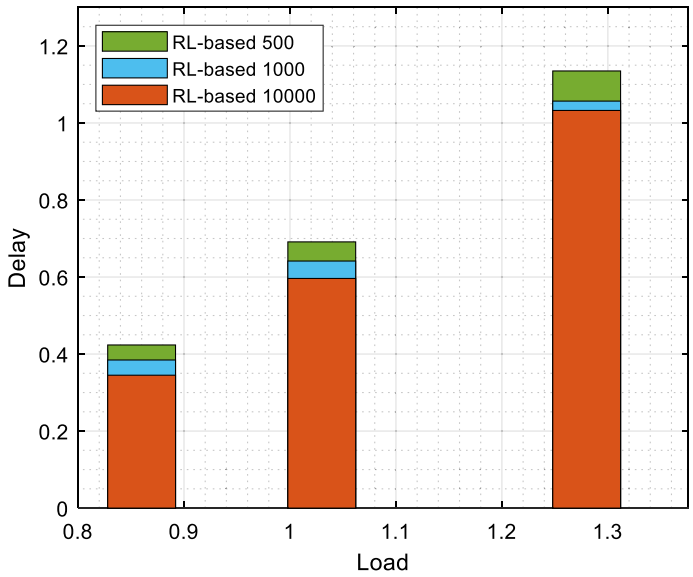

Fig. 9 The impact of the number of iterations on delay optimization. The effect of the number of iterations on the delay optimization with the data when the load is 0.86, 1.03, and 1.28. The three colors indicate 500 iterations, 1000 iterations, and 10,000 iterations, respectively. The *x*-axis is the loads, and the *y*-axis is the average delay

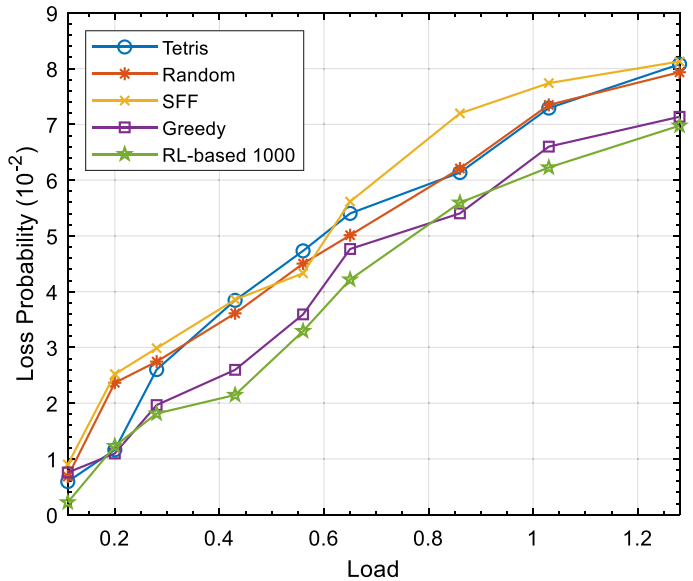

Fig. 10 Loss probability with $D_{MAX} = 3$ ms. The loss probabilities of the RL-based, Greedy, Random, SFF, and Tetris schemes with maximum delay constraint $D_{MAX} = 3$ ms. The *x*-axis is the load, and the *y*-axis is the loss probability in Eq. (5)

other three schemes. After relaxing D_{MAX}, the RL-based scheme is not significantly better than Greedy.

In Fig. 11, we follow strictly the delay constraint of URLLC. When the load exceeds 0.65, Tetris and Random cannot meet the 1-ms delay constraint, so there is no loss

