IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 22, 2018, accepted March 29, 2018, date of publication April 2, 2018, date of current version April 25, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2822398

A Service-Oriented Deployment Policy of End-to-End Network Slicing Based on Complex Network Theory

WANQING GUAN, XIANGMING WEN, LUHAN WANG, ZHAOMING LU, AND YIDI SHEN

Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing 100876, China
Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China
Beijing Advanced Innovation Center for Future Internet Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Wanqing Guan (guanwanqing95@bupt.edu.cn)

This work was supported in part by the National Science and Technology Major Project under Grant 2018ZX03001016, in part by the China Ministry of Education-CMCC Research under Grant MCM20160104, in part by the 111 Project B17007, in part by the Beijing Municipal Science and Technology Commission Research under Project Z171100005217001, and in part by the Fundamental Research Funds for the Central Universities.

ABSTRACT For fifth-generation wireless communication systems, network slicing has emerged as a key concept to meet the diverse requirements of various use cases. By slicing an infrastructure network into multiple dedicated logical networks, wireless networks can support a wide range of services. However, how to fast deploy the end-to-end slices is the main issue in a multi-domain wireless network infrastructure. In this paper, a mathematical model is used to construct network slice requests and map them to the infrastructure network. The mapping process consists of two steps: the placement of virtual network functions and the selection of link paths chaining them. To efficiently utilize the limited physical resources, we pay attention to the service-oriented deployment by offering different deployment policies for three typical slices: eMBB slices, mMTC slices, and uRLLC slices. Furthermore, we adopt complex network theory to obtain the topological information of slices and infrastructure network. With the topological information, we define a node importance metric to rank the nodes in node mapping. To evaluate the performance of deployment policy we proposed, extensive simulations have been conducted. The results have shown that our algorithm performed better in terms of resource efficiency and acceptance ratio. In addition, the average execute time of our algorithm is in a linear growth with the increase of infrastructure network size.

INDEX TERMS 5G, network slicing, complex network theory, service-oriented deployment, end-to-end slices.

I. INTRODUCTION

With the explosive growth of mobile data traffic, the massive terminal connection and the rise of various kinds of new applications, future wireless network needs to be agile, programmable and open. Moreover, fifth-generation (5G) networks are nowadays expected to satisfy different requirements of numerous new services and support vertical markets such as automotive, energy, food and agriculture, healthcare, etc [1]. A wide range of verticals with diverse requirements spur 5G networks to be flexible, scalable, manageable, customized, and allow multi-tenancy and multi-service support [2]. In order to realize the above vision, network slicing (NS) has been proposed as a concept for slicing a common underlying physical network into multiple end-to-end (E2E) logical networks which are mutually isolated, managed independently and created on demand [3]. In NS based 5G system, resources of multi-domain infrastructure network can be efficiently allocated to multiple network slices according to the requirements of use cases [4]. As shown in the Fig. 1, NS aims to logically separate the set of virtual network functions (VNFs) within the physical infrastructure to build dedicated and customized logical networks. The International Telecommunication Union (ITU) [5] has identified three broad use case families: enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), and ultra-reliable and low-latency communication (uRLLC).

Although the concept of NS is still nascent, the potential technologies for achieving it, software-defined networking (SDN) and network functions virtualization (NFV), have

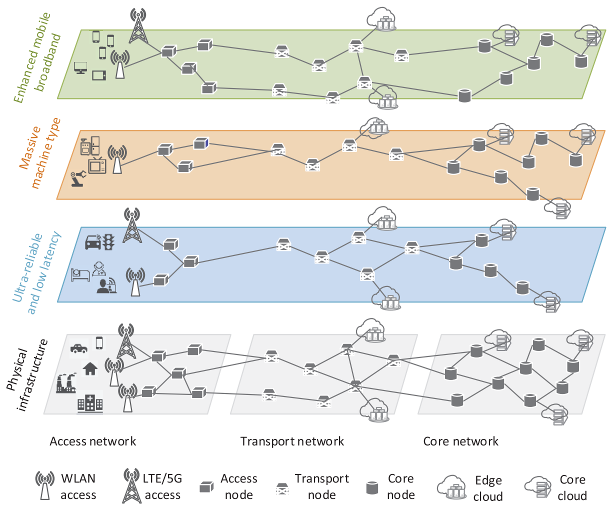


FIGURE 1. **5G network slicing architecture.**

many mature researches with concrete solutions and readily available platforms. SDN architecture is an appropriate technology for the configuration and control of the forwarding planes of the underlying resources, while NFV can manage the life cycle of network slices and orchestrate VNFs efficiently. Actually, Ordonez-Lucena *et al.* [6] have presented a deployment example of NS by integrating SDN and NFV. [7] first propose an E2E 5G system framework and illustrate the techniques in the radio access network and the core network. However, as far as we know, few researches have provided efficient and practical deployment policies for E2E slices in wireless networks, and the technology development on NS is mostly focused on slicing the core network.

Reference [8] defines that network slices are chains of VNFs and logical/physical resources meeting the service requirements. Therefore, the realization of NS in essence is the deployment and orchestration of VNFs, and there are plentiful related researches on VNF placement. References [9], [10] propose the algorithms of VNFs placement, which only for a single type of VNF chain request without considering diverse service requirements. Besides, the mapping from slices to the infrastructure networks is a typical virtual network embedding (VNE) problem [11]. As these two problems have been studied maturely in recent years, deployment of VNFs brings several advantages in terms of the resource utilization and cost saving. Currently, little works in existing literature have been done on the deployment of E2E network slices although it is necessary for the realization of NS. On one hand, E2E slices need to be instantiated rapidly and should support cross-domain deployment. On the other hand, the resources of infrastructure network need to be allocated dynamically in order to support a diverse set of use cases. Owing that different use case families have different demands on the multi-domain resources, NS deployment should be service-oriented. The above-mentioned two respects raise extreme challenges for NS deployment in wireless communication systems.

In order to address these challenges, we propose a service-oriented deployment policy of E2E network slicing based on complex network (CN) theory. CN theory aims at analyzing topological characteristics and predict dynamical behaviors of networked system [12]. Empirical studies have demonstrated that many real-life communication networks exhibit small-world and scale-free topological properties [13], [14]. Moreover, [15] demonstrates that wireless networks are scale-free with a proper algorithm. Using CN theory analyze the impact of topology on E2E NS deployment is reasonable and reliable. The contributions of this work can be concluded as following:

(1) A mathematical model is created to describe E2E NS deployment. The mathematical model consists of infrastructure network model, network slice request (NSR) model and slice deployment model. These models are through three domains: access network, transport network and core network.

(2) For meeting different service requirements, three dedicated deployment policies are provided for three use case families respectively. By analyzing different features of three use cases, three service-oriented deployment algorithms and three optimization objectives are used for eMBB slice, mMTC slice and uRLLC slice accordingly.

(3) In order to improve the utilization of underlying resource and increase revenues of service providers, the topological properties of infrastructure are analyzed based on CN theory. The topological properties network are combined with the local resources of nodes in the deployment of NS.

