IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 29, 2018, accepted October 3, 2018, date of publication October 17, 2018, date of current version November 9, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2876544

Adaptive Virtual Resource Allocation in 5G Network Slicing Using Constrained Markov Decision Process

LUN TANG, QI TAN, YINGJIE SHI, CHENMENG WANG, AND QIANBIN CHEN, (Senior Member, IEEE)

Key Laboratory of Mobile Communication, School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

Corresponding author: Qianbin Chen (cqb@cqupt.edu.cn)

This work was supported in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-M201800601, in part by the National Natural Science Foundation of China under Grant 61571073, and in part by the Chongqing Science and Technology Commission through the Generic Key Technology Innovation Projects for Key Industries under Grant cstc2015zdcy-ztzx40008.

ABSTRACT Network virtualization technology is generally envisaged as a promising technology to consequently satisfy various types of service requirements. On the other hand, non-orthogonal multiple access (NOMA) technology has the potential to significantly increase the spectral efficiency of the system. However, previous works that jointly address these two issues have not considered the dynamic resource allocation issue in this context. In this paper, we propose a slice-based virtual resources scheduling scheme with NOMA technology to enhance the quality-of-service (QoS) of the system. We formulate the power granularity allocation and subcarrier allocation strategies into a constrained Markov decision process problem, aiming at the maximization of the total user rate. In order to further avoid the curse of dimensionality and the expectation calculation in the optimal value function, we develop an adaptive resource allocation algorithm based on approximate dynamic programming to solve the problem. Extensive simulation works have been conducted under various system settings, and the results demonstrate that the proposed algorithm can significantly reduce the outage probability and increase the user data rate.

INDEX TERMS 5G slice, adaptive virtual resource allocation, constrained Markov decision process, approximate dynamic programming, NOMA.

I. INTRODUCTION

The fifth generation (5G) mobile cellular systems are meant to create an all-encompassing and inter-connected environment and revolutionize the way of communication, in which the tremendous growth in the number of users can be natively supported while a certain level of quality-of-service (QoS) is guaranteed [1]. The rising popularity of smart phones and portable wireless devices result in a variety of QoS requirements of 5G mobile network. The services of 5G cellular networks are typically classified into three main categories [2], namely Enhance Mobile Broadband (eMBB), massive Machine Type of Communication (mMTC) and Ultra Reliable Low Latency Communication(uRLLC). In order to satisfy various QoS requirements of different services, the system architecture must be extremely flexible and adaptive [3]. To support the aforementioned heterogeneous services, wireless network virtualization is considered to be the key enabler to enhance wireless networks with the desired flexibility to provide the required QoS for each category of service [4]. The wireless network virtualization enables the radio resources to be abstracted and sliced into several virtual resources, so that the network infrastructure can be shared by different services [5]. Therefore, network slices, the most attractive and efficient solution to provide the required QoS, is considered the key enabler to support such functional and operational diversity. Specifically, 3GPP defines the network slice as "A logical network that provides specific network capabilities and network characteristics" [6]. A network slice consists of a radio access network (RAN) and a core network (CN) [7]. In this paper, we employ RAN slicing to partition different components of the radio access resources. From a service perspective,

the expected behavior of RAN is parameterized with a set of QoS attribute [8]. During the past years, the 5G network slicing concept has been thoroughly studied. In the context of resource isolation for network slicing, Nojima *et al.* [9] introduce two resource allocation methods, which are the conventional ordinary packet scheduling algorithm with a slight modification and dynamic resource block (RB) allocation. Parsaeefard *et al.* [10] propose a joint BS assignment, subcarrier allocation, and power allocation algorithm to maximize the network sum rate, while satisfying the minimum reserved traffic rate of each slice and considering the downlink dynamic resource allocation in multi-cell virtualized wireless networks (VWNs) through orthogonal frequency division multiple access(OFDMA)system. Tang *et al.* [11] investigate the joint sub-carrier allocation and caching placement approach for two network slices, and an average delay optimization problem for one slice with user data rate guarantee for the other slice is formulated. The work in [12] establishes an analytical model for the admissibility region of 5G networks based on the semi-Markov decision process, and Q-learning was utilized to design an adaptive dynamic resource allocation optimization scheme to improve system performance.

Moreover, efficient spectrum exploitation is required in response to a surge in network traffic volume and densification of end devices [13]. However, the traditional technologies of slicing such as Spectrum Planning do not consider the power domain of resources [14], as such, this paper adopts NOMA system to improve the spectral efficiency. In NOMA, multiple users can access the same system, the same code domain, and the same frequency resources simultaneously. By assigning different power levels to multiple users and implementing Successive Interference Cancelation (SIC), multi-user detection can be achieved [15]. The works in [16] and [17] study virtual resource allocation in NOMA system. In [18], users access different network slices according to service types, and the system performs power allocation with given channel assignment under average CSI for different slices. This research proposes a method to jointly optimize power allocation and channel assignment by incorporating the matching algorithm with the optimal power allocation, taking the max-min fairness into consideration. The scheme outperforms the traditional multiple access schemes (MA) resource scheduling method by significantly improving the performance of the worst users.

Generally speaking, the system needs to allocate resources dynamically according to different services demands. However, most of the existing research works only investigate one type of service requirement on the same time slot for resource allocation. For the issues above, this paper proposes a CMDP-based network slice adaptive virtual resource allocation algorithm. Moreover, most studies cannot guarantee the efficiency of resource allocation when the number of users takes large, which may lead to the curse of dimensionality. Therefore, we adopt the approximate dynamic programming to avoid this problem. The main contributions of this paper are summarized as follows.

- This paper focuses on the dynamic virtual resource allocation problem in 5G slicing network with the downlink NOMA system, considering the user outage probability and the queue caching in slice to guarantee the Qos of slices. So as that, this paper formulates this problem as a CMDP to establish a dynamic optimization model using the total rate of slices as a reward.
- For the purpose of avoiding calculating the expectation in the optimal value function, the post-decision state is defined. Based on the post-decision state, this paper defines the basis functions of the allocation actions, which are defined as action of adjusting power allocation granularity and action of subcarrier allocation, using the approximate dynamic programming (ADP) theory. Therefore, the post-decision state space could be replaced and the computation complexity could be reduced, which means that the curse of dimensionality problem is avoided.
- An adaptive virtual resource allocation algorithm based on ADP is proposed. In this algorithm, the resource allocation strategy, which aims to allocate power and subcarrier, is dynamically adjusted through constant interaction with the external environment. The numerical results demonstrate that the system performance is optimized.

The rest of the paper is organized as follows. In section II, the system model is briefly described. In Section III, we introduce the slice scheduling model and the user rate of NOMA system. In section IV, the virtual resource adaptive allocation algorithm based on ADP is described in detail. Simulation results are shown in Section V. Finally, conclusions are drawn in Section VI.

II. SYSTEM MODEL

This section describes the system model adopted in this paper. As shown in figure 1, at first, users access different service slices, and then slices allocate users to different resource blocks (RB), meanwhile, the resource scheduler allocates resources in real time based on the monitored information of slices during the period. This paper considers downlink of the NOMA system, and users who access the same subcarrier with different channel gains. Due to the diversity of services, the resource manager will customizes the allocation strategy for each slice according to its demands. For the sake of the dynamic changing characteristics of slice load, the resource manager adjusts the allocation strategy for each slice in real time on the basis of the resource utilization and the slice queue.

