# Deep Reinforcement Learning-based Optimization for End-to-end Network Slicing with Control- and User-Plane Separation

Yunfeng Wang, Liqiang Zhao, *Member, IEEE,* Xiaoli Chu, *Senior Member, IEEE,* Shenghui Song, *Senior Member, IEEE,* Yansha Deng, *Member, IEEE,* Arumugam Nallanathan, *Fellow, IEEE,* Kai Liang, *Member, IEEE*

*Abstract*—Control- and user-plane separation (CUPS) and network slicing are two key technologies to support increasing network traffic and diverse wireless services. However, the benefit of CUPS in decoupling the network coverage and data service functions has not been fully utilized to facilitate network slicing. In this paper, we present a novel CUPS-based end-to-end (CUPS-E2E) network slicing scheme. First, the base stations (BS) are classified into control BSs (CBS) that provide control plane (CP) coverage and traffic BSs (TBS) that deliver user plane (UP) traffic. Next, upon CBSs and TBSs being virtualized, we define four typical end-to-end (E2E) network slices: one for CP coverage, one for high-throughput services, one for computation-intensive services, and the other for delay-sensitive services. The utilities of the four E2E network slices are defined based on their coverage, throughput, computing capability and delay requirements, respectively. Then, a deep deterministic policy gradient (DDPG)-based algorithm is employed to maximize the long-term sum-utility of the four E2E network slices by jointly optimizing the allocation of communication and computing resources to the four network slices and the activation of virtual TBSs, while meeting the service requirements of all users. Simulation results show that our proposed CUPS-E2E network slicing scheme in conjunction with a DDPG-based sum-utility maximization algorithm can support the CP wide-coverage and massive access requirements as well as the UP high-throughput, computation-intensive and delay-sensitive services simultaneously, and outperforms the existing E2E network slicing schemes in terms of the sum-utility, coverage percentage, throughput and delay.

*Index Terms*—End-to-end network slicing, control- and user-plane separation, utility, deep deterministic policy gradient.

Yunfeng Wang and Kai Liang are with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: yunfengwang@stu.xidian.edu.cn; kliang@xidian.edu.cn).

Liqiang Zhao is with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China, and Guangzhou Institute of Technology, Xidian University, Guangzhou 510100, China (e-mail: lqzhao@mail.edu.xidian.cn).

Xiaoli Chu is with the Department of Electronic and Electrical Engineering, The University of Sheffield S10 2TN, U.K. (e-mail: x.chu@sheffield.ac.uk).

Shenghui Song is with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: eeshsong@ust.hk.)

Yansha Deng and Arumugam Nallanathan are with the Queen Mary University of London, London E1 4NS, U.K. (e-mail: yansha.deng@kcl.ac.uk; a.nallanathan@qmul.ac.uk).

## I. INTRODUCTION

AS a key technology for the next generation mobile networks to support diverse application scenarios and provide customized services, network slicing defines multiple logical end-to-end (E2E) networks (each including the core network (CN) and radio access network (RAN) ) on a shared physical network infrastructure [1]. Each logical network contains a set of virtual network functions (VNF) and resources.

Meanwhile, the drawbacks of traditional mobile networks due to the tight coupling between network coverage and data service become increasingly evident with the exponential growth of wireless traffic, which has called for the control- and user-plane separation (CUPS) [2]. CUPS decouples the control plane (CP) and the user plane (UP), where the control base stations (CBS) in the CP transmit control signals to users using low frequency bands, while the traffic base stations (TBS) in the UP transmit data signals using high frequency bands. As a result, CUPS can alleviate the interference between control signals and data signals [3] and can reduce the energy consumption of the network by switching off TBSs with no or few active users in coverage [4].

### A. Related Works

VNF deployment and resource allocation for network slicing have recently attracted a lot of research interests. Sattar *et al.* [5] investigated the placement of CP functions and UP functions (UPF) for CN slicing, and proposed a mixed-linear integer programming-based algorithm to minimize the E2E delay while maintaining intra-slice isolation among VNFs of the same CN slice. Guan *et al.* [6] used complex network theory to optimize the VNF deployment for Ultra Reliable Low Latency Communication (URLLC), Enhanced Mobile Broadband (eMBB), and massive Machine Type Communication (mMTC) E2E network slices.

Regarding resource allocation for network slicing, Sun *et al.* [7] proposed a Stakelberg game-based radio resource allocation scheme to minimize the content download latency for the low-delay slice while guaranteeing the data rate for the high-rate slice with minimal transmit power consumption. Zhou *et al.* [8] maximized a time-averaged utility by jointly optimizing the allocation of virtual bandwidth and power for URLLC and eMBB RAN slices via Lyapunov optimization. An Upper-tier First with Latency-bounded Over-provisioning Prevention

(UFLOP) algorithm was proposed in [9] to optimize the allocation of computing and communication resources for the eMBB, URLLC and mMTC E2E network slices in a multi-tenant 5G network, while satisfying the latency constraints and Service-level Agreements (SLA) of multiple tenants.

CUPS has been researched for mobility management, e.g., in a two-tier downlink ultra-dense network (UDN) [10], where the small base stations (SBS) in the UP send data packets to nearby user devices, and the macro base stations (MBSs) in the CP manage the radio resource control. Sun *et al.* [11] proposed a probability suffix tree (PST)-based predictive mobility management algorithm for the urban UDN with CUPS. Yang *et al.* [12] have shown that the CUPS architecture outperforms the conventional network architecture in terms of the coverage probability and the handover cost in UDNs. Liang *et al.* [13] employed the stochastic geometry and queuing theory to analyze the coverage probability of the CP assuming non-line-of-sight (NLoS) transmissions and the energy efficiency of the UP under the co-existence of both NLoS and line-of-sight (LoS) transmissions.

Although the feasibility of combining CUPS with RAN slicing was demonstrated through testbed experiments for three types of RAN slices (i.e., continuous CP coverage, UP video services and UP audio services) [14], the benefit of CUPS in decoupling the CP coverage and UP traffic has not been sufficiently studied or exploited to achieve customized E2E network slicing. Moreover, [10]–[14] considered the CP and the UP separately. Despite that the influence of the CP coverage on the spectrum efficiency and the energy efficiency of the UP for the CUPS-based cellular RAN was studied in [15], the relevance between the CP and the UP in utility modeling for E2E network slicing has not been fully researched.

Nowadays, the resource allocation in mobile networks involves communication, storage and computing resources [16], and the resource allocation problems become increasingly complex. Traditional optimization algorithms show limitations in solving high-complexity resource allocation problems. Fortunately, deep reinforcement learning (DRL), where an agent interacts with the environment constantly to learn the best action that maximizes a cumulative reward under a given state through the trial and error [17], has been employed to solve complex resource allocation problems. Popular DRL algorithms include: deep Q-learning (DQL) [18], policy gradient (PG), actor-critic (AC), and deep deterministic policy gradient (DDPG) [19]. Therein, the DQL algorithm was mainly used to solve resource allocation problems with discrete action spaces [20]–[22]. For resource allocation problems with continuous action spaces, the DDPG algorithm was used, e.g., to solve the joint optimization of continuous energy harvesting (EH) time and transmit power allocation for maximizing the long-term throughput of EH-assisted communications [23]. Jiang *et al.* [24] used DDPG algorithm to maximize the number of the access success devices in cellular internet of things networks by dynamic optimize the access class barring factor. However, it is important to note that the relevance between the CP and the UP under CUPS will lead to new challenges in the utility modeling and DRL-based resource allocation for E2E network slicing.

## B. Contributions

In this paper, we aim to solve the CUPS-based E2E (CUPS-E2E) network slicing optimization of the four typical E2E network slices: one typical mMTC slice for CP wide-coverage, two typical eMBB slices for UP high-throughput services (such as panorama virtual reality [25]) and UP computation-intensive services (such as augmented reality [26]), respectively, and one typical URLLC slice for UP delay-sensitive services (such as immersive virtual reality [27] and industry automation [28]). We define the utilities of the four E2E network slices based on their coverage percentage, throughput, computing capability and delay requirements, respectively, while taking the relevance between the CP coverage and the UP services into consideration. Therein, the coverage percentage is defined as the instantaneous fraction of all users covered by the CP at the corresponding time. Then, we propose to maximize the long-term sum-utility of the four E2E network slices by jointly optimizing the virtual TBSs (vTBS) activation and the allocation of communication and computing resources while meeting the service requirements of users, which is formulated into a non-convex optimization problem. It is important to note that the joint optimization of vTBS activation and resource allocation should be solved timely to meet the dynamic user requirements and wireless channel states, which can hardly be achieved by the traditional convex optimization methods. We solve the sum-utility maximization problem with a high-dimensional action space by devising a DRL-based algorithm, in which the agent learns the mapping function between the users' real-time requests as well as wireless channel states and the action. The main contributions of this work are summarized as follows:

- We study a CUPS-based system that decouples the CP coverage and the UP services in E2E network slicing. The CUPS-E2E network slicing overcomes the restrictions caused by the tight coupling between the CP and the UP of traditional networks and allows for the customization of CP coverage and UP services for each E2E network slice. More specifically, the CP coverage and the UP services are supported by dedicated virtual CBSs (vCBS) (operating in low frequency bands) and vTBSs (operating in high frequency bands), respectively. The wide-coverage slice builds on the virtualized CP only, whereas the high-throughput, computation-intensive and delay-sensitive slices involve both the virtualized UP and CP, where the activation of vTBSs in the UP is determined by the CP.

- In order to better characterize the heterogeneous performance metrics of the E2E network slices, we define the utilities of the four slices in terms of their gains in coverage percentage, throughput, computing capability and delay minus their corresponding costs, respectively. The costs for each slice include the backhaul link capacity and the downlink bandwidth. In addition, the costs of the wide-coverage slice include the vCBS transmit power consumption; while the costs of the high-throughput, computation-intensive and delay-sensitive slices include both the vTBS transmit power and vTBS computing

power comsumptions. These slice-specific utilities enable the joint optimization of communication and computing resources allocation, as well as a comprehensive performance evaluation, for each E2E network slice.

- In our proposed CUPS-E2E network slicing model, if a user is not covered by the CP, then it will not be served by the UP, i.e., it will not be allocated with any UP communication or computing resource. This dependence of UP services on the CP coverage is embodied in our utility model. Accordingly, the utilities of the high-throughput, computation-intensive, and delay-sensitive slices contain only the throughput, computing capability, and delay of the users covered by the CP, respectively.
- The CUPS-E2E network slicing optimization is formulated into a problem that maximizes the long-term sum-utility of the four E2E network slices by jointly optimizing the vTBS activation and the allocation of communication and computing resources while meeting the service requirements of all users. The formulated problem is non-convex and hard to solve using traditional optimization methods. We design a DDPG-based sum-utility maximization algorithm to find the optimal vTBS activation and resource allocation policy in the continuous action space. More specifically, we obtain the optimal vTBS activation and the optimal allocation of communication and computing resources to the four E2E network slices, including the CP subcarrier and vCBS transmit power for the wide-coverage slice, as well as the backhaul link capacity, UP subcarrier, vTBS transmit power and vTBS computing capability for the high-throughput, computation-intensive and delay-sensitive slices.

The remainder of this paper is organized as follows. The CUPS-based system model is proposed in Section II. Section III formulates the sum-utility maximization problem. The DDPG-based sum-utility maximization algorithm is presented in Section IV. In Section V, simulation results are presented. Finally, the paper is concluded in Section VI.

## II. System Model

This section presents a CUPS-based system, where the E2E network model contains the CN and the RAN. The CN mainly consists of the Access and Mobility Management Function (AMF), Session Management Function (SMF) and UPF, etc. Therein, the AMF is responsible for the access and mobility of users, while the SMF and UPF are in charge of the UP services [29].

In the RAN, we assume that there are $N_C$ physical CBSs and $N_T$ physical TBSs distributed within a sufficiently large area $S_{pl}$, and the symbols $\boldsymbol{N_C}$ and $\boldsymbol{N_T}$ are used to denote sets of the physical CBSs and TBSs, respectively. The CBSs send control signals at the low frequency bands to provide the CP coverage for users and control the switching on-off for the TBSs, whereas the TBSs process and send UP data at the high frequency bands to provide the UP services for users. Moreover, the densities of the physical CBSs and TBSs are $\lambda_C = N_C/S_{pl}$ and $\lambda_T = N_T/S_{pl}$, respectively.

