Distributed Deep Reinforcement Learning Based Mode Selection and Resource Allocation for VR Transmission in Edge Networks

Jie Luo[1(✉)], Bei Liu[2], Hui Gao[3], and Xin Su[2]

[1] China Academy of Telecommunications Technology, Beijing, China
nsluojie1016@163.com
[2] Tsinghua University, Beijing, China
[3] Beijing University of Posts and Telecommunications, Beijing, China

Abstract. Wireless virtual reality (VR) is expected to be one of the most pivotal applications in 5G and beyond, which provides an immersive experience and will greatly renovate the way people communicate. However, the challenges of VR service transmission to provide high quality of experience (QoE) and a huge data rate remain unsolved. In this paper, we formulate an optimization of the mode selection and resource allocation to maximize the QoE of VR users, aiming at the optimal transmission of VR service based on the cloud-edge-end architecture. Moreover, a distributed game theory based deep reinforcement learning (DGTB-DRL) algorithm is proposed to solve the problem, which can achieve a Nash equilibrium (NE) rapidly. The simulation results demonstrate that the proposed method can achieve better performance in terms of training efficiency, QoE utility values.

Keywords: Virtual reality · Reinforcement learning · Resource allocation · Mobile edge network

1 Introduction

Based on the existing three service scenarios in wireless communication, enhanced mobile broadband (eMBB), massive machine type communications (mMTC) and ultra-reliable low latency communications (uRLLC), virtual reality (VR) develops rapidly and is considered to be one of the most promising application in the next generation, which will greatly renovate the way people communicate [1]. With the growth of multiple access for various scenarios, video applications represented by VR are required with an exponential increase of data rate, which brings up challenges. It is foreseeable that the system capacity will desire to be multiple 1000. Additionally, VR services are obliged to provide low latency and high quality of experience (QoE) to prevent users from feeling dizzy.

This work was supported by National Key R&D Program of China (2020YFB1806702).

H. Gao et al. (Eds.): ChinaCom 2021, LNICST 433, pp. 153–167, 2022.
https://doi.org/10.1007/978-3-030-99200-2_13

Benefit from mobile edge computing (MEC) which has been considered as an essential network architecture for future wireless networks [2], many works focused on reducing the amount of data transmission by delivering computation and communication resources to the network edge. The authors in [3] introduced MEC into the internet of things (IoT) network and proposed a deep reinforcement learning (DRL) based scheme to optimize the communication and computing resource allocation. To apply MEC in vehicle networks, [4] proposed to utilize DRL to find the policies of computation offloading and resource allocation in stochastic traffic and other uncertain communication conditions.

Additionally, there are dozens of works that concentrate on QoE optimization of cross-layer transmission in the wireless network. The author in [5] proposed an artificial intelligence (AI) aided joint bit rate selection and radio resource allocation scheme in fog-computing based radio access networks (F-RANs) based on multi-agent hierarchy DRL. [6] came up with an online learning method to solve the fast device-to-device (D2D) clustering and mode selection joint problem in both large-scale and small-scale scenarios, while [7] proposed a deep learning approach with an adaptive VR framework to conquer association, offloading and caching problem in real-time VR rendering tasks. To optimize VR content delivery while meeting high transmission rate requirements, [8] formed a Lagrangian dual decomposition approach to solve multiple dimensional knapsack problem, which realized a communications-caching-computing tradeoff for mobile VR devices.

However, there is little research to develop a distributed DRL method based on game theory to solve the joint optimization issue. As the matter of fact, the distributed learning methods have better flexibility and adaptability than the centralized ones [7]. Taking the adaptation capability and the scalability into account, in this paper, we focus on the transmission of VR service based on the cloud-edge-end architecture and formulate a joint optimization of the mode selection and resource allocation to maximize the QoE of all VR users. Furthermore, we propose a distributed game theory based DRL (DGTB-DRL) algorithm to conquer the above highly complex problem, which can achieve a Nash equilibrium (NE) within less time. At last, the simulation results show the superiority of the proposed method in terms of training efficiency, QoE utility values.

The remainder of this paper is organized as follows. Section 2 and Sect. 3 introduce the system model and present the problem formulation. The proposed algorithm is presented in Sect. 4. In Sect. 5, simulation results of the proposed method are presented and analyzed. Finally, the conclusion is given in Sect. 6.

2 System Model

2.1 Network Model

In this paper, we consider a mobile edge network composed of Y cloud nodes, M edge nodes and U user equipments as a cloud-edge-end collaborative framework, as shown in Fig. 1. The original VR game resources are centrally managed by

the cloud server on the cloud nodes. Equipped with edge servers, edge nodes served as access points are distributedly deployed in the VR environment on a large scale. End nodes are defined as VR headsets worn by real users.

