Received June 9, 2020, accepted June 21, 2020, date of publication July 2, 2020, date of current version July 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3006502

An End-to-End Network Slicing Algorithm Based on Deep Q-Learning for 5G Network

TAIHUI LI, XIAORONG ZHU, AND XU LIU
Wireless Communication Key Lab of Jiangsu Province, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Corresponding author: Xiaorong Zhu (xrzhu@njupt.edu.cn)

This work was supported in part by the Natural Science Foundation of China under Grant 61871237, in part by the Program to Cultivate Middle-aged and Young Science Leaders of Universities of Jiangsu Province and the Key Research and Development Plan of Jiangsu Province under Grant BE2019017, and in part by the Project of Nanjing University of Posts and Telecommunications under Grant NY217028 and Grant NY215100.

ABSTRACT As one of key technologies of the fifth-generation (5G) communication system, network slicing can share the underlying infrastructure with different application requirements and ensure that the slices can be isolated from each other. This paper proposes an end-to-end (E2E) network slicing resource allocation algorithm based on Deep Q-Networks (DQN), which is suitable for multi-slice and multi-service scenarios. This algorithm jointly considers the radio access network slices and core network slices to dynamically allocate resources to maximize the number of access users. First we build such a model, which is a mixed integer programming problem and it needs to be dynamically adjusted according to the changes of environment. We propose to use DQN algorithm to solve this problem, which can perceive changes in the environment and make dynamic decisions. Under each decision, we need to calculate the reward value of DQN, so we divide the problem into the core side and the access side. Then the dynamic knapsack algorithm and the link mapping algorithm are used to obtain the reward. The simulation results show that the average access rate of DQN scheme is higher than 97%. Compared with the optimal allocation scheme of access side, the average access rate is increased by 9% for delay constrained slices and 5% for rate constrained slices in a dynamic environment.

INDEX TERMS 5G network, network slicing, resource allocation, deep Q-networks.

I. INTRODUCTION

The fifth-generation (5G) network will support a large number of diversified business scenarios from vertical industries, such as intelligent security, high-definition video, telemedicine, smart home, autopilot and augmented reality, which usually have different communication requirements. For example, the requirements are different in terms of mobility, billing, security, policy control, delay and reliability [1]. Traditional mobile communication network is mainly used to serve a single mobile broadband service, which cannot adapt to the diversified business scenarios of 5G in the future. If a special physical network is built for each business scenario, it will inevitably lead to problems such as complex network operation and maintenance, high cost and poor scalability. Therefore, in order to support a variety of business scenarios with different performance requirements on one physical network, network slicing technology emerges at the right moment. Through network slicing technology, operators can divide physical network infrastructure into multiple virtual networks according to the needs of different users to meet the diversified business needs of 5G.

Next Generation Mobile Networks (NGMN) defines network slicing as a virtual network with multiple independent business operations running on a general physical infrastructure, and introduces the concept of network slicing into mobile communication networks for the first time. Different tenants share the network, computing, and storage resources by creating isolated virtual networks on the common underlying physical infrastructure. Each network slice is a logically independent E2E network, which consists of a set of Network Functions (NF) and corresponding resources to provide E2E on-demand services for specific business scenarios [2].

Network slicing provides on-demand services with different characteristics and requirements for vertical industries [3]. Each service usually contains network functions in a

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Khandaker.

fixed order, called service function chain (SFC). Each service function in SFC can only be provided by some given network nodes. In order to achieve network slicing, the network must select the network function nodes according to the SFC and determine the routing strategy of the function nodes in the specified order. Therefore, some scholars regard network slicing as an SFC and combine the service provision and slicing construction processes to provide the real-time required by the business.

On the access side, network slices need to virtualize the resources to the corresponding slices and users [4]. On the core side, network slices need to virtualize the network elements as virtual network function (VNF) to assign to the corresponding slices [5]. Resource virtualization is a process to realize the abstraction, slicing, isolation and sharing of radio resources. This paper proposes an E2E framework for wireless resource virtualization and allocation. E2E refers to the entire communication process from the access side to the core side. The main contributions of this work can be summarized as follows:

- In this paper, we propose an E2E network slicing framework for 5G resource allocation. This framework consist of access network and core network. In the access network, the infrastructure provider dynamically allocate wireless resources to slices and BSs. In the core network, VNFs of SFC automatically matches physical nodes.
- To meet different service requirements, this paper considers two types of slices: rate constraints and delay constraints. Under the premise of meeting different needs, solve the E2E maximum access rate of the entire system. For different types of slices, design different algorithms and measurement indicators to solve.
- This paper proposes a DQN algorithm which enables dynamic and real-time update of wireless resource allocation and mapping of service links based on feedback from the environment. The feedback in this paper is the maximum access rate of the system. In order to solve the feedback, the E2E slice is divided into the access side and the core side. The dynamic knapsack algorithm and the link mapping algorithm are used to get the solutions to obtain the reward.
- Extensive simulations are performed to evaluate the performance of the algorithm proposed in this paper. The results show that our proposed DQN resource allocation strategy performs better in terms of access rate.

The remainder of this paper is organized as follows. In Section 2, we review related studies. In Section 3, we introduce the E2E system model in this research with some equations and propose the DQN algorithm. In Section 4, we divide the problem into two sub-problems to solve. A performance evaluation of our proposed DQN algorithm is presented in Section 5. Section 6 concludes this paper.

II. RELATED WORKS

Numerous studies have conducted in-depth research on network slicing. In view of resource allocation, the research of network slicing is divided into access side network slicing, core side network slicing and E2E network slicing.

For wireless access network slicing, [6] proposes a scheme for mobility management. In this scheme both co-layer interference and cross-layer interference are taken into account to allocate the power and subchannel. [7] considers the existence of Enhanced Mobile Broadband (eMBB) and Ultra Reliable Low Latency Communications (uRLLC) slices in the radio access network (RAN). Under the constraints of limited physical resources, the slicing requests is appropriate accepted can maximize the revenue of operators. The objective function is transformed into mixed integer nonlinear programming, which is solved by continuous convex approximation and semidefinite relaxation. References [8], [9] use the queuing model to minimize the transmission power and total bandwidth of the access network for uRLLC slices. Reference [10] studied the application of deep reinforcement learning (DRL) in solving some typical resource management in network slicing scenarios, including radio resource slicing and priority-based core network slicing. In [11], an efficient slicing scheme combining offline reinforcement learning and heuristic algorithm is proposed to allocate wireless resources for eMBB and vehicle to X (V2X) slices. DQN was used for dynamic wireless virtual resource allocation in [12]–[14]. Reference [12] allocate radio resources by bandwidth ratio and each user can occupy the resources of all base stations(BSs). In [13], the allocation of resource blocks is refined. Each user can only connect to one BS, and the bandwidth and time slot of the occupied resource blocks are calculated according to the needs of users. In [14], DQN algorithm is improved to dynamic wireless resource allocation, and the discrete normalized advantage function (DNAF) was introduced into DQL, so that the DQL algorithm could converge faster.

References [6]–[14] only consider the performance of the RAN to allocate wireless resources, but the allocation of wireless resources will also affect the performance of the core network slice. Therefore, it is necessary to do some research on the resource allocation of the core network in [15]–[18]. The placement of intermediate boxes of service chain was considered in [15], which is also the premise of SFC mapping. References [16], [17] focus on the reliability of SFC deployment. In [16], a SFC deployment scheme which maximizes capital and operational expenditure is proposed to ensure reliability. In [17], a reliability-aware and delay-constrained (READ) routing optimization framework is established. READ includes a complex mixed integer linear programming that produces optimal VNF placement and traffic routing policies to maximize the reliability of network services and minimize E2E service latency. In [18], the VNF scheduling and flow control problems are formulated as mixed integer linear problems, which minimizes the entire VNF occupation time / wait time.