There are $N_u$ users distributed in the given area with the density of $\lambda_u = N_u/S_{pl}$, and the symbol $\boldsymbol{N_u}$ is used to denote

the set of users. If a user has a data service request, then the user is deemed as an active user. All users will receive control signals from their serving CBSs, but only the active users can receive data services from their serving TBSs. The physical network model is shown in Fig. 1.
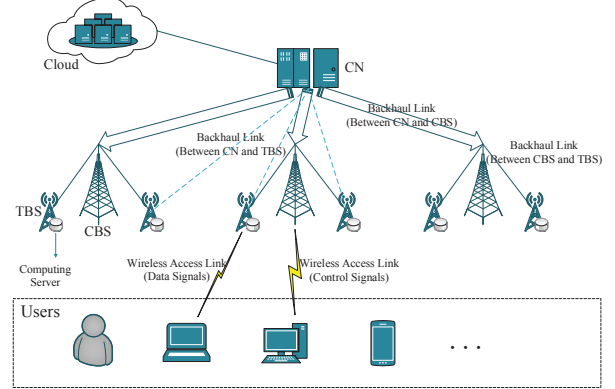


Fig. 1: Physical Network Model

The network operator provides one wide-coverage slice and three data service slices for users' high-throughput, computation-intensive and delay-sensitive requests, respectively. Firstly, the CN and the backhaul links that connect the CN, the CBSs and the TBSs will be abstracted and virtualized into several virtual CNs (vCN) and virtual backhaul links, respectively. Next, each physical TBS is mapped to several vTBSs by virtualization, and each vTBS will be related to one specific UP service. Each physical CBS is mapped to four vCBSs by virtualization, wherein one vCBS will be associated with the CP coverage, while the other three will control the activation of vTBSs. Accordingly, the CP is mapped to four virtualized CPs, which are deployed in the four E2E network slices, respectively; the UP is mapped to three virtualized UPs, which are deployed in the high-throughput, computation-intensive and delay-sensitive slices, respectively. Meanwhile, the communication and computing resources in the system are abstracted into virtual resources to facilitate resource sharing. The network slicing model is shown in Fig. 2. The t-th time slot has the time interval of $[t, t+1)$, where $t \in \{0, 1, 2, ..., T-1\}$ and the duration of each slot is one second.
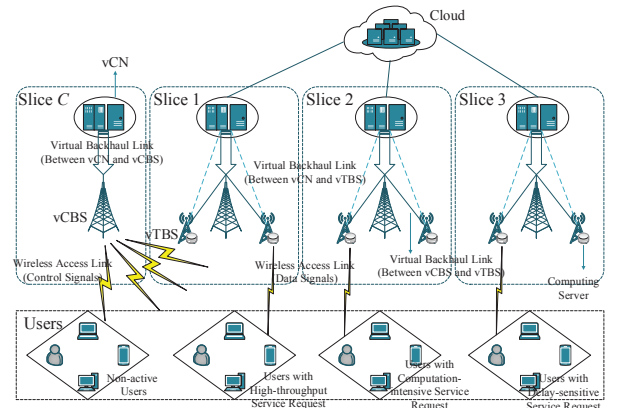


Fig. 2: Network Slicing Model

## A. Network Coverage

The network coverage is supported by the wide-coverage slice (i.e., slice $C$), which involves only the virtualized CP.

The coverage control signals are transmitted from the AMF of the vCN to the vCBSs associated with the CP coverage, and then the vCBSs send the control signals to all users [30].

Each physical CBS is connected with the CN by a backhaul link with the capacity of $Rb_B^C$, and the maximum transmit power of each physical CBS is $P_B^C$. The total downlink bandwidth of the CP is $B_C$, which is evenly divided into $N_{sub}^C$ subcarriers forming the set of CP subcarriers $\boldsymbol{N_{sub}^C} = \{1, 2, \cdots, N_{sub}^C\}$, and each CP subcarrier has a bandwidth of $B_C/N_{sub}^C$. And we define $\boldsymbol{y^C}(t) = \left\{ y_{k,l}^C(t) \right\}$ $\left( k \in \boldsymbol{N_u}, l \in \boldsymbol{N_{sub}^C} \right)$ as the set of binary CP subcarrier allocation indicators, where $y_{k,l}^C(t) = 1$ if CP subcarrier $l$ is allocated to user $k$ at slot $t$ and $y_{k,l}^C(t) = 0$ otherwise. We assume that the transmission of control signal for each user will occupy a small backhaul link capacity of $Rb^C$ and only one CP subcarrier because its traffic is low. Then, we denote the vCBS transmit power allocated to user $k$ at slot $t$ by $P_k^C(t)$, and the set of vCBS transmit power levels at slot $t$ is given by $\boldsymbol{P^C}(t) = \left\{ P_k^C(t) \right\}$ $(k \in \boldsymbol{N_u})$.

The distance between CBS $j$ $(j \in \boldsymbol{N_C})$ and user $k$ $(k \in \boldsymbol{N_u})$ is denoted by $r_{j,k}^C$. Since the number of vCBSs in slice $C$ equals to the number of physical CBSs and does not change over time, we still denote the set of vCBSs in slice $C$ by $\boldsymbol{N_C}$. Each user is associated with its nearest vCBS, and a binary vCBS association indicator $g_{j,k}^C$ $(j \in \boldsymbol{N_C}, k \in \boldsymbol{N_u})$ is defined such that $g_{j,k}^C = 1$ if user $k$ is associated with vCBS $j$, namely $j = \underset{1 \leq j \leq N_C}{\operatorname{argmin}} r_{j,k}^C$, and $g_{j,k}^C = 0$ otherwise.

We assume that the downlink channels of CBSs exhibit independent Rayleigh fading and have a path loss exponent of $\alpha^C$. Therefore, the downlink SINR between user $k$ $(k \in \boldsymbol{N_u})$ and its associated vCBS $j$ $(j \in \boldsymbol{N_C})$ on CP subcarrier $l$ $(l \in \boldsymbol{N_{sub}^C})$ at slot $t$ is

$$SINR_{k,l}^C(t) = \frac{y_{k,l}^C(t) g_{j,k}^C P_k^C(t) h_{j,k,l}^C(t) (r_{j,k}^C)^{-\alpha^C}}{\sum\limits_{\substack{i \in \boldsymbol{N_C} \\ i \neq j}} \sum\limits_{\substack{w \in \boldsymbol{N_u} \\ w \neq k}} y_{w,l}^C(t) g_{i,w}^C P_w^C(t) h_{i,k,l}^C(t) (r_{i,k}^C)^{-\alpha^C} + \sigma^2},$$
(1)

where $r_{j,k}^C$ and $r_{i,k}^C$ denote the transmission distances between user $k$ and its associated vCBS or interfering vCBSs, respectively, and $\sigma^2$ denotes the power of the additive white noise. While $h_{j,k,l}^C(t)$ and $h_{i,k,l}^C(t)$ are the channel power gains of the desired transmission path and interference path at slot $t$, respectively, following two independent unit-mean exponential distributions. The sets of channel power gains of the desired transmission path and interference path at slot $t$ are denoted by $\boldsymbol{h^C}(t) = \left\{ h_{j,k,l}^C(t) \right\}$ and $\boldsymbol{h_{inte}^C}(t) = \left\{ h_{i,k,l}^C(t) \right\}$ $(i, j \in \boldsymbol{N_C}, k \in \boldsymbol{N_u}, l \in \boldsymbol{N_{sub}^C})$, respectively. Besides, $y_{k,l}^C(t)$ and $y_{w,l}^C(t)$ represent that the desired signal and interference signals are transmitted in the common CP subcarrier $l$ at slot $t$, while $g_{j,k}^C$ and $g_{i,w}^C$ are the binary vCBS association indicators of user $k$ and the interference users, respectively.

Due to the fact that only one CP subcarrier is allocated to a user, the downlink SINR between user $k$ and its associated vCBS at slot $t$ is given by

$$SINR_k^C(t) = \sum_{l \in \boldsymbol{N_{sub}^C}} y_{k,l}^C(t) \cdot SINR_{k,l}^C(t),$$
(2)

where the binary CP subcarrier allocation indicator $y_{k,l}^C(t)$ has to satisfy that $\sum\limits_{l \in \boldsymbol{N_{sub}^C}} y_{k,l}^C(t) = 1$.

For any user, if its downlink SINR is above a pre-defined threshold $x_0$, the user is covered by the CP. Then, we define a binary coverage indicator $d_k^C(x_0, t)$ $(k \in \boldsymbol{N_u})$ to represent whether user $k$ is covered by the CP at slot $t$ or not, i.e., $d_k^C(x_0, t) = 1$ if $SINR_k^C(t) > x_0$ and $d_k^C(x_0, t) = 0$ otherwise.

Accordingly, the number of users covered by the CP at slot $t$ is given by

$$N_u^{\text{cov}}(x_0, t) = \sum_{k \in \boldsymbol{N_u}} d_k^C(x_0, t),$$
(3)

and the coverage percentage of slice $C$ at slot $t$ is

$$p_C^{\text{cov}}(x_0, t) = \frac{N_u^{\text{cov}}(x_0, t)}{N_u}.$$
(4)

Next, we will analyze the costs of slice $C$, including the backhaul link capacity, downlink bandwidth, and vCBS transmit power consumption.

The backhaul link capacity of slice $C$ is given by

$$Rb_{tot}^C = Rb^C N_u.$$
(5)

For the downlink bandwidth of slice $C$, let us analyze it from the perspective of the CP subcarrier allocation. Firstly, any CP subcarrier $l$ will be idle if it is not allocated to any user at slot $t$, i.e., the CP subcarrier allocation indicator $y_{k,l}^C(t)$ satisfies that $y_{k,l}^C(t) = 0$ and $1 - y_{k,l}^C(t) = 1$ for any user $k$. At this time, the cumulative product of $1 - y_{k,l}^C(t)$ satisfies that $\prod\limits_{k \in \boldsymbol{N_u}} \left[ 1 - y_{k,l}^C(t) \right] = 1$. However, CP subcarrier $l$ will be occupied if it is allocated to at least one user at slot $t$, i.e., there exist a user $k$ $(k \in \boldsymbol{N_u})$ enables that $y_{k,l}^C(t) = 1$ and $1 - y_{k,l}^C(t) = 0$. At this time, the cumulative product of $1 - y_{k,l}^C(t)$ satisfies that $\prod\limits_{k \in \boldsymbol{N_u}} \left[ 1 - y_{k,l}^C(t) \right] = 0$. From the above analysis, we find that the expression $\prod\limits_{k \in \boldsymbol{N_u}} \left[ 1 - y_{k,l}^C(t) \right]$ can be used to indicate whether CP subcarrier $l$ is idle at slot $t$ or not, and the number of the idle CP subcarriers is $\sum\limits_{l \in \boldsymbol{N_{sub}^C}} \left( \prod\limits_{k \in \boldsymbol{N_u}} \left[ 1 - y_{k,l}^C(t) \right] \right)$. Therefore, the number of the occupied CP subcarriers can be calculated as the total number of the CP subcarriers $N_{sub}^C$ minus the number of the idle CP subcarriers, and the downlink bandwidth of slice $C$ at slot $t$ is the total bandwidth of the occupied CP subcarriers, i.e.,

$$B_{tot}^C(t) = \frac{B_C}{N_{sub}^C} \cdot \left[ N_{sub}^C - \sum_{l \in \boldsymbol{N_{sub}^C}} \left( \prod_{k \in \boldsymbol{N_u}} \left[ 1 - y_{k,l}^C(t) \right] \right) \right].$$
(6)

The total vCBS transmit power consumption of slice $C$ at slot $t$ is expressed as

$$P_{tot}^C(t) = \sum_{k \in \boldsymbol{N_u}} P_k^C(t).$$
(7)

## B. Data Service

The high-throughput slice (i.e., slice 1), the computation-intensive slice (i.e., slice 2) and the delay-sensitive slice (i.e., slice 3) provide data services for active users with different service requirements if the users are covered by the CP [31].

We take the video services as an example, the whole procedure of data services in slice $s$ ($s \in \{1, 2, 3\}$) includes the backhaul link transmission, video processing, and wireless access link transmission. If the active users have video requests, the requested videos will be first delivered from the cloud to the vTBSs through the virtual backhaul links. Then, the videos will be processed and rendered by the vTBSs, which consumes computing resource. After the video processing, the vTBSs will deliver the videos to the corresponding users who request them over the wireless access links [25].

In the CN, the UPF is in charge of forwarding the video data requested by the active users from the data center in the cloud to the vTBSs over the virtual backhaul links, and the SMF is responsible for managing and controlling the data forwarding of the UPF [29].

The RAN of slice $s$ ($s \in \{1, 2, 3\}$) involves both the virtualized UP and CP, and the vCBSs in the virtualized CP control the activation of the vTBSs in the virtualized UP according to the users' requesting states. In this process, the vCBSs send control signals to the vTBSs to switch them on or off [2]. The communication between the vCBSs and the vTBSs is through the virtual backhaul links between them. For simplicity, we ignore this process when analyzing the utilities of the data service slices because its traffic is extreme low.

We define $\boldsymbol{a_T^{(s)}}(t) = \left\{ a_j^{(s)}(t) \right\}$ ($j \in \boldsymbol{N_T}, s \in \{1, 2, 3\}$) [1] as the set of vTBSs activation indicators, therein, $a_j^{(s)}(t)$ is a binary variable, where $a_j^{(s)}(t) = 1$ if vTBS $j$ in slice $s$ is activated at slot $t$ and $a_j^{(s)}(t) = 0$ otherwise. Therefore, the number of activated vTBSs in slice $s$ at slot $t$ is expressed as

$$N_T^{(s)}(t) = \sum_{j \in \boldsymbol{N_T}} a_j^{(s)}(t). \tag{8}$$

and we denote the set of activated vTBSs in slice $s$ at slot $t$ by $\boldsymbol{N_T^{(s)}}(t)$.