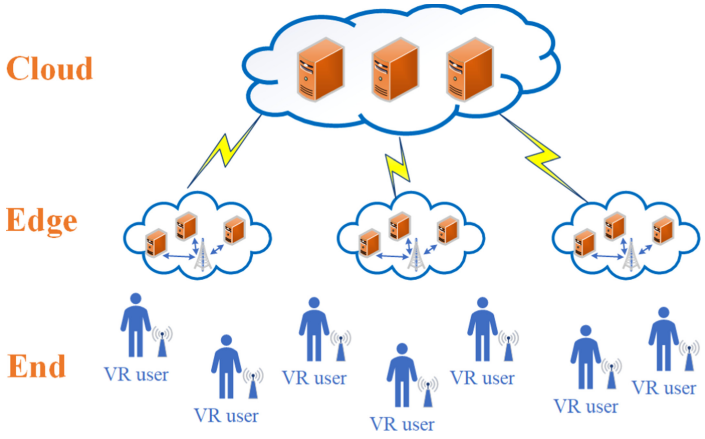

Fig. 1. Cloud-edge-end VR system model.

Considering the different quality of service requirements and the computation capabilities of different users, the VR user can choose three modes to transmit the contents, (1) cloud mode, (2) edge mode, and (3) local mode. The cloud nodes transmit the contents directly in cloud mode while the edge nodes serve users simply in edge mode. Specially, there are F femto base stations (BS) for short distance transmission in local mode. Therefore, we denotes the set of all transmitting nodes by $\mathcal{TN} = \{pn_1, \ldots, pn_y, mn_1, \ldots, mn_m, fn_1, \ldots, fn_f\}$, with the set of the transmitting nodes' indices $\mathcal{N} = \{0, 1, \ldots, L-1\}$, where $L = Y + M + F$.

2.2 Communication Model

Assume that different transmitting nodes may associate with different sets of end nodes with K shared orthogonal channels. A binary mode selection indicator is given by $\mathcal{V} = \{v_i^l(t), i \in \mathcal{U}, l \in \mathcal{N}\}$, where $\mathcal{U} = \{1, \ldots, U\}$. And we have $v_i^l(t) \in \{0, 1\}$, where $v_i^l(t) = 1$ indicates that the i^{th} user selects the $\mathcal{TN}_l$ as its transmitting node and $v_i^l(t) = 0$ otherwise. We assume that each user can only choose one mode at any time, the following constraint should be met,

$$\sum_{l=0}^{L-1} v_i^l(t) \leq 1, \forall i \in \mathcal{U}. \tag{1}$$

Meanwhile, assume that spectrum resource is divided into $\mathcal{K} = \{1, \ldots, K\}$ channels for users. We denote a binary channel allocation indicator by $\mathcal{C} = \{c_i^k(t), i \in \mathcal{U}, k \in \mathcal{K}\}$. We have $c_i^k(t) \in \{0, 1\}$, where $c_i^k(t) = 1$ indicates that the

i^{th} user select channel $\mathcal{C}_k$ at time t. Considering the limitation of simultaneous transmission, we assume that each user can only choose one channel at any time, the following constraint should be met,

$$\sum_{k=1}^{K} c_i^k(t) \leq 1, \forall i \in \mathcal{U}. \tag{2}$$

All transmitting nodes share the same downlink spectrum. At the same time, to avoid interference between users, the spectrum is orthogonal allocated to each client node under the edge node. Thus, there is interference among cloud nodes, edge nodes and femto BSs. Let $h_i^{k,l}(t)$ be the channel gain between transmitting node $\mathcal{TN}_l$ and the i^{th} user allocated channel $\mathcal{C}_k$ at time t. The received signal to interference plus noise ratio (SINR) is given by

$$SINR_i^{k,l}(t) = \frac{v_i^l(t)c_i^k(t)p_i^{k,l}(t)h_i^{k,l}(t)}{Int_i^{k,l} + \sigma^2}, \tag{3}$$

where $p_i^{k,l}(t)$ is the transmit power used on channel $\mathcal{C}_k$ gain between node $\mathcal{TN}_l$ and the i^{th} user at time t, $Int_i^{k,l} = \sum_{j \neq l} v_i^j(t)c_i^k(t)p_i^{k,l}(t)h_i^{k,j}(t)$ denotes the interference and σ^2 denotes the noise power.