Just considering access network or core network resource allocation cannot achieve the best performance of E2E slicing, so that [19]–[23] research resource allocation from E2E network slicing. Reference [19] analyzed the application of E2E slicing in Industry 4.0, and proposed a slicing method for packet-switched industrial communication protocols. In [20], E2E network slicing modeling is carried out for the key tasks of high speed and high reliability, and the prototype hardware implementation is completed. References [21], [22] provide different E2E deployment strategies for three typical slices: eMBB slices, MMTC slices, and uRLLC slices. In [21], complex network theory is used to obtain the topology information of slices and infrastructure networks. The mapping process includes two steps: the placement of VNF and the selection of link paths. In [22], an E2E slicing two-layer framework for mobile edge computing (MEC) is proposed, in which the core network and transmission network are considered upwards, and the wireless access network is considered downwards. The framework converts tenant requirements into resource requirements. Reference [23] proposed an E2E network slicing framework that considers QoS, and proposed a dynamic wireless resource fragmentation scheme for the wireless field to maximize network utility. For the wired network field, when multiple traffic flows through network function virtualization (NFV) nodes. The BR-GPS dual resource slicing scheme is proposed to minimize the packet queuing delay of each flow on the node's outbound link. But the impact between wired and wireless networks is not considered.

Above literatures are all about the resource allocation in the network slice. However, most of these studies are based on the access network resource allocation or the service link mapping of the core network. The impact of core-side link mapping is not taken into account in wireless resource allocation. In this paper, DQN intelligent coordination slice resources are adopted to improve the E2E access success rate of system users.

III. SYSTEM MODEL

A. E2E MODEL

The E2E slicing model is shown in Fig. 1. The model consists of two parts, the access side and the core side. The access side mainly selects the BS for users, and the core side maps the user service chain. The ellipse represents a cell with multiple BSs in Fig. 1. There are three types of slices in the figure. Different types of devices will be connected to different types of slices. Different slice resources are represented by different stripes. Each type of slice has specific SFC that needs to be executed. The link from the user to the core network constitutes an E2E communication link.

In this paper, All the BSs is denoted by $N = \{1, 2, \ldots, |N|\}$. The transmission power of the user is denoted by P. The slices is denoted by $M = \{1, 2, \ldots, |M|\}$.

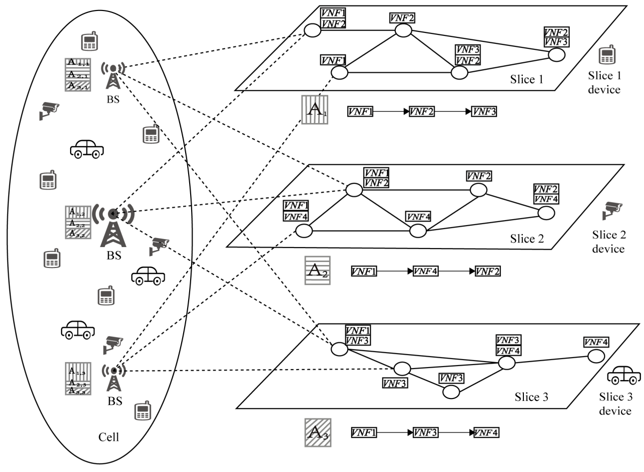


FIGURE 1. E2E network slicing model.

$U = \{1, 2, \ldots |U|\}$ be a set of all users. A set of users under slice m are denoted by U_m, and a specific user under slice m is denoted by u_m. The resources allocated to the slice m is denoted by A_m. $A_{m,n}$ represents the resources allocated to the BS n by the slice m. We should allocate wireless resources to slices and BSs from the perspective of E2E slicing.

From the network model in Fig. 1, it is known that when the user determines which BS to access, the initial VM of the core network link is determined. Each VM carries specific VNFs. For each slice, it needs to implement specific network functions, and the VNF functions it requires are determined and arranged in a certain order. This kind of VNF link arranged in a particular order is called SFC. As shown in Fig. 1, The SFCs of the three slices are:

Slice 1: $VNF1 - VNF2 - VNF3$

Slice 2: $VNF1 - VNF4 - VNF2$

Slice 2: $VNF1 - VNF3 - VNF4$

B. PROBLEM STATEMENT

The access network of this model considers the uplink cellular network. We assumes that the network has a perfect synchronization system and channel estimation. The system radio resource is represented as bandwidth B, which is divided into L in the frequency domain, each of which has bandwidth B_l. In the time domain, the radio resource is divided into scheduling frames, each consists of T subframes. The length of subframe is Δ_t. Therefore, the length of a scheduling frame is $\Delta_t \times T$. Hence, the smallest resource block (RB) is denoted as RB_l^t. We assume that each user only accesses one BS, and each user belongs to one type of slice. This paper considers rate-constrained and delay-constrained two types of slices. For rate-constrained slices, the minimum data rate of user u_m is denoted by $v_{u_m}^{\min}$. For delay-constrained slices, the maximum latency of user u_m is denoted by $\tau_{u_m}^{\max}$.

Assume that user u_m occupies one RB of BS n, $h_{u_m,n}$ represents the channel gain, and the path loss model refers to reference [24]. The data rate that a user occupy an RB is expressed as follows:

$$RB_{u_m,n} = RB_l^t \log_2\left(1 + \frac{P\left|h_{u_m,n}\right|^2}{\sigma B_l}\right) \tag{1}$$

where σ represents the spectral density of noise. The user has a priority order to select the BS, which obtain a large $RB_{u_m,n}$ value is preferentially selected. For rate-constrained users, the number of RBs for u_m of the BS n calculates as follow:

$$Nv = \left\lceil \frac{v_{u_m}^{\min}}{RB_{u_m,n}} \right\rceil \tag{2}$$

$$v_{u_m} = Nv \cdot RB_{u_m,n} \tag{3}$$

The v_{u_m} represents the data rate of u_m when we have Nv number of RBs. For delay-constrained users, the traffics is regarded as a queuing model. Assume that the packet arriving rate of the user is λ_u and the length of the packet is L_u bits. When the number of RB is Nt, we calculate the average delay by referring the [11] as follows:

$$\tau_{u_m} = \frac{1}{\frac{RB_{u_m,n} \cdot Nt}{L_u} - \lambda_u} = \frac{L_u}{RB_{u_m,n} \cdot Nt\text{-}\lambda_u \cdot L_u} \tag{4}$$

To meet the delay $\tau_{u_m}^{\max}$, The minimum number of RB can be calculated as follows:

$$Nt = \left\lceil \frac{L_u + \lambda_u \cdot L_u \cdot \tau_{u_m}^{\max}}{\tau_{u_m}^{\max} \cdot RB_{u_m,n}} \right\rceil \tag{5}$$

The higher $RB_{u_m,n}$ value of the user u_m access BS n, the higher priority of the user to choose the BS. During initialization, it is assumed that the user chooses the BS with the highest priority. The number of RB required by the user is $Nnum$. For rate-constrained slices, $Nnum = Nv$. For delay-constrained slices, $Nnum = Nt$. The resources allocated to the slice m can be expressed as:

$$A_m = L \cdot T \cdot \frac{\sum_{n \in N} \sum_{u_m \in U_m} Nnum}{\sum_{m \in M} \sum_{n \in N} \sum_{u_m \in U_m} Nnum}, \quad \forall\, m \tag{6}$$

During initialization, the resource allocation only considers the influence of the access side. The resources allocated to the BS n by the slice m are calculated as follows:

$$A_{m,n} = A_m \cdot \frac{\sum_{u_m \in U_m} Nnum}{\sum_{n \in N} \sum_{u_m \in U_m} Nnum}, \quad \forall\, m,\ n. \tag{7}$$

We use the binary variable $x_{u,n}$ to indicate whether the user u accesses the BS n, the access is 1, otherwise it is 0, which means as follows:

$$x_{u,n} = \begin{cases} 1 & user\ u\ access\ BS\ n \\ 0 & otherwise \end{cases}$$

After the initial wireless resource allocation is completed, the results will be used to achieve E2E resource allocation. On the wireless access side, the user selects a BS. On the core side, the user needs to achieve service chain mapping. Only the E2E link is mapped, the user successfully access.