In addition, we deem the active users requesting the high-throughput, computation-intensive or delay-sensitive services at slot $t$ as the users in slice 1, slice 2 or slice 3 at current slot. We denote $o_k^{(s)}(t)$ as a binary user service requesting indicator, where $o_k^{(s)}(t) = 1$ if user $k$ is in slice $s$ at slot $t$ and $o_k^{(s)}(t) = 0$ otherwise, and the set of user service requesting indicators is denoted by $\boldsymbol{o_u^{(s)}}(t) = \left\{ o_k^{(s)}(t) \right\}$ ($k \in \boldsymbol{N_u}, s \in \{1, 2, 3\}$). Hence, the number of active users in slice $s$ at slot $t$ is given by

$$N_u^{(s)}(t) = \sum_{k \in \boldsymbol{N_u}} o_k^{(s)}(t). \tag{9}$$

and we denote the set of active users in slice $s$ at slot $t$ by $\boldsymbol{N_u^{(s)}}(t)$.

Each physical TBS is connected with the CN by a backhaul link with the capacity of $Rb_B^T$. We assume that the total downlink bandwidth of the UP is $B_U$ and the maximum

transmit power of each physical TBS is $P_B^T$. The communication resource, including the backhaul link capacity, downlink bandwidth and vTBS transmit power will only be allocated to the active users covered by the CP.

We denote the backhaul link capacity allocated to user $k$ (covered by the CP) in slice $s$ at slot $t$ by $Rb_k^{(s)}(t)$, and $\boldsymbol{Rb^{(s)}}(t) = \left\{ Rb_k^{(s)}(t) \right\}$ ($k \in \boldsymbol{N_u^{(s)}}(t), s \in \{1, 2, 3\}$) is the set of backhaul link capacities of slice $s$ at slot $t$.

The downlink bandwidth in the UP is evenly divided into $N_{sub}^U$ subcarriers forming the set of UP subcarriers $\boldsymbol{N_{sub}^U} = \{1, 2, ..., N_{sub}^U\}$, and each UP subcarrier has a bandwidth of $B_U / N_{sub}^U$. We define $\boldsymbol{y^{(s)}}(t) = \left\{ y_{k,l}^{(s)}(t) \right\}$ ($k \in \boldsymbol{N_u^{(s)}}(t), l \in \boldsymbol{N_{sub}^U}, s \in \{1, 2, 3\}$) as the set of binary UP subcarrier allocation indicators, where $y_{k,l}^{(s)}(t) = 1$ if UP subcarrier $l$ is allocated to user $k$ (covered by the CP) in slice $s$ at slot $t$ and $y_{k,l}^{(s)}(t) = 0$ otherwise.

The vTBS transmit power allocated to user $k$ (covered by the CP) in slice $s$ on UP subcarrier $l$ at slot $t$ is denoted by $p_{k,l}^{(s)}(t)$, and $\boldsymbol{p^{(s)}}(t) = \left\{ p_{k,l}^{(s)}(t) \right\}$ ($k \in \boldsymbol{N_u^{(s)}}(t), l \in \boldsymbol{N_{sub}^U}, s \in \{1, 2, 3\}$) is the set of vTBS transmit power levels in slice $s$ at slot $t$. Therefore, the overall vTBS transmit power allocated to user $k$ (covered by the CP) in slice $s$ at slot $t$ is the sum of vTBS transmit power on all the UP subcarriers allocated to the user, i.e.,

$$P_k^{(s)}(t) = \sum_{l \in \boldsymbol{N_{sub}^U}} y_{k,l}^{(s)}(t) \cdot p_{k,l}^{(s)}(t) \tag{10}$$

Besides, we denote the set of downlink channel power gains of slice $s$ at slot $t$ by $\boldsymbol{h^{(s)}}(t) = \left\{ h_{j,k,l}^T(t) | k \in \boldsymbol{N_u^{(s)}}(t) \right\}$ ($j \in \boldsymbol{N_T^{(s)}}(t), l \in \boldsymbol{N_{sub}^U}, s \in \{1, 2, 3\}$), and $h_{j,k,l}^T(t)$ is the downlink channel power gain between TBS $j$ and user $k$ on UP subcarrier $l$ at slot $t$.

The distance between TBS $j$ ($j \in \boldsymbol{N_T}$) and user $k$ ($k \in \boldsymbol{N_u}$) is denoted by $r_{j,k}^T$. Each active user is associated with its nearest vTBS, and a binary vTBS association indicator $g_{j,k}^{(s)}(t)$ ($j \in \boldsymbol{N_T^{(s)}}(t), k \in \boldsymbol{N_u^{(s)}}(t), s \in \{1, 2, 3\}$) is defined such that $g_{j,k}^{(s)}(t) = 1$ if user $k$ (covered by the CP) in slice $s$ is associated with vTBS $j$ at slot $t$, namely $j = \underset{1 \leq j \leq N_T^{(s)}(t)}{\arg\min} r_{j,k}^T$, and $g_{j,k}^{(s)}(t) = 0$ otherwise.

We assume that the downlink channels of vTBSs exhibit independent Rayleigh fading and have a path loss exponent $\alpha^T$. Therefore, the SINR between user $k$ $\left( k \in \boldsymbol{N_u^{(s)}}(t) \right)$ and its associated vTBS $j$ $\left( j \in \boldsymbol{N_T^{(s)}}(t) \right)$ on UP subcarrier $l$ ($l \in \boldsymbol{N_{sub}^U}$) in slice $s$ at slot $t$ is

$$SINR_{k,l}^{(s)}(t) =$$

$$\frac{y_{k,l}^{(s)}(t) g_{j,k}^{(s)}(t) p_{k,l}^{(s)}(t) h_{j,k,l}^T(t) \left( r_{j,k}^T \right)^{-\alpha^T}}{\sum\limits_{\substack{i \in \boldsymbol{N_T^{(s)}}(t) \\ i \neq j}} \sum\limits_{\substack{w \in \boldsymbol{N_u^{(s)}}(t) \\ w \neq k}} y_{w,l}^{(s)}(t) g_{i,w}^{(s)}(t) p_{w,l}^{(s)}(t) h_{i,k,l}^T(t) \left( r_{i,k}^T \right)^{-\alpha^T} + \sigma^2}, \tag{11}$$

---

[1] Note that the superscript ($s$) in all symbols represents slice $s$ ($s \in \{1, 2, 3\}$).

where $r_{j,k}^T$ and $r_{i,k}^T$ denote the transmission distances between user $k$ and its associated vTBS or interfering vTBSs, respectively, and $\sigma^2$ denotes the power of the additive white noise. While $h_{j,k,l}^T(t)$ and $h_{i,k,l}^T(t)$ are the channel power gains of the desired transmission path and interference path on the common UP subcarrier $l$ at slot $t$, which follow two independent unit-mean exponential distributions. Specifically, the set of interfering channel power gains is denoted by $\boldsymbol{h}_{inte}^{(s)}(t) = \left\{ h_{i,k,l}^T(t) | k \in \boldsymbol{N_u^{(s)}}(t) \right\}$ $\left( i \in \boldsymbol{N_T^{(s)}}(t), l \in \boldsymbol{N_{sub}^U}, s \in \{1,2,3\} \right)$. Furthermore, $y_{k,l}^{(s)}(t)$ and $y_{w,l}^{(s)}(t)$ represent that the desired signal and interference signals are transmitted in the common UP subcarrier $l$ in slice $s$ at slot $t$, while $g_{j,k}^{(s)}(t)$ and $g_{i,w}^{(s)}(t)$ are the vTBS association indicators of user $k$ and the interference users in slice $s$ at slot $t$, respectively. Accordingly, we obtain the downlink rate of user $k$ on UP subcarrier $l$ in slice $s$ at slot $t$ as

$$r_{k,l}^{(s)}(t) = \frac{B_U}{N_{sub}^U} \cdot \log_2\left[1 + SINR_{k,l}^{(s)}(t)\right]. \tag{12}$$

and the overall downlink rate of user $k$ in slice $s$ at slot $t$ is the sum of the downlink rate on all the UP subcarriers allocated to the user, i.e.,

$$R_k^{(s)}(t) = \sum_{l \in \boldsymbol{N_{sub}^U}} y_{k,l}^{(s)}(t) \cdot r_{k,l}^{(s)}(t). \tag{13}$$

The throughput of user $k$ (covered by the CP) in slice $s$ at slot $t$ is $\min\left\{ R_k^{(s)}(t), Rb_k^{(s)}(t) \right\}$ [32]. Therefore, we have the total throughput of E2E network slice $s$ at slot $t$, i.e.,

$$R_{tot}^{(s)}(t) = \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} \min\left\{ R_k^{(s)}(t), Rb_k^{(s)}(t) \right\}. \tag{14}$$

The computing resource of the network is represented by computing capability in CPU cycle/s [33], and the maximum computing capability of each physical TBS is $F_B^T$. The vTBS computing capability allocated to user $k$ (covered by the CP) in slice $s$ at slot $t$ is denoted by $f_k^{(s)}(t)$, and $\boldsymbol{f_k^{(s)}}(t) = \left\{ f_k^{(s)}(t) \right\}$ $\left( k \in \boldsymbol{N_u^{(s)}}(t), s \in \{1,2,3\} \right)$ is the set of vTBS computing capabilities in slice $s$ at slot $t$. Therefore, we obtain the total vTBS computing capability of E2E network slice $s$ at slot $t$ as

$$f_{tot}^{(s)}(t) = \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} f_k^{(s)}(t). \tag{15}$$

To analyze the delay performance of the E2E network slicing, three data queues are discussed. Let $\boldsymbol{G^{(s)}}(t) = \left\{ G_k^{(s)}(t) \right\}$, $\boldsymbol{Z^{(s)}}(t) = \left\{ Z_k^{(s)}(t) \right\}$ and $\boldsymbol{H^{(s)}}(t) = \left\{ H_k^{(s)}(t) \right\}$ $\left( k \in \boldsymbol{N_u^{(s)}}(t), s \in \{1,2,3\} \right)$ denote the current backhaul link data queue backlogs, computing queue backlogs and wireless access link data queue backlogs in slice $s$ at slot $t$, respectively. We define $\boldsymbol{Ah^{(s)}}(t) = \left\{ Ah_k^{(s)}(t) \right\}$ $\left( k \in \boldsymbol{N_u^{(s)}}(t), s \in \{1,2,3\} \right)$ as the random data arrivals on the backhaul links in slice $s$ at slot $t$, where $\boldsymbol{Ah^{(s)}}$ is assumed to be independent and identically distributed (i.i.d.) over time and follows a Poisson arrival process with the average rate $\gamma_a^{(s)}$ (in bits/s). Therefore, we model the backhaul link data queuing process for user $k$

(covered by the CP) in slice $s$ by

$$G_k^{(s)}(t+1) = \max\left\{ G_k^{(s)}(t) - Rb_k^{(s)}(t), 0 \right\} + Ah_k^{(s)}(t), \forall k, s. \tag{16}$$

Since the data will be processed at the vTBS after the backhaul link transmission, the arrival of the computing queue is the departure of the backhaul link data queue. Then, the computing queuing process for user $k$ (covered by the CP) in slice $s$ is modeled by

$$Z_k^{(s)}(t+1) = \max\left\{ Z_k^{(s)}(t) - f_k^{(s)}(t)/c_k^{(s)}, 0 \right\} + Rb_k^{(s)}(t), \forall k, s. \tag{17}$$

where $c_k^{(s)}$ is the number of CPU cycles required per bit in CPU cycle/bit [33] and $\left[ f_k^{(s)}(t)/c_k^{(s)} \right]$ is the computing rate, i.e., the number of bits that computed per second for user $k$ in slice $s$ at slot $t$, which is expressed in bit/s.

The data will be transmitted to users over the wireless access links after the vTBS processing, hence, the arrival of the wireless access link queue is the departure of the computing queue. We model the wireless access link data queuing process for user $k$ (covered by the CP) in slice $s$ by

$$H_k^{(s)}(t+1) = \max\left[ H_k^{(s)}(t) - R_k^{(s)}(t), 0 \right] + f_k^{(s)}(t)/c_k^{(s)}, \forall k, s. \tag{18}$$

According to Little's Theorem [34], the user's average queuing delay is proportional to the average queue length, therefore, we can represent the delay of user $k$ in slice $s$ by the sum of the queue length $Y_k^{(s)}(t) = \left[ G_k^{(s)}(t) + Z_k^{(s)}(t) + H_k^{(s)}(t) \right]$ [35]. The access delay is not considered here because of its small proportion [36]. Hence, we optimize the delay of the E2E network slicing by minimizing the total queue length. The total queue length of slice $s$ at slot $t$ is given by

$$\begin{aligned} Y_{tot}^{(s)}(t) &= \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} Y_k^{(s)}(t) \\ &= \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[ G_k^{(s)}(t) + Z_k^{(s)}(t) + H_k^{(s)}(t) \right]. \end{aligned} \tag{19}$$

and the average delay of slice $s$ can be calculated as

$$D_{ave}^{(s)}(t) = \frac{Y_{tot}^{(s)}(t)}{\gamma_a^{(s)} N_u^{(s)}(t)}. \tag{20}$$

Specifically, since slice 1, slice 2 and slice 3 can only serve users covered by the CP, no communication or computing resource will be allocated to the users not covered, i.e., if $d_k^C(x_0, t) = 0$, then $Rb_k^{(s)}(t) = 0$, $y_{k,l}^{(s)}(t) = 0$, $p_{k,l}^{(s)}(t) = 0$ and $f_k^{(s)}(t) = 0$; and there is no data arrival for the users not covered, i.e., if $d_k^C(x_0, t) = 0$, then $Ah_k^{(s)}(t) = 0$.