(4) Extensive simulations are carried out to evaluate the performance of our service-oriented deployment policy. The results show that our deployent policy performs better in terms of resource efficiency and acceptance ratio, and requires less time to execute.

The remainder of this paper is organized as follows. In Section II we discuss the related work. Infrastructure network Model, NSR model and slice deployment model are presented in Section III. Section IV introduces the details of three deployment algorithms based on CN theory. In Section V, we present the simulation results and analyses. Finally, Section VI draws the conclusions of the paper.

II. RELATED WORK

In this section, we briefly introduce the structural characteristics in CN theory which are useful enablers for the purpose of obtaining the topological information. We also give a short summary of works in E2E network slicing for 5G networks. In addition, we review some existing works on VNF placement and VNE problem, showing their contribution to deployment of NS.

A. CN THEORY

According to the analysis of many real-world networks, several topological properties have been proposed to capture structural characteristics of various networks in the past decades. For instance,

1) DEGREE

The degree of a node measures the number of edges that connect to it, which reflects the level of influence. The node degree can be formulated as

$$d_i = \sum_{j \in N} \delta_{ij}. \tag{1}$$

The parameter δ_{ij} takes the value 1 if node i and node j are directly connected, otherwise it takes the value 0.

2) BETWEENNESS CENTRALITY

The betweenness centrality quantifies how much a node is found between the path linking other pair of nodes. The betweenness centrality is defined as the fraction of shortest paths between any pair of nodes that travel through the node, which can be denoted by

$$b_i = \sum_{s \neq i \neq t} \frac{\sigma_{st}(i)}{\sigma_{st}}. \tag{2}$$

In this equation, σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(i)$ is the number of those paths that pass through node i.

B. E2E NETWORK SLICING

The concept of network slicing has captured many attentions. Many researches focus on radio resource virtualization [16], [17], management and orchestration of network functions [18], [19], mapping of service and service description [16]. Since NS allows operators to customize networks according to various service demands, both industry and academia introduce many realization models of NS. Although NS has been implemented in fixed network, E2E NS of the mobile network still lags behind. There is still lacking efficient deployment and management of E2E slices. Zhou *et al.* [20] illustrate NS as a service (NSaaS) and introduce the business model of NSaaS. Reference [21] introduces PERMIT slice orchestration system which is first to consider E2E slicing.

C. VNF PLACEMENT

Recently, with the rise of NFV concept [22], [23] and open-source platform, such as OPNFV [24], OpenMANO [25] and OpenBATON [26], there are significant efforts to VNF management and orchestration [9]. In order to dynamically allocate virtualization resources to VNFs, there are also many algorithms for the deployment of the service function chain (SFC). These algorithms [27], [28] provide solutions to optimize VNF placement problem. Clayman *et al.* [29] describe an architecture based on a distributed orchestrator which enables automated placement of virtual elements. Similar with the dynamic VNF placement algorithm, Ghaznavi *et al.* [30] introduce Elastic Virtual Network Function Placement (EVNFP) problem and present a model for minimizing operational costs in providing VNF services. Targeting to achieve optimal resource utilization of infrastructure resources, these works do not consider the characteristics of infrastructure network, and ignore the investigation of the placement algorithm matching the resource demands of providers.

D. VIRTUAL NETWORK EMBEDDING

The topic of VNE problem has received significant attention of many researchers, and the amount of literature on VNE topic is considerable. Fischer *et al.* [31] provide a comprehensive survey on VNE algorithms. Dealing with the virtual resources allocation both in nodes and links, VNE can be divided in two sub-problems: Virtual Node Mapping (VNoM) and Virtual Link Mapping (VLiM). Solving VNoM and VLiM in an isolated way or a coordinated way divides VNE methods into uncoordinated VNE method and coordinated VNE method. The example of uncoordinated VNE method was proposed in [32] while coordination of VNE was first achieved in [33]. In addition, M. Chowdhury et al. introduce ViNEYard [34] for single domain and PolyViNE [35] for multi-domain VNE. Considering that network slices are the chains of VNFs, the large number of VNE methods provide some valuable algorithms that can be used in deployment of network slices.

III. MATHEMATICAL MODEL OF NETWORK SLICING

In this section, we describe the mathematical model of network slicing including infrastructure network model, network slice request (NSR) model and slice deployment model.

A. INFRASTRUCTURE NETWORK MODEL

The deployment of NS requires topological information of infrastructure network including the structural characteristics of physical nodes (e.g. base stations (BS), optical switches (OS), core nodes (CN)). The infrastructure network can be abstracted as undirected weighted graph, which can be denoted as $G_I = (N_I, E_I, C_I, B_I)$. Similar to some previous literatures, we only take into consideration the capacity of nodes and bandwidth of links. N_I stands for the set of physical nodes, which can be partitioned into the set of base stations N_I^{BS}, the set of optical switches N_I^{OS} and the set of core nodes N_I^{CN}, $N_I = N_I^{BS} \cup N_I^{OS} \cup N_I^{CN}$. E_I stands for the set of physical links including the wireless wave links E_I^{wl} and wired optical links E_I^{ol}, $E_I = E_I^{wl} \cup E_I^{ol}$.

$$E_I^{wl} = \bigcup_{n_I^{BS} \in N_I^{BS}} E_I^{wl}\left(n_I^{BS}\right) \tag{3}$$

where $E_I^{wl}\left(n_I^{BS}\right)$ is the subset of wireless link e_I^{wl} responsible of connecting n_I^{BS} and other nodes. C_I stands for the capacity of physical nodes, which includes BS wireless channel capacity C_I^{BS} and computing resource of core cloud C_I^{CN}, $C_I = C_I^{BS} \cup C_I^{CN}$. B_I stands for the bandwidth set of physical links, including the available bandwidth set of wireless wave links B_I^{wl} and wired optical links B_I^{ol}, $B_I = B_I^{wl} \cup B_I^{ol}$.

B. NETWORK SLICE REQUEST MODEL

In our model, the set of NSRs consists of three types of slices for three use case families, which can be denoted by R_{NS}, $R_{NS} = R_e \cup R_m \cup R_u$. R_e represents eMBB slice, R_m represents mMTC slice and R_u represents uRLLC slice. Each request is regarded as $G_R = (N_R, E_R, C_R, B_R, T_R)$ where N_R represents nodes of network slice, E_R represents links, C_R denotes capacity, B_R denotes bandwidth and T_R is the duration of the NSR remaining in the infrastructure network. Thus, $G_{R_e} = \left(N_{R_e}, E_{R_e}, C_{R_e}, B_{R_e}, T_{R_e}\right)$ is for request R_e, and similarly G_{R_m} and G_{R_u} is for requests R_m and R_u respectively.