As a result, the downlink data rate of the i^{th} user from node $\mathcal{TN}_l$ on channel $\mathcal{C}_k$ can be calculated by

$$r_i^{k,l}(t) = W \log_2 \left(1 + SINR_i^{k,l}(t)\right), \tag{4}$$

where W denoted the channel bandwidth.

For convenience, we focus on the chosen channel (the rate is abbreviated as $r_i^l(t)$) and assume that $q_i^l(t)$ denotes the total bits of transmission, the transmission time is given by

$$TR_i^l(t) = \frac{q_i^l(t)}{r_i^l(t)}, \forall i, l. \tag{5}$$

Therefore, we have the total downlink data rate of the i^{th} user which is expressed by

$$r_i(t) = \sum_{l=0}^{L-1}\sum_{k=1}^{K} r_i^{k,l}(t) = \sum_{l=0}^{L-1}\sum_{k=1}^{K} W \log_2 \left(1 + SINR_i^{k,l}(t)\right). \tag{6}$$

2.3 Computing Model

In this system, the frame calculation of video rendering is also indispensable for VR users. The computing task requested by users is scheduled by the servers on the transmitting nodes. At time t, $C_i^l(t)$ denotes the computational resource which is assigned to the i^{th} user from node $\mathcal{TN}_l$. Thus, the time consumed for computing tasks is given by

$$TC_i^l(t) = \frac{D_i^l(t)}{C_i^l(t)\beta}, \forall i, l, \tag{7}$$

where $D_i^l(t)$ denotes the data size at time t, β is the computation capacity of the server per CPU cycle. Let C_{sum} denote the total computational resource, the following constraint should be met,

$$\sum_{i=1}^{U}\sum_{l=0}^{L-1} C_i^l(t) \leq C_{sum}. \tag{8}$$

2.4 Quality of Experience Model

Most existing QoE model building methods rely on the prior assumption that the QoE score and quality of service (QoS) parameters have specific mathematical expressions. A commonly used model for QoE prediction is given by a rational model with a logarithmic function, which is defined by network-level parameters (packet loss rate) and application-level parameters (i.e., send bitrate, frame rate). Based on [9], the time of video stalling should be kept at a low level to enhance the user's experience. We convert logarithm to linear weighting and make a reasonable migration assumption. Similar to [10], the time-average stalling probability is defined as

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\frac{1}{LK}\sum_{l=0}^{L-1}\frac{TC_i^l(\tau)}{TR_i^l(\tau)}, \tag{9}$$

and the time average bitrate is calculated by

$$\lim_{t\to\infty}\frac{1}{t}\sum_{\tau=0}^{t-1}\frac{1}{LK}\sum_{l=0}^{L-1}\sum_{k=1}^{K} r_i^{k,l}(\tau). \tag{10}$$

In order to jointly consider the above factors, we define the average QoE for the i^{th} user at processing period t as

$$QoE_i(t) = \frac{1}{LK}\sum_{l=0}^{L-1}\sum_{k=1}^{K}\left[\omega_1 r_i^{k,l}(\tau) - \omega_2\frac{TC_i^l(\tau)}{TR_i^l(\tau)}\right], \tag{11}$$

where ω_1 and ω_2 are non-negative weights representing the relative importance of QoE.

3 Problem Formulation

Consider that all end nodes are desired to get the maximum transmission rate from transmitting nodes while keeping a quite low latency. We assume that the SINR of the i^{th} user $SINR_i(t)$ should not be less than the minimum QoS threshold ξ_i,

$$SINR_i(t) = \sum_{l=0}^{L-1}\sum_{k=1}^{K} SINR_i^{k,l}(t) \geq \xi_i, \tag{12}$$

Furthermore, taking transmission cost into consideration, we define λ_l as the unit price of the $\mathcal{TN}_l$ transmit power. Distinctly, λ_l has a negative correlation with ω_1. Thus, we make $\omega_2 = \lambda_l$.

Similarly, the i^{th} user obtains the achieved profit which is given by ρ_i. We have $\omega_1 = \rho_i$.

Our goal is to develop an effective joint mode selection and resource allocation scheme to maximize the QoE of the VR users. The optimization problem can be formulated as follows,

$$\begin{aligned} &\mathrm{P0}: \max_{\mathcal{V},\mathcal{C}} QoE_i(t) \\ &s.t. C1: \sum_{l=0}^{L-1} v_i^l(t) \leq 1, \sum_{k=1}^{K} c_i^k(t) \leq 1, \forall i \in \mathcal{U} \\ &\quad C2: \sum_{i=1}^{U} \sum_{l=0}^{L-1} C_i^l(t) \leq C_{sum} \\ &\quad C3: \sum_{l=0}^{L-1} \sum_{k=1}^{K} SINR_i^{k,l}(t) \geq \xi_i, \forall i \in \mathcal{U} \end{aligned} \tag{13}$$

where the first constraint $C1$ denotes the mode and the channel limitation. $C2$ limits the computation resource of computing servers, and the third constraint $C3$ means that each user should achieve the minimum QoS threshold.