In addition, the resource manager can adaptively learns and adjusts power and subcarrier allocation strategy based on the monitored rate and queue information of slices. If the current queue of slices is too long, the resource manager will determine the power allocation strategy for slices. At the same time, it must be considered that if too many users use the same subcarrier, it will causes a lot of interference, which

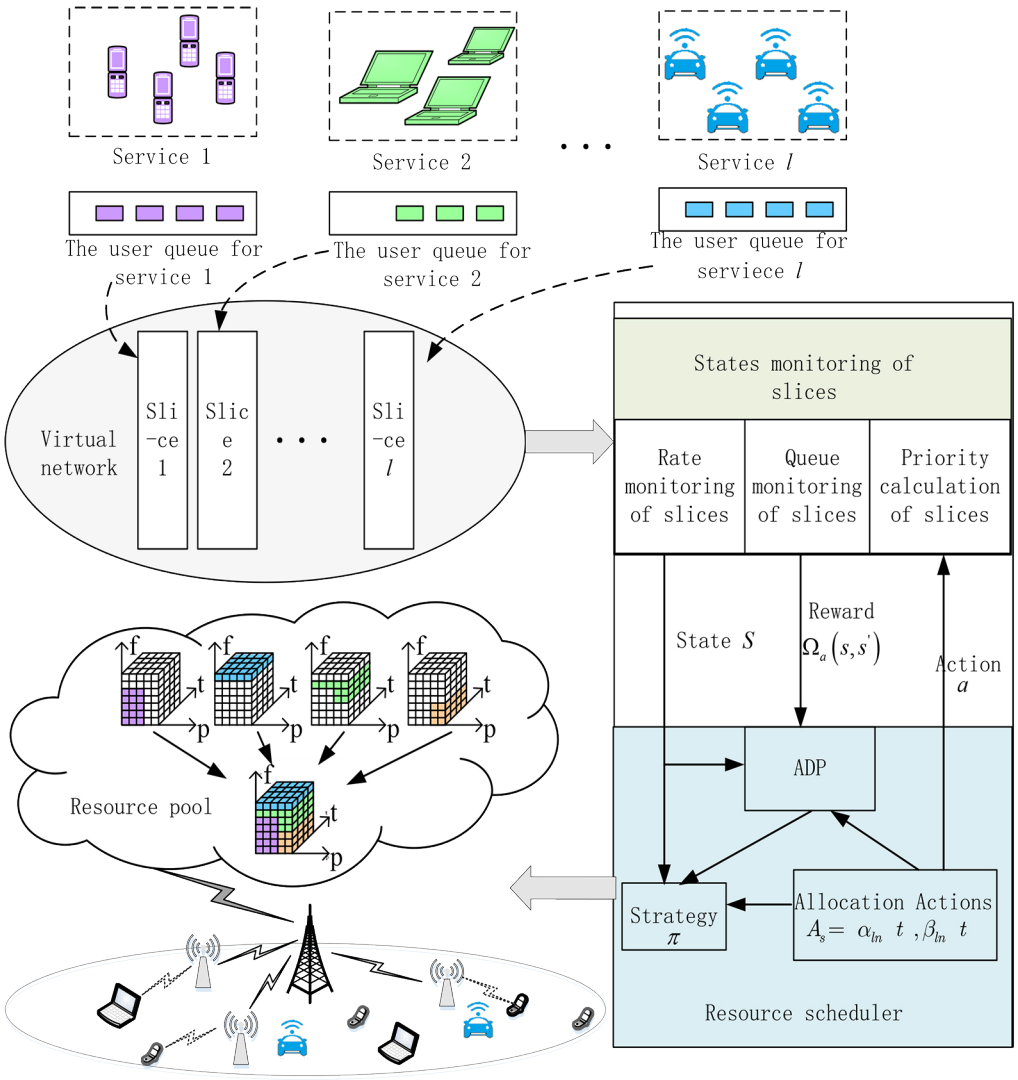


FIGURE 1. System scenario

will leads to an increase in the user's outage probability. Therefore, in order to guarantee the quality of service of slices, the resource manager also needs to adjust the subcarrier allocation strategy of slices.

III. NETWORK MODEL

A. SCHEDULING MODEL FOR SLICES

In this system, the network will determine the service types of users at first, and then users will be connected to the corresponding network slices according to users' different demands. Each slice serves the same service type of user, so each slice has its specific virtual resource requirements, including bandwidth and power requirements. This article develops a special resource allocation policy for each slice.

Assume that there are L network slices that support L different types of services, and there are $k_l, l = \{1, 2, \ldots\ldots, L\}$ users per slice. The user arrivals at time slot t is $A(t)$, $A(t) \in \mathcal{A}$, where $\mathcal{A}$ represents the maximum number of users in the physical region. At time slot t, the user arrival amount is equal to the sum of user arrivals for all slices:

$$A(t) = \sum_{l=1}^{L} A_l(t) \tag{1}$$

where $A_l(t)$ represents the number of users arriving on the slice l at time slot t.

As shown in figure 2, the request queue of slice l is $Q_l^F(t)$, $Q_l^F(t) < \infty, \forall l \in L$. At first, users access the corresponding network slices according to different services requirements at time t, and then the slice manager allocates all slice users to different resource blocks (RBs) according to the opportunity scheduling policy [13]. Successive interference cancelation (SIC) is applied at the receivers to enable multiple users multiplexed on the same subchannel with different power levels under NOMA system. According to protocol, the multiplexed users with higher channel gain can correctly decode and remove the interference from users with lower channel

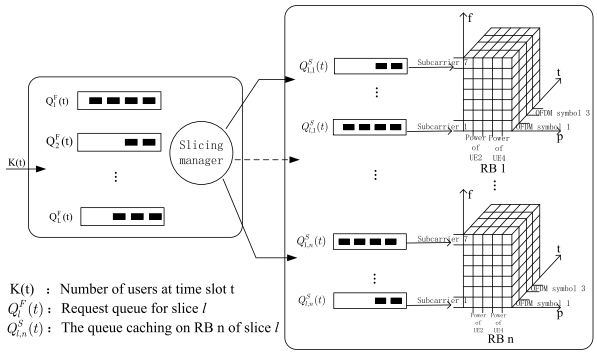


FIGURE 2. Scheduling model for slices.

gains on the same subchannel [21]. Generally, users with low channel gain are allocated higher power, and users with high channel gain are allocated lower power [20]. Therefore, the scheduling strategy ensures that the channel gains of users accessing the same subcarrier are different. The request queues of users accessing the Resource Block(RB) n of slice l is defined as $Q_{l,n}^S(t)$, $Q_{l,n}^S(t) < \infty$. Define P_{ln} as the probability that the user of slice l is processed on RB n, $n \in \mathcal{N}$, where $\mathcal{N}$ represents the set of RBs, and $|\mathcal{N}| = N$. The total probability that the user of slice l is processed at time slot t can be expressed as

$$\sum_{n \in N_l} P_{ln} = 1 \tag{2}$$

Among them, N_l represents a set of RBs allocated to slice l. According to equation (2), the system will provide services for all users scheduled to the RB. Defined $Q^S(t) = (Q_{l,n}^S(t))$ as the user queue length of slice l on RB n within time slot t. The queue caching at time t can be expressed as

$$Q(t) = \sum_{l \in L} \sum_{n \in N} Q_{l,n}^S(t) \tag{3}$$

From equation (3), further define the average queue caching $\bar{Q}$ as:

$$\bar{Q} = \limsup_{T \to \infty} \frac{1}{T} \sum_{t \in T} E\{Q(t)\} \tag{4}$$

B. USER RATE OF NOMA SYSTEM

The slice manager considers the channel conditions of each user and allocates users to different RBs. In this paper, all channels in the system are independent and identically distributed Rayleigh fading channels, and the channel noise is additive white Gaussian noise. We assume that the bandwidth B is divided into a set of subcarriers $\mathcal{M} = \{1, 2, \ldots\ldots, M\}$, and there are $k_{l,m}$ users of slice l are multiplexed on the subcarrier m. Define the total transmit power of subcarrier m as P_m, i.e.$P_m = \sum_{i=1}^{k_{l,m}} p_{i,m}$. This paper considers the downlink transmission, and all users are indexed based on their channel gains in an increasing order such as $|h_{1,m}|^2 < \cdots < |h_{i-1,m}|^2 < |h_{i,m}|^2 < \cdots < |h_{l,m}|^2$, where i represents the index of users in the subcarrier m. According to the user's channel gain, different users can be distinguished by power level. In the downlink of NOMA, the superimposed signal on the transmit terminal through subcarrier m is S_m, which can be expressed as:

$$S_m = \sum_{i=1}^{k_{l,m}} \sqrt{p_{i,m}} x_{i,m} \tag{5}$$

Here, $x_{i,m}$ denotes the transmission signal of user i on subcarrier m. and $p_{i,m}$ denotes the power allocated to user i on subcarrier m. At the receiver, the received signal according to NOMA system of user i on subcarrier m can be expressed as::

$$y_{i,m} = h_{i,m} S_m + \omega_{i,m} \tag{6}$$

In (6), $h_{i,m}$ is the realistic Rayleigh fading channel coefficient between the BS to the i^{th} user on subcarrier m, and $\omega_{i,m}$ is a zero-mean complex additive white Gaussian noise (AWGN) random variable with variance σ_m^2.