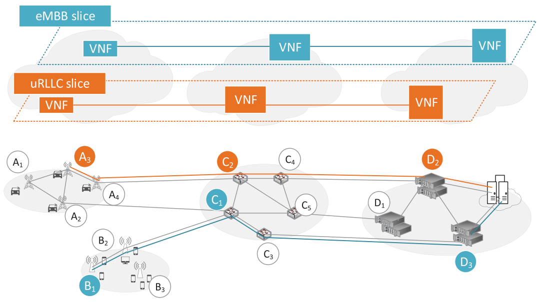


FIGURE 2. The schematic diagram of deploying NSRs.

C. SLICE DEPLOYMENT MODEL

Slice deployment is a process in which nodes of slice requests are mapped onto substrate nodes and links are mapped onto substrate paths on the premise of meeting service demands of slices. The mapping process consists of two stages, the node mapping and the link mapping. The schematic diagram of deploying NSRs is demonstrated in Fig. 2. As shown in Fig. 2, VNFs of eMBB slice are mapped in nodes B_1, C_1, D_3 and the links chaining VNFs are mapped in path $B_1 \rightarrow B_2 \rightarrow C_1 \rightarrow C_3 \rightarrow D_3$. Similarly, VNFs of uRLLC slice are mapped in nodes A_3, C_2, D_2 and the links chaining VNFs are mapped in path $A_3 \rightarrow A_4 \rightarrow C_2 \rightarrow C_4 \rightarrow D_2$. A node of NSR can only be mapped on a node of infrastructure network, and a node of infrastructure network can only host a node from the same of NSR. Below, there is a summary of parameters and variables that are used for the formulation of the mathematical model and the introduction of the decision variables:

1) PARAMETERS

- $C_I^{CN}\left(n_I^{CN}\right)$: Computing resource of core node n_I^{CN}, $n_I^{CN} \in N_I^{CN}$, $\sum_{n_I^{CN} \in N_I^{CN}} C_I^{CN}\left(n_I^{CN}\right) = C_I^{CN}$;
- $C_R^{CN}\left(n_R^{CN}\right)$: Computing resource requirement of the virtual core node $n_R^{CN} \in N_R^{CN}$;
- $B_R^{wl}\left(e_R^{wl}\right)$: Bandwidth requirement of the virtual wireless link $e_R^{wl} \in E_R^{wl}$;
- $B_R^{ol}\left(e_R^{ol}\right)$: Bandwidth requirement of the virtual optical link $e_R^{ol} \in E_R^{ol}$.

2) VARIABLES

- $C_I^{wl}\left(e_I^{wl}\right)$: Channel capacity of the wireless link e_I^{wl}, $e_I^{wl} \in E_I^{wl}$;
- $B_I^{wl}\left(e_I^{wl}\right)$: Bandwidth assigned to the wireless link e_I^{wl};
- $\mu_{n_R^{BS}, n_I^{BS}}$: Binary variable, if n_R^{BS} of G_R is mapped to n_I^{BS} of G_I, $\mu_{n_R^{BS}, n_I^{BS}} = 1$; Otherwise, $\mu_{n_R^{BS}, n_I^{BS}} = 0$;
- $\mu_{n_R^{OS}, n_I^{OS}}$: Binary variable, if n_R^{OS} of G_R is mapped to n_I^{OS} of G_I, $\mu_{n_R^{OS}, n_I^{OS}} = 1$; Otherwise, $\mu_{n_R^{OS}, n_I^{OS}} = 0$;
- $\mu_{n_R^{CN}, n_I^{CN}}$: Binary variable, if n_R^{CN} of G_R is mapped to n_I^{CN} of G_I, $\mu_{n_R^{CN}, n_I^{CN}} = 1$; Otherwise, $\mu_{n_R^{CN}, n_I^{CN}} = 0$;
- $\nu_{e_R^{wl}, e_I^{wl}}$: Binary variable, if a virtual link e_I^{wl} of G_R traverse the physical wireless link e_I^{wl}, $\nu_{e_R^{wl}, e_I^{wl}} = 1$; Otherwise, e_I^{wl}, $\nu_{e_R^{wl}, e_I^{wl}} = 0$;
- $\nu_{e_R^{ol}, e_I^{ol}}$: Binary variable, if a virtual link e_I^{ol} of G_R traverse the physical optical link e_I^{ol}, $\nu_{e_R^{ol}, e_I^{ol}} = 1$; Otherwise, e_I^{ol}, $\nu_{e_R^{ol}, e_I^{ol}} = 0$;
- $\xi_{n_R^{BS}, e_I^{wl}}$ Binary variable, if n_R^{BS} of G_R is served by $e_I^{wl} \in E_I^{wl}\left(n_R^{BS}\right)$, $\xi_{n_R^{BS}, e_I^{wl}} = 1$; Otherwise, $\xi_{n_R^{BS}, e_I^{wl}} = 0$;

IV. DEPLOYMENT POLICY OF NS BASED ON CN

The main objective of deployment process is minimizing the deployment cost on the premise of meeting NS requirements. In addition to the main objective, three types of slices have their peculiar objectives because of different service demands. Besides, we propose an VNFs placement algorithm based on CN theory which was used to analyze the topological properties of nodes in the underlying network.

A. MAPPING ALGORITHM

1) PLACING VNFS

Placing VNFs means to select the physical nodes of substrate network as host for the virtual nodes of NSRs under the condition of satisfying the capacity requirements. According to the literatures of VNE problem, the local resources for nodes are measured by

$$NR\,(i) = C\,(i) \cdot \sum_{l \in s(i)} BW\,(l) \tag{4}$$

where $C\,(i)$ represents the capacity of node i, $s\,(i)$ represents the set of links that directly connected to the node i, $BW\,(l)$ represents the current available bandwidth of link l. The main shortcoming of this measurement is ignoring the topological characteristics of nodes. Hence, we combine the degree and betweenness centrality of nodes to measure node importance in the step of placing VNFs.

First, the degree and betweenness centrality of nodes are normalized. Considering that the degree of node is not exceed $N-1$ when the total number of nodes is N, the normalization of the degree can be expressed by

$$d_i' = \frac{d_i}{N-1}. \tag{5}$$

Similarly, the betweenness centrality of node can be normalized by using

$$b'_i = \frac{2b_i}{(N-1)(N-2)} \tag{6}$$

because the maximum of b_i is $(N-1)(N-2)/2$. In the case of reaching maximum, each node pair of the network has at least one shortest path that travel through the node. Based on these normalized metrics of nodes, the weighting parameters of node i can be given by $\frac{d'_i+b'_i}{2}$. Therefore, combining the local resource and weighting of each node, the node importance of i can be given by

$$NI(i) = NR(i) \times \left(\frac{d'_i+b'_i}{2}\right). \tag{7}$$

According to the node importance, we use graphical breadth-first-search (BFS) algorithm to sort nodes, and map the virtual nodes to physical nodes based on BFS. The sorting algorithm of virtual nodes are listed in Algorithm 1. Based on the sorting algorithm, the node mapping algorithm is introduced in Algorithm 2. The node degree describes the number of its neighborhood nodes and node betweenness centrality describes the importance of a node with respect to the shortest path. Nodes with higher degree and betweenness centrality mean that they are frequently used and many shortest paths pass through them. Hence, with BFS algorithm, giving priority to the nodes with higher node importance will reduce link resources usage in the infrastructure network.