Afterwards, we analyze the costs of the data service slices, including the backhaul link capacity, downlink bandwidth, vTBS transmit and vTBS computing power consumptions.

The backhaul link capacity of slice $s$ at slot $t$ is given by

$$Rb_{tot}^{(s)}(t) = \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} Rb_k^{(s)}(t), \tag{21}$$

For the downlink bandwidth of slice $s$, let us analyze it from the perspective of the UP subcarrier allocation. Firstly, any UP subcarrier $l$ will be occupied in slice $s$ if it is allocated to at

least one user (covered by the CP) in slice $s$ at slot $t$, i.e., there exist a user $k$ $\left(k \in \boldsymbol{N_u^{(s)}}(t) \text{ and } d_k^C(x_0,t)=1\right)$ enables that $y_{k,l}^{(s)}(t) = 1$ and $1-y_{k,l}^{(s)}(t) = 0$. At this time, the cumulative product of $1-y_{k,l}^{(s)}(t)$ satisfies that $\prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1-y_{k,l}^{(s)}(t)\right] = 0$ and $1- \prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1-y_{k,l}^{(s)}(t)\right]=1$. However, UP subcarrier $l$ will be idle if it is not allocated to any user in slice $s$ at slot $t$, i.e., the UP subcarrier allocation indicator $y_{k,l}^{(s)}(t)$ satisfies that $y_{k,l}^{(s)}(t) = 0$ and $1 - y_{k,l}^{(s)}(t) = 1$ for any user $k$ in slice $s$. At this time, the cumulative product of $1 - y_{k,l}^{(s)}(t)$ satisfies that $\prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1-y_{k,l}^{(s)}(t)\right] = 1$ and $1- \prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1-y_{k,l}^{(s)}(t)\right] = 0$. From the above analysis, we find that the expression $1 - \prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1-y_{k,l}^{(s)}(t)\right]$ can be used to indicate whether UP subcarrier $l$ is occupied in slice $s$ at slot $t$ or not, and the number of the occupied UP subcarriers is $\sum_{l \in \boldsymbol{N_{sub}^U}} \left(1 - \prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1-y_{k,l}^{(s)}(t)\right]\right)$. Therefore, the downlink bandwidth of slice $s$ at slot $t$ is the total bandwidth of the occupied UP subcarriers, i.e.,

$$B_{tot}^{(s)}(t) = \frac{B_U}{N_{sub}^U} \cdot \sum_{l \in \boldsymbol{N_{sub}^U}} \left(1 - \prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1-y_{k,l}^{(s)}(t)\right]\right). \quad (22)$$

The vTBS transmit power consumption of slice $s$ at slot $t$ is expressed as

$$P_{tot}^{(s)}(t) = \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} P_k^{(s)}(t). \quad (23)$$

The vTBS computing power consumption of user $k$ in slice $s$ at slot $t$ is [33]

$$P_{ck}^{(s)}(t) = \kappa \cdot \left[f_k^{(s)}(t)\right]^3, \quad (24)$$

accordingly, the vTBS computing power consumption of slice $s$ at slot $t$ is given by

$$P_{ctot}^{(s)}(t) = \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} P_{ck}^{(s)}(t). \quad (25)$$

## III. Sum-utility Maximization Problem

In this section, we will define the utilities of the wide-coverage, high-throughput, computation-intensive and delay-sensitive slices, respectively, and then the sum-utility maximization problem will be formulated.

### A. Utility Definition

From the perspective of the network operator, the utility of CUPS-E2E network slicing is defined as the income minus the costs of the physical network. For slice $C$, the utility is defined as the difference between the income quantified by the number of users covered by the CP minus the costs in terms of the backhaul link capacity, the downlink bandwidth and the vCBS transmit power consumption in slice $C$, i.e.,

$$U^C(t) = m_C N_u^{\text{cov}}(x_0,t) - \xi \cdot \max\left\{\rho-p_C^{cov}(x_0,t), 0\right\} \\ - \left[\omega R b_{\text{tot}}^C + \delta B_{\text{tot}}^C + \beta P_{\text{tot}}^C(t)\right], \quad (26)$$

where $m_C$ is the income from each user covered by the CP, while $\omega$, $\delta$ and $\beta$ are the costs per unit backhaul link capacity, per unit downlink bandwidth and per unit power consumption, respectively. To guarantee the fundamental coverage for the users, the CP coverage percentage has to satisfy the constraint: $p_C^{cov}(x_0,t) \geq \rho$, $\forall t, x_0 \leq -5\text{dB}$, where $\rho$ is the threshold of the CP coverage percentage. The coverage penalty $\xi \cdot \max\{\rho-p_C^{cov}(x_0,t), 0\}$ will be paid if the constraint on the CP coverage percentage is violated, where $\xi$ is a positive penalty coefficient.

For slice 1, providing high throughput has the priority. Therefore, the utility of slice 1 is defined as the difference between the income from the total throughput provided and the costs of slice 1, including the backhaul link capacity, downlink bandwidth, the vTBS transmit power and vTBS computing power consumptions in slice 1, i.e.,

$$U^{(1)}(t) = m_1 R_{tot}^{(1)}(t) \\ - \left\{\omega R b_{tot}^{(1)}(t) + \delta B_{tot}^{(1)}(t) + \beta \left[P_{tot}^{(1)}(t) + P_{ctot}^{(1)}(t)\right]\right\}, \quad (27)$$

where $m_1$ is the income per unit throughput.

Providing high computing capability is the primary target for slice 2. Accordingly, the utility of slice 2 is defined as the income from the total computing capability provided minus the costs in terms of the backhaul link capacity, downlink bandwidth, the vTBS transmit power and vTBS computing power consumptions in slice 2, i.e.,

$$U^{(2)}(t) = m_2 f_{tot}^{(2)}(t) \\ - \left\{\omega R b_{tot}^{(2)}(t) + \delta B_{tot}^{(2)}(t) + \beta \left[P_{tot}^{(2)}(t) + P_{ctot}^{(2)}(t)\right]\right\}, \quad (28)$$

where $m_2$ is the income per unit computing capability.

The main objective of slice 3 is to provide users as lower delay as possible. Consequently, the utility of slice 3 is defined as the difference between the income defined by the queuing delay gain [8] and the costs in terms of the backhaul link capacity, downlink bandwidth, the vTBS transmit power and vTBS computing power consumptions in slice 3, i.e.,

$$U^{(3)}(t) = \left[\psi N_u^{(3)}(t) - m_3 Y_{tot}^{(3)}(t)\right] - \eta \cdot \max\left\{D_{ave}^{(3)}(t) - \zeta, 0\right\} \\ - \left\{\omega R b_{tot}^{(3)}(t) + \delta B_{tot}^{(3)}(t) + \beta \left[P_{tot}^{(3)}(t) + P_{ctot}^{(3)}(t)\right]\right\}, \quad (29)$$

where $m_3$ is the income per unit queue length, i.e., the unit price of delay gain, and $\psi$ is an initial maximum benefit coefficient of slice 3 to guarantee the non-negativity of the utility [37]. To ensure the quality of service (QoS) requirement for users in the delay-sensitive slice 3, the average delay of slice 3 has to meet the constraint: $D_{ave}^{(3)}(t) \leq \zeta$, $\forall t$, where $\zeta$ is the threshold of the average delay of slice 3. If the constraint on the average delay is violated, the average delay penalty $\eta \cdot \max\left\{D_{ave}^{(3)}(t) - \zeta, 0\right\}$ will be paid, where $\eta$ is a positive penalty coefficient.

Hence, we have the sum-utility of the entire network as the sum of $U^C(t)$, $U^{(1)}(t)$, $U^{(2)}(t)$ and $U^{(3)}(t)$, i,e,

$$U(t) = U^C(t) + U^{(1)}(t) + U^{(2)}(t) + U^{(3)}(t). \quad (30)$$

## B. Sum-utility Maximization Problem Formulation

We aim to maximize the expected cumulative discounted sum-utility of CUPS-E2E network slicing, while satisfying the vTBS activation and resource allocation constraints, including the vCBS transmit power, CP subcarrier, backhaul link capacity, UP subcarrier, vTBS transmit power and vTBS computing capability allocations. We formulate the sum-utility maximization problem as follows

$$\min_{\{\boldsymbol{V}(t)\}} \sum_{n=0}^{\infty} -\gamma^n U(t+n)$$

$$s.t. \; C1 : \sum_{k \in \boldsymbol{N_u}} g_{j,k}^C \cdot y_{k,l}^C(t) \le 1, \forall j \in \boldsymbol{N_C}, \forall l \in \boldsymbol{N_{sub}^C},$$

$$C2 : \sum_{l \in \boldsymbol{N_{sub}^C}} y_{k,l}^C(t) = 1, \forall k \in \boldsymbol{N_u},$$

$$C3 : P_k^C(t) \le P_{max}^C, \forall k \in \boldsymbol{N_u},$$

$$C4 : \sum_{k \in \boldsymbol{N_u}} P_k^C(t) \cdot g_{j,k}^C \le P_B^C, \forall j \in \boldsymbol{N_C},$$

$$C5 : \sum_{s \in \{1,2,3\}} \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} Rb_k^{(s)}(t) \cdot g_{j,k}^{(s)}(t) \le Rb_B^T, \forall j \in \boldsymbol{N_T^{(s)}}(t),$$

$$C6 : \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} g_{j,k}^{(s)}(t) \cdot y_{k,l}^{(s)}(t) \le 1, \forall j \in \boldsymbol{N_T^{(s)}}(t),$$
$$\forall l \in \boldsymbol{N_{sub}^U}, s \in \{1,2,3\},$$

$$C7 : \sum_{s \in \{1,2,3\}} \left( 1 - \prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[ 1 - y_{k,l}^{(s)}(t) \right] \right) \le 1, \forall l \in \boldsymbol{N_{sub}^U},$$

$$C8 : P_k^{(s)}(t) \le P_{max}^{(s)}, \forall k \in \boldsymbol{N_u^{(s)}}(t), s \in \{1,2,3\},$$

$$C9 : \sum_{s \in \{1,2,3\}} \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} P_k^{(s)}(t) \cdot g_{j,k}^{(s)}(t) \le P_B^T, \forall j \in \boldsymbol{N_T^{(s)}}(t),$$

$$C10 : \sum_{s \in \{1,2,3\}} \sum_{k \in \boldsymbol{N_u^{(s)}}(t)} f_k^{(s)}(t) \cdot g_{j,k}^{(s)}(t) \le F_B^T, \forall j \in \boldsymbol{N_T^{(s)}}(t),$$

$$C11 : a_T^{(s)}(t), y_{k,l}^{(s)}(t) \le \{0,1\}, \forall k \in \boldsymbol{N_u^{(s)}}(t),$$
$$\forall l \in \boldsymbol{N_{sub}^U}, s \in \{1,2,3\},$$

$$C12 : p_C^{cov}(x_0, t) \ge \rho, \forall x_0 \le -5\text{dB},$$

$$C13 : D_{ave}^{(3)}(t) \le \zeta, \tag{31}$$

where $\gamma \in [0,1]$ is the discount factor, and variable set $\boldsymbol{V}(t) = \left[ \boldsymbol{y^C}(t), \boldsymbol{P^C}(t), \boldsymbol{a_T^{(s)}}(t), \boldsymbol{Rb^{(s)}}(t), \boldsymbol{y^{(s)}}(t), \boldsymbol{p^{(s)}}(t), \boldsymbol{f^{(s)}}(t) \right]$. Besides, $P_{max}^C$ and $P_{max}^{(s)}$ are the maximum vCBS and vTBS transmit power that can be allocated to a single user in slice $C$ and slice $s$, respectively. In the constraints, $C3$ and $C8$ are the vCBS and vTBS transmit power constrains, respectively. $C1$ and $C6$ guarantee that one CP subcarrier or UP subcarrier can only be allocated to at most one user associated with the same vCBS or vTBS at the same slot, respectively, while $C2$ ensures that only one CP subcarrier can be allocated to one user. $C7$ guarantees that one UP subcarrier can only be allocated to users in at most one slice, which guarantees the bandwidth isolation among network slices. From the analysis in Section II-B, the expression $\left(1 - \prod_{k \in \boldsymbol{N_u^{(s)}}(t)} \left[1 - y_{k,l}^{(s)}(t)\right]\right)$ can be used as an indicator to represent whether UP subcarrier

$l$ is allocated to slice $s$ at slot $t$ or not, therefore, the sum of the indicators of three data service slices means the number of slices that the same UP subcarrier allocated to at the same slot, which should not be more than 1. $C4$, $C5$, $C9$ and $C10$ guarantee that the total vCBS transmit power, backhaul link capacity, vTBS transmit power and vTBS computing capability allocated to all the users associated with one CBS or TBS should not exceed the total available vCBS transmit power, backhaul link capacity, vTBS transmit power and vTBS computing capability of the CBS or TBS, respectively. $C11$ represents the value range of the binary variables $a_T^{(s)}(t)$ and $y_{k,l}^{(s)}(t)$. $C12$ and $C13$ are the CP coverage percentage constraint of slice $C$ and the average delay constraint of slice 3, respectively.