On the core side, each network slice is composed of virtual machine (VM) with different VNFs, and users on the same slice need to implement the same SFC. Assume that the topology diagram of the VM in each slice is known, and the type and number of VNF in each VM are also known. When a user's SFC request comes, it is necessary to find a VM host for every VNF in the SFC, the premise of which is to meet the capacity requirements of virtual nodes and the bandwidth requirements of virtual links. As shown in Fig. 2, different colored cylinders represent different types of VNF. The SFC request is $VNF4 \rightarrow VNF1 \rightarrow VNF3 \rightarrow VNF2$. For each user in the slice we should find the VM to place the SFC.

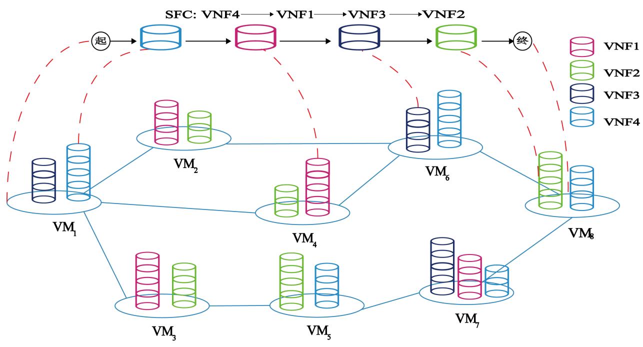


FIGURE 2. Core side service chain mapping system model.

The network topology graph composed of slice m can be represented by an undirected graph $G = (V, E)$, where V represents the set of virtual machine nodes in the slice, and E represents the set of connection bandwidth between links. SFC is represented by a directed graph, which connects VNF in a certain order. We need to find VM nodes and links in the network for each SFC to map. For a better description, the core side variables are summarized in Table 1.

TABLE 1. Definition of symbols.

Notation	Description
$h_{n,m}$	The VM number that BS n connects slice m
q_b	Bandwidth constraint of SFC
q_t	Delay constraint of SFC
F	$F = \{f_1, f_2 ... f_I\}$,It represents the composition of the SFC, f_1 represents the starting VM, $f_{i,i\neq 1}$ represents the type of VNF required for SFC node i
P^m	$P^m = \{1, 2 ... p ... \lvert P^m \rvert\}$. The set of SFC in slice m. p is the pth SFC in slice m.
V^m	$V^m = \left\{v_j^m \vert j = 1, 2, 3, ... J^m\right\}$. The set of VM in slice m. v_j^m is the jth VM in slice m
$c_{j,k}^m$	The number of VNF_k in v_j^m.
e_{j_1,j_2}^m	The bandwidth of the between $v_{j_1}^m$ and $v_{j_2}^m$.
d_{j_1,j_2}^m	The transmission delay of the connection between $v_{j_1}^m$ and $v_{j_2}^m$.
$y_{i,j}^p$	If f_i is hosted on v_j^m in the SFC p, $y_{i,j}^p$ equals 1. Otherwise 0
l_{j_1,j_2}^p	If the SFC p maps the link between $v_{j_1}^m$ and $v_{j_2}^m$, l_{j_1,j_2}^p equals 1. Otherwise 0
$z_{u_m,p}$	If the SFC p of user u_m maps successfully, $z_{u_m,p}$ equals 1. Otherwise 0

The BS which the user access determines that the VM position mapped by f_1. In this paper, the initial function f_1 of each SFC maps on $h_{n,m}$.

$$when\, j = \sum_{n=1}^{N} x_{u,n} h_{n,m} = f_1, \quad y_{1,j}^p = 1$$

In this paper, the bandwidth requirement of slices correspond to the number of RBs needed on the access side. Therefore, the value of q_b is equal to $Nnum$. q_t represents the delay demand of the user. The rate constrainted slice has no rigid requirement for delay so that q_t is set to a large positive integer. The delay constraint slice is set according to the actual demand.

C. E2E OPTIMIZATION

After introduction of the E2E slicing problem, it is necessary to match the users' E2E link by using the initially allocated wireless resources. We should determine which BS the user access and which link the core side maps to maximize the access rate of the entire system. Then adjust the resource allocation according to the access rate calculated and calculate the maximum access rate at this time, and iterate until the resource allocation can get maximum E2E access rate. We formulate this problem as follow:

$\mathcal{P}1$:

$$\max \quad \frac{\sum_{m=1}^{M}\sum_{u=1}^{U_m}\sum_{n=1}^{N} x_{u,n} \cdot z_{u_m,p}}{\sum_{m=1}^{M} U_m} \tag{8}$$

$$\text{s.t.} \quad \sum_{n=1}^{N} x_{u,n} \leq 1, \ \forall u \in U \tag{8a}$$

$$x_{u,n} \cdot \left(1 - x_{u,n}\right) = 0, \ \forall u \in U, \forall n \in N \tag{8b}$$

$$\sum_{u=1}^{U_m} x_{u,n} Nnum \leq A_{m,n}, \ \forall n \in N, \forall m \in M \tag{8c}$$

$$\tau_{u_m} \leq \tau_{u_m}^{\max}, \ \forall u \in U \tag{8d}$$

$$v_{u_m} \geq v_{u_m}^{\min}, \ \forall u \in U \tag{8e}$$

$$\sum_{j=1}^{J^m} y_{i,j}^p = 1, \quad \forall p \in P^m, \forall i \in F \tag{8f}$$

$$\sum_{p=1}^{P^m}\sum_{i=2}^{I} y_{i,j}^p \cdot f_i \leq c_{j,k}^m, \quad \forall j \in V^m \tag{8g}$$

$$\sum_{p=1}^{P^m} l_{j_1,j_2}^p q_b \leq e_{j_1,j_2}^m, \quad \forall j_1, j_2 \in V^m, j_1 \neq j_2 \tag{8h}$$

$$\sum_{j_1=1}^{J^m}\sum_{j_2=1}^{J^m} l_{j_1,j_2}^p d_{j_1,j_2}^m \leq q_t, \quad \forall j_1, j_2 \in V^m, j_1 \neq j_2 \tag{8i}$$

$$\sum_{j_1=1}^{J^m} l_{j,j_1} - \sum_{j_2=1}^{J^m} l_{j_2,j} = \begin{cases} 1, & i = 1, y_{i,j}^p = 1 \\ 0, & i = 2 \ldots I-1, y_{i,j}^p = 1 \\ -1, & i = I, y_{i,j}^p = 1 \end{cases} \tag{8j}$$

$$z_{u_m,p} = \begin{cases} 0, & if\ all\ l_{j_1,j_2}^p = 0 \\ 1, & else \end{cases} \tag{8k}$$

When the allocated resources are known. The variables $x_{u,n}$, l_{j_1,j_2}^p and $y_{i,j}^p$ need to be optimized to get the solutions. Only user access successfully in access side and core side at the same time. We can say that the user E2E communication succeeds. We define the E2E access rate as the number of users who successfully access simultaneously on the access side and the core side divided by the total number of users. (8) is the expression of E2E access rate. (8a) to (8e) are constraints on the access side. (8a) means that a user can only access one BS, and (8b) states that $x_{u,n}$ is a binary variable. (8c) shows that the resources occupied by users cannot exceed the allocated resources. (8d) and (8e) guarantee the delay and rate constraint, respectively. (8f) to (8k) are constraints on the core side. (8f) shows that each node of the SFC can only be mapped to one VM. (8g) indicates the capacity constraint of each VM. (8h) indicates the bandwidth constraint of each link. (8i) shows the delay constraint of the SFC. (8j) guarantees conserved node traffic. (8k) illustrates the relationship between SFC mapping success and link mapping.