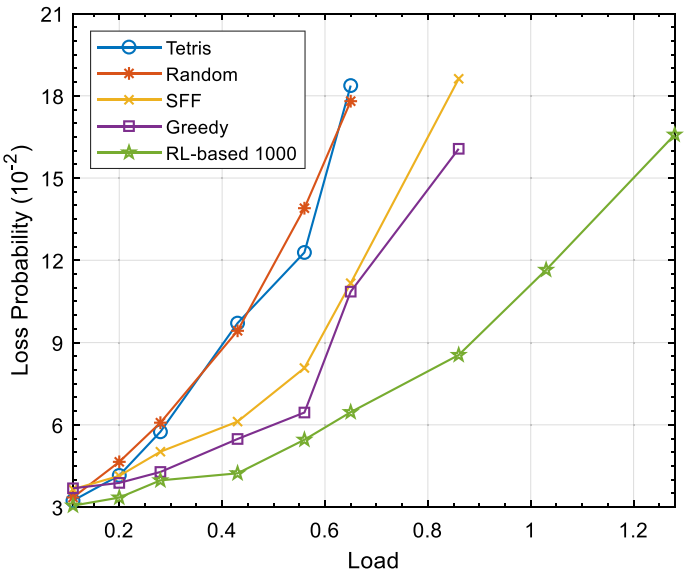

Fig. 11 Loss probability with $D_{MAX} = 1$ ms. The loss probabilities of the RL-based, Greedy, Random, SFF, and Tetris schemes with maximum delay constraint $D_{MAX} = 1$ ms. The *x*-axis is the load, and the *y*-axis is the loss probability in Eq. (5)

probability. For the same reason, when the load exceeds 0.86, the loss probability of SFF and Greedy cannot be detected. The RL-based scheme can effectively guarantee the URLLC delay constraint when the load does not exceed 130%. It is found that when the delay constraint is tightened, the loss probability of SFF is better than Tetris and Random. This is because URLLC data are relatively short, and SFF is biased toward short data.

5 Conclusion

To carry critical data such as emergency rescue and road safety, the cellular mobile communication network requires more sophisticated and faster resource allocation methods. In this paper, a novel RL-based hybrid spectrum and power resource allocation scheme has been proposed for the URLLC service in mmWave cell. This scheme guarantees the low latency characteristics of URLLC under high load and shortage of radio resources. We compressed the state space and the action space. The policy gradient and multipath DNN are used to update the policy. The design of the reward function considers all URLLC data to make the RL-based scheme more visionary. This process enables the RL-based scheme to be successfully installed in the roadside BS and to ensure converge quickly. The experimental results show the RL-based scheme can achieve a higher overall delay performance than the conventional schemes, especially when the resource demand load of URLLC data is 85% to 130%. In future work, we will investigate the prediction of high-speed mobile user behavior trends, the investigation of the optimal multipath DNN structure, and the impact of fading on URLLC.

Abbreviations

BEP: Block error probability; BS: Base station; DNN: Deep neural network; FC: Fully connected; i.i.d.: Independent and identically distributed; IIoT: Industrial Internet of Things; IoV: Internet of Vehicles; LTE: Long-Term Evolution; MDP: Markov decision process; mmWave: Millimeter-wave; NR: New radio; QoS: Quality of service; RB: Resource block; RL:

Reinforcement learning; ReLU: Rectified linear unit; SFF: Shortest Frame First; URLLC: Ultra-reliable and low-latency communication.

Acknowledgements
Not applicable.

Authors' contributions
QH was responsible for the theoretical analysis, algorithm design, experimental simulation, and manuscript writing of this research. XZX contributed to the organization structure of the paper and provided suggestions for experimental simulation. MC contributed to theoretical analysis and provided suggestions for English writing. All authors read and approved the final manuscript.

Funding
This work was supported in part by the Canada Research Chair, Tier 1, held by Mohamed Cheriet, the National Nature Science Foundation of China under Grant No. 61502067, the Key Research Project of Chongqing Education Commission under Grant No. KJZD-K201800603, the Doctoral Candidate Innovative Talent Project of CQUPT under Grant No. BYJS2017003, and the China Scholarship Council under Grant No. 201908500144.

Availability of data and materials
The author keeps the analysis and simulation data sets, but the data sets are not public.

Competing interests
The authors declare that they have no competing interests.

Author details
[1] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China. [2] École de Technologiesupérieure, Université du Québec, Montréal, QC H3C 1K3, Canada.

Received: 13 October 2020 Accepted: 2 December 2020
Published online: 10 December 2020