## IV. DDPG-BASED SUM-UTILITY MAXIMIZATION

In this section, we will firstly give a brief introduction about DDPG algorithm. Then, we will present the state space, action space, and reward function of the sum-utility maximization problem. Next, a DDPG-based sum-utility maximization algorithm is designed to solve the sum-utility maximization problem for CUPS-E2E network slicing. Finally, the complexity of our algorithm will also be analyzed.

### A. DDPG Algorithm

As a deep reinforcement learning algorithm, DDPG can be applied for high-dimensional continuous joint vTBS activation and resource allocation for CUPS-E2E network slicing, which involves two online neural networks and two target neural networks: online critic network, online actor network, target critic network and target actor network with the network parameters $\theta^Q$, $\theta^\mu$, $\theta^{Q'}$ and $\theta^{\mu'}$, respectively [38].

The objective function of DDPG algorithm is defined as the expectation of the discounted accumulative reward, i.e.,

$$J(\theta^\mu) = E\left[ r_0 + \gamma r_1 + \gamma^2 r_2 + \cdots + \gamma^n r_n \right], \tag{32}$$

where $r_0, r_1, \cdots, r_n$ denote the rewards at each time step.

The online actor network is used to learn the action selection under a given state, which maps the current state $s^t$ to a certain action $a^t$ based on the present policy $\mu(s^t | \theta^\mu)$. The online critic network evaluates the action by a state-action value function $Q(s^t, a^t | \theta^Q)$, the input of which includes both the current state and the action obtained by the online actor network. And the online critic network parameter $\theta^Q$ is updated by the stochastic gradient descent method to minimize the loss function of the online critic network. The loss function $L$ is represented by the mean square error, i.e.,

$$L = E[(y^t - Q(s^t, a^t | \theta^Q))^2], \tag{33}$$

where $y^t$ is the target state-action value, which contains the current reward and the state-action value of the next time step. Deep neural network (DNN) is used to approximate the policy function $\mu(s^t | \theta^\mu)$ and state-action value function $Q(s^t, a^t | \theta^Q)$. To make the state-action value more stable during the training stage, the target actor network and target critic network are

used to select the action and calculate the state-action value of the next time step, respectively. Therefore, $y^t$ is given by

$$y^t = r^t + \gamma Q'(s^{t+1}, \mu'(s^{t+1}|\theta^{\mu'})|\theta^{Q'}), \qquad (34)$$

The online actor network parameter $\theta^\mu$ is updated by the gradient descent method, i.e.,

$$\nabla_{\theta^\mu} J \approx E[\nabla_a Q(s^t, \mu(s^t|\theta^\mu)|\theta^Q) \cdot \nabla_{\theta^\mu} \mu(s^t|\theta^\mu)]. \qquad (35)$$

To explore potentially better strategies, a stochastic noise $\mathcal{N}_o$ is introduced to affect the action selection during the network model training stage, and the action selection with the stochastic noise is expressed as

$$a^t = \mu(s^t|\theta^\mu) + \mathcal{N}_o, \qquad (36)$$

The target critic network parameter $\theta^{Q'}$ and target actor network parameter $\theta^{\mu'}$ are updated by the soft update method with a small constant $\tau$, i.e.,

$$\begin{cases} \theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}, \\ \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}. \end{cases} \qquad (37)$$

### B. DDPG-based Sum-utility Maximization Algorithm

We design a DDPG-based sum-utility maximization algorithm to find the optimal joint vTBSs activation, communication and computing resources allocation policy for CUPS-E2E network slicing while ensuring that the long-term sum-utility is maximum. The state space, action space and reward function are as follows:

*1) State Space:* The state space $S^t$ includes the wireless channel states, user's service requesting states and data arrivals. Therein, the channel states contains the desired channel power gains and interfering channel power gains in slice $C$ and slice $s$ $(s = 1, 2, 3)$ at current slot:

$$S^t = [h^C(t), h^C_{inte}(t), h^{(s)}(t), h^{(s)}_{inte}(t), o^{(s)}_u(t), Ah^{(s)}(t)]. \qquad (38)$$

*2) Action Space:* The network operator has to decide the activation of vTBSs and the allocation of communication and computing resources for CUPS-E2E network slicing. For network coverage slice (slice $C$), the network operator has to decide the allocation of CP subcarrier and vCBS transmit power. For data service slices (slice 1, slice 2 and slice 3), the network operator has to decide the allocation of backhaul link capacity, UP subcarrier, vTBS transmit power and vTBS computing capability. Accordingly, we have the action space vector $A^t$ as:

$$A^t = [y^C(t), P^C(t), a^{(s)}_T(t), Rb^{(s)}(t), y^{(s)}(t), p^{(s)}(t), f^{(s)}(t)]. \qquad (39)$$

Especially, we relax the binary variables in $y^C(t)$, $a^{(s)}_T(t)$ and $y^{(s)}(t)$ to continuous variables with the range of $[0, 1]$ to satisfy the continuity condition of the action space in DDPG algorithm. Importantly, the network operator will check whether the selected actions content the constraints of the sum-utility maximization problem or not at each time slot, and the actions which violate the constraints will be discarded or modified. Specifically, some of the selected actions $P^C(t)$, $Rb^{(s)}(t)$, $p^{(s)}(t)$ and $f^{(s)}(t)$ will be modified to the lower

values to satisfy the resource allocation constraints if they exceed the limitations in constraints $C3$, $C4$, $C5$, $C8$, $C9$ and $C10$. Besides, if some of the selected actions $y^C(t)$ and $y^{(s)}(t)$ violate the CP subcarrier allocation indicator constraints $C1$ and $C2$ or the UP subcarrier allocation indicator constraints $C6$ and $C7$, only one of the CP subcarrier allocation indicators for a user or one of the UP subcarrier allocation indicators will be reserved and the others will be modified to zero to satisfy the constrains $C1$, $C2$, $C6$ and $C7$.

*3) Reward Function:* The reward function is denoted by the sum-utility of CUPS-E2E network slicing, which is determined by the current state $S^t$ and action $A^t$:

$$Re^t = U(S^t, A^t). \qquad (40)$$

---

**Algorithm 1** Training Stage of DDPG-based Sum-utility Maximization Algorithm

---

1: Initialization:
   Initialize the parameters of online critic network $\theta^Q$ and online actor network $\theta^\mu$;
   Initialize the parameters of target critic network $\theta^{Q'} \leftarrow \theta^Q$ and online actor network $\theta^{\mu'} \leftarrow \theta^\mu$;
   Initialize the sizes of replay buffer $D$ and mini-batch $M$;
2: **for** $episode = 1, 2, \cdots K$ **do**
3:     Receive initial observation state $S^1$: $[h^C(1), h^C_{inte}(1), h^{(s)}(1), h^{(s)}_{inte}(1), o^{(s)}_u(1), Ah^{(s)}(1)]$;
4:     **for** $t = 1, 2, \cdots T$ **do**
5:         Select action $A^t$ based on the current deterministic policy $\mu$ with the stochastic noise according to (36);
6:         Take action $A^t$, obtain the immediate reward $Re^t$ and generate the next state $S^{t+1}$;
7:         Store the data set $(S^t, A^t, Re^t, S^{t+1})$ in $D$;
8:         **if** Replay buffer $D$ is full **do**
9:             Select $M$ data sets randomly from $D$ and build a mini-batch $(S_i, A_i, Re_i, S_{i+1})$ $(i = 1, 2, \cdots M)$;
10:           Set $Q_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$;
11:           Update $\theta^Q$ by gradient descent method to minimize the loss $\nabla_{\theta^Q} J \approx \frac{1}{M} \sum_i^M [(Q_i - Q(s_i, a_i|\theta^Q)) \cdot \nabla_{\theta^Q} Q(s_i, a_i|\theta^Q)]$;
12:           Update $\theta^\mu$ by gradient descent method according to (35)
   $\nabla_{\theta^\mu} J \approx \frac{1}{M} \sum_i^M [\nabla_a Q(s_i, a_i|\theta^Q) \cdot \nabla_{\theta^\mu} \mu(s_i|\theta^\mu)]$;
13:           Soft update $\theta^{Q'}$ and $\theta^{\mu'}$ according to (37);
14:           Update the state: $S^t \leftarrow S^{t+1}$;
15:         **end if**
16:     **end for**
17: **end for**
18: **return**

---

Our algorithm contains the training stage and validating stage as shown in Algorithm 1 and Algorithm 2, respectively. The agent learns the mapping function between the users' UP service requests as well as wireless channel states and the action, including the vTBS activation and resource allocation in the training stage. The vTBS activation and resource allocation are determined based on the current deterministic policy $\mu$ according to the current state $S^t$ at each time step. With the selected vTBS activation and resource allocation,

**Algorithm 2** Validating Stage of DDPG-based Sum-utility Maximization Algorithm

---

1: Receive initial observation state $\boldsymbol{S^1}$: $[\boldsymbol{h^C}(1), \boldsymbol{h_{inte}^C}(1),$ $\boldsymbol{h^{(s)}}(1), \boldsymbol{h_{inte}^{(s)}}(1), \boldsymbol{o_u^{(s)}}(1), \boldsymbol{Ah^{(s)}}(1)]$;

2: **for** $t = 1, 2, \cdots T$ **do**

3:     Select action $\boldsymbol{A^t}$ based on the trained deterministic policy $\mu$: $\boldsymbol{A^t} = \mu(\boldsymbol{S^t}|\theta^\mu)$;

4:     Take action $\boldsymbol{A^t}$, obtain the immediate reward $\boldsymbol{Re^t}$ and generate the next state $\boldsymbol{S^{t+1}}$;

5:     Update the state: $\boldsymbol{S^t} \leftarrow \boldsymbol{S^{t+1}}$;

6: **end for**

7: **return** Validating Results

---

the immediate reward $\boldsymbol{Re^t}$ and the state of the next step $\boldsymbol{S^{t+1}}$ will be generated. In the training stage, the data set $(\boldsymbol{S^t}, \boldsymbol{A^t}, \boldsymbol{Re^t}, \boldsymbol{S^{t+1}})$ of each time step will be stored in a space named replay buffer as the training data. A mini-batch of the training data is sampled from the reply buffer at each time step to update the parameters of the online critic network $\theta^Q$ and online actor network $\theta^\mu$. In the validating stage, the joint vTBS activation and resource allocation are determined by the trained deterministic policy $\mu$, and the immediate reward is obtained at the same time.

### C. Complexity Analysis

In this subsection, we analyse the computation complexity of the training process of the DDPG-based Sum-utility Maximization Algorithm. DNNs are used as function approximators for both the actor and critic networks of the DDPG-based algorithm. The DNNs have one input layer, one output layer, and several hidden layers, which are fully connected. Accordingly, the computation complexity analysis is mainly based on the DNN model [39].

The actor network takes the users' requests and wireless channel states as input and outputs the actions including the resource allocation and vTBS activation policy, while the critic network inputs the state and action information and outputs the corresponding state-action value to evaluate the action selected by the action network. Let $N_L^a$ and $N_L^c$ denote the numbers of hidden layers of the actor and critic networks, respectively, and the numbers of neurons in the $m$th hidden layer of the actor and critic networks are denoted by $n_m^a$ and $n_m^c$, respectively. In addition, we denote the dimensions of the state and action spaces by $|\mathcal{S}_1|$ and $|\mathcal{A}_1|$, respectively, which can be obtained based on (38) and (39), and assume that the parameters of the DNNs will converge after $F_1^{conv}$ episodes and $N_1^{conv}$ time slots. Accordingly, the complexity of the DDPG-based algorithm is expressed as

$$\mathcal{O}\left(N_1^{conv}F_1^{conv}\left(n_1^a \cdot |\mathcal{S}_1| + \sum_{m=1}^{N_L^a-1} n_m^a n_{m+1}^a + n_{N_L^a}^a \cdot |\mathcal{A}_1|\right) + \right.$$
$$\left. N_1^{conv}F_1^{conv}\left(n_1^c \cdot (|\mathcal{S}_1| + |\mathcal{A}_1|) + \sum_{m=1}^{N_L^c-1} n_m^c n_{m+1}^c + n_{N_L^c}^c\right)\right). \tag{41}$$

Next, we analyze the complexity of the DQL-based algorithm. Firstly, the DQL algorithm requires to discretize the action space of the sum-utility maximization problem. We denote the dimension of the state space and the size of the discretized action space of the DQL-based algorithm by $|\mathcal{S}_2|$ and $|\mathcal{A}_2|$, namely the dimensions of input and output of the DQL network, respectively. Therein, $|\mathcal{S}_2| = |\mathcal{S}_1|$, and $|\mathcal{A}_2|$ depends on the resolution of discretization of the action space. Let $N_L^Q$ and $n_m^Q$ denote the number of hidden layers of the DQL network and the number of neurons in the $m$th hidden layer, respectively, and assume that the parameters of the DQL network will converge after $F_2^{conv}$ episodes and $N_2^{conv}$ time slots. Therefore, the complexity of the DQL-based algorithm is expressed as

$$\mathcal{O}\left(N_2^{conv}F_2^{conv}\left(n_1^Q \cdot |\mathcal{S}_2| + \sum_{m=1}^{N_L^Q-1} n_m^Q n_{m+1}^Q + n_{N_L^Q}^Q \cdot |\mathcal{A}_2|\right)\right). \tag{42}$$

Based on the above analysis, we can see that the complexity of the DDPG-based algorithm is higher than that of the DQL-based algorithm. This is because the complexity analysis of the DDPG-based algorithm covers both the actor and critic networks. However, the DQL-based algorithm divides the continuous action space into a discrete action space, and restricts the resource allocation for CUPS-E2E network slicing to the finite discrete quantities, which deteriorates the sum-utility obtained by the DQL-based algorithm as compared with the DDPG-based algorithm.