IV. ALGORITHM DESIGN

A. AUTOMATIC RESOURCE ALLOCATION FOR E2E SLICING

According to the result of initial allocation of radio resources, we can calculate the E2E maximum access rate by solving P1. In this section, the resources of the slices and BSs need to be dynamically changed, so as to maximize the access rate of the E2E slices of the whole system. In this paper, DQN algorithm is used to adjust the radio resources.

The basic idea of DQN algorithm is the same as that of Q-learning algorithm, but the difference is that its Q-value is not calculated by state action pairs, but by a neural network. DQN can save the information of each interaction with the environment to an experience pool, and then select data from the experience pool to update the network parameters. In this paper, an improved DQN algorithm is adopted, which use two neural networks. The current neural network is used to update the network parameters and generate experience pool data. The other target neural network is used to calculate the Q-value. The parameters of the target neural network are copied from the current neural network at regular intervals. Two neural networks can reduce the correlation between the two networks and speed up the convergence.

Some important elements in DQN are designed as follows:

States: $s = (R_m, S_m)$. R_m represents the probability of successful access on the access side of slice m. S_m represents the ratio of the users who achieve E2E access to the users access on the access side. Users can access successfully is affected by two aspects: one is the access side resources are sufficient, the other is the core side nodes and link resources are sufficient. A higher S_m value means that most of the users can access successfully on the core side, a smaller value means that the users can't access because of the core side.

$$R_m = \frac{\sum_{n=1}^{N} \sum_{u_m=1}^{U_m} x_{u,n}}{U_m}, \quad \forall m \tag{9}$$

$$S_m = \frac{\sum_{n=1}^{N} \sum_{u_m=1}^{U_m} x_{u,n} \cdot z_{u_m,p}}{\sum_{n=1}^{N} \sum_{u_m=1}^{U_m} x_{u,n}}, \quad \forall m \tag{10}$$

Action: The action are a set of discrete percentages $a = [-10\%, -8\%, -6\%, -4\%, -2\%, 0,2\%, 4\%, 6\%, 8\%, 10\%]$. The negative value indicates a decrease in resources, 0 indicates the slice resources keep the same, and the positive value indicates an increase in resources.

Reward: The reward is defined as the total access rate of the system.

$$r = \frac{\sum_{m=1}^{M} \sum_{u=1}^{U_m} \sum_{n=1}^{N} x_{u,n} \cdot z_{u_m,p}}{\sum_{m=1}^{M} U_m} \tag{11}$$

Q-value update: We use the Bellman equation to update the value. the expression is as follows:

$$Q(s,a) = r + \gamma_{a' \in A}^{\max} \left(Q\left(s', a'\right)\right) \tag{12}$$

where $\gamma \in [01]$ is a discount factor.

Next state: After selecting the action, the slice resource is updated to A_m^{t+1}, and the resource needs to be reassigned to BS to determine $A_{m,n}^{t+1}$, so that the reward and the R_m^{t+1}, S_m^{t+1} can be calculated by solving P1.

Resource update: After the action is selected, the slice-level resources are updated as follows:

$$A_m^{t+1} = A_m^t (1 + a_m), \quad \forall m \tag{13}$$

By using (13) to allocate resources to slices, the sum of resources allocated by all slices may be greater or less than the total resources. Therefore, we use (14) to normalize the slice allocation resources so that the sum of resources remains the same. If a new slice occurs, (6) is used to initialize resource allocation for all slices.

$$A_m^{t+1} = A_m^{t+1} \times \frac{L \cdot T}{\sum_{m=1}^{M} A_m^{t+1}}, \quad \forall m \tag{14}$$

After the slice-level resource update is completed, the slice resources need to be feedback to the BS. The relative access success rate of users of slice m in BS n is defined as follows:

$$S_{m,n} = \frac{\sum_{u_m=1}^{U_m} x_{u,n} \cdot z_{u_m,p}}{\sum_{u_m=1}^{U_m} x_{u,n}}, \quad \forall m, n \tag{15}$$

The wireless resources allocated to BS by slicing is updated as follows:

$$\begin{cases} A_{m,n}^{t+1} = A_{m,n}^t + \left(A_m^{t+1} - A_m^t\right) \cdot \dfrac{S_{m,n}}{\sum_{n=1}^{N} S_{m,n}}, & if\, a_m \geq 0 \\ A_{m,n}^{t+1} = A_{m,n}^t + \left(A_m^{t+1} - A_m^t\right) \cdot \dfrac{1 - S_{m,n}}{\sum_{n=1}^{N} \left(1 - S_{m,n}\right)}, & else \end{cases} \tag{16}$$

At this point, the slice-level and BS-level resource updates are completed. First, we can use the allocated resources to bring into P1 and solve the E2E resource allocation and link mapping. Then we can get the r_t and the R_m^{t+1}, S_m^{t+1}, $S_{m,n}^{t+1}$.

Q network structure: In DQN, the neural network is used instead of the Q table in Q learning. The Q network in this paper is set as a forward feedback neural network. The input of the network is the state of the slice, and the output is the Q value of the state action pair. The number of hidden layers and the number of neurons are obtained through trial and testing. The initialization parameters of the current neural network and the target neural network are the same.

DQN process: In our proposed DQN resource allocation algorithm, training and testing are simultaneously performed. In the training phase, substitute the initial allocated resources into the problem P1 and calculate the current state. Then we use the greedy strategy to explore an action to get a new resource allocation strategy and substitute the allocation strategy into the problem P1 again to get the reward at this time and the next state. And store the current state, the next state, action and reward into the experience pool. And finally repeat the above behavior. After a period of time, the data of the mini-batch is selected from the experience pool for network training. The test is always running during the training process. According to the state obtained by the resource allocation strategy at the previous moment, the current trained network is used for action selection to obtain a new resource allocation strategy. As the network is trained better and better, the decisions made will be better and better.

Fig. 3 describes the process of DQN dynamic resource allocation. The model is composed of two parts. The first part is that the network determines the resource allocation ratio and adjusts network parameters based on environmental feedback. The second part is the environment, which adopts heuristic algorithm to achieve resources E2E match and evaluate access rate to feedback to the network. The entire system has been running to allocate resource in order to adapt the changes in the environment. The detailed process is shown in Algorithm 1.