Sequentially, mode-selection action taken by users may consume cost, the reward of the i^{th} user should be given as the QoE utility minus the action-selection cost Ψ_i, that is,

$$R_i(t) = QoE_i(t) - \Psi_i \tag{14}$$

where $\Psi_i \geq 0$ acts as a punishment for the negative reward. To achieve the minimum QoS of all users, the punishment should be set large enough. Besides, the joint optimization problem is to maximize the long-term reward. We define the long-term reward Φ_i as the weighted sum of the instantaneous rewards over a finite period T. Hence, we can transfer P0 to P1 as

$$\begin{aligned} &\mathrm{P1}: \max_{\mathcal{V},\mathcal{C}} \sum_{t=0}^{T-1} \gamma^t R_i(t) \\ &s.t. C1: \sum_{l=0}^{L-1} v_i^l(t) \leq 1, \sum_{k=1}^{K} c_i^k(t) \leq 1, \forall i \in \mathcal{U} \\ &\quad C2: \sum_{i=1}^{U} \sum_{l=0}^{L-1} C_i^l(t) \leq C_{sum} \\ &\quad C3: \sum_{l=0}^{L-1} \sum_{k=1}^{K} SINR_i^{k,l}(t) \geq \xi_i, \forall i \in \mathcal{U} \end{aligned} \tag{15}$$

where $\gamma \in [0, 1)$ denotes the discount rate to determine the weight of the future reward. When $\gamma = 0$, we only focus on the immediate reward. $\gamma \leq 1$ means that future rewards are smaller than the rewards in the earlier periods.

4 DGTB-DRL for the Optimization Problem

In particular, it is worth noting that the action space of the issue increases exponentially with the growth of the number of transmitting nodes and channels. Owing to the non-convex and combinatorial characteristics, it is a great challenge to find a globally optimal strategy for the joint mode selection and resource allocation problem. Besides, the traditional centralized learning method may need global information of all nodes to achieve the optimal solution. Hence, we develop a distributed game theory based DRL (DGTB-DRL) method for the optimization problem in the mobile edge network.

Suppose that all users do not know the network environment and the quality of transmitting nodes. At any time t, the reward of each user relies on the current state of the network environment and the action of other users. Thus, the learning game meets Markov property [11]. We formulate the joint optimization problem as a regular Markov decision process (MDP). The corresponding quadruple $\langle \mathcal{S}, \mathcal{A}_i, \mathcal{P}, \mathcal{R}_i \rangle$ is defined as follows.

1. State space $\mathcal{S}$: For the sake of convenience, let $s(t)$ denote the state of link quality at time t,

$$s(t) = \{s_1(t), s_2(t), \ldots, s_U(t)\}, \tag{16}$$

where $s_i(t) \in \{0, 1\}$ is a binary indicator. $s_i(t) = 1$ means that the i^{th} user satisfies the minimum QoS threshold ξ_i and $s_i(t) = 0$ otherwise. It is critical that the number of possible states is as large as 2^U.

2. Action space $\mathcal{A}_i$: Considering the uniqueness of chosen transmitting node and channel for each user, we define the action space as

$$a_i^l(t) = \{v_i^l(t), c_i^l(t)\}, \tag{17}$$

where $v_i^l(t) \in \{0, 1\}$ and $c_i^k(t) \in \{0, 1\}$ are binary indicators and $\boldsymbol{v_i^l(t)} \in \{v_i^0(t), \ldots, v_i^{(L-1)}(t)\}$, $\boldsymbol{c_i^k(t)} \in \{c_i^1(t), \ldots, c_i^K(t)\}$.

3. State transition probability $\mathcal{P}$: The state is transferred by taking the action. We have state transition probability as follows:

$$P_{ss'}(\vec{a}) = P\left[\mathcal{S}_{t+1} = s' \mid \mathcal{S}_t = s, \mathcal{A}_t = a\right], \tag{18}$$

where $\vec{a} = (a_1, \ldots, a_U)^T$ is the joint action of all users.