According to Shannon's capacity formula, by utilizing the SIC technique at the receivers, the maximum achievable date rate of the i^{th} user on subcarrier m can be written as

$$r_{i,m} = B_m log_2 \left(1 + \frac{p_{i,m} \Gamma_{i,m}}{1 + \sum_{j=i+1}^{k_{l,m}} p_{j,m} \Gamma_{i,m}} \right) \tag{7}$$

From (7), it can be seen that for each user, the power allocated for other users will have a great influence on its rate. Here, B_m is the bandwidth of subcarrier m, and $\Gamma_{i,m}$ is the carrier-to-interference and noise ratio (CINR) of i^{th} user, which can be expressed as:

$$\Gamma_{i,m} = |h_{i,m}|^2 / \sigma_m^2 \tag{8}$$

We assumed that the number of subcarriers accessed by the slice l is M, so the total rate of slice l can be written as:

$$\begin{aligned} R_l^M &= \sum_{m=1}^{M} \sum_{k=1}^{k_{l,m}} r_{k,m} \\ &= \sum_{m=1}^{M} \sum_{k=1}^{k_{l,m}} B_m \log_2 (1 + \frac{p_{i,m} \Gamma_{i,m}}{1 + \sum_{j=i+1}^{k_{l,m}} p_{j,m} \Gamma_{i,m}}) \end{aligned} \tag{9}$$

The global outage probability of subcarrier m in the NOMA system is defined as:

$$\begin{aligned} P_{total}^m &= 1 - \Pr\{\forall r_{i,m} > R_l^{\min}\}, \\ & i = \{1, 2, \cdots, k_{l,m}\}, \quad l = \{1, 2, \cdots, L\} \end{aligned} \tag{10}$$

IV. PROBLEM FORMULATION AND ALGORITHM DESIGN

Using the user rate and scheduling model for slices presented in the above, we formulate the virtual resource allocation problem in slice-based network of NOMA system by applying the CMDP, and then propose a resource adaptive allocation algorithm based on ADP in this section.

A. VIRTUAL RESOURCE ALLOCATION MODELING BASED ON CMDP

In previous researches, the work on [22] investigates a scheme to improves the power efficiency of NOMA by 45 to 54 percent through maintaining each required minimum capacity with the aim that minimize the total transmit power in VWN. In [23], a joint power and subcarrier allocation scheme based on successive convex approximation (SCA) and complementary geometric programming (CGP) was proposed with the constrains of rate and subcarrier reservations. However, in order to meet the delay requirements and the outage probability demands, only this paper works on the constrained Markov decision process (CMDP) method for the resource allocation problem of the slice network under the NOMA system. In this section, we first define the state, action spaces, state transition equation and reward function. At last, the optimization problem based on the CMDP are defined.

1) SYSTEM STATE

At each time slot, the system state is related to the state and action of the previous time slot. In this paper, we denote the system state at time slot t consists of internal state $u(t)$ and external state $w(t)$. The composite state of the proposed CMDP formulation for the resource allocation is defined as follows

$$S(t) = \{u(t), \omega(t)\} \tag{11}$$

Among them, the internal state $u(t)$ describes the subcarrier and power allocation of the slice l on the RB n within the time period $[t, t+1]$. Thus, the internal state $u(t)$can be expressed as

$$u(t) = (u_{ln}(t))_{l\in L, n\in N} = (N_{ln}(t), P^m_{ln}(t))_{l\in L, n\in N, m\in M} \tag{12}$$

From (12), $N_{ln}(t)$ represents the number of subcarriers assigned to slice l by resource block RB n, and $P^m_{ln}(t)$ represents the power granularity of allocated subcarrier m.

We define the external state $w(t)$ as the weight of service slices. At each time slot t, after the resource allocation is completed, the weight of each slice is updated according to the external environment, and the allocation action at the next time slot is affected by the current slices weights. In this paper, the weight of each slice is determined by service demands of slices and states of the slice queues, which can be expressed as

$$w(t) = (w_l(t))_{l\in L} \tag{13}$$

Wherein, $\omega_l(t)$ is the weight of service slice l updated at time slot t. Specifically, the real-time weight of slice l on each subcarrier is calculated by using (14).

$$\omega_l(t) = -\frac{\lg \delta_l}{\tau_l} \times D_{HOL,l} \times \frac{\overline{R_l}(t)}{R_l(t)} \tag{14}$$

Among them, δ_l is a constant between 0 and 1, τ_l represents the packet loss rate related to the quality of service (QoS), which indicating the maximum delay threshold allowed by the slice l. $D_{HOL,l}$ represents the delay of the head of the queue, $\overline{R_l}(t)$ is the ideal instantaneous rate of slice l in the current time slot, and $R_l(t)$ is the actual rate of slice l in the current time slot.

2) ALLOCATION ACTIONS

When slicing queue is too long, the power granularity may be adjusted to provide services for more users, or users can be assigned to other subcarriers for services. However, if too many users are connected to the same subcarrier, the outage probability will be increased, while if too many subcarriers are allocated, the queue caching of slices will become larger, so it is important to balance the outage probability and queue caching. Define A_s as action space when the state is s:

$$A_s = \{\alpha_{ln}(t), \beta_{ln}(t)\} \tag{15}$$

In (15), $\alpha_{ln}(t)$ is the adjustment action of power allocation granularity of the slice l on RB n at time slot t, and $\beta_{ln}(t)$ is the adjustment action of subcarrier allocation of slice l at the time slot t.

- **Action of Adjusting Power Allocation Granularity**
  In this paper, we put the power discretization into power granularity, which can control the number of users in each sub-carrier to meet the slicing queue requirements. When the queue length of slices exceeds the maximum value, the power allocation granularity of subcarrier m will be reduced, so that more users are accessed within subcarrier m. Otherwise, the power allocation granularity of the subcarrier m may be increased, and this action can improve the QoS of slices. The action of adjusting power allocation granularity is defined as $\alpha_{ln}(t)$, so that $P^m_{ln}(t+1) = P^m_{ln}(t) \pm \alpha_{ln}(t)$. Assuming that the number of service users on subcarrier m of resource block n is k, the total power of the resource block n satisfies the following conditions:

$$\sum_{i=1}^{k} i\left(P^m_{ln}(t) \pm \alpha_{ln}(t)\right) \le P_{\max} \tag{16}$$

  After the power allocation granularity is determined at time slot t, the maximum number of serving users on subcarrier m at time slot t defined as $k^m_{\max}(t)$:

$$k^m_{\max}(t) = \left[-\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{2P_{\max}}{P^m_{ln}(t) \pm \alpha_{ln}(t)}}\right] \tag{17}$$

  According to the power allocation granularity, the queue of slice l at time slot t can be expressed as:

$$Q^S_{ln}(t) = Q^S_{ln}(t-1) - \sum_{m=1}^{M} k^m_{\max}(t) + A_l(t) \tag{18}$$

- **Action of Subcarrier Allocation**
  Another allocation action is subcarriers allocation which means adjustment of subcarrier number of slices. According to formula (7), when power allocation granularity reduced, user noise and outage probability will

be increased, and the rate will be decreased, which seriously affects the QoS of slices. For this problem, when the queue length of slice l exceeds the maximum threshold value, and the power allocation granularity has reached the minimum value, the system will considers to allocate additional subcarriers for users of slice l. In this paper, the adjustment action of subcarrier allocation is defined as $\beta_{ln}(t)$, which is expressed as

$$\beta_{ln}(t) = \{-1, +1\} \tag{19}$$

It should be noted that the power granularity will not changed after perform this action.