B. SOLUTIONS OF THE SUB-PROBLEMS

In order to solve the DQN resource allocation scheme, we must solve the feedback of the environment when the resources are adjusted. This feedback is the reward value of DQN. After the DQN network action is executed, the new resource allocation ratio can be obtained and substituted into P1 to solve the environmental feedback. P1 is a

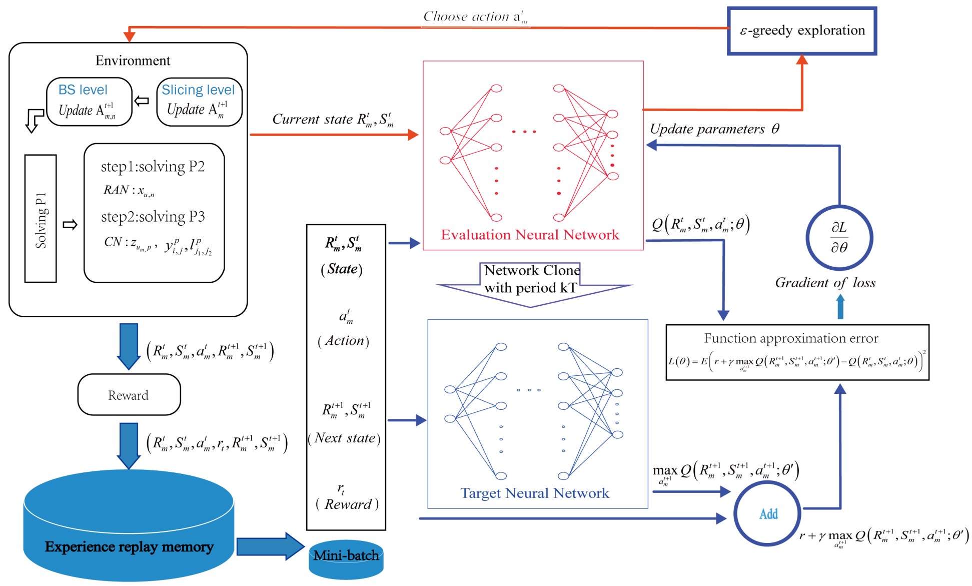


FIGURE 3. **DQN dynamic resource allocation model.**

0-1 program problem, and there are multiple variables. This problem is difficult to solve. In this paper, we split P1 into two sub-problems on the access side and the core side. First solve the maximum access rate of the sub-problem on the access side, then substitute the result into the core side to solve the maximum access rate on the core side. Let the two sub-problems get the maximum access rate to ensure maximum E2E access rate.

1) ACCESS SIDE SUB-PROBLEM

On the access side, when the wireless resource of the slice is determined, the slices are isolated from each other. Therefore, the objective functions and constraints on the access side can be decoupled to each slice. For each type of slice, the target on the access side is the maximum number of access users. The access side sub-problem of rate constrainted slice P2 as follows. The delay constrainted slice change the rate constraint to the delay constraint.

$$\mathcal{P}2: \quad \max \quad \sum_{u=1}^{U_m} \sum_{n=1}^{N} x_{u,n} \tag{17}$$

$$\text{s.t.} \quad \sum_{n=1}^{N} x_{u,n} \leq 1, \ \forall u \in U \tag{17a}$$

$$x_{u,n} \cdot \left(1 - x_{u,n}\right) = 0, \ \forall u \in U, \forall n \in N \tag{17b}$$

$$\sum_{u=1}^{U_m} x_{u,n} Nnum \leq A_{m,n}, \ \forall n \in N \tag{17c}$$

$$v_{u_m} \geq v_{u_m}^{\min}, \ \forall u \in U \tag{17d}$$

The above sub-problem is a NP-Hard problem. Suppose that the BS is a backpack, the user is an item, the weight of the item is *Nnum*, and the revenue is the system capacity. This problem can be converted into a 0-1 multiple backpack problem. However, users access different BS will change the number of RB required so that it can not be solved by knapsack algorithm. Therefore, we propose a dynamic programming algorithm based on backpack algorithm. The algorithm idea is shown in Algorithm 2.

The idea of Algorithm 2 is to put all users in a public candidate pool, and each BS will select users based on the 0-1 backpack algorithm to maximize the BS capacity. If a user is selected by multiple BSs, the user selects the BS with the highest priority. The BSs select users from the public candidate pool according to the above method until the users don't select by multiple BSs. The complexity of Algorithm 2 is $O(U \times L \times T)$, which means the product of the number of users and the number of blocks.

2) CORE SIDE SUB-PROBLEM

After the user selects BS on the access side, the core side needs to realize SFC mapping to complete the users' E2E communication. The BS connected by the user determines the first VM position of the SFC map. Then users need to complete the matching of the entire SFC starting from the first node. On the core side, each SFC needs to be mapped to maximize user capacity so that P1 can be solved. The SFC mapping of each type of slice is isolated and unaffected. First, we need to classify all SFCs according to the slice type. Then, we can get the SFC mapping sub-problem P3 is as follows:

In order to quickly solve the above problem and obtain a better solution, the heuristic algorithm is designed as algorithm 3.

Algorithm 3 adopts different mapping algorithms for two types of slices on the core side. $m = 1$ means rate constrained slice, and $m = 2$ means delay constrained slice.

Algorithm 1 Dynamic Resource Allocation Based on DQN

Input: BS and users information, user RB requirements, Total bandwidth, core-side topology

Output: Slice resource allocation strategy $A_m, A_{m,n}$, E2E link mapping strategy $x_{u,n}$, $y_{i,j}^p$, $l_{j1,j2}^p$

Initialize experience pool E_D, mini-batch M_D, current Q_e network and the target Q_t network, the discount factor γ and epsilon ε
Calculate the initial slice resource A_m by (6), calculate BS level resource $A_{m,n}$ by (7), solve P1 to get the E2E mapping result, use (9), (10) to get the DQN initial state $s^1 = (R_m, S_m)$
while true **do**
  Generate a random number rand()
  if rand()<ε **then**
    Random generating action
  else
    Select the action that has maximum Q-value by current Q_e network
  end if
  Update the next time A_m^{t+1} by (13), (14), update $A_{m,n}^{t+1}$ by (15), (16), solve P1 to get the E2E mapping result, calculate (9), (10), (11) to get R_m^{t+1}, S_m^{t+1}, r^t
  Store $\left(R_m, S_m, a^t, r^t, R_m^{t+1}, S_m^{t+1}\right)$ in experience pool
  if activated by timer **then**
    Pick M_D of samples from the experience pool
    for all samples **do**
      Input R_m, S_m, a^t into Q_e to calculate current Q-value
      Input R_m^{t+1}, S_m^{t+1} into Q_t, calculate target Q-value by (12)
      Update the current Q_e network parameters with the minimum mean square error of the current Q-value and the target Q-value
    end for
  end if
  Copy Q_e network parameters to Q_t at certain intervals
end while

Algorithm 2 Access Side Dynamic Programming Algorithm

Input: User information, BS level resource allocation $A_{m,n}$, and the number of resource blocks *Nnum* required by the user

Output: user selects BS policy $x_{u,n}$

Initializes a public set of candidate users $\bar{U}_m = \{1, 2, \ldots. |U_m|\}$ for all BSs
Initializes a preemptive set of users $\tilde{U}_{m,n} = \varnothing n \in N$ for each BS
while true **do**
  for all BSs $n \in N$ **do**
    The 0-1 knapsack algorithm is used to select the access users of BS n from $\bar{U}_m$, denoted as $\tilde{U}_{m,n}$
  end for
  if the user is not selected by multiple BSs, $\bigcap_{n\in N} \tilde{U}_{m,n} = \varnothing$ **then**
    Output $x_{u,n}$
  else
    Calculate the *Nnum* of the user in different BS
    Adds the user to $\tilde{U}_{m,n}$ of BS with the minimum *Nnum* and remove it from other BSs
    Remove the selected users from $\bar{U}_m$
  end if
end while

First, the SFCs of the same slice are prioritized. The SFC mapping is performed point by point. Each candidate node is evaluated and the node with the highest evaluation score is selected to be mapped in turn. We can say that the SFC maps success until the all NFV maps success and the bandwidth and delay constraints are meet. The complexity of Algorithm 3 is proportional to the size of the network, the number of SFCs and the length of SFC. For the bandwidth constrained evaluation function, we should consider the path is short, the average remaining bandwidth is large, and the remaining resources of the node are enough. For the delay constrained evaluation function, we should conside the link delay is short and the remaining resources of the node are enough. In this way, the node resources can be balanced, which facilitates the access of the subsequent links. Therefore, we can access more users.