4. Reward function $\mathcal{R}_i$: When the i^{th} user makes a decision and takes the action $a_i^l(t)$, it receives an immediate reward which is referred to (14). We have $\boldsymbol{R_i(t)} \in \{R_1(t), \ldots, R_U(t)\}$. Assume that the whole system is stationary, the

policy π_i is defined as a time-invariant mapping $\mathcal{S} \rightarrow \mathcal{A}_i$. According to formulation (15), the long-term reward can be formulated as follows.

$$\mathcal{R}_i(s, \pi_i, \pi_{-i}) = \sum_{t=0}^{T-1} \gamma^t R_i(s(t), \pi_i(t), \pi_{-i}(t) \mid s(0)), \tag{19}$$

where $\pi_{-i} = (\pi_1, \ldots, \pi_{i-1}, \pi_{i+1}, \ldots, \pi_U)$ is the policy of the rest $U-1$ agents.

Notice that the reward of each user depends not only on its decisions but also on the decisions of other users. Therefore, the centralized optimization for such complex problems is unsuitable. To conquer the above shortcoming, we propose a distributed learning method. When the network changes dynamically, the system needs to rerun the entire scheme to reach the Nash equilibrium (NE) with traditional methods. Using the DGTB-DRL method proposed in this paper, each user can predict the utility generated by each action from multiple states with little consumption of computation and latency.

Hence, we formulate a stochastic game $\left\langle \mathcal{U}, \{\mathcal{A}_i\}_{i\in\mathcal{U}}, \{R_i\}_{i\in\mathcal{U}} \right\rangle$ that is involved in all users, where the utility function of the game is equivalent to Eq. (19) for convenience. Besides, $(\vec{a}_i, \vec{a}_{-i}) \in \mathcal{A}_i$ is defined as the feasible solution space of this game where $\vec{a}_{-i}$ are actions of the other users.

To solve the proposed problem, we used the pure-strategy NE theorem, that is:

$$\mathcal{R}_i\left(s, \vec{a}_i^*, \vec{a}_{-i}^*\right) \geq \mathcal{R}_i\left(s, \vec{a}_i, \vec{a}_{-i}^*\right), \tag{20}$$

The game reaches its NE state if and only if the inequalities above stay true.

Next, we resort to the DRL method to obtain an NE strategy, where we adopt the double-dueling DQN model to interact with the dynamic environment. Figure 2 shows the procedure of the proposed method.

To solve the problem of overestimation in typical DQN, the online network and the target network are linked up to calculate the target instead of using the target network alone, which is the essence of double DQN [12]. The target is expressed by

$$y_i = R_i + \gamma \hat{Q}_i \left(s', \arg\max_{a_i \in \mathcal{A}_i} Q_i\left(s', a_i; \theta\right); \theta^- \right). \tag{21}$$

Moreover, the dueling architecture [13] is introduced to improve the training efficiency of DQN by estimating a part of the value of actions, which can be expressed by

$$Q(s, a) = A(s, a) + V(s) \tag{22}$$

The overall algorithm is illustrated in Algorithm 1.

According to the increasing utility and the existence of its upper bound, the purpose game will achieve convergence eventually within finite steps. Therefore, the purpose DGTB-DRL algorithm can guarantee to achieve a pure strategy NE.

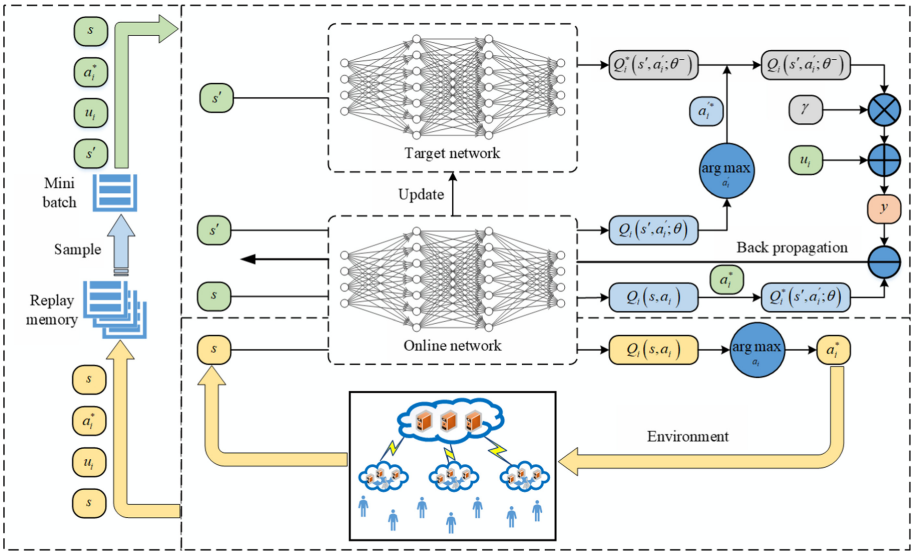

Fig. 2. Deep reinforcement learning with double-dueling DQN model.