3) STATE TRANSITION EQUATION

According to the above, the state transition equation of the system can be expressed as:

$$s(t+1) = s(s(t), a_s(t), w(t+1)) \tag{20}$$

From equation (20), the state of the system at time slot $t+1$ is obtained by the state and action at time slot t and the system external state $w(t+1)$ at $t+1$. In (20), $a_s(t)$ represents the resource allocation action for each slice at time slot t, i.e. adjusting the power allocation granularity and adjusting the number of subcarriers. In addition, the action $a_s(t)$ is affected by the global system states at the previous time slot, that is, the internal state $u(t)$ and the external state $w(t)$.

The transfer relation equation (21) of the internal state $u(t)$ is shown in the bottom of the next page.

Among (21), Ψ^N and Ψ^P represent the transfer equations of $N_{ln}(t)$ and $P^m_{ln}(t)$, respectively. It is clear that, when the power allocated to one subcarrier satisfies the constraint, and the QoS and queue delay of slices can be satisfied by adjusting the power allocation granularity, the system will only chooses the action of adjusting power allocation granularity. If the action of adjusting power allocation granularity cannot satisfy these requirements, or the power cannot satisfy the constrain, the system will selects the action of adjusting subcarriers numbers. So the internal state $u(t)$ of the system will be definitely changed, that is, the internal transition probability of the system can be expressed as $P(u(t), u(t+1)) = 1$.

The external state $w(t)$ of the system depends on the user's arrival process and the slice state after performing actions, so the external state $w(t)$ is constantly updated during interactive learning with external environment.

4) REWARD FUNCTION

After the system takes allocation actions at each time slot t, it will get a reward value, and system will perform correlation action according to the reward value at next time slot. At time slot t, the resource manager observes internal state $u(t)$ of the system and the external state $w(t)$ of each slice at first, and then selects the feasible action $a_s(t)$ according to the resource allocation policy $\pi_s(t)$. The strategy $\pi \in \Pi_s$ measured by the value function $V^\pi(s)$, which can be expressed as

$$V^\pi(s) = E^\pi \left\{ \sum_{t=0}^{\infty} \gamma^t \Omega_a(s, s') \,|s(t) = s, a_s(t) = a \right\} \tag{22}$$

Where γ is discount factor to guarantee the convergence of value function and $\Omega_a(s, s')$ is the reward function. Obviously, the value function of formula (22) needs to calculate the expectation, which is related to the reward value at each epoch, thus, this paper adopt the iterative to obtain the optimal value function for the state s. The expression is:

$$V^*(s) = \max_{a_s(t) \in A_s} E^\pi \left\{ \Omega_a(s, s') + \gamma V^*(s') \,|s(t) = s, a_s(t) = a \right\} \tag{23}$$

Correspondingly, the optimal policy for state s can be expressed as:

$$\pi^*(s) = \arg\max V^*(s) = \arg\max E \left\{ \Omega_a(s, s') + \gamma V^*(s') \,|s(t) = s, a_s(t) = a \right\} \tag{24}$$

Defined reward function as $\Omega_a(s, s')$, which is related to rate and weight of the slice l. The reward function can be expressed as (25), shown in the bottom of the next page.

The objective of the resource manager is to maximize the reward function under the constraints of the average queue caching and outage probability, which can be expressed as

$$\max E^\pi \left\{ \sum_{t=0}^{\infty} \Omega_a(s, s') \right\} \tag{26}$$

$$s.t.\ C1: \bar{Q}(t) < Q_{\max}$$

$$C2: \frac{1}{L} \sum_{l \in L} \bar{P}_l^{out}(t) < P_{\max}^{out} \tag{27}$$

From C_1, $\bar{Q}(t)$ is the average queue caching of the system at time slot t, $Q_{\max}$ is the maximum queue caching allowed by the system. From another constraint C_2, $\bar{P}_l^{out}(t)$ is the average outage probability of slice l at time slot t, and $\frac{1}{L} \sum_{l \in L} \bar{P}_l^{out}(t)$ is the average outage probability of the system, $P_{\max}^{out}$ represents the maximum allowable outage probability.This optimization objective function ensures that the queue delay of slices does not exceed the maximum and the QoS of slices simultaneously.

B. ADAPTIVE VIRTUAL RESOURCE ALLOCATION ALGORITHM DESIGN

The traditional method updates the value function iteratively, and for each state, state transition probability are needed. However, the system transition probability cannot be obtained in this paper. In addition, when the state space scale is too large, it is easy to encounter the curse of dimensionality. For the above problems, this paper adopts Approximate Dynamic Programming (ADP) theory, which can not only avoid calculating the expectation caused by the transition probability, but

also overcoming the curse of dimensionality and improve the efficiency of the algorithm because of the post-decision state. In this paper, the self-adaptive resource allocation algorithm based on ADP is proposed. Different from the order of traditional dynamic programming iterations, this method iterates the Bellman equations along the time sequence.

1) POST-DECISION STATES

The post-decision state $S^a(t)$ is defined to avoid calculate the expectation of the optimal value function in e.q.(23). The system state transition equation is equivalently split into the following two steps [21]:

$$\begin{aligned} S^a(t) &= u^a_{ln}(t) = (N^a_{ln}(t), P^a_{ln,m}(t))_{l\in L, n\in N, m\in n} \\ &= u_{ln}(t+1) = (N_{ln}(t+1), P^m_{ln}(t+1))_{l\in L, n\in N, m\in n} \\ &= S^a(S(t), a_s(t)) \qquad (28) \\ S(t+1) &= (u(t+1), w(t+1)) \\ &= S^w(S^a(t), w(t+1)) \qquad (29) \end{aligned}$$

The states $S^a(t)$ and $S(t+1)$ in equations (28) and (29) represent the post-decision state at time slot t for slice l on RB n and the pre-decision state at time slot $t+1$, respectively. The transition of the system state is shown in figure 3. It can be seen that $S^a(t)$ is the system state after the implementation of the allocation action $a_s(t)$ and before the external state

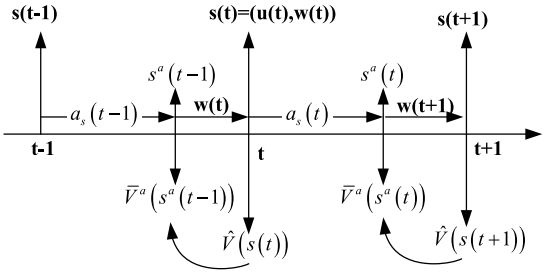


FIGURE 3. State transition.

$w(t+1)$ transfer, which describes allocation of resources. The pre-decision state $S(t+1)$ is the system state before the implementation of the action $a_s(t+1)$ at time slot $t+1$, which reflects the QoS of the slices including rate and delay. According to the definition of the post-decision state, the optimal value function equation in (23) can be rewritten as

$$\overset{*}{V}(s) = \max_{a_s(t)\in A_s} \left\{\Omega^a(S^a(t)) + \gamma \bar{V}^a(S^a(t))\right\} \qquad (30)$$