## V. SIMULATION RESULTS AND DISCUSSIONS

In this section, we present the simulation results to validate our analysis. The simulation experiments are performed using Python 3.7 with TensorFLow. According to [18], [32] and [35], we set the parameters as follows unless special stated: $\alpha^C = 4$, $\alpha^T = 3.2$, $\sigma^2 = -90$dBm, $\kappa = 10^{-26}(W \cdot s^3/\text{cycle}^3)$, $\gamma = 0.9$, $\tau = 0.01$, $\rho = 0.70$, $\zeta = 0.6$ms, $\xi = 0.3$, $\eta = 0.3$. We set $\lambda_C = 0.0001(\text{CBSs}/m^2)$, $\lambda_T = 0.0200(\text{TBSs}/m^2)$, $\lambda_u = 0.5000(\text{users}/m^2)$, $S_{pl} = 10km \times 10km$. Besides, we assume that the maximum transmit powers of one physical CBS and TBS are $P_B^C = 400W$ and $P_B^T = 30W$, respectively, and the maximum computing capability of one physical TBS is $F_B^T = 20$M CPU cycle/s. The capacities of the backhaul link connecting the CN with one physical CBS and TBS are $Rb_B^C = 20$Mbps and $Rb_B^T = 500$kbps, respectively. The downlink bandwidth in the UP is $B_U = 30$MHz. Moreover, according to the unit price settings of the utility in [8], [40], [41] and [42], we set the unit prices of the incomes and costs as follows: $m_C = 3000(/\text{user})$, $m_1 = 0.001(\text{s/bit})$, $m_2 = 0.00005(\text{s/CPU cycle})$, $m_3 = 1(/\text{bit})$, $\omega = 0.0002(\text{s/bit})$, $\delta = 0.005(/\text{Hz})$ and $\beta = 50(/\text{W})$.

To verify the performance of the proposed scheme, we provide two coupled network slicing schemes and two benchmark resource allocation algorithms for comparison. Firstly, we compare the proposed CUPS-E2E network slicing scheme with two coupled schemes: 1) Conventional E2E network slicing scheme where the CP and the UP are tightly coupled [17]: the control signals and data signals are transmitted by

the same physical base station (BS) in the common bandwidth. 2) Software-Defined Hyper-Cellular Networks (SD-HCN)-based E2E network slicing scheme: traditional BSs are classified in to CBSs and TBSs, and the control and data signals are transmitted by the dedicated CBSs and TBSs, respectively in the common bandwidth [43]. Then, to reflect the advantage of the DDPG-based algorithm in solving the sum-utility maximization problem, we compare two typical algorithms: the asynchronous advantage actor-critic (A3C)-based algorithm and the fixed resource allocation algorithm with the DDPG-based algorithm in terms of the throughput, computing capability and sum-utility. Both A3C and DDPG are actor-critic algorithms and can solve problems with continuous action spaces, therefore, we choose A3C algorithm as a comparison algorithm. With the fixed resource allocation algorithm, the communication and computing resources are empirically allocated to each slice and keep constant no matter how the data service requesting states change over time [44]. Besides, the convergence of the DQL-based algorithm is also compared with that of the DDPG-based and A3C-based algorithms.
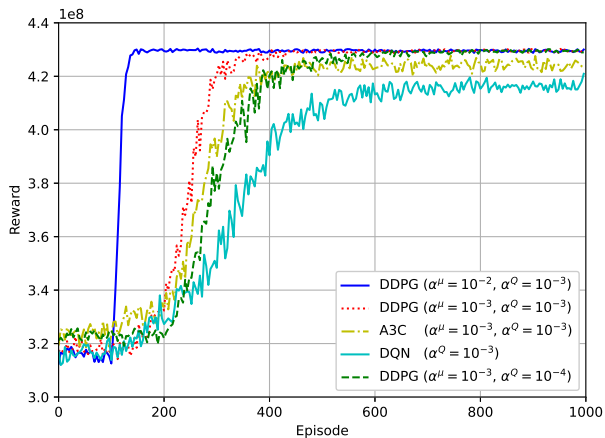
The convergence of the DDPG-based algorithm is shown in Fig. 3 and Fig. 4. Therein, Fig. 3 shows the convergence for the DDPG-based sum-utility maximization algorithm under different values of the actor's learning rate $\alpha^\mu$ and critic's learning rate $\alpha^Q$ as well as the convergence for the DQL-based and A3C-based algorithms. As can be seen from Fig. 3, under the same critic's learning rate $\alpha^Q = 10^{-3}$, the convergence of the algorithm is faster when $\alpha^\mu = 10^{-2}$ than it when $\alpha^\mu = 10^{-3}$. Similarly, under the same actor's learning rate $\alpha^\mu = 10^{-3}$, the algorithm has a faster convergence when the critic's learning rate is larger ($\alpha^Q = 10^{-3}$). Fig. 4 shows the impact of the sizes of replay buffer and mini-batch on the convergence. In Fig. 4, we can see that under the same size of mini-batch, the algorithm has a faster convergence when the size of replay buffer is smaller. Similarly, under the same size of replay buffer, the convergence is faster when the size of mini-batch is 32 than it when the size of mini-batch is 50. Besides, Fig. 3 and Fig. 4 show that the convergence of the DDPG-based algorithm is slightly faster than the DQL-based algorithm and the A3C-based algorithm under the same $\alpha^\mu$ and $\alpha^Q$ or the same sizes of replay buffer and mini-batch.



Fig. 3: The reward under different actor's learning rates and critic's learning rates



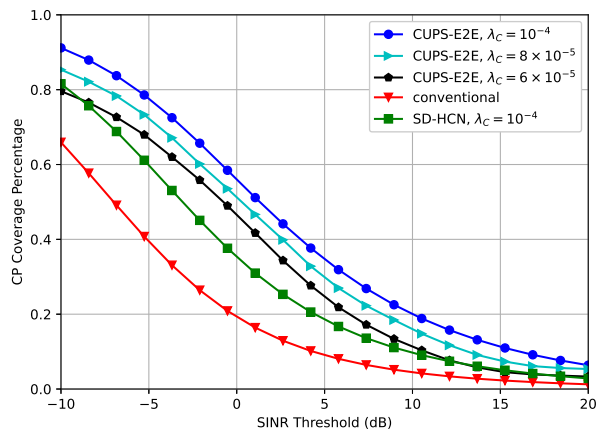Fig. 5: CP coverage percentage versus the SINR threshold



Fig. 4: The reward under different sizes of replay buffer and mini-batch
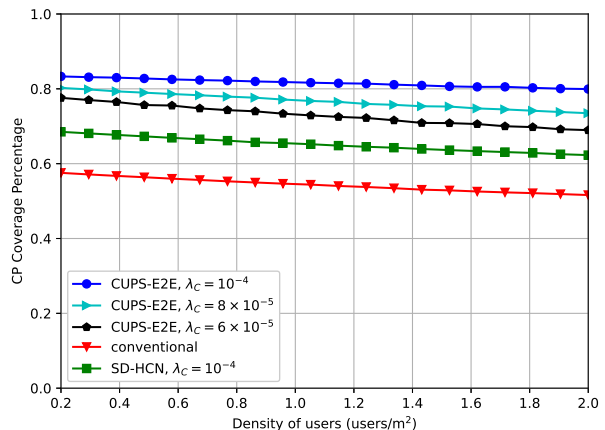


Fig. 6: CP coverage percentage versus the density of users

Fig. 5 and Fig. 6 depict the CP coverage percentage of slice $C$ versus the SINR threshold $x_0$ and the density of users, respectively, for the proposed CUPS-E2E network slicing scheme under various $\lambda_C$ and other network slicing schemes. The simulation results consider the distances among the users and the CBSs when calculating the SINR for each user. Compared with the conventional E2E network slicing scheme, the SD-HCN-based E2E network slicing scheme could provide a better coverage performance. This is because the coverage control signals are transmitted via dedicated CBSs in the SD-HCN-based E2E network slicing scheme, and the dedicated CBSs would provide a higher transmit power. Moreover, both Fig. 5 and Fig. 6 show that the CP coverage percentage of the proposed CUPS-E2E network slicing scheme is higher than that of the SD-HCN-based E2E network slicing and conventional E2E network slicing schemes. For the proposed scheme, coverage control signals and data signals are transmitted in different frequency bands, and the interference between them is reduced significantly, which improves the CP coverage percentage of the network. Besides, Fig. 6 shows that the CP coverage percentage decreases slightly with the increasing density of users because the growing number of users in the given area reduces the vCBS transmit power allocated to each user, but the proposed CUPS-E2E network slicing scheme can still provide a high CP coverage percentage (over 70%) when the density of users is greater than 1.0 users/m$^2$, which guarantees massive user access even when the density of users is high.
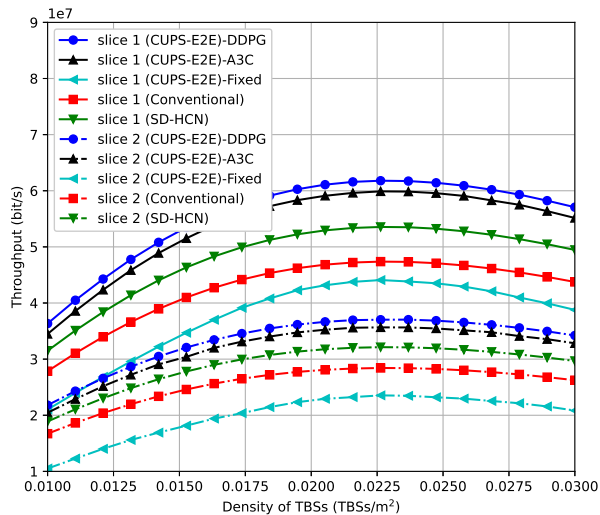


Fig. 7: Throughput versus the density of TBSs

Fig. 7 plots the throughput of two typical eMBB slices versus the density of TBSs for the proposed CUPS-E2E network slicing scheme and other benchmark algorithms or schemes. We observe that when the density of TBSs is below 0.0225 TBSs/m$^2$, the interference from other TBSs is limited, and the throughput increases with the density of TBSs because of the reduction of path loss and the raising of vTBS transmit power allocated to each user. With the increasing density of TBSs (from 0.0225 to 0.0300 TBSs/m$^2$), the interference enhances and the throughput shows a decreasing trend. The throughput of the CUPS-E2E network slicing scheme precedes the SD-HCN-based E2E network slicing and conventional E2E network slicing schemes, which shows the advantage of the proposed scheme. We can also note that the throughput obtained by DDPG-based algorithm is 3.2%, 40.7% higher as compared with the A3C-based algorithm and the fixed resource allocation algorithm, respectively. In addition, it is worth noting that the throughput of slice 1 is also much higher than that of slice 2, which satisfies the throughput performance requirement of slice 1.
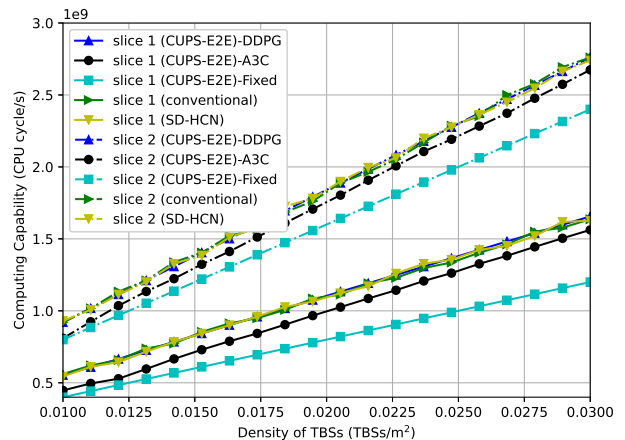


Fig. 8: Computing capability versus the density of TBSs

Fig. 8 shows the computing capability of two typical eMBB slices versus the density of TBSs for all algorithms and schemes. Upon increasing density of TBSs, the number of users associated with one physical TBS decreases, and the TBS is able to provide a higher computing capability for each user. Therefore, the computing capability is also increased almost linearly with the density of TBSs. Compared with the A3C-based algorithm and the fixed resource allocation algorithm, we can obtain better computing performance by the DDPG-based algorithm. The computing capability of the proposed CUPS-E2E network slicing scheme is same as that of the SD-HCN-based E2E network slicing and conventional E2E network slicing schemes because the separation of the CP and the UP will not affect the allocation of computing resource. Moreover, the computing capability of slice 1 is much lower than slice 2, which contents the computing capability requirement of slice 2.