Algorithm 1 The Sorting Algorithm for Virtual Nodes in NSRs

Input: N_R: the set of virtual nodes in NSR
Output: N'_R: the sequence of sorted virtual nodes
Calculating NI value of each virtual node.
Sorting the virtual nodes by NI value in non-increasing order.
Selecting the virtual node with highest NI value as R.
Using R as the root node, traverse the graph of NSR using BFS algorithm, and get the BFS tree T.
Sorting the virtual nodes in each layer of T according to NI value in non-increasing order.
Return N'_R.

Algorithm 2 The Node Mapping Algorithm Based on BFS

Input: R_{NS}: the arrived NSR
Output: M_{node}: the results of node mapping
Sort virtual nodes with Algorithm 1.
Sort physical nodes according to their NI values in non-increasing order.
for each virtual node **do**
  if it is root R **then**
    it is mapped into the physical node with the greatest value of NI.
  else
    find the parent node P of it.
    find the mapped physical node I for P.
    find the neighbor nodes of I as the candidate physical nodes C.
    choose one of C which owns the greatest value of NI in the premise of satisfying the capacity requirements.
  end if
  return M_{node}.
end for

Algorithm 3 The Link Mapping Algorithm Based on KSP

Input: R_{NS}: the arrived NSR
Output: M_{link}: the results of link mapping
Sort the virtual links according to bandwidth in non-increasing order.
for each virtual link l. **do**
  calculate the bandwidth requirement $BW(l)$.
  remove the physical links that can not meet the bandwidth requirement.
  according to M_{node}, find the mapped physical nodes of l.
  find the physical shortest path between these two physical nodes by using Floyd algorithm.
  return M_{link}.
end for

2) CHAINING VNFS

The procedure of creating paths that interconnect the VNFs placed nodes would be achieved on the basis of k-shortest paths (KSP) algorithm. KSP algorithm is used to select suitable physical paths in the premise of satisfying the bandwidth resource requirements. After removing the link paths that do not satisfy the requirements, Floyd algorithm is used to calculate the shortest path. More details are shown in Algorithm 3.

B. OBJECTIVES OF THREE TYPES OF SLICES

In this section, we introduce the objectives of NSRs deployment problem. No matter which kind of slice is required, the ultimate goal is to take advantage of infrastructure resources efficiently. Hence, the main objective can be expressed by

$$\min \left[\sum_{n_R \in N_R} C_R(n_R) \cdot \mu_{n_R,n_I} + \sum_{e_R \in E_R} B_R(e_R) \cdot \nu_{e_R,e_I} \right]. \tag{8}$$

Furthermore, the objectives of different kinds of slices are also determined by their service demands. These objectives are listed below.

1) EMBB SLICE

The eMBB usage scenario covers a range of cases, including wide-area coverage and hotspot. For the hotspot case, i.e. for an area with high user density, very high traffic capacity is needed, while the requirement for mobility is low and user data rate is higher. This kind of slice does not require strict delay and plentiful resources. Hence, the deployment objective of eMBB slices should be maximizing the remaining resources of physical nodes, which can be represented by

$$\max \left[\sum_{n_I \in N_I} C_I(n_I) - \sum_{n_R \in N_R} C_R(n_R) \cdot \mu_{n_R,n_I} \right] \tag{9}$$

2) MMTC SLICE

The mMTC usage scenario is characterized by a very large number of connected devices typically transmitting a relatively low volume of non-delay sensitive data. This use case has plenty of connections, which results in the requirement of high computing resources and low congestion rate. Therefore, the deployment objective should be minimizing the usage of bandwidth on physical links. In other words, the remaining bandwidth on physical links should be maximized. Thus, the deployment objective of mMTC slice can be denoted as

$$\max \left[\sum_{e_I \in E_I} B_I(e_I) - \sum_{e_R \in E_R} B_R(e_R) \cdot \nu_{e_R,e_I} \right] \tag{10}$$

3) URLLC SLICE

The uRLLC usage scenario has stringent requirements for capabilities such as throughput, latency and availability. Some examples include wireless control of industrial manufacturing, remote medical surgery, transportation safety, etc. The QoS guarantee of this use case is low latency, which causes that the deployment objective should be minimizing the delay of slices. We transfer delay time into the number of hops, so minimizing the delay means minimizing each physical path length. Hence, deployment objective of uRLLC slices is

$$\min \sum_{e_R^{wl} \in E_R^{wl}} \nu_{e_R^{wl},e_I^{wl}} + \sum_{e_R^{ol} \in E_R^{ol}} \nu_{e_R^{ol},e_I^{ol}} \tag{11}$$

These objectives are subject to

$$\sum_{n_R^{BS} \in N_R^{BS}} \mu_{n_R^{BS},n_I^{BS}} = 1, \quad \forall n_I^{BS} \in N_I^{BS} \tag{12}$$

$$\sum_{n_I^{BS} \in N_I^{BS}} \mu_{n_R^{BS},n_I^{BS}} \le 1, \quad \forall n_R^{BS} \in N_R^{BS} \tag{13}$$

Eq. 12 ensures that each virtual BS only should be mapped to a physical BS. Eq. 13 ensures that each physical BS only should undertake a virtual BS for each NSR.

$$\sum_{e_I^{wl} \in E_I^{wl}} B_I^{wl}\left(e_I^{wl}\right) \le B_I^{wl} \tag{14}$$

$$\sum_{e_I^{wl} \in E_I^{wl}(n_I^{BS})} C_I^{wl}\left(e_I^{wl}\right) \le C_I^{BS}\left(n_I^{BS}\right), \quad \forall n_R^{BS} \in N_R^{BS} \tag{15}$$

$$\sum_{n_R^{BS} \in N_R^{BS}} \mu_{n_R^{BS},n_I^{BS}} \cdot C_R^{BS}\left(n_R^{BS}\right) \le C_I^{BS}\left(n_I^{BS}\right), \ \forall n_I^{BS} \in N_I^{BS} \tag{16}$$

$$\sum_{n_I^{BS} \in N_I^{BS}} \sum_{e_I^{wl} \in E_I^{wl}(n_I^{BS})} \mu_{n_R^{BS},n_I^{BS}} \cdot \xi_{n_R^{BS},e_I^{wl}} \cdot C_I^{wl}\left(e_I^{wl}\right) \ge C_R^{BS}\left(n_R^{BS}\right), \quad \forall n_R^{BS} \in N_R^{BS} \tag{17}$$

Eq. 14 ensures that the bandwidth occupied by all the wireless channels should not exceed the total available bandwidth B_I^{wl} for each BS. Eq. 15 ensures that the channel capacity sum of wireless links in each BS should not exceed capacity of this BS allocated by the control plane. Eq. 16 ensures that the capacity sum of all virtual BS undertaken in this BS should not exceed its allocated capacity. Eq. 17 ensures that the allocated capacity for each virtual BS should not be less than its capacity requirement.