According to (30), $\bar{V}^a(S^a(t))$ is an estimate value of the value function, which can reflects the statistics information of the external state. Therefore, it is no need to calculate the expectation in equation (30). $\Omega^a(S^a(t))$ is the utility function

$$\begin{aligned} u_{ln}(t+1) &= (N_{ln}(t+1), P^m_{ln}(t+1))_{l\in L, n\in N, m\in n} \\ &= (\Psi^N(u_{ln}(t), \beta_{ln}(t)), \Psi^P(u_{ln}(t), \alpha_{ln}(t)), \\ &= \begin{cases} (N_{ln}(t), P^m_{ln}(t) \pm \alpha_{ln}(t)) & P_{\min} \le \sum_{i=1}^{k} i\left(P^m_{ln}(t) \pm \alpha_{ln}(t)\right) \le P_{\max} \\ (N_{ln}(t) \pm \beta_{ln}(t), P^m_{ln}(t) \pm \alpha_{ln}(t)) & P_{\min} > \sum_{i=1}^{k} i\left(P^m_{ln}(t) \pm \alpha_{ln}(t)\right) \\ & or \\ \sum_{i=1}^{k} i\left(P^m_{ln}(t) \pm \alpha_{ln}(t)\right) > P_{\max} \end{cases} \end{aligned} \qquad (21)$$

$$\begin{aligned} \Omega_a(s, s') &= \sum_{l=1}^{L} \omega_l(t) R_l(t) = \sum_{l=1}^{L} \omega_l(t) \sum_{m=1}^{M} \sum_{k=1}^{k_{l,m}} r_{k,m}(t) \\ &= \begin{cases} \sum_{n=1}^{N} \sum_{m=1}^{N_{ln}(t)} \sum_{k=1}^{k_{n,m}} B_m \log_2\left(1 + \frac{k(P^m_{ln}(t) \pm \alpha_{ln}(t))\Gamma_{i,m}}{1 + \sum_{j=1}^{k_{l,m}} j(P^m_{ln}(t) \pm \alpha_{ln}(t))\Gamma_{j,m}}\right), & P_{\min} \le \sum_{i=1}^{k} i(P^m_{ln}(t) \pm \alpha_{ln}(t)) \le P_{\max} \\ & P_{\min} > \sum_{i=1}^{k} i(P^m_{ln}(t) \pm \alpha_{ln}(t)) \\ \sum_{n=1}^{N} \sum_{m=1}^{N_{ln}(t)\pm 1} \sum_{k=1}^{k_{n,m}} B_m \log_2\left(1 + \frac{k(P^m_{ln}(t) \pm \alpha_{ln}(t))\Gamma_{i,m}}{1 + \sum_{j=1}^{k_{l,m}} j(P^m_{ln}(t) \pm \alpha_{ln}(t))\Gamma_{j,m}}\right), & or \\ & \sum_{i=1}^{k} i(P^m_{ln}(t) \pm \alpha_{ln}(t)) > P_{\max} \end{cases} \end{aligned} \qquad (25)$$

which is redefined with the post-decision state, as shown in the formula (31):

$$\Omega^a(S^a(t)) = E\{\Omega_a(s, s')\} = \sum_{l=1}^{L} \omega_l(t+1) R_l^a(S^a(t)) \quad (31)$$

Within (31), the $R_l^a(S^a(t))$ can be expressed as (32), shown in the bottom of this page.

The queue information and the outage probability with the post-decision state are defined as follows:

$$Q^a(S^a(t)) = \bar{Q}(t) = \lim_{T\to\infty} \frac{1}{T} Q(t) \quad (33)$$

$$P^a(S^a(t)) = \lim_{T\to\infty} \frac{1}{T}\frac{1}{N} \sum_{i\in A} P_i^{out}(S^a(t)) = \frac{1}{N} \sum_{i\in A} \bar{P}_i^{out}(t) \quad (34)$$

2) ADAPTIVE VIRTUAL RESOURCE ALLOCATION ALGORITHM DESIGN

This section designs an adaptive virtual resources allocation algorithm based on post-decision state. The post-decision state can avoid expectation calculation, and the estimation value $\bar{V}^a(S^a(t))$ is imported into the objective function, so the system can obtain the optimal policy at each period in the process of the continuous update of the estimation value. For the optimization model (i.e. equation (26)), this paper uses the Lagrange algorithm to bring the constraint into the objective function, which is expressed as (35), shown in the bottom of this page.

In (35), λ_1 and λ_2 represent the Lagrangian multipliers, and the optimization problem model can be transformed into equation (36):

$$(\pi^*, \lambda_1^*, \lambda_2^*) = \underset{\lambda_1\geq 0, \lambda_2 \geq 0, \pi}{\arg\max} \; L(\pi, \lambda_1, \lambda_2) \quad (36)$$

We update the Lagrangian multipliers at each time slot, which can be expressed as:

$$\lambda_1^{t+1} = \lambda_1^t + \varsigma_1(\bar{Q}^{\pi^*_{\lambda_1,\lambda_2}}(t) - Q_{\min}) \quad (37)$$

$$\lambda_2^{t+1} = \lambda_2^t + \varsigma_2(\frac{1}{N}\sum_{i\in A} \bar{P}_i^{\pi^*_{\lambda_1,\lambda_2}}(t) - P_{\max}^{out}) \quad (38)$$

TABLE 1. Basis Function Definition.

Basis Function	Description
$P_{\ln}^m(t) + \alpha_{\ln}(t)$	Power allocation granularity for slice l
$N_{\ln}(t) + \beta_{\ln}(t)$	Number of subcarriers to slice l
$(P_{\ln}^m(t) + \alpha_{\ln}(t))^2$	The square of power allocation granularity for slice l
$(N_{\ln}(t) + \beta_{\ln}(t))^2$	The square of number of subcarriers to slice l
$(N_{\ln}(t) + \beta_{\ln}(t))(P_{\ln}^m(t) + \alpha_{\ln}(t))$	The product of power allocation granularity for slice l and number of subcarriers to slice l

Bring the Lagrangian function into the optimal value function, and the optimal value function is transformed to the formula (39).

$$\overset{*}{V}(s) = \underset{a_s(t)\in A_s}{max} \left\{L^a(S^a(t)) + \gamma \bar{V}^a(S^a(t))\right\} \quad (39)$$

The traditional method to obtain estimates value uses the look-up table method, but this method has poor scalability and is not suitable for large-scale state spaces. Therefore, this paper uses ADP to solve this problem. ADP, also known as forward dynamic programming (FDP), overcomes the problem of "the curse of dimensionality" in large-scale CMDP [20].

In the CMDP model of this paper, both the state transition and the reward function are related to the action of resource allocation. Different actions will determine the state of the next time slot and the value of reward function. According to the action that the system will choose, we defined the basis function as shown in Table 1. According to the basis function in Table 1, we introduce the parameter vector $\eta^T = \{\eta_1, \eta_2, \eta_3, \cdots, \eta_\hbar\}$, $h \in \hbar$ represents the feature of the resource allocation action. Define the basis function vector $\chi = \{\chi_1(S^a), \chi_2(S^a), \cdots, \chi_\hbar(S^a)\}$, an approximate function can be obtained using the linear function which is expressed as

$$\bar{V}^a(S^a(t)) = \eta^T \chi = \sum_{h\in\hbar} \eta_h \chi_h(S^a(t)) \quad (40)$$

$$R_l^a(S^a(t)) = \sum_{n=1}^{N} \sum_{m=1}^{N_{ln}(S^a(t))} \sum_{k=1}^{k_{n,m}} B_m \log_2\left(1 + \frac{k\left[p_{ln}^m(S^a(t)) \pm \alpha_{ln}(S^a(t))\right]\Gamma_{i,m}}{1 + \sum_{j=k+1}^{k_{l,m}} j\left[p_{ln}^m(S^a(t)) \pm \alpha_{ln}(S^a(t))\right]\Gamma_{j,m}}\right) \quad (32)$$

$$\begin{aligned} L(\pi, \lambda_1, \lambda_2) &= \sum_{t=0}^{\infty} \Omega_a(s, s') - \lambda_1(\bar{Q}(t) - Q_{\min}) - \lambda_2(\frac{1}{N}\sum_{i\in A}\bar{P}_i^{out}(t) - P_{\max}^{out}) \\ &= \lim_{T\to\infty}\frac{1}{T}\left[\sum_{t=0}^{\infty}\gamma^t\Omega_a(s, s') - \lambda_1(Q(t) - TQ_{\min}) - \lambda_2(\frac{1}{N}\sum_{i\in A}P_i^{out}(t) - TP_{\max}^{out})\right] \\ &= \Omega^a(S^a(t)) - \lambda_1(Q^a(S_{ln}^a(t)) - Q_{\min}) - \lambda_2(P^a(S_{ln}^a(t)) - P_{\max}^{out}) \end{aligned} \quad (35)$$