5 Simulation Results and Discussions

In this section, simulations are executed to evaluate the performance of the proposed algorithm which shows the advantages of the proposed method in terms of training efficiency and system utility. Similar to [14], we consider a mixed VR scenario consisting of 2 cloud nodes, 8 edge nodes 30 end nodes whose correlated transmission radiuses are 500 m, 100 m and 30 m, respectively. We conduct the simulation on a 64-bit computer with 16 GB of RAM and Intel i7 1.8 GHz. The other simulation parameters of the proposed solution are defined in Table 1.

Table 1. Simulation parameters.

Parameter	Value
Channel bandwidth W	180 MHz
Orthogonal channels K	20
Transmit power	40, 30, 20 dBm
Path loss model of cloud/edge node	$34 + 40\log(d)$
Spectral noise power density N_0	-174 dBm/Hz
Minimum QoS threshold ξ_i	5 dB
Action-selection cost Ψ_i	0.001
Non-negative weight ω_1	0.5
Non-negative weight ω_2	0.0005

Algorithm 1. DGTB-DRL for the optimization problem

Input: The list of allowed actions taken by all users.
Output: The strategy of all users that meets the QoS threshold.
Initialization
Initialize the replay memory $\mathcal{D}$ with capacity L_{rp}.
Initialize the online DQN Q and the target DQN $\hat{Q}$ with $\theta^{-} = \theta$.
Run:
while episode $\leq T_1$ (total episodes in a trial) **do**
  Observe the network state s.
  while step $\leq T_2$ (total steps in an episode) **do**
    Each user selects an action a_i using ϵ-greedy policy from $\hat{Q}$.
    Each user obtains the current immediate reward R_i .
    Each user gets the new state s' by communications and sets $s \leftarrow s'$.
    Store transition tuple (s, a_i, R_i, s') in $\mathcal{D}$.
    Update the online DQN Q and the target DQN $\hat{Q}$.
    Sample a mini-batch from $\mathcal{D}$ randomly.
    Calculate the target based on (21).
    Calculate the loss and minimize the loss function through gradient descent
    Every T_0 steps, update the target DQN $\hat{Q}$ with $\theta^{-} = \theta$
    if the system state is $s = (1, \ldots, 1)$ **then**
      Break.
    end if
  end while
end while
Return: The optimal allocation scheme, mode selection strategy and the total reward.

In our experiments, the deep neural networks in DGTB-DRL are composed of an input layer, 3 hidden layers and an output layer using the ReLU function for activation function. We initialize 500 episodes and 500 steps for a trial, 8 for mini-batch size and 500 for replay memory $\mathcal{D}$. The ϵ-greedy policy is utilized linearly with ϵ from 0 to 0.9.

5.1 Evaluation of Different Learning Parameters

Firstly, the proposed method is evaluated with various learning rates η. Figure 3 demonstrates the training efficiency with varying η. At the early phase of the process, training steps are huge in all trials. As the number of episodes grows up, training steps tend to converge with narrow fluctuation. Furthermore, as η decreases, the speed of convergence increases that shows all users satisfy the QoS threshold through exploration and exploitation. However, when $\eta = 0.0001$, the learning convergence becomes unstable at the end of the process. Considering comprehensive performance, η is hence chosen to be 0.01.

Next, we evaluate the performance of the proposed method with different optimization strategies. As shown in Fig. 4, as well as learning rates, training steps are very large in all cases at the early phase. As the number of episodes

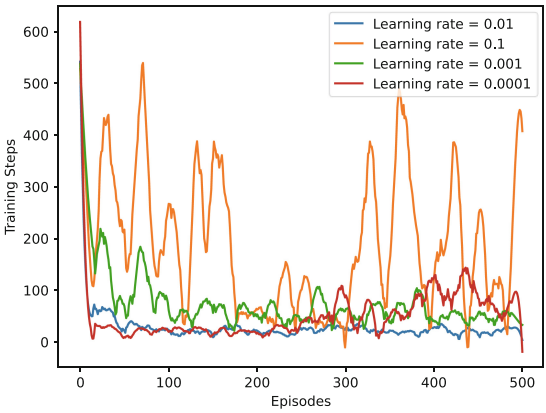

Fig. 3. Training steps of different learning rates η.

grows up, the RMSProp optimizer and Adagrad optimizer show better convergence than Adam. Consequently, we choose the Adagrad optimization strategy in our DGTB-DRL method.