$$\sum_{n_R^{OS} \in N_R^{OS}} \mu_{n_R^{OS},n_I^{OS}} = 1, \quad \forall n_I^{OS} \in N_I^{OS} \tag{18}$$

$$\sum_{n_I^{OS} \in N_I^{OS}} \mu_{n_R^{OS},n_I^{OS}} \le 1, \quad \forall n_R^{OS} \in N_R^{OS} \tag{19}$$

Eq. 18 ensures that each virtual OS only should be mapped to a physical OS. Eq. 19 ensures that each physical OS only can undertake a virtual OS for each NSR.

$$\sum_{n_R^{CN} \in N_R^{CN}} \mu_{n_R^{CN},n_I^{CN}} = 1, \quad \forall n_I^{CN} \in N_I^{CN} \tag{20}$$

$$\sum_{n_I^{CN} \in N_I^{CN}} \mu_{n_R^{CN},n_I^{CN}} \le 1, \quad \forall n_R^{CN} \in N_R^{CN} \tag{21}$$

$$\sum_{n_R^{CN} \in N_R^{CN}} \mu_{n_R^{CN},n_I^{CN}} \cdot C_R^{CN}\left(n_R^{CN}\right) \le C_I^{CN}\left(n_I^{CN}\right), \quad \forall n_I^{CN} \in N_I^{CN} \tag{22}$$

$$\sum_{n_I^{CN} \in N_I^{CN}} \mu_{n_R^{CN},n_I^{CN}} \cdot C_I^{CN}\left(n_I^{CN}\right) \ge C_R^{CN}\left(n_R^{CN}\right), \quad \forall n_R^{CN} \in N_R^{CN} \tag{23}$$

Eq. 20 ensures that each virtual CN only should be mapped to a physical CN. Eq. 21 ensures that each physical CN only can undertake a virtual CN for each NSR. Eq. 22 ensures that the computing resource of each physical CN can satisfy the total requirement of all virtual CNs mapped in it. Eq. 23 ensures that the computing resource of selected CN should not be less than the computing resource requirement of virtual CNs.

C. DEPLOYMENT STRATEGIES OF THREE TYPES OF SLICES

As a functional block in the management framework of 5G network slicing [36], the deployment strategies we proposed are conducted under the assumption that the arrived NSR has been identified as one of those three types. In our proposed deployment policy of NS, we give three different strategies to three types of slices. After the arriving of NSRs, these requests are classified, then implemented by different mapping algorithms respectively. Meanwhile, the resource efficiency (RE) and acceptance ratio (AR) of NSRs are calculated. Resource efficiency is defined as the revenues and cost ratio. The achieved revenues of accepting a NSR by the infrastructure network can be defined as the sum of nodes capacity and link bandwidth requirements of a NSR. And the cost can be defined as the sum of nodes capacity and link bandwidth resources of the infrastructure network. Hence, the resource efficiency can be formulated as follows:

$$RE = \frac{\sum\limits_{n \in N_R} C_R(n) + \sum\limits_{l \in E_R} B_R(l)}{\sum\limits_{n \in N_R} C_R(n) + \sum\limits_{l \in E_R} B_R(l) \times hop(l)} \tag{24}$$

Where $C_R(n)$ represents the capacity of node n and $B_R(l)$ represents the bandwidth of link l, $hop(l)$ represents the mapping path length of link l. Furthermore, acceptance ratio is the ratio of the number of NSRs which have been successfully mapped and the total number. Hence, it can be formulated as

$$AR = \frac{\sum_{t=0}^{T} NUM_{acc}}{\sum_{t=0}^{T} NUM_{arr}} \tag{25}$$

In the above formula, NUM_{acc} represents the number of NSRs that have been accepted while NUM_{arr} denotes the number of NSRs that have been arrived. Details are presented in Algorithm 4. For the deployment of NSRs, we consider two conditions which are the static deployment and dynamic deployment. The former means that slices are permanent once they are deployed successfully. However, in the latter condition, NSRs have life time and the resource allocated to NSRs will be recycled at the end of life time. For dynamic deployment, step 2 in Algorithm 4 will check if there are any NSRs which need to be recycled and calculate the resources after the recycle of NSRs. However, for static deployment, there are no any NSRs which need to be recycled during the deployment of the sequence of NSRs.

Algorithm 4 The NSRs Implementation Algorithm

Input: $G_I = (N_I, E_I, C_I, B_I)$ and $R_{NS} = R_e \cup R_m \cup R_u$

Output: AR, RE, M_{node} and M_{link}

1: **while** $R_{NS} \neq \emptyset$ **do**
2: Calculating the resources of infrastructure network G_I.
3: **if** NSR is eMBB slice R_e **then**
4: deploy it using Algorithm 5.
5: **else if** NSR is mMTC slice R_m **then**
6: deploy it using Algorithm 6.
7: **else**
8: deploy it using Algorithm 7.
9: **end if**
10: **if** M_{node} and M_{link} are not null **then**
11: update the resources of infrastructure network G_I.
12: calculating resource efficiency RE.
13: **else if** M_{node} and M_{link} are null **then**
14: calculating acceptance ratio AR.
15: **end if**
16: **end while**

1) ALGORITHM A FOR EMBB SLICE

In the deployment algorithm for eMBB slice, the virtual BSs are first sorted and mapped according to the node mapping algorithm (Algorithm 2). After mapping the virtual BSs, the virtual CNs are mapped similarly considering the computing resource requirements. Then, the mapping of virtual OSs are finished when searching the shortest paths between each BS-CN pair. Finally, the virtual links are mapped with the link mapping algorithm (Algorithm 3). Details are shown in Algorithm 5.

Algorithm 5 Deployment Algorithm A for eMBB Slice

Input: G_I and R_e

Output: M_{node} and M_{link}

1: sort the virtual nodes by Algorithm 1.
2: do node mapping of BSs by Algorithm 2.
3: do node mapping of CNs by Algorithm 2.
4: do link mapping by Algorithm 3.
5: do node mapping of OSs according to the link mapping.
6: return the mapping results.

2) ALGORITHM B FOR MMTC SLICE

In the deployment algorithm for mMTC slice, the virtual CNs are mapped firstly. Next, take these CNs as the source endpoints and find out the candidate BSs as the target endpoints. Then, we select the shortest path from the set of candidate paths between CN and candidate BSs to map the virtual link. Details are shown in Algorithm 6.

Algorithm 6 Deployment Algorithm B for mMTC Slice

Input: G_I and R_m

Output: M_{node} and M_{link}

1: sort the virtual nodes by Algorithm 1.
2: do node mapping of CNs by Algorithm 2.
3: select available BSs as candidate BSs.
4: search the set of candidate paths between CN and candidate BSs.
5: do link mapping based on candidate paths by Algorithm 3.
6: do node mapping of OSs according to the link mapping.
7: do node mapping of BSs.
8: return the mapping results.