$\mathcal{P}3:$

$$\max \quad \sum_{u=1}^{U_m}\sum_{n=1}^{N} x_{u,n}\cdot z_{u_{m,p}} \tag{18}$$

$$\text{s.t.} \quad \sum_{j=1}^{J^m} y_{i,j}^p = 1, \quad \forall p \in P^m, \forall i \in F \tag{18a}$$

$$\sum_{p=1}^{P^m}\sum_{i=2}^{I} y_{i,j}^p \cdot f_i \le c_{j,k}^m, \quad \forall j \in V^m \tag{18b}$$

$$\sum_{p=1}^{P^m} l_{j_1,j_2}^p q_b \le e_{j_1,j_2}^m, \quad \forall j_1, j_2 \in V^m, j_1 \ne j_2 \tag{18c}$$

$$\sum_{j_1=1}^{J^m}\sum_{j_2=1}^{J^m} l_{j_1,j_2}^p d_{j_1,j_2}^m \le q_t, \quad \forall j_1, j_2 \in V^m, j_1 \ne j_2 \tag{18d}$$

$$\sum_{j_1=1}^{J^m} l_{j,j_1} - \sum_{j_2=1}^{J^m} l_{j_2,j} = \begin{cases} 1, i = 1, y_{i,j}^p = 1 \\ 0, i = 2\ldots I-1, y_{i,j}^p = 1 \\ -1, i = I, y_{i,j}^p = 1 \end{cases} \tag{18e}$$

$$z_{u_{m,p}} = \begin{cases} 0, \textit{if all } l_{j_1,j_2}^p = 0 \\ 1, \textit{else} \end{cases} \tag{18f}$$

Algorithm 3 Core Side SFC Mapping Algorithm

Input: SFC request P^m, network topology information V^m, $c_{j,k}^m$, e_{j_1,j_2}^m, d_{j_1,j_2}^m

Output: SFC mapping results $y_{i,j}^p$, l_{j_1,j_2}^p, $z_{u_m,p}$, Map successful users $Snum$

1: **for** two types of slices **do**
2: Sort the SFC request to get new P^m
3: **for** SFCs $p \in P^m$ **do**
4: Remove the edges in the network topology that do not meet the q_b to get a new network topology, define variable v_{st}^m=f_1
5: Find the candidate VM node set V' start from v_{st}^m
6: **if** candidate node set $V' = \phi$ **then**
7: SFC mapping failed, goto step 3
8: **end if**
9: **if** $m = 1$ **then**
10: Calculate the shortest hops hop_i to all candidate nodes $v_i{}' \in V'$ using Dijkstra's algorithm, calculate the remaining average bandwidth of the shortest hops B_i,calculate the number of remaining functions of candidate nodes rec_i
11: Evaluate $v_i{}' \in V'$ nodes $w(v_i{}') = \frac{1}{hop_i} \cdot \frac{B_i}{\max_{j \in V'}(B_j)} \cdot \frac{rec_i}{\max_{j \in V'}(rec_j)}$
12: **else**
13: Calculate the shortest delay $delay_i$ to all candidate nodes $v_i{}' \in V'$ using Dijkstra algorithm,calculate the number of remaining functions of candidate nodes rec_i
14: Evaluate $v_i{}' \in V'$ nodes $w(v_i{}') = (1 - \frac{delay_i}{\max_{j \in V'}(delay_j)})(\frac{rec_i}{\max_{j \in V'}(rec_j)})$
15: **end if**
16: Select VM node with large $w(v_i{}')$ to place function which denoted as v_{nt}^m. And Let $\mathrm{l}_{st,nt}^p = 1$, $v_{st}^m = v_{nt}^m$
17: **if** SFC mapping ends and delay q_t requirement is met **then**
18: Update network information, $z_{u_m,p} = 1$ and $Snum = Snum + 1$
19: **else**
20: Goto step 11
21: **end if**
22: **end for**
23: **end for**

V. SIMULATION AND RESULT ANALYSIS

In order to evaluate the performance of the scheme in this paper, we use MATLAB and apply framework design algorithms such as Tenseflow and Keras to build the simulation model of the proposed algorithm. The simulation parameters are given based on 5G network standard. Considers that rate constrained and delay constrained users are uniform distributed in a cell with a radius of 500m. The cell has 1 macro BS and 4 micro BSs. Each slice needs to meet the minimum QoS requirements such as rate or delay constraints. Our proposed DQN resource allocation algorithm is always learning the changes of the environment and making corresponding resource allocation adjustments in the real environment. But in order to observe whether our trained network can make good decisions according to changes in the environment, we observe 300 slicing periods in each simulation, and one slicing cycle is 5 ms. This algorithm is also applicable to multi-cell scenario. The resource allocation process is the same as a single cell. The simulation parameters are shown in Table 2.

TABLE 2. Simulation parameters configuration.

Parameters name	Value
System bandwidth B	20 MHz
Number of slices M	2
Number of BS N	5
Number of users U	200
Number of users in each slice	60˜120
User transmit power P	200 mw
Cell coverage area	500 $m \times 500\ m$
BS transmit power	Macro BS:40 dBm; Micro BS:30 dBm
Noise spectrum density σ	$-174\ dBm/Hz$
Minimum rate constraint $v_{u_m}^{\min}$	100 Mb/s
Maximum delay constraint $\tau_{u_m}^{\max}$	1 ms
Packet length L_u	100 kb
Packet arrival rate λ_u	120˜150 $packet/s$
Number of VM per slice J	8
Number of VNF species K	6
Experience pool size	10000
Mini-batch size	100
Discount factor	0.01
Greed factor ε	0.2
Minimum mean square error target	0.0001
Slicing period	5 ms

In this paper, we have proposed a DQN resource allocation scheme. We compare our proposed scheme with the other two schemes, the optimal access side resource allocation (ASO) and the resource allocation based on the BS coverage (BSC). The ASO scheme indicates that each user chooses the closest BS to allocate resources on the access side. The BSC scheme indicates that each BS allocates resources according to the coverage area of the BS. We will observe the E2E access rate over time in the three schemes.