Fig. 9 plots the average delay of slice 3 versus the density of TBSs for all algorithms and schemes. We note that when the density of TBS is below 0.0225 TBSs/m$^2$, the average delay decreases rapidly with the increasing density of TBSs because of the promotion of the backhaul link capacity, computing capability and downlink rate of each user and the reductions of the queue backlogs. The backhaul link capacity and computing capability allocated to each user will increase continuously with the denser deployment of the TBSs (from 0.0225 to 0.0300 TBSs/m$^2$), while the downlink rate of each user will decrease because of the enhancement of the interference. Therefore, the backlogs of the backhaul link queues and computing queues will continue to reduce and the backlogs of the wireless access link queues will increase. Accordingly,
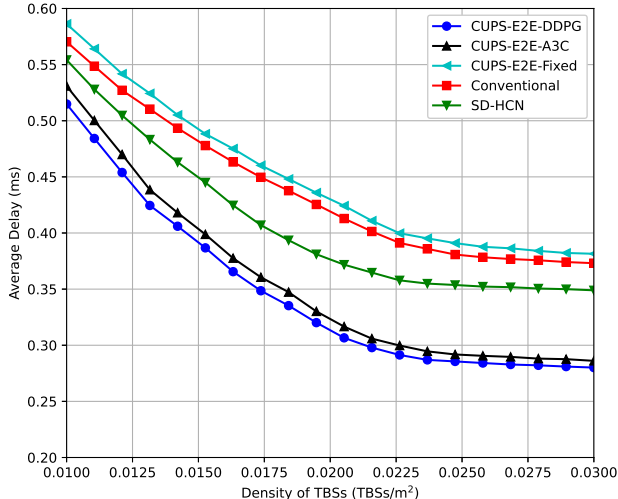
Fig. 9: Average delay versus the density of TBSs

the average delay will decrease slowly and tend to be stable when the density of TBSs exceeds 0.0225 TBSs/m$^2$. Fig. 9 also shows that we can obtain better delay performance by the DDPG-based algorithm as compared with the A3C-based algorithm and the fixed resource allocation algorithm. Besides, the average delay of the proposed CUPS-E2E network slicing scheme is lower than that of the SD-HCN-based E2E network slicing and conventional E2E network slicing schemes because the improvement of the downlink rate of each user in the proposed scheme makes the queue backlogs lower than the SD-HCN-based E2E network slicing and conventional E2E network slicing schemes.

Fig. 10(a) depicts the utility versus the density of CBSs. When the density of CBSs is below $6 \times 10^{-5}$ CBSs/m$^2$, the interference from other CBSs is limited and the number of users covered by the CP increases with the density of CBSs. Therefore, the utility of slice $C$ is also incremental upon increasing density of CBSs. When the CBSs are deployed densely (from $6 \times 10^{-5}$ to $10^{-4}$ CBSs/m$^2$), the CP coverage percentage saturates and the utility of slice $C$ keeps steady. Moreover, the impact of the density of CBSs on the utilities of the data service slices is also reflected in Fig. 10(a): when the CBSs are deployed sparsely (below $5 \times 10^{-6}$ CBSs/m$^2$), there are only a few number of users covered by the CP, and the communication and computing resources allocated to each user will not change almost with the number of covered users. Hence, the total throughput and the total computing capability increase with the number of covered users, and the utilities of slice 1 and slice 2 increase near-linearly upon increasing density of CBSs. The number of the covered users is further increasing with the denser deployment of CBSs, and the communication and computing resources allocated to each user decrease relatively because of the inherent resource limitation. Consequently, the total throughput and the total computing capability keep almost stable with the increasing number of covered users and therefore the utilities of slice 1 and slice 2 keep steady nearly with the increment of the density of CBSs (over $5 \times 10^{-6}$ CBSs/m$^2$) as well. However,

the tendency of the utility of slice 3 with the density of CBSs is different from that of the two eMBB slices. The utility of slice 3 decreases first with the increasing density of CBSs. This is because the total queue length grows with the incremental number of the covered users when the density of CBSs is below $6 \times 10^{-5}$ CBSs/m$^2$, and especially the reduction of the computing capability and downlink rate of the individual user further increases the total queue length when the density of CBSs is between $5 \times 10^{-6}$ CBSs/m$^2$ and $6 \times 10^{-5}$ CBSs/m$^2$. And with the further increment of the density of CBSs (over $6 \times 10^{-5}$ CBSs/m$^2$), the utility of 3 tends to be stable due to the saturation of the number of covered users. Finally, the sum-utility of the entire network increases rapidly when the density of CBSs is below $5 \times 10^{-6}$ CBSs/m$^2$. And the sum-utility keeps almost steady when the density of CBSs exceeds $5 \times 10^{-6}$ CBSs/m$^2$ because of the mutual cancellation of the utilities of slice $C$ and slice 3.

Fig. 10(b) illustrates the utility versus the density of TBSs. Upon increasing density of TBSs, the utility of slice 1 increases gradually first and then decreases when the TBSs are deployed densely, which is in accord with the throughput performance of slice 1 as depicted in Fig. 7. While the utility of slice 2 increases linearly with the increasing density of TBSs, which conforms to the computing capability performance of slice 2 as illustrated in Fig. 8. And the utility of slice 3 increases evidently first because of the reduction of the slice 3's delay and then increases slowly due to the delay of tends to be stable with the increasing density of TBSs, which conforms to the delay performance of slice 3 as illustrated in Fig. 9. Additionally, the utility of slice $C$ keeps steady with the increasing density of TBSs. Fig. 10(a) and Fig. 10(b) together show that the sum-utilities of the SD-HCN-based E2E network slicing and conventional E2E network slicing schemes are worse than that of the proposed CUPS-E2E network slicing scheme. Therefore, the proposed scheme guarantees that the network operator obtains better sum-utility. Conventional E2E network slicing scheme is not compared in Fig. 10(a) because the control signals are not transmitted by the dedicated CBSs in this scheme. We can also note that the network operator could get a higher sum-utility by the DDPG-based algorithm obviously as compared with the fixed resource allocation algorithm, and the sum-utility obtained by the DDPG-based algorithm is slightly better than that obtained by the A3C-based algorithm. The results demonstrate that the DDPG-based algorithm performs better than others in solving the sum-utility maximization problem.

## VI. Conclusion

In this paper, we have studied the sum-utility maximization for CUPS-E2E network slicing, where the CP coverage and UP services are decoupled for four E2E network slices: one for CP coverage, one for UP high-throughput services, one for UP computation-intensive services, and the other for UP delay-sensitive services. The utilities of the four E2E network slices are defined considering their heterogeneous performance metrics. To maximize the sum-utility of the entire network, we have employed a DDPG-based sum-utility maximization
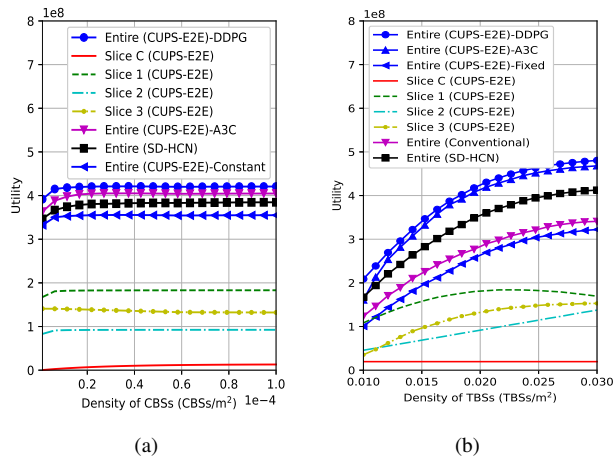
Fig. 10: Utility versus the density of CBSs and TBSs

algorithm that jointly optimizes the vTBS activation and the allocation of CP subcarrier, vCBS transmit power, backhaul link capacity, UP subcarrier, vTBS transmit power and vTBS computing capability. Simulation results have shown that the proposed CUPS-E2E network slicing scheme in conjunction with a DDPG-based sum-utility maximization algorithm can support the wide-coverage and massive access requirements, as well as the high-throughput, computation-intensive and delay-sensitive services simultaneously, and outperforms the SD-HCN-based E2E network slicing and conventional E2E network slicing schemes in terms of the sum-utility, coverage percentage, throughput and delay.

REFERENCES

[1] I. Afolabi, M. Bagaa, W. Boumezer and T. Taleb, Toward a Real Deployment of Network Services Orchestration and Configuration Convergence Framework for 5G Network Slices, *IEEE Network*, vol. 35, no. 1, pp. 242-250, Jan. 2021.

[2] A. Mohamed, O. Onireti, M. A. Imran, A. Imran and R. Tafazolli, "Control-Data Separation Architecture for Cellular Radio Networks: A Survey and Outlook," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 446-465, Firstquarter 2016.

[3] A. Mohamed, O. Onireti, M. A. Imran, A. Imran and R. Tafazolli, "Predictive and Core-Network Efficient RRC Signalling for Active State Handover in RANs With Control/Data Separation," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1423-1436, Mar. 2017.

[4] E. Ternon, P. Agyapong, L. Hu and A. Dekorsy, "Database-aided energy savings in next generation dual connectivity heterogeneous networks," 2014 *IEEE Wireless Communications and Networking Conference (WCNC)*, Istanbul, 2014, pp. 2811-2816.

[5] D. Sattar and A. Matrawy, "Optimal Slice Allocation in 5G Core Networks," *IEEE Networking Letters*, vol. 1, no. 2, pp. 48-51, Jun. 2019.

[6] 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.

[7] Y. Sun, M. Peng, S. Mao and S. Yan, "Hierarchical Radio Resource Allocation for Network Slicing in Fog Radio Access Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3866-3881, Apr. 2019.

[8] G. Zhou, L. Zhao, K. Liang, G. Zheng and L. Hanzo, "Utility Analysis of Radio Access Network Slicing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 1163-1167, Jan. 2020.

[9] H. Chien, Y. Lin, C. Lai and C. Wang, "End-to-End Slicing With Optimized Communication and Computing Resource Allocation in Multi-Tenant 5G Systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 2079-2091, Feb. 2020.

[10] H. Ibrahim, H. ElSawy, U. T. Nguyen and M. Alouini, "Mobility-Aware Modeling and Analysis of Dense Cellular Networks With *C*-Plane/ *U*-Plane Split Architecture," *IEEE Transactions on Communications*, vol. 64, no. 11, pp. 4879-4894, Nov. 2016.

[11] Y. Sun, Y. Chang, M. Hu and T. Zeng, "A Universal Predictive Mobility Management Scheme for Urban Ultra-Dense Networks With Control/Data Plane Separation," *IEEE Access*, vol. 5, pp. 6015-6026, 2017.

[12] B. Yang, X. Yang, X. Ge and Q. Li, "Coverage and Handover Analysis of Ultra-Dense Millimeter-Wave Networks With Control and User Plane Separation Architecture," *IEEE Access*, vol. 6, pp. 54739-54750, 2018.

[13] K. Liang, G. Liu, L. Zhao, X. Chu, S. Wang and L. Hanzo, "Performance Analysis of Cellular Radio Access Networks Relying on Control- and User-Plane Separation," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 7, pp. 7241-7245, Jul. 2019.

[14] H. Zhao, L. Zhao, K. Liang and C. Pan, "Radio access network slicing based on C/U plane separation," *China Communications*, vol. 14, no. 12, pp. 134-141, Dec. 2017.

[15] H. Tu, L. Zhao, C. Feng and C. Guo, "NOMA-Enabled Cellular Radio Access Networks Relying on Control- and User-Plane Separation," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 11116-11120, Oct. 2021.

[16] C. Wang, Y. He, F. R. Yu, Q. Chen and L. Tang, "Integration of Networking, Caching, and Computing in Wireless Systems: A Survey, Some Research Issues, and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 7-38, Firstquarter 2018.

[17] Vincent Franois-Lavet; Peter Henderson; Riashat Islam; Marc G. Bellemare; Joelle Pineau, *An Introduction to Deep Reinforcement Learning*, now, 2018.

[18] H. Ge, Y. Song, C. Wu, J. Ren and G. Tan, "Cooperative Deep Q-Learning With Q-Value Transfer for Multi-Intersection Signal Control," *IEEE Access*, vol. 7, pp. 40797-40809, 2019.

[19] K. Li, Y. Emami, W. Ni, E. Tovar and Z. Han, "Onboard Deep Deterministic Policy Gradients for Online Flight Resource Allocation of UAVs," *IEEE Networking Letters*, vol. 2, no. 3, pp. 106-110, Sep. 2020.

[20] H. Ye, G. Y. Li and B. F. Juang, "Deep Reinforcement Learning Based Resource Allocation for V2V Communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3163-3173, Apr. 2019.

[21] G. M. S. Rahman, M. Peng, S. Yan and T. Dang, "Learning Based Joint Cache and Power Allocation in Fog Radio Access Networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4401-4411, Apr. 2020.