3) ALGORITHM C FOR URLLC SLICE

In the deployment algorithm for uRLLC slice, we first find out the set of candidate BS-CN pairs and search all the possible routing paths between these candidate pairs as the candidate path set. According to the number of virtual links of NSR, we select the shortest paths from the set of candidate paths to map the virtual links. Then we map the virtual BSs and CNs into the physical endpoints of the selected paths. Details are shown in Algorithm 7.

Algorithm 7 Deployment Algorithm C for uRLLC Slice

Input: G_I and R_u

Output: M_{node} and M_{link}

sort the virtual nodes by Algorithm 1.
select available BSs as candidate BSs.
select available CNs as candidate CNs.
search the set of candidate paths between candidate CN and candidate BS.
do link mapping based on candidate paths by Algorithm 3.
do node mapping of OSs according to the link mapping.
do node mapping of BSs.
do node mapping of CNs.
return the mapping results.

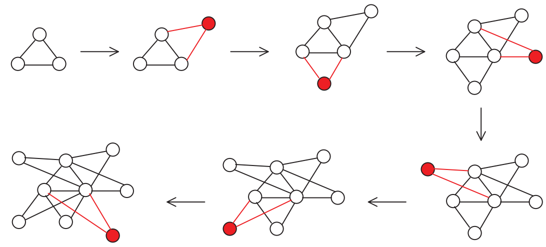


FIGURE 3. The generative process of BA scale-free network.

TABLE 1. The settings of parameters.

Parameter Items	The Range
Substrate network:	
The number of physical BS	30,60,90
The number of physical OS	40,80,120
The number of physical CN	30,60,90
The distribution of node capacity	U[20,50]
The distribution of link bandwidth	U[20,50]
Network slice requests:	
The nodes number of each NSR	20
The distribution of node capacity requirement	U[5,25]
The distribution of link bandwidth requirement	U[5,25]
The number of NSRs arrived	U[5,35]
The time duration of each NSR	T[15,45]
The time point of NSRs randomly arrived	T[1,50]

V. SIMULATION RESULTS AND ANALYSIS

In this section, the simulation environment settings and the simulation results are discussed. The integral linear programming (ILP) model presented in previous section is implemented using YALMIP platform embedded in MATLAB which integrates the commercial programming solver CPLEX 12.7. The Algorithm 1-7 are achieved using MATLAB 2015b and simulations are carried out on a laptop with four 2.4GHz CPU cores and 12GB memories.

A. EXPERIMENTAL ENVIRONMENT SETTINGS

In the growth of communication network, a forthcoming node has the tendency to connect itself to the nodes with large degrees, and the node-degree distribution follows a power-law form. Hence, we apply the algorithms of BarabÃąsi-Albert (BA) scale-free networks [37] to generate the topology of NSRs and infrastructure network. The generating principle is as follows:

- Growth: Starting from a connected network of small size with $n_0 \geq 1$ nodes, introduce one new node to the existing network each time, and this incoming new node is connected to n existing nodes, where $1 \leq n \leq n_o$.
- Preferential Attachment: The incoming new node is connected to each of the n existing nodes, according to the following probability: $\prod_i = \frac{k_i}{\sum_{j=1}^{N_c} k_j}$, where N_c denotes the total number of current existing nodes and k_i is the degree of node i.

Obviously, after t steps of adding new nodes, the BA network will have $N = n_0 + t$ nodes. Fig. 3 illustrates the growing process of a BA scale-free network.

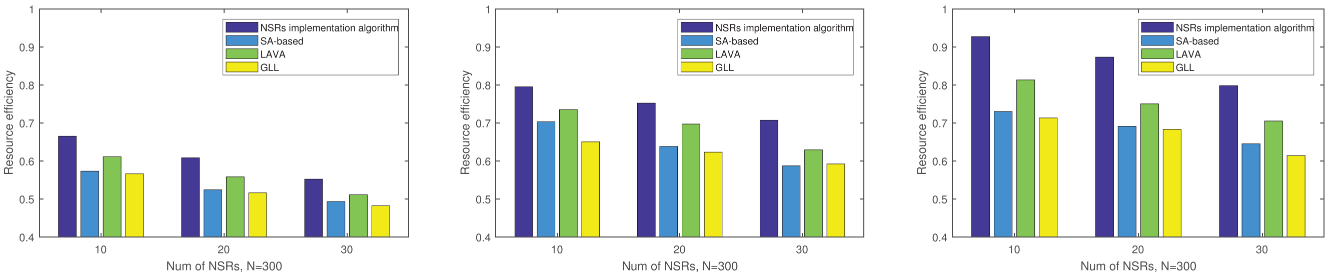
The specific parameter settings are shown in Table.1. We assume that the total number of physical nodes in infrastructure network can be $N = 100, 200, 300$. These nodes are divided into three sets: BS nodes, OS nodes and CN nodes. For the static deployment, the sequences of NSRs consist of three different types of slices and the numbers of eMBB slices, mMTC slices and uRLLC slices are randomly decided. For dynamic deployment, the difference is that the time duration of slice is ranged from 15 units to 45 units and the arrive time of each NSR is between 1 and 50.

B. EXPERIMENTAL RESULTS AND ANALYSIS

We use the resource efficiency and accptance ratio which were introduced in the previous section to evaluate the performance of the NSRs implementation algorithm. As only a few deployment algorithm of E2E slices can be found in current literatures, our deployment algorithms are compared with a simulated annealing (SA) [38] based approach, the VNE algorithm LAVA [39] and the VNF placement algorithm GLL [40]. The SA-based approach here is similar with SA-based middel box placement algorithm in [27]. LAVA is a service-oriented substrate resource slicing strategy, and GLL is a greedy algorithm which is based on functions being mapped to the node with highest available capacity. Moreover, we analyze the execution time of three different deployment strategies while the number of NSRs increases. To eliminate the randomicity and improve the accuracy, the simulations have been operated 50 times independently and took the average.

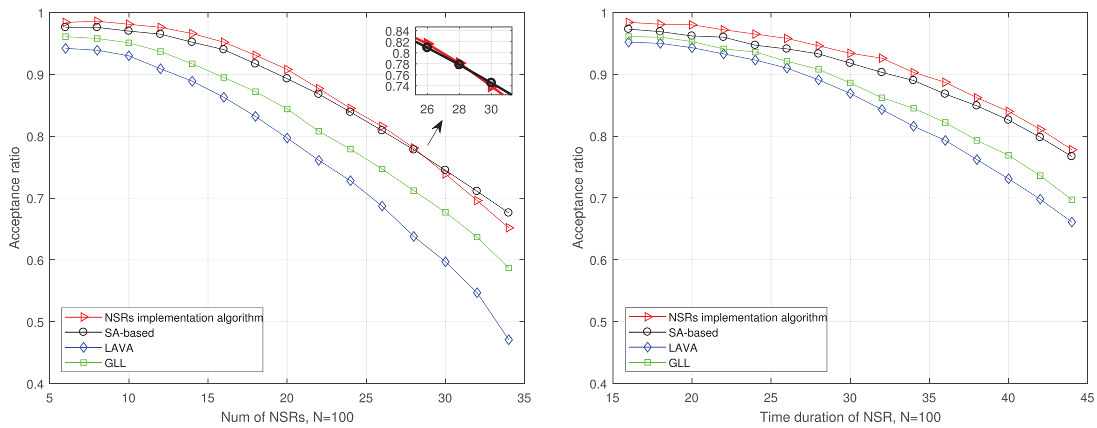
1) RESOURCE EFFICIENCY

In Fig. 4, we compare the RE results of eMBB slice, mMTC slice, and uRLLC slice sequences respectively. Each sequence of NSRs contains a single type of slices under the condition of static deployment. We calculate the results at $N = 300$ to guarantee sufficient physical resource. It can be seen that RE gradually decreases along with the increasing of



(a) (b) (c)

FIGURE 4. Resource efficiency of different types of slices. (a) eMBB slices. (b) mMTC slices. (c) uRLLC slices.