Fig. 4 shows the different E2E access rates of the three schemes in the static environment which users are stationary. The E2E access success rate of the proposed DQN scheme is over 98% in the static. The E2E access rate of the ASO and BSC schemes is not higher than 94%. ASO scheme is slightly higher than the BSC scheme. The distribution of users may not be distributed according to the coverage size of the BS in the static environment. The performance of the DQN scheme is better than the other schemes. In Fig. 4, we can find that the DQN scheme can still automatically adjust the resources until reaching the best solution in the static environment. But other schemes can't achieve dynamically adjust so that the access rate won't change. The most important thing is that the DQN designed in this paper can directly make use of the access side and the core side situation to automatically adjust the allocation of resources at the next moment. In other words,

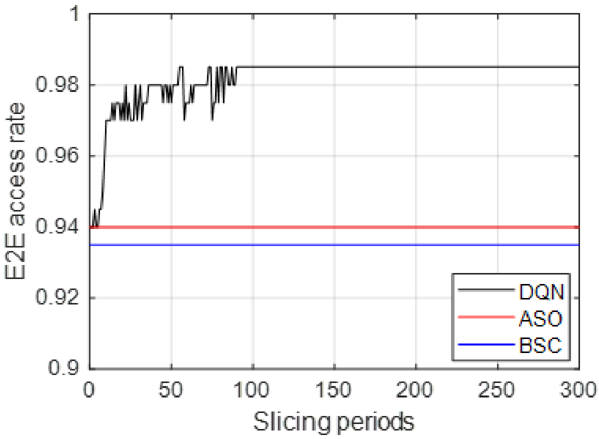


FIGURE 4. Static environment access rate changes with time.

DQN allocates resources from an E2E perspective which is different from the other two schemes. It only considers the access side.

We have confirmed that the DQN scheme has certain advantages for static resource adjustment. However, users and devices are dynamically mobile in the real environment. In order to study the feasibility of DQN scheme in a mobile scenario. Fig. 5 depicts that users move randomly in the cell. The moving speed is less than the maximum rate we define. The maximum rates we studied were $v_{\max} = 30\ m/\min$, $v_{\max} = 60\ m/\min$ and $v_{\max} = 90\ m/\min$ respectively. We also compare DQN scheme with ASO and BSC schemes.

When the $v_{\max} = 30\ m/\min$ or $v_{\max} = 60\ m/\min$, the E2E access rate of user is higher than 95%. Compared with other schemes the DQN schemes is increased by at least 4%. When the $v_{\max} = 60\ m/\min$, the access rate of the three schemes has declined, but the DQN scheme is still better than other schemes. With the speed increases, we can find that the fluctuation of the access rate of the DQN scheme is smaller than that of the other two schemes. Because the DQN scheme comprehensively considers the whole E2E resources for dynamic adjustment. The ASO scheme fluctuates greatly with the change of the dynamic environment, because it does not consider the influence of the core side and simply satisfies the maximum access of the access side.

In order to compare the advantages of the proposed scheme more clearly, Fig. 6 compares rate constrained slices and delay constrained slices when users' maximum movement speed is less than 60. We evaluate the performance of 300 slicing cycles in the dynamic environment. Compare the average values of R_m, S_m and E2E success rate of three schemes.

Fig. 6 show the schemes of ASO and BSC can ensure a large access rate on the access side. But the average value of S_m is much lower than that of the DQN scheme, this means users can't access on the core side without take into account the influence of E2E. We can find that the average access rate of DQN scheme is higher than 97% for all slices. Compared

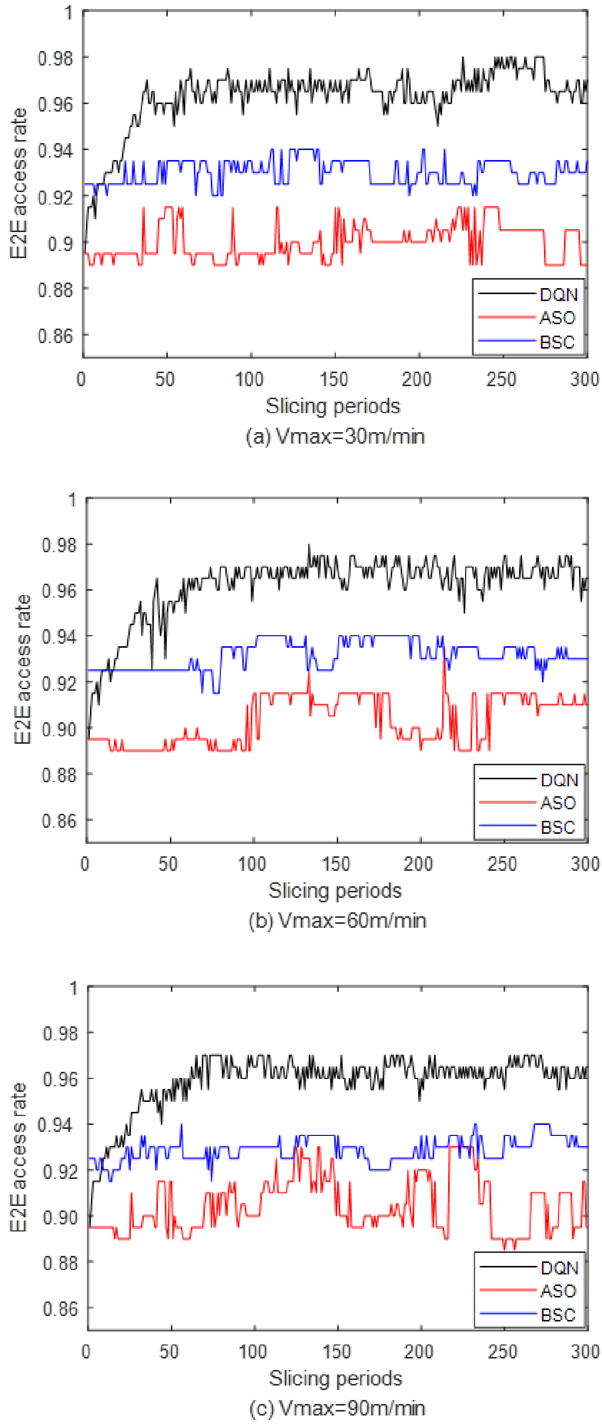


FIGURE 5. Dynamic environment access rate changes with time.

with ASO scheme, the average access rate is increased by 9% for delay constrained slices and 5% for rate constrained slices. The performance of the DQN scheme on the access side is not as good as that of the other two schemes, but users have a high probability of success on the core side. From Fig. 6, it is shown that the DQN dynamic resource allocation scheme can significantly improve the capacity of E2E system.

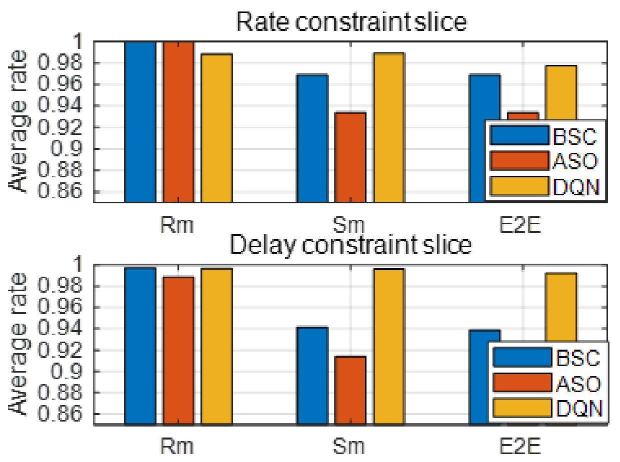


FIGURE 6. Comparison access rates for the two slices.

VI. CONCLUSION

In this paper, we proposed the DQN based autonomous resources allocation framework for the next generation mobile networks. The proposed scheme take into account the influence of E2E to ensure the maximum access rate of the whole system. In this system, we consider rate constrained slices and delay constrained slices. Different slices have different constraints and resource requirements. Reasonable resource allocation and dynamic adjustment between slices make the system access more users. DQN was used by the slices to adjust resources. The reward of DQN is the E2E access rate, which was solved by break the P1 into two sub-problem. Different algorithms are designed to solve sub-problem so that we can get the reward of DQN and autonomously adjust resource. The simulation results show that DQN can dynamically change the resource allocation according to the system access rate in static or dynamic environment. The system access rate can be higher than 98% in static environment which is the best compared with ASO and BSC schemes. When users moves below 60 $m/\min$, the average access rate of DQN scheme is higher than 97%. Compared with ASO schemes, the average access rate is increased by 9% for delay constrained slices and 5% for rate constrained slices.