Based on the approximation value function of equation (40), the optimal value function (39) can be converted to formula (41).

$$\overset{*}{V}(s) = \max_{a_s(t)\in A_s} \left\{ L^a(S^a(t)) + \gamma \sum_{h\in\hbar} \eta_h \chi_h(S^a(t)) \right\} \quad (41)$$

For equation (41), the system only need to update the parameter vector η to get the estimated value. This paper uses a gradient algorithm to update the parameter vector η, and the objective is to minimize the mean square error between the estimated value and the sample function, as shown in equation (42):

$$\eta^* = \arg\min E\{\frac{1}{2}\left[\bar{V}^a(\eta) - \hat{V}\right]^2\} \quad (42)$$

Where $\bar{V}^a(\eta)$ is the estimated value and $\hat{V}$ is the sample value. The random gradient expression of approximate value function $\bar{V}^a(\eta)$ with parameter vector η can be expressed as:

$$\begin{aligned}\nabla_\eta \bar{V}^a(S^a(t)\,|\eta) &= (\frac{\partial \bar{V}^a(S^a(t)\,|\eta)}{\partial \eta_1}, \cdots, \frac{\partial \bar{V}^a(S^a(t)\,|\eta)}{\partial \eta_{|\hbar|}})^T \\ &= (\chi_1(S^a(t)), \cdots, \chi_{|\hbar|}(S^a(t)))^T \\ &= \chi(S^a(t)) \end{aligned} \quad (43)$$

The parameter vector η is updated along the gradient direction and the update expression is expressed as:

$$\eta \leftarrow \eta - \mu_{t-1}(\bar{V}^a(S^a(t-1)\,|\eta) - \hat{V}(S(t)))\chi(S^a(t-1)) \quad (44)$$

Where μ is the step size, with the following constrains:

$$\begin{cases} \sum_{t=1}^{\infty} \mu_{t-1} = \infty \\ \sum_{t=1}^{\infty} (\mu_{t-1})^2 < \infty \end{cases} \quad (45)$$

The sample value function can be expressed as

$$\hat{V}(S(t)) = \max_{a_s(t)\in A_s} \left\{ L^a(S^a(t)) + \gamma V^a(S^a(t)) \right\} \quad (46)$$

Bring the constantly updated parameter vector into the optimal value function, i.e. equation (39). The current optimal policy can be rewritten as:

$$\pi^*(s) = \arg\max \left\{ L^a(S^a(t)) + \gamma \sum_{h\in\hbar} \eta_h \chi_h(S^a(t)) \right\} \quad (47)$$

The resource adaptation allocation algorithm is summarized in Algorithm 1.

From this algorithm, the execution process of the proposed resource adaptive allocation algorithm is online, and the computational complexity is $O(|\hbar| \cdot T)$, where $|\hbar|$ denotes the feature number of the resource allocation action and T represents the length of time period. Since $|\hbar|$ is much smaller than the size of the post-decision state space, the value function approximation strategy can significantly reduce the storage and computational overhead of the algorithm when compared to the lookup table update method. Hence, the computational complexity is not a challenge to system and the algorithm is applicable for practical application.

Algorithm 1 Adaptive Virtual Resource Allocation Algorithm Based on ADP.

Input:. $x_h(S^a(t))$: Basis function, γ: Discount factor
Output:. η: Parameter vector, λ_1, λ_2 : Lagrange multiplier

```
Initialization: t ← 0, η ← 0, λ1, λ2 ← 0
for (t = 1; t <= T; t++) do
while V_t − V_{t−1} > ε1
while V̄^a(η) − V̂ > ε2
Update V̂ according to (46)
if t > 0 then
Update: η according to (44)
end if
Sampling external random variables ω(t + 1)
Update V̄^a(S^a(t)) according to (40) based on η
end while
Calculate objective function based on π*(s)
Update λ1, λ2 according to (37) and (38)
endwhile
end for
```

V. SIMULATION RESULTS

In this section, we evaluate the performance of our proposed Power Granularity Uncertain - Approximate Dynamic Programming (PGU-ADP) algorithm with reference to [22] and [23] and compare our solution with PGC-F (Power Granularity Certain - Fairness) and PGU-QL (Power Granularity Uncertain - Q-Learning), which have been presented in [12] and [18]. In order to verify the performance of the algorithm, the simulation experiment is divided into two phases, which are the learning phase and the testing phase. The simulation time lasted 600 periods, including 400 periods for the learning phase and 200 periods for the testing phase. A longer learning time allows the system to have enough time to learn the parameter vector β in the approximation value function during the learning phase. Additionally, we consider

TABLE 2. Simulation parameters.

M (Subcarrier number)	64
P (Transmit power)	33 dBm
Path loss	$133.6 + 35log_{10}(d)$
Number of transmission antennas	1
Number of receiving antennas	1
Base station service area	500m
Number of users superimposed on a single subcarrier	1 4
Action of power allocation granularity adjustment	$\alpha = \{\ 0.25,\quad 0.5,\quad 1\ \}$
Action of subcarriers allocation	$\beta = 1$
Demands of slice 1	(5ms, 200kbits/s)
Demands of slice 2	(10ms, 500kbits/s)
Demands of slice 3	(50ms, 1Mbits/s)

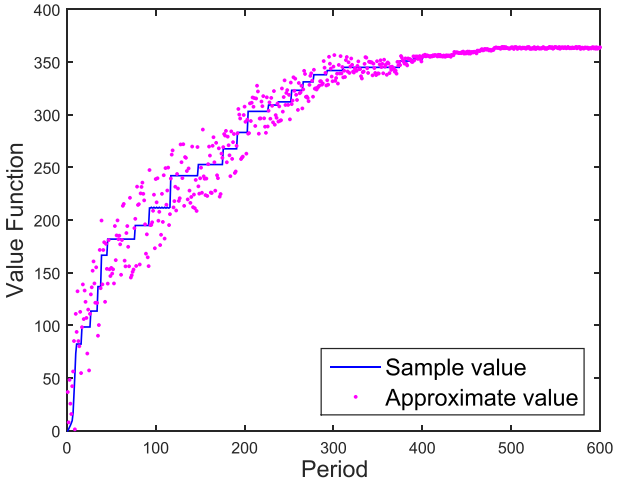


FIGURE 4. Value function.

3 types of slices having different demands. Simulation parameters are shown in Table 2.

Figure 4 depicts the comparison of the approximate values and sample values. As we can observe, the approximate value and the sample value have a large gap during the first 400 periods, which is the learning phase. With the periods increases, the gap between the approximate value and the sample value gradually decreases and approaches coincidence around the 400^{th} period. During the next 200 periods, which is the testing phase, the approximate value and the sample value almost coincide. As can be seen from Figure 4, the approximate values will converge to the sample value because of continuous learning and updating of the parameter vector.

In Figure 5, we have compared the bandwidth resource utilization when the power allocation granularity is 0.25, 0.5, and 1, respectively. Among Figure5, we define the resource utilization as R'/R, in which R represents the total resources, and $R^{'}$ represents resources that have been allocated to the user without interruption. It is clear that, the resource utilization rate increases fast when the number of users is small, but if the number of users continues to increase, the outage probability would increases and the resource utilization rate would decreases. When $\alpha = 0.5$, the rate of increase in resource utilization is not as fast as when α is equal to 1, but in this action,system can serve for more users. When α is equal to 0.25, the resource utilization and service users reached the maximum value of 0.9483, but with the number of users continues increase, resource utilization fall down rapidly, even below the number when α is equal to 0.5. In summary, when $\alpha = 1$, resource utilization increases quickly when the number of users is less than 95, and when $\alpha = 0.25$, resource utilization reaches the pick value when the number of users in the range from 165 to 195, but the rate of decline is quickly when the number of users exceeding this range. In addition, when $\alpha = 0.5$, the resource utilization rate can almost keep stable after the number of users is greater than 100.