(a) (b)

FIGURE 5. Acceptance ratio of different deployment algorithms. (a) Static deployment. (b) Dynamic deployment.

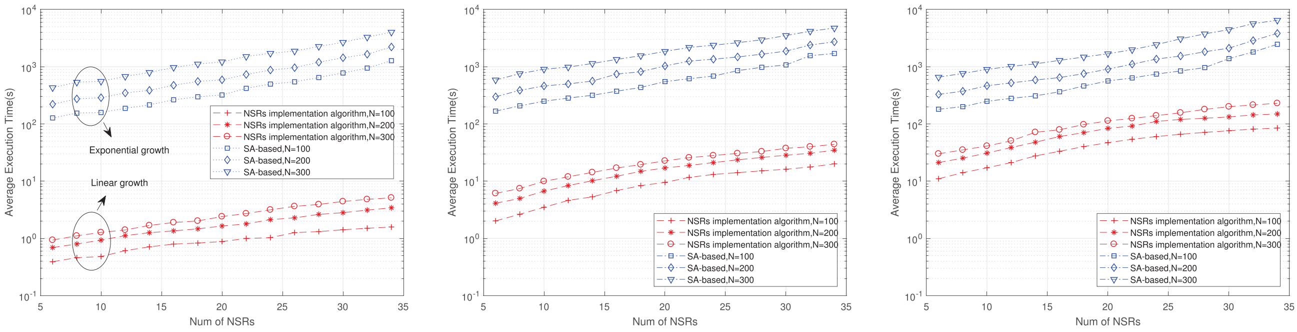
the number of NSRs. The main reason is that the bigger the number of NSRs, the longer the average mapping path length and the more link bandwidth resources are consumed. The results show that NSRs implementation algorithm and LAVA perform better than two heuristic algorithms, SA-based and GLL. Thus, it can be concluded that service-oriented resource slicing approach can improve the efficiency of resource using. Since we have improved the node's ranking method with the topological characteristics based on CN theory, our algorithm performs even better than LAVA. In addition, we can see that the RE of uRLLC slices is highest compared to mMTC slices and eMBB slices. This is due to the fact that the average mapping path length of links in the sequence of uRLLC slices is lowest, which is the result of the deployment objective of uRLLC slices.

2) ACCEPTANCE RATIO

According to equation Eq. 22, the AR is proportional to the number of successfully deployed slices. We analyze the relationship between the number of NSRs and the AR in the condition of static deployment. Furthermore, we calculate the AR when the time duration of each NSR is changed in the condition of dynamic deployment. The AR results of these two conditions are plotted respectively in Fig. 5. As illustrated in Fig. 5(a), the AR of four algorithms decrease gradually with the increasing number of NSRs and time duration of each NSR. And the AR of our algorithm is higher than others when the number of NSRs is lower than 28 approximately. However the AR of our algorithm is lower than SA-based algorithm when the number of NSRs is more than 28. That is because, our algorithm focuses on increasing the RE and provide different objectives for three types of slices. This more strict deployment policy leads to lower AR when the number of NSRs is higher. It also can be observed that the AR of GLL is higher than LAVA. The reason is that LAVA allocates the substrate resources according to the service class, which is more strict than GLL. In Fig. 5(b), the infrastructure network recycles resources with the ending of slices, which causes more available resources. Hence, the number of NSRs that are accepted successfully is higher than static deployment. When the time duration is long, the resources are not frequently updated and it causes the decreasing of AR.

3) EXECUTION TIME

To evaluate the execution time, we use the proposed algorithms and SA-based approach to deploy three types of slices



(a) (b) (c)

FIGURE 6. Execution time performance of different deployment algorithms. (a) Algorithm A for eMBB slices. (b) Algorithm B for mMTC slices. (c) Algorithm C for uRLLC slices.

seperately in the condition of static deployment. In Fig. 6, with the increasing number of slices, both average execution time of our algorithms and SA-based algorithm increase. And the SA-based algorithm takes much more time than the proposed algorithm for three types of slices. It can also be noticed that average execution time of our algorithm is different for three types of slices. For eMBB slices, it requires quite less time and the execution time for uRLLC slices is highest. That is because, algorithm B and algorithm C need to search the set of candidate paths. In addition, algorithm C searches the paths between candidate BS nodes and candidate CN nodes, which cause more time consumption than algorithm B. For the proposed algorithm, the search of paths between BS domain and CN domain becomes the main factor of execution time. However, for SA-based algorithm, it takes too much time to get an optimal location for each node and an optimal path for each link. Moreover, as the total number of physical nodes increases, the execution time increments of our deployment algorithm are less than the SA-based algorithm. As a heuristic method, time to find the optimal solution for SA-based algorithm increases exponentially when the underlying infrastructure network scales up. Oppositely, there is an approximately liner increase in the execution time of our algorithm. Hence, our algorithm performs better in terms of the execution time.

VI. CONCLUSION

The 5G network is envisioned to support a range of verticals and use cases, which causes that network slicing draws a lot of attention. In 5G network slicing, the deployment of different type of slices is one of the most challenging tasks aimed at enabling 5G architecture to accommodate diverse use cases. In this paper, we introduce the metric of node importance based on complex network theory. Based on topological information, we present the mathematical model of deploying end-to-end slices. And the network slice request implementation algorithm was proposed, which includes three different deployment algorithms for eMBB slices, mMTC slices and uRLLC slices respectively. Extensive simulations was conducted to validate the performance of our proposed algorithm and analyze the average execution time. Results analysis of simulations have shown that our proposed algorithm achieve higher resource efficiency and acceptance ratio. Analysis of execution time prove that E2E network slices can be deployed rapidly with our proposed service-oriented deployment policy.