REFERENCES

[1] S. Zhang, "An overview of network slicing for 5G," *IEEE Wireless Commun.*, vol. 26, no. 3, pp. 111–117, Jun. 2019.

[2] P. Rost, C. Mannweiler, D. S. Michalopoulos, C. Sartori, V. Sciancalepore, N. Sastry, O. Holland, S. Tayade, B. Han, D. Bega, D. Aziz, and H. Bakker, "Network slicing to enable scalability and flexibility in 5G mobile networks," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 72–79, May 2017.

[3] S. E. Elayoubi, S. B. Jemaa, Z. Altman, and A. Galindo-Serrano, "5G RAN slicing for verticals: Enablers and challenges," *IEEE Commun. Mag.*, vol. 57, no. 1, pp. 28–34, Jan. 2019.

[4] H. Halabian, "Distributed resource allocation optimization in 5G virtualized networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 3, pp. 627–642, Mar. 2019.

[5] A. Fendt, S. Lohmuller, L. C. Schmelz, and B. Bauer, "A network slice resource allocation and optimization model for End-to-End mobile networks," in *Proc. IEEE 5G World Forum (5GWF)*, Silicon Valley, CA, USA, Jul. 2018, pp. 262–267.

[6] H. Zhang, N. Liu, X. Chu, K. Long, A.-H. Aghvami, and V. C. M. Leung, "Network slicing based 5G and future mobile networks: Mobility, resource management, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 138–145, Aug. 2017.

[7] J. Tang, B. Shim, and T. Q. S. Quek, "Service multiplexing and revenue maximization in sliced C-RAN incorporated with URLLC and multicast eMBB," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 4, pp. 881–895, Apr. 2019.

[8] C. She, C. Yang, and T. Q. S. Quek, "Cross-layer optimization for ultra-reliable and low-latency radio access networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, p. 127141, Jan. 2018.

[9] C. She, C. Yang, and T. Q. S. Quek, "Joint uplink and downlink resource configuration for ultra-reliable and low-latency communications," *IEEE Trans. Commun.*, vol. 66, no. 5, pp. 2266–2280, May 2018.

[10] R. Li, Z. Zhao, Q. Sun, C.-L. I, C. Yang, X. Chen, M. Zhao, and H. Zhang, "Deep reinforcement learning for resource management in network slicing," *IEEE Access*, vol. 6, pp. 74429–74441, 2018.

[11] H. D. R. Albonda and J. Perez-Romero, "An efficient RAN slicing strategy for a heterogeneous network with eMBB and V2X services," *IEEE Access*, vol. 7, pp. 44771–44782, 2019.

[12] G. Sun, Z. T. Gebrekidan, G. O. Boateng, D. Ayepah-Mensah, and W. Jiang, "Dynamic reservation and deep reinforcement learning based autonomous resource slicing for virtualized radio access networks," *IEEE Access*, vol. 7, pp. 45758–45772, 2019.

[13] G. Sun, K. Xiong, G. O. Boateng, D. Ayepah-Mensah, G. Liu, and W. Jiang, "Autonomous resource provisioning and resource customization for mixed traffics in virtualized radio access network," *IEEE Syst. J.*, vol. 13, no. 3, pp. 2454–2465, Sep. 2019.

[14] C. Qi, Y. Hua, R. Li, Z. Zhao, and H. Zhang, "Deep reinforcement learning with discrete normalized advantage functions for resource management in network slicing," *IEEE Commun. Lett.*, vol. 23, no. 8, pp. 1337–1341, Aug. 2019.

[15] J. Liu, Y. Li, Y. Zhang, L. Su, and D. Jin, "Improve service chaining performance with optimized middlebox placement," *IEEE Trans. Serv. Comput.*, vol. 10, no. 4, pp. 560–573, Jul. 2017.

[16] J. Sun, G. Zhu, G. Sun, D. Liao, Y. Li, A. K. Sangaiah, M. Ramachandran, and V. Chang, "A reliability-aware approach for resource efficient virtual network function deployment," *IEEE Access*, vol. 6, pp. 18238–18250, 2018.

[17] L. Qu, C. Assi, K. Shaban, and M. J. Khabbaz, "A reliability-aware network service chain provisioning with delay guarantees in NFV-enabled enterprise datacenter networks," *IEEE Trans. Netw. Service Manage.*, vol. 14, no. 3, pp. 554–568, Sep. 2017.

[18] L. Qu, C. Assi, and K. Shaban, "Delay-aware scheduling and resource optimization with network function virtualization," *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3746–3758, Sep. 2016.

[19] A. E. Kalor, R. Guillaume, J. J. Nielsen, A. Mueller, and P. Popovski, "Network slicing in industry 4.0 applications: Abstraction methods and end-to-end analysis," *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5419–5427, Dec. 2018.

[20] V. Petrov, M. A. Lema, M. Gapeyenko, K. Antonakoglou, D. Moltchanov, F. Sardis, A. Samuylov, S. Andreev, Y. Koucheryavy, and M. Dohler, "Achieving End-to-End reliability of mission-critical traffic in softwarized 5G networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 3, pp. 485–501, Mar. 2018.

[21] W. Guan, X. Wen, L. Wang, Z. Lu, and Y. Shen, "A service-oriented deployment policy of End-to-End network slicing based on complex network theory," *IEEE Access*, vol. 6, pp. 19691–19701, 2018.

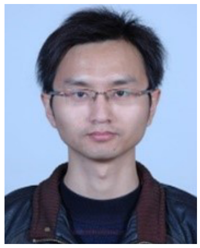
[22] H.-T. Chien, Y.-D. Lin, C.-L. Lai, and C.-T. Wang, "End-to-End slicing as a service with computing and communication resource allocation for multi-tenant 5G systems," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 104–112, Oct. 2019.

[23] Q. Ye, J. Li, K. Qu, W. Zhuang, X. S. Shen, and X. Li, "End-to-end quality of service in 5G networks: Examining the effectiveness of a network slicing framework," *IEEE Veh. Technol. Mag.*, vol. 13, no. 2, pp. 65–74, Jun. 2018.

[24] W. Jiang, G. Feng, S. Qin, T. S. P. Yum, and G. Cao, "Multi-agent reinforcement learning for efficient content caching in mobile D2D networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1610–1622, Mar. 2019.

TAIHUI LI received the B.S. degree from the Changsha University of Science and Technology, Changsha, China, in 2018. She is currently pursuing the M.S. degree in communication and information system with the Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China. Her research interests include mobile communications and resource allocation. She received the Third-Class scholarship from NJUPT and the Second Price in Smart Service System (3S) from China Communication Association.

XIAORONG ZHU received the Ph.D. degree in wireless communications from Southeast University, Nanjing, China, in 2008. She was a Postdoctoral Researcher with The Chinese University of Hong Kong, in 2008 and 2009. She is currently a Professor with the Nanjing University of Posts and Telecommunications. Her research interests include 5G/6G networks and the Internet of Things.



XU LIU received the Ph.D. degree in communication and information system from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2010. He is currently an Associate Professor with the College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications. His current research interests include wireless communication and signal processing.