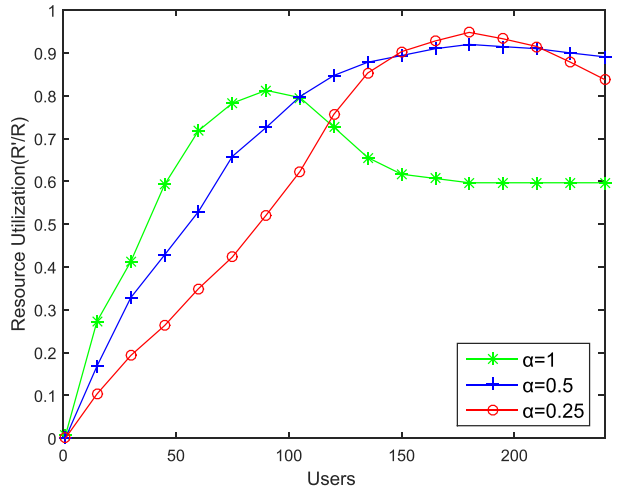


FIGURE 5. Resource utilization under different allocation actions.

Figure 6 illustrates the variation trend in outage probability with the number of users, and $P_{\max}$ denotes the maximum outage probability. In general, the number of service users when $P_{max} = 0.3$ is less than the number when $P_{max} = 0.6$. Under the condition that $P_{\max} = 0.3$, it can be seen that as the allocation granularity α decreases, the outage probability increases. It is clear that the outage probability increase fastest when and when $\alpha = 0.25$, however, when the number of users reaches 190, the outage probability is stable at around 0.3. Under the condition of $P_{\max} = 0.6$, the outage probability increases rapidly, and the difference of rising trend of different α is small.

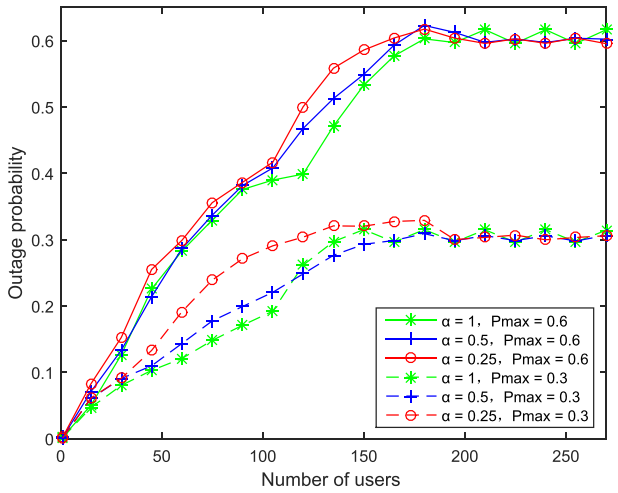


FIGURE 6. Outage Probability with Different Allocation Actions and Constrains.

In order to demonstrate the performance of the proposed PGU-ADP algorithm, this paper compares the proposed algorithm with these two algorithm which is proposed by Bega *et al.* [12] and Zhu *et al.* [18].

Figure 7 is a comparison of the number of users in the actual service slices when the power allocation granularity

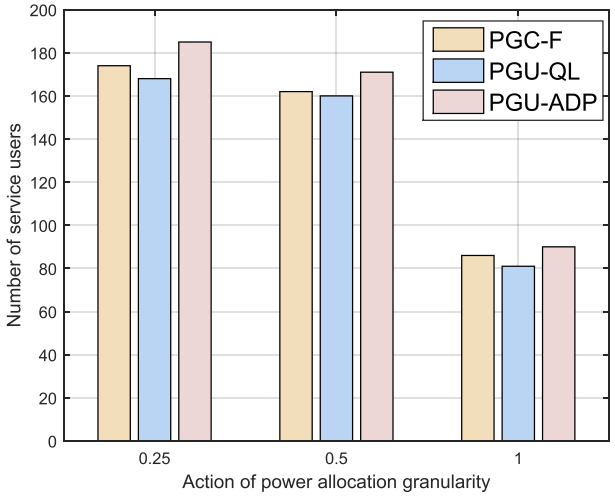


FIGURE 7. Number of Service Users under Different Schemes.

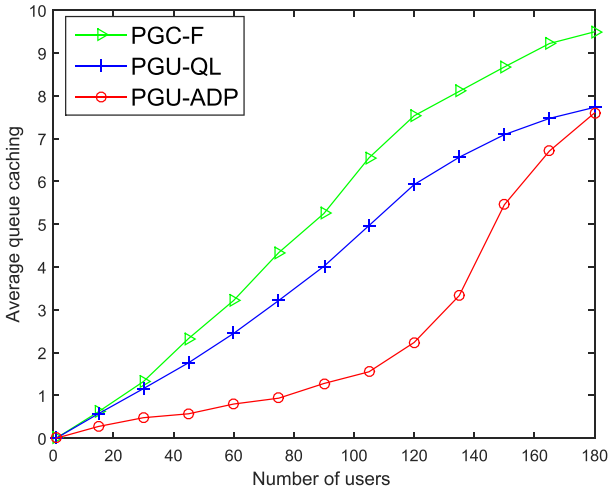


FIGURE 9. Average queue caching under different schemes.

is 0.25, 0.5, and 1 respectively. In general, the number of service users under different power allocation granularity is different. It is clear that, when $\alpha = 1$, the number of serve users is the least. When α becomes 0.5, twice the number of services when $\alpha = 1$. With the decrease of α, the number of service users has increased, but the increase is slow due to the increase of the outage probability. It is clear that, the PGU-ADP algorithm can serves more users than the other two algorithms.

Figure 8 shows the comparison of total rates under different schemes with the number of users increases. As we can be observed in Figure 8, the total rate of the PGC-F algorithm increases steadily, and when the number of users reaches around 100, the rate of increase becomes slow due to the limitation of the number of service users. The PGU-QL scheme cannot converge when the number of users is little, the increase rate becomes faster once it has reached a certain number of users, and almost equal to the rates when using PGU-ADP scheme when the number of users is 200. From Figure 8, the proposed PGU-ADP algorithm has the fastest convergence rate and has the maximum sum rate. However, when the number of users reaches 180, the rate of increase becomes slower.

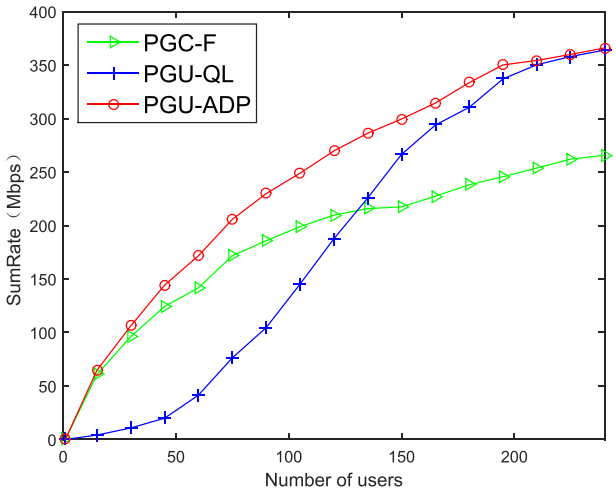


FIGURE 8. Sum rates under Different Schemes.

Figure 9 shows the comparison of the average queue caching under different algorithms with the number of users increases. Setting the constraint condition of the queue caching to $Q_{\max} = 30$, it can be clearly seen that the queue caching of the PGC-F scheme is the largest, and the PGU-QL scheme gets slower as users increase. With PGU-ADP scheme, when the number of users is greater than 140, the queue caching will increases rapidly due to the increase of the outage probability, but there is a obvious superiority when the number of users is less than 140.