[22] C. Qiu, H. Yao, F. R. Yu, F. Xu and C. Zhao, "Deep Q-Learning Aided Networking, Caching, and Computing Resources Allocation in Software-Defined Satellite-Terrestrial Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5871-5883, Jun. 2019.

[23] L. Li, H. Xu, J. Ma, A. Zhou and J. Liu, "Joint EH Time and Transmit Power Optimization Based on DDPG for EH Communications," *IEEE Communications Letters*, vol. 24, no. 9, pp. 2043-2046, Sep. 2020.

[24] N. Jiang, Y. Deng, A. Nallanathan and J. Yuan, "A Decoupled Learning Strategy for Massive Access Optimization in Cellular IoT Networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 3, pp. 668-685, Mar. 2021.

[25] T. Dang and M. Peng, "Joint Radio Communication, Caching, and Computing Design for Mobile Virtual Reality Delivery in Fog Radio Access Networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 7, pp. 1594-1607, Jul. 2019.

[26] A. Al-Shuwaili and O. Simeone, "Energy-Efficient Resource Allocation for Mobile Edge Computing-Based Augmented Reality Applications," *IEEE Wireless Communications Letters*, vol. 6, no. 3, pp. 398-401, Jun. 2017.

[27] K. S. Kim et al., "Ultrareliable and Low-Latency Communication Techniques for Tactile Internet Services," *Proceedings of the IEEE*, vol. 107, no. 2, pp. 376-393, Feb. 2019.

[28] S. Dhok, P. Raut, P. K. Sharma, K. Singh and C. -P. Li, "Non-Linear Energy Harvesting in RIS-Assisted URLLC Networks for Industry Automation," *IEEE Transactions on Communications*, vol. 69, no. 11, pp. 7761-7774, Nov. 2021.

[29] M. Bagaa, T. Taleb, A. Laghrissi, A. Ksentini and H. Flinck, "Coalitional Game for the Creation of Efficient Virtual Core Network Slices in 5G Mobile Systems" *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 469-484, Mar. 2018.

[30] Y. Liu, W. Huangfu, H. Zhang and K. Long, "An Efficient Stochastic Gradient Descent Algorithm to Maximize the Coverage of Cellular

Networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 7, pp. 3424-3436, Jul. 2019.

[31] Y. Wu, X. Fang and X. Wang, "Mobility Management through Scalable C/U-Plane Decoupling in IoV Networks," *IEEE Communications Magazine*, vol. 57, no. 2, pp. 122-129, Feb. 2019.

[32] R. Li, X. Liu, K. Luo, T. Jiang and S. Jin, "Decoupled Access in HetNets With Backhaul Constrained Small Base Stations," *IEEE Access*, vol. 6, pp. 27028-27038, 2018.

[33] Y. Mao, J. Zhang, S. H. Song and K. B. Letaief, "Stochastic Joint Radio and Computational Resource Management for Multi-User Mobile-Edge Computing Systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5994-6009, Sep. 2017.

[34] D. P. Bertsekas and R. G. Gallager, *Data Networks* (*2nd edition*). Prentice Hall, 1992.

[35] G. Zhou, L. Zhao, Y. Wang, G. Zheng and L. Hanzo, "Energy Efficiency and Delay Optimization for Edge Caching Aided Video Streaming," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 14116-14121, Nov. 2020.

[36] W. Wang, V. K. N. Lau and M. Peng, "Delay-Aware Uplink Fronthaul Allocation in Cloud Radio Access Networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 7, pp. 4275-4287, Jul. 2017.

[37] M. Xiao, J. Wu, C. Liu and L. Huang, "TOUR: Time-sensitive Opportunistic Utility-based Routing in delay tolerant networks," 2013 Proceedings IEEE INFOCOM, Turin, 2013, pp. 2085-2091.

[38] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, and Y. Tassa, "Continuous control with deep reinforcement learning," *Computer Science*, vol. 8, no. 6, pp. 114, Jul. 2019.

[39] Y. Liu, J. Yan and X. Zhao, "Deep Reinforcement Learning based Latency Minimization for Mobile Edge Computing with Virtualization in Maritime UAV Communication Network," *IEEE Transactions on Vehicular Technology*, doi: 10.1109/TVT.2022.3141799.

[40] Y. Qian et al., "Cache-Enabled MIMO Power Line Communications With Precoding Design in Smart Grid," *IEEE Transactions on Green Communications and Networking*, vol. 4, no. 1, pp. 315-325, Mar. 2020.

[41] Q. Zhang, W. Ma, Z. Feng and Z. Han, "Backhaul-Capacity-Aware Interference Mitigation Framework in 6G Cellular Internet of Things," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 10071-10084, Jun. 2021.

[42] L. Tang, X. Yang, X. Wu, T. Cui and Q. Chen, "Queue Stability-Based Virtual Resource Allocation for Virtualized Wireless Networks With Self-Backhauls," *IEEE Access*, vol. 6, pp. 13604-13616, 2018.

[43] Q. Li, W. Shi, X. Ge and Z. Niu, "Cooperative Edge Caching in Software-Defined Hyper-Cellular Networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2596-2605, Nov. 2017.

[44] J. Gang and V. Friderikos, "Inter-Tenant Resource Sharing and Power Allocation in 5G Virtual Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7931-7943, Aug. 2019.

**Liqiang Zhao** (Member, IEEE) received the B.Sc. degree in electrical engineering from Shanghai Jiao Tong University, China, in 1992, and the M.Sc. degree in communications and information systems and the Ph.D. degree in information and communications engineering from Xidian University, China, in 2000 and 2003, respectively.

From 1992 to 2005, he was a Research Engineer with the 20th Research Institute, Chinese Electronics Technology Group Corporation (CETC), China. From 2005 to 2007, he was an Associate Professor with the State Key Laboratory of Integrated Service Networks (ISN), Xidian University. He was appointed as a Marie Curie Research Fellow at the Centre for Wireless Network Design (CWiND), University of Bedfordshire, in June 2007, to conduct research in the GAWIND project funded under EU FP6 HRM Program. His activities focused on the area of automatic wireless broadband access network planning and optimization. Since June 2008, he has returned Xidian University, first as an Associate Professor and later as a Professor. He has hosted/participated many national research projects, such as the National Natural Science Foundation, the 863 Program, and the National Science Technology Major Projects, and several international research projects, including the EU FP6, FP7 plans for international cooperation exchange projects, and some research projects from companies, such as Huawei. He has more than 100 published in authorized academic periodicals both in and abroad and in international science conferences, wherein 30 of which are retrieved in SCI, and more than 70 of them are EI indexed, and six national invention patents. His current research interests include mobile communication systems, spread spectrum communications, WiMAX, WLAN, wireless sensor networks, broadband wireless communications, and space communications. In 2008, he was awarded by the Program for New Century Excellent Talents in University, Ministry of Education, China.

**Yunfeng Wang** received the B.Sc. degree in Telecommunications Engineering from Inner Mongolia University, Huhhot, China, in 2018. He is currently pursuing the Ph.D. degree with the School of Telecommunication Engineering, Xidian University, Xi'an, China. His research interests include end-to-end network slicing, deep reinforcement learning, control- and user- plane separation, and collaboration of cloud-network-edge.

**Xiaoli Chu** (Senior Member, IEEE) received the B.Eng. degree in electronic and information engineering from Xi'an Jiao Tong University in 2001 and the Ph.D. degree in electrical and electronic engineering from the Hong Kong University of Science and Technology in 2005. She is a Professor with the Department of Electronic and Electrical Engineering, University of Sheffield, U.K. From 2005 to 2012, she was with the Centre for Telecommunications Research, Kings College London. She has coauthored over 150 peer-reviewed journal and conference papers. She coauthored/co-edited the books Fog-Enabled Intelligent IoT Systems (Springer, 2020), Ultra Dense Networks for 5G and Beyond (Wiley, 2019), Heterogeneous Cellular Networks: Theory, Simulation and Deployment (Cambridge University Press, 2013), and 4G Femtocells: Resource Allocation and Interference Management (Springer, 2013). She is co-recipient of the IEEE Communications Society 2017 Young Author Best Paper Award. She received the IEEE Communications Letters Exemplary Editor Award in 2018. She was Co-Chair of Wireless Communications Symposium for IEEE ICC 2015, a Workshop Co-Chair for IEEE GreenCom 2013, and has co-organized eight workshops at IEEE ICC, GLOBECOM, WCNC, and PIMRC. She is Senior Editor for the IEEE Wireless Communications Letters and an Editor for the IEEE Communications Letters.

**Shenghui Song** (Senior Member, IEEE) is now an Assistant Professor jointly appointed by the Division of Integrative Systems and Design (ISD) and the Department of Electronic and Computer Engineering (ECE) at the Hong Kong University of Science and Technology (HKUST). His research is primarily in the areas of Wireless Communications and Machine Learning with current focus on Distributed Intelligence (Federated Learning), Machine Learning for Communications (Model and Data-driven Approaches), and Integrated Sensing and Communication. He was named the Exemplary Reviewer for IEEE COMMUNICATIONS LETTER and served as the Tutorial Program Co-Chairs of the 2022 IEEE International Mediterranean Conference on Communications and Networking. Dr. Song is also interested in the research on Engineering Education and is now serving as an Associate Editor for the IEEE TRANSACTIONS ON EDUCATION. He has won several teaching awards at HKUST, including the Michael G. Gale Medal for Distinguished Teaching in 2018, the Best Ten Lecturers in 2013, 2015, and 2017, the School of Engineering Distinguished Teaching Award in 2012, the Teachers I Like Award in 2013, 2015, 2016, and 2017, and the MSc(Telecom) Teaching Excellent Appreciation Award for 2020-21. Dr. Song was one of the honorees of the Third Faculty Recognition at HKUST in 2021.

**Arumugam Nallanathan** (Fellow, IEEE) was an Assistant Professor with the Department of Electrical and Computer Engineering, National University of Singapore, from August 2000 to December 2007. He was with the Department of Informatics, Kings College London from December 2007 to August 2017, where he was Professor of wireless communications from April 2013 to August 2017 and a Visiting Professor since September 2017. He has been a Professor of wireless communications and the Head of Communication Systems Research (CSR) Group, School of Electronic Engineering and Computer Science, Queen Mary University of London, since September 2017. He has published nearly 500 technical papers in scientific journals and international conferences. His research interests include artificial intelligence for wireless systems, beyond 5G wireless networks, the Internet of Things (IoT) and molecular communications. Prof. Nallanathan has served as the Chair for the Signal Processing and Communication Electronics Technical Committee of IEEE Communications Society and the Technical Program Chair and a member of Technical Program Committees in numerous IEEE conferences. He received the IEEE Communications Society SPCE Outstanding Service Award 2012 and the IEEE Communications Society RCC Outstanding Service Award 2014. He was a co-recipient of the Best Paper Awards presented at the IEEE International Conference on Communications 2016 (ICC2016), IEEE Global Communications Conference 2017 (GLOBECOM2017), and IEEE Vehicular Technology Conference 2018 (VTC2018). He is an IEEE Distinguished Lecturer. He has been selected as a Web of Science Highly Cited Researcher in 2016. He was an Editor of the IEEE Transactions on Wireless Communications from 2006 to 2011, the IEEE Transactions on Vehicular Technology from 2006 to 2017, and the IEEE Signal Processing Letters. He is an Editor of the IEEE Transactions on Communications and a Senior Editor of the IEEE Wireless Communications Letters.

**Kai Liang** received the B.S. degree in communication engineering from Xi'an University of Architecture and Technology, Xi'an, China, in 2007, and the Ph.D. degree in communication and information system from Xidian University, Xi'an, China, in 2016.

He is a Lecturer with the State Key Laboratory of Integrated Service Networks, Xidian University. From September 2014 to September 2015, he received the funding from the program of China Scholarships Council to visit the Network Convergence Laboratory, University of Essex, Colchester, U.K. His current research interests include network slicing, multiaccess edge computing, wireless powered communications, broadband wireless access, wireless network design, coordinated multipoint-to-multiuser communication systems, and fog computing. He has co-organized two workshops at IEEE GLOBECOM 2019 and 2020.

**Yansha Deng** (Member, IEEE) received the Ph.D. degree in electrical engineering from the Queen Mary University of London, U.K., in 2015. From 2015 to 2017, she was a Post-Doctoral Research Fellow with King's College London, U.K, where she is currently a Senior Lecturer (an Associate Professor) with the Department of Engineering. Her research interests include molecular communication and machine learning for 5G/6G wireless networks. She was a recipient of the Best Paper Awards from ICC 2016 and GLOBECOM 2017 as the first author and IEEE Communications Society Best Young Researcher Award for the Europe, Middle East, and Africa Region 2021. She also received the Exemplary Reviewers of the IEEE TRANSACTIONS ON COMMUNICATIONS in 2016 and 2017 and IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS in 2018. She has served as a TPC Member for many IEEE conferences, such as IEEE GLOBECOM and ICC. She is currently an Associate Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS and IEEE TRANSACTIONS ON MOLECULAR, BIOLOGICAL AND MULTI-SCALE COMMUNICATIONS, a Senior Editor of the IEEE COMMUNICATIONS LETTERS, and the Vertical Area Editor of IEEE Internet of Things Magazine.