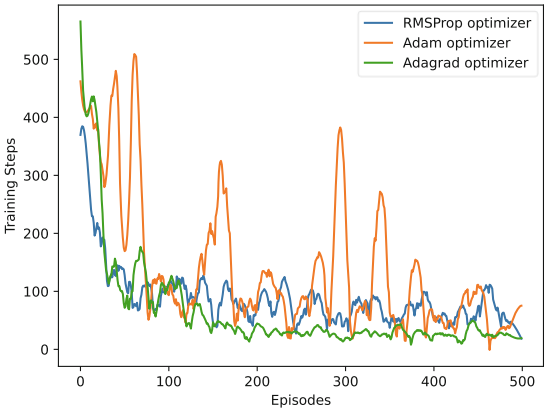

Fig. 4. Training steps of different optimization strategies.

5.2 Simulation of Different Scenarios

Secondly, simulations of various scenarios are executed to verify the proposed method. The number of end nodes is varied in this experiment. Figure 5 indicates that the efficiency of the DGTB-DRL method is decreasing with the growth of the number of users. Concretely, when $U = 10$ the speed of convergence is faster than $U = 20$ and $U = 30$, which means that as U increases, it takes more time for agents to get their optimal action vector and achieve the NE due to the increase of interference.

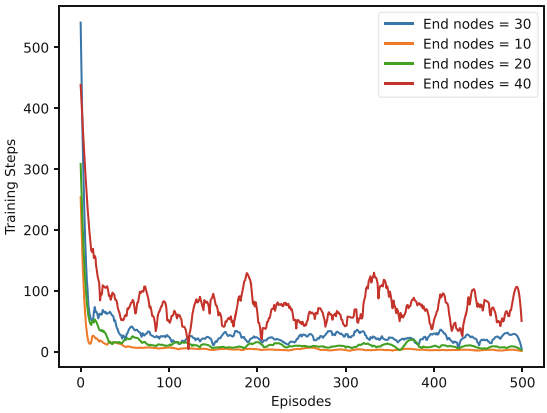

Fig. 5. Training steps with various numbers of end nodes.

Moreover, Fig. 6 demonstrates the performance of the DGTB-DRL method with different minimum QoS thresholds of users ξ_i. When $\xi_i = 0\,\mathrm{dB}$ and $\xi_i = -5\,\mathrm{dB}$, the curves of learning efficiency perform well. However, with the growth of QoS requirements, more steps need to be tried for the proposed method to meet users' experience.

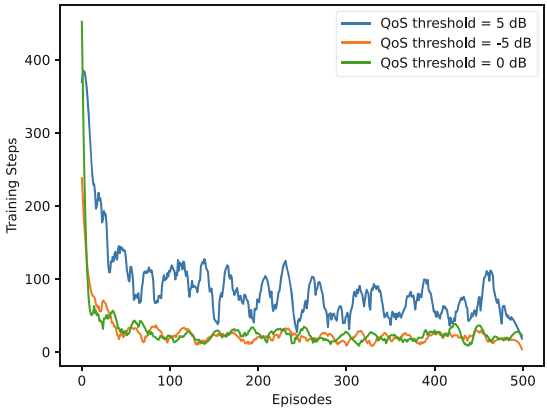

Fig. 6. Training steps with different minimum QoS thresholds of users.

5.3 Performance of Different Algorithms

Ultimately, the performance of training efficiency and total QoE utility value of the DGTB-DRL method are compared with several mainstream optimization algorithms which are presented in Figs. 7 and 8. For the convenience of comparison, the curves of training steps in Fig. 7 are fitted. The proposed DGTB-DRL

algorithm shows better performance in terms of training speed and convergence. Moreover, the simple genetic algorithm (SGA) as the classic method to solve the optimization problem is also considered with varying numbers of users. On account of computational complexity, the experiment of SGA terminates at 30. As the number of users increases, the total QoE utility increases in all optimization methods. Specifically, three learning methods perform better than SGA and achieve approximately the same QoE utility. However, the DQN method and Q-learning method obtain less QoE utility when users of the system get higher, which shows the superiority of the proposed DGTB-DRL method.

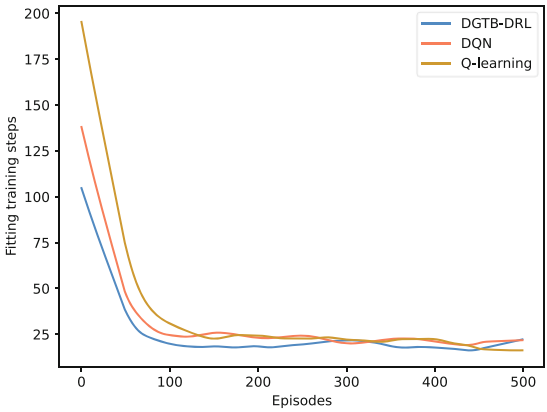

Fig. 7. Training steps of different learning methods.