VI. CONCLUSION

In this paper, we proposed an adaptive virtual resources allocation algorithm by using the approximate dynamic programming (ADP) theory for downlink of NOMA system under average delay constraint and average outage probability constraint. The algorithm analyzed slice state (i.e. slice rate and slice queue length) for the diversity of slice service requirements, and used CMDP theory to construct a dynamic optimization model of resource adaptive allocation problem. The system in this paper allocated virtual resources by adjusting the power granularity and the number of allocated subcarriers, and considered the total rate as a reward. Additionally, in order to avoid the curse of dimensionality and transfer probability, we defined the post-decision state and basis function for allocation actions. Simulation results demonstrated that the PGU-ADP algorithm can improve the performance of the system and meet the QoS requirements.

REFERENCES

[1] V. W. S. Wong, *Key Technologies for 5G Wireless Systems*. Cambridge, U.K.: Cambridge Univ. Press, 2017.
[2] 5G PPP Architecture Working Group. (Aug. 2017). *View on 5G Architecture*. [Online]. Available: https://5g-ppp.eu/wpcontent/uploads/2014/02/5G-PPP-5GArchitecture-WP-July-2016.pdf

[3] A. A. Gebremariam, M. Chowdhury, M. Usman, A. Goldsmith, and F. Granelli, "SoftSLICE: Policy-based dynamic spectrum slicing in 5G cellular networks," in *Proc. IEEE Int. Conf. Commun.*, Kansas, MO, USA, May 2018, pp. 1–6, doi: 10.1109/ICC.2018.8422148.

[4] A. Kaloxylos, "A survey and an analysis of network slicing in 5G networks," *IEEE Commun. Standards Mag.*, vol. 2, no. 1, pp. 60–65, Mar. 2018, doi: 10.1109/MCOMSTD.2018.1700072.

[5] O. Sallent *et al.*, "On radio access network slicing from a radio resource management perspective," *IEEE Wireless Commun.*, vol. 24, no. 5, pp. 166–174, Oct. 2017, doi: 10.1109/MWC.2017.1600220WC.

[6] *System Architecture for the 5G System Release-15, V.2.0.1*, document 3GPP TS 23.501, Dec. 2017.

[7] D. Nojima *et al.*, "Resource isolation in RAN part while utilizing ordinary scheduling algorithm for network slicing," in *Proc. Veh. Technol. Conf.*, Porto, Portugal, Jun. 2018, pp. 1–5, doi: 10.1109/VTCSpring.2018.8417638.

[8] S. Parsaeefard, R. Dawadi, M. Derakhshani, and T. Le-Ngoc, "Joint user-association and resource-allocation in virtualized wireless networks," *IEEE Access*, vol. 4, pp. 2738–2750, 2016, doi: 10.1109/ACCESS.2016.2560218.

[9] L. Tang, X. Zhang, H. Xiang, Y. Sun, and M. Peng, "Joint resource allocation and caching placement for network slicing in fog radio access networks," in *Proc. Int. Workshop Signal Process. Adv. Wireless Commun.*, Sapporo, Japan, Jul. 2017, pp. 1–6, doi: 10.1109/SPAWC.2017.8227791.

[10] D. Bega *et al.*, "Optimising 5G infrastructure markets: The business of network slicing," in *Proc. IEEE Conf. Comput. Commun.*, Atlanta, GA, USA, May 2017, pp. 1–9, doi: 10.1109/INFOCOM.2017.8057045.

[11] Q. Zhang, Q. Zhu, M. F. Zhani, and R. Boutaba, "Dynamic service placement in geographically distributed clouds," in *Proc. IEEE Conf. Comput. Commun.*, Atlanta, GA, USA, Jun. 2017, pp. 526–535, doi: 10.1109/INFOCOM.2017.8057045.

[12] S. M. R. Islam, N. Avazov, O. A. Dobre, and K.-S. Kwak, "Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 721–742, 2nd Quart., 2017, doi: 10.1109/COMST.2016.2621116.

[13] H. Zhang, Y. Qiu, K. Long, G. K. Karagiannidis, X. Wang, and A. Nallanathan, "Resource allocation in NOMA-based fog radio access networks," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 110–115, Jun. 2018, doi: 10.1109/MWC.2018.1700326.

[14] R. Dawadi, S. Parsaeefard, M. Derakhshani, and T. Le-Ngoc, "Power-efficient resource allocation in NOMA virtualized wireless networks," in *Proc. IEEE Global Commun. Conf.*, Washington, DC, USA, Dec. 2016, pp. 1–6, doi: 10.1109/GLOCOM.2016.7842162.

[15] D. Tweed and T. Le-Ngoc, "Dynamic resource allocation for uplink MIMO NOMA VWN with imperfect SIC," in *Proc. IEEE Int. Conf. Commun.*, Kansas City, MO, USA, May 2018, pp. 1–6, doi: 10.1109/ICC.2018.8422363.

[16] J. Zhu, J. Wang, Y. Huang, S. He, X. You, and L. Yang, "On optimal power allocation for downlink non-orthogonal multiple access systems," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2744–2757, Dec. 2017, doi: 10.1109/JSAC.2017.2725618.

[17] J. Datta and H.-P. Lin, "Detection of uplink NOMA systems using joint SIC and cyclic FRESH filtering," in *Proc. Wireless Opt. Commun. Conf.*, Hualien, Taiwan, Apr. 2018, pp. 1–4, doi: 10.1109/WOCC.2018.8372742.

[18] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Hoboken, NJ, USA: Wiley, 2007, pp. 129–144.

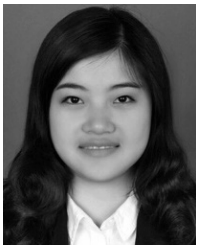
[19] T. Chen, A. Mokhtari, X. Wang, A. Ribeiro, and G. B. Giannakis, "Stochastic averaging for constrained optimization with application to online resource allocation," *IEEE Trans. Signal Process.*, vol. 65, no. 12, pp. 3078–3093, Jun. 2017, doi: 10.1109/TSP.2017.2679690.

[20] F. Fang, H. Zhang, J. Cheng, and V. C. M. Leung, "Energy-efficient resource scheduling for NOMA systems with imperfect channel state information," in *Proc. IEEE Int. Conf. Commun.*, Paris, France, May 2017, pp. 1–5, doi: 10.1109/ICC.2017.7996360.

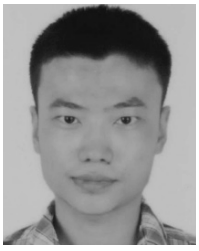
[21] R. Rai, H. Zhu, and J. Wang, "Resource scheduling in non-orthogonal multiple access (NOMA) based cloud-RAN systems," in *Proc. Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf.*, New York, NY, USA, Oct. 2017, pp. 418–422, doi: 10.1109/UEMCON.2017.8249102.

LUN TANG received the Ph.D. degree in communication and information system from Chongqing University, Chongqing, China. He is currently a Professor with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications. His current research interests include the fifth-generation cellular networks, interference management, and small cell networks.

QI TAN received the B.S. degree in telecommunication engineering from the Chongqing University of Posts and Telecommunications, China, in 2017, where she is currently pursuing the M.S. degree with the Key Laboratory of Mobile Communication Technology. Her research interests include virtual resource allocation for the fifth-generation network slicing.



YINGJIE SHI received the M.S. degree in information and telecommunication engineering from the Chongqing University of Posts and Telecommunications, China, in 2018. Her research interests include resource allocation for the fifth-generation network slicing.



CHENMENG WANG received the M.S. degree in information and telecommunication engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2014, where he is currently pursuing the Ph.D. degree with the School of Communication and Information Engineering. From 2015 to 2017, he was a Visiting Student with Carleton University, Ottawa, ON, Canada. His current research interests include small cell networks, mobile edge computing systems, resource allocation, and applications of convex optimization in mobile networks.

QIANBIN CHEN (M'03–SM'14) received the Ph.D. degree in communication and information system from the University of Electronic Science and Technology of China, Chengdu, China, in 2002. He is currently a Professor with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, where he is also the Director of the Chongqing Key Laboratory of Mobile Communication Technology. He has authored or co-authored over 100 papers in journals and peer-reviewed conference proceedings. He has also co-authored seven books. He holds 47 granted national patents.