References

1. E. Basar, Reconfigurable intelligent surface-based index modulation: a new beyond MIMO paradigm for 6G. IEEE Trans. Commun. **68**(5), 3187–3196 (2020)
2. H. Ji, S. Park, J. Yeo et al., Ultra-reliable and low-latency communications in 5G downlink: physical layer aspects. IEEE Wirel. Commun. **25**(3), 124–130 (2018)
3. D. Feng, C. She, K. Ying et al., Toward ultrareliable low-latency communications: typical scenarios, possible solutions, and open issues. IEEE Veh. Technol. Mag. **14**(2), 94–102 (2019)
4. Q. Huang, X. Xie, H. Tang et al., Machine-learning-based cognitive spectrum assignment for 5G URLLC applications. IEEE Netw. **33**(4), 30–35 (2019)
5. H. Yang, K. Zheng, L. Zhao et al., Twin-timescale radio resource management for ultra-reliable and low-latency vehicular networks. IEEE Trans. Veh. Technol. **69**, 1023–1036 (2019)
6. Z. Wang, T. Lv, Z. Lin et al., Outage performance of URLLC NOMA systems with wireless power transfer. IEEE Wirel. Commun. Lett. (2019). https://doi.org/10.1109/lwc.2019.2956536
7. 3GPP, Study on physical layer enhancements for NR ultra-reliable and low latency case (URLLC), document TR38.824 V16.0.0 (2019)
8. X. Zhang, J. Wang, H.V. Poor, Heterogeneous statistical-QoS driven resource allocation over mmWave massive-MIMO based 5G mobile wireless networks in the non-asymptotic regime. IEEE J. Sel. Areas Commun. **37**, 2727–2743 (2019)
9. C. Zhao, Y. Cai, A. Liu et al., Mobile edge computing meets mmWave communications: joint beamforming and resource allocation for system delay minimization. IEEE Trans. Wirel. Commun. **19**, 2382–2396 (2020)
10. X. Lu, V. Petrov, D. Moltchanov et al., 5G-U: Conceptualizing integrated utilization of licensed and unlicensed spectrum for future IoT. IEEE Commun. Mag. **57**(7), 92–98 (2019)
11. R. Xie, J. Wu, R. Wang et al., A game theoretic approach for hierarchical caching resource sharing in 5G networks with virtualization. China Commun. **16**(7), 32–48 (2019)
12. F. Zhou, W. Li, L. Meng et al., Capacity enhancement for hotspot area in 5G cellular networks using mmWave aerial base station. IEEE Wirel. Commun. Lett. **8**(3), 677–680 (2019)
13. H. Huang, Y. Song, J. Yang et al., Deep-learning-based millimeter-wave massive MIMO for hybrid precoding. IEEE Trans. Veh. Technol. **68**(3), 3027–3032 (2019)
14. C. Qiu, H. Yao, F.R. Yu et al., Deep q-learning aided networking, caching, and computing resources allocation in software-defined satellite-terrestrial networks. IEEE Trans. Veh. Technol. **68**(6), 5871–5883 (2019)
15. K. Shimotakahara, M. Elsayed, K. Hinzer et al., High-reliability multi-agent Q-learning-based scheduling for D2D microgrid communications. IEEE Access **7**, 74412–74421 (2019)
16. W. AlSobhi, A.H. Aghvami, QoS-Aware resource allocation of two-tier HetNet: a Q-learning approach, in *26th International Conference on Telecommunications (ICT)* (IEEE, 2019), pp. 330–334.
17. A.T.Z. Kasgari, W. Saad, Model-free ultra reliable low latency communication (URLLC): a deep reinforcement learning framework, in *ICC 2019–2019 IEEE International Conference on Communications (ICC)* (IEEE, 2019), pp. 1–6.
18. N.B. Khalifa, M. Assaad, M. Debbah, Risk-sensitive reinforcement learning for urllc traffic in wireless networks, in *2019 IEEE Wireless Communications and Networking Conference (WCNC)* (IEEE, 2019), p. 1–7.
19. H. Yang, X. Xie, M. Kadoch, Intelligent resource management based on reinforcement learning for ultra-reliable and low-latency IoV communication networks. IEEE Trans. Veh. Technol. **68**(5), 4157–4169 (2019)

20. X. Chen, C. Wu, T. Chen et al., Age of information aware radio resource management in vehicular networks: a proactive deep reinforcement learning perspective. IEEE Trans. Wirel. Commun. **19**(4), 2268–2281 (2020)
21. R.S. Sutton, D.A. McAllester, S.P. Singh, et al. Policy gradient methods for reinforcement learning with function approximation, in *Advances in Neural Information Processing Systems* (2000), p. 1057–1063
22. Ciosek K, Whiteson S. Expected policy gradients, in *Thirty-Second AAAI Conference on Artificial Intelligence* (2018)
23. R. Grandl, G. Ananthanarayanan, S. Kandula et al., Multi-resource packing for cluster schedulers. ACM SIGCOMM Comput. Commun. Rev. **44**(4), 455–466 (2014)

Publisher's Note