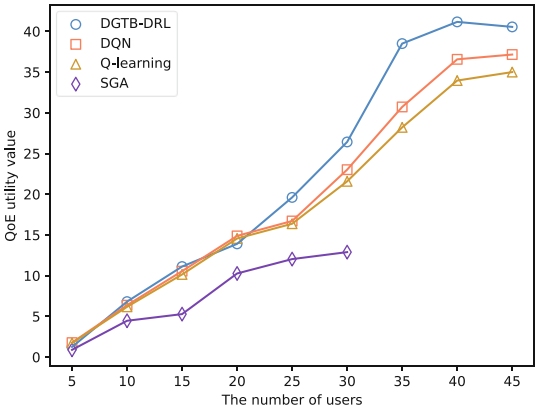

Fig. 8. Total QoE utility of different optimization methods.

6 Conclusion

In this paper, we focus on the transmission of VR services based on the cloud-edge-end architecture towards future generation communication. Foremost, we formulate a joint optimization of the mode selection and resource allocation to maximize the QoE of VR users. Then we propose the DGTB-DRL algorithm to conquer the high complexity problem, which can achieve a Nash equilibrium (NE) and maintain the adaptation capability and scalability. The numerical results show the superiority of the proposed method compared with prevalent schemes in terms of training efficiency, QoE utility values.

Acknowledgment. This work was supported by National Key R&D Program of China under Grant 2020YFB1806702.

References

1. Zong, B., Fan, C., Wang, X., Duan, X., Wang, B., Wang, J.: 6G technologies: key drivers, core requirements, system architectures, and enabling technologies. IEEE Veh. Technol. Mag. **14**(3), 18–27 (2019)
2. Truong, H.-L., Karan, M.: Analytics of performance and data quality for mobile edge cloud applications. In: Proceedings of the IEEE 11th International Conference on Cloud Computing, San Francisco, pp. 660–667 (2018)
3. Min, M., Xiao, L., Chen, Y., Cheng, P., Wu, D., Zhuang, W.: Learning-based computation offloading for IoT devices with energy harvesting. IEEE Trans. Veh. Technol. **68**(2), 1930–1941 (2019)
4. Liu, Y., Yu, H., Xie, S., Zhang, Y.: Deep reinforcement learning for offloading and resource allocation in vehicle edge computing and networks. IEEE Trans. Veh. Technol. **68**(11), 11158–11168 (2019)
5. Chen, J., Wei, Z., Li, S., Cao, B.: Artificial intelligence aided joint bit rate selection and radio resource allocation for adaptive video streaming over F-RANs. IEEE Wireless Commun. **27**(2), 36–43 (2020)
6. Feng, L., Yang, Z., Yang, Y., Que, X., Zhang, K.: Smart mode selection using online reinforcement learning for VR broadband broadcasting in D2D assisted 5G hetNets. IEEE Trans. Broadcast. **66**(2), 600–611 (2020)
7. Guo, F., Yu, F.-R., Zhang, H., Ji, H., Leung, V.-C.-M., Li, X.: An adaptive wireless virtual reality framework in future wireless networks: a distributed learning approach. IEEE Trans. Veh. Technol. **69**(8), 8514–8528 (2020)
8. Dang, T., Peng, M.: Joint radio communication, caching, and computing design for mobile virtual reality delivery in fog radio access networks. IEEE J. Sel. Areas Commun. **37**(7), 1594–1607 (2019)
9. Tao, X., Jiang, C., Liu, J., Xiao, A., Qian, Y., Lu, J.: QoE driven resource allocation in next generation wireless networks. IEEE Wireless Commun. **26**(2), 78–85 (2019)
10. Luo, J., Yu, F.R., Chen, Q., Tang, L.: Adaptive video streaming with edge caching and video transcoding over software-defined mobile networks: a deep reinforcement learning approach. IEEE Trans. Wireless Commun. **19**(3), 1577–1592 (2019)
11. Neyman, A., Sorin, S.: Stochastic Games and Applications. Kluwer, Dordrecht (2003)

12. Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double Q-learning. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence (2016)
13. Wang, Z., Freitas, N., Lanctot, M.: Dueling network architectures for deep reinforcement learning (2015). arXiv:1511.06581
14. Zhao, N., Liang, Y., Niyato, D., Pei, Y., Wu, M., Jiang, Y.: Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks. IEEE Trans. Wireless Commun. **18**(11), 5141–5152 (2019)