REFERENCES

[1] *5G Empowering Vertical Industries*, 5GPPP and ERTICO, 2016. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2016/02/BROCHURE_5PPP_BAT2_PL.pdf
[2] *View on 5G Architecture*, 5GPPP, 2016. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-5G-Architecture-WP-July-2016.pdf
[3] *Description of Network Slicing Concept*, NGMN-Alliance, 2016, vol. 1. [Online]. Available: https://www.ngmn.org/fileadmin/user_upload/160113_Network_Slicing_v1_0.pdf
[4] H. Zhang, N. Liu, X. Chu, K. Long, A.-H. Aghvami, and V. C. M. Leung, "Network slicing based 5g and future mobile networks: Mobility, resource management, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 138–145, Aug. 2017.
[5] "Imt vision–framework and overall objectives of the future development of IMT for 2020 and beyond," Tech. Rep., 2015. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.2083-0-201509-I!!PDF-E.pdf
[6] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 80–87, May 2017.
[7] Q. Li, G. Wu, A. Papathanassiou, and U. Mukherjee. (2016). "An end-to-end network slicing framework for 5G wireless communication systems." [Online]. Available: https://arxiv.org/abs/1608.00572
[8] M. R. Sama, X. An, Q. Wei, and S. Beker, "Reshaping the Mobile core network via function decomposition and network slicing for the 5G era," in *Proc. IEEE Wireless Commun. Netw. Conf. Workshops (WCNCW)*, Apr. 2016, pp. 90–96.
[9] F. Bari, S. R. Chowdhury, R. Ahmed, R. Boutaba, and O. C. M. B. Duarte, "Orchestrating virtualized network functions," *IEEE Trans. Netw. Service Manag.*, vol. 13, no. 4, pp. 725–739, Dec. 2016.
[10] R. Riggio, A. Bradai, D. Harutyunyan, T. Rasheed, and T. Ahmed, "Scheduling wireless virtual networks functions," *IEEE Trans. Netw. Service Manag.*, vol. 13, no. 2, pp. 240–252, Jun. 2016.
[11] S. Herker, A. Khan, and X. An, "Survey on survivable virtual network embedding problem and solutions," in *Proc. Int. Conf. Netw. Services*, 2013, pp. 1–6.
[12] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Phys. Rep.*, vol. 424, nos. 4–5, pp. 175–308, 2006.
[13] B. Zhang, R. Liu, D. Massey, and L. Zhang, "Collecting the internet AS-level topology," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 1, pp. 53–61, 2005.
[14] J. Wu, C. K. Tse, and F. C. M. Lau, "Optimizing performance of communication networks: An application of network science," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 62, no. 1, pp. 95–99, Jan. 2015.

[15] Y. B. Kim, B. Hong, and W. Choi, "Scale-free wireless networks with limited degree information," *IEEE Wireless Commun. Lett.*, vol. 1, no. 5, pp. 428–431, Oct. 2012.

[16] N. Nikaein *et al.*, "Network store: Exploring slicing in future 5G networks," in *Proc. 10th Int. Workshop Mobility Evol. Internet Archit.*, 2015, pp. 8–13.

[17] X. Costa-Pèrez, J. Swetina, T. Guo, R. Mahindra, and S. Rangarajan, "Radio access network virtualization for future mobile carrier networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 27–35, Jul. 2013.

[18] P. Rost *et al.*, "Mobile network architecture evolution toward 5G," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 84–91, May 2016.

[19] K. Samdanis, X. C. Perez, and V. Sciancalepore, "From network sharing to multi-tenancy: The 5G network slice broker," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 32–39, Jul. 2016.

[20] X. Zhou, R. Li, T. Chen, and H. Zhang, "Network slicing as a service: Enabling enterprises' own software-defined cellular networks," *IEEE Commun. Mag.*, vol. 54, no. 7, pp. 146–153, Jul. 2016.

[21] T. Taleb, B. Mada, M. Corici, A. Nakao, and H. Flinck, "PERMIT: Network slicing for personalized 5G mobile telecommunications," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 88–93, May 2017.

[22] "Network Functions Virtualisation (NFV); architectural framework," ETSI and GSNFV, Tech. Rep., 2013. [Online]. Available: http://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.01.01_60/gs_NFV002v010101p.pdf

[23] R. Mijumbi *et al.*, "Network function virtualization: State-of-the-art and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 236–262, 1st Quart., 2016.

[24] LinuxFoundation. *Opnfv*. Accessed: Jun. 7, 2017. [Online]. Available: https://www.opnfv.org/

[25] Telefonica. *Openmano*. Accessed: Jun. 7, 2017. [Online]. Available: https://github.com/nfvlabs/

[26] FraunhoferFOKUS. *Openbaton*. Accessed: Jun. 7, 2017. [Online]. Available: http://openbaton.github.io/

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

[28] L. Wang, Z. Lu, X. Wen, R. Knopp, and R. Gupta, "Joint optimization of service function chaining and resource allocation in network function virtualization," *IEEE Access*, vol. 4, pp. 8084–8094, 2016.

[29] S. Clayman, E. Maini, A. Galis, A. Manzalini, and N. Mazzocca, "The dynamic placement of virtual network functions," in *Proc. IEEE Netw. Oper. Manage. Symp. (NOMS)*, May 2014, pp. 1–9.

[30] M. Ghaznavi, A. Khan, N. Shahriar, K. Alsubhi, R. Ahmed, and R. Boutaba, "Elastic virtual network function placement," in *Proc. IEEE 4th Int. Conf. Cloud Netw. (CloudNet)*, Oct. 2015, pp. 255–260.

[31] A. Fischer, J. F. Botero, M. T. Beck, H. de Meer, and X. Hesselbach, "Virtual network embedding: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 1888–1906, 4th Quart., 2013.

[32] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking virtual network embedding: Substrate support for path splitting and migration," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 17–29, Apr. 2008.

[33] N. M. M. K. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual network embedding with coordinated node and link mapping," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 783–791.

[34] M. Chowdhury, M. R. Rahman, and R. Boutaba, "ViNEYard: Virtual network embedding algorithms with coordinated node and link mapping," *IEEE/ACM Trans. Netw.*, vol. 20, no. 1, pp. 206–219, Feb. 2012.

[35] M. Chowdhury, F. Samuel, and R. Boutaba, "PolyViNE: Policy-based virtual network embedding across multiple domains," in *Proc. 2nd ACM SIGCOMM Workshop Virtualized Infrastruct. Syst. Archit.*, 2010, pp. 49–56.

[36] W. Guan, X. Wen, L. Wang, and Z. Lu, "Network slicing management of 5G ultra-dense networks based on complex network theory," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2017, pp. 1–6.

[37] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[38] E. Aarts and J. Korst, "Simulated annealing and boltzmann machines," Tech. Rep., 1988. [Online]. Available: https://www.osti.gov/biblio/5311236

[39] C. Yu, W. Hou, Y. Guan, Y. Zong, and P. Guo, "Virtual 5G network embedding in a heterogeneous and multi-domain network infrastructure," *China Commun.*, vol. 13, no. 10, pp. 29–43, Oct. 2016.

[40] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. de Turck, and S. Davy, "Design and evaluation of algorithms for mapping and scheduling of virtual network functions," in *Proc. 1st IEEE Conf. Netw. Softwarization (NetSoft)*, Apr. 2015, pp. 1–9.

WANQING GUAN received the B.S. degree in electronic information science and technology from Beijing Forestry University, Beijing, China. She is currently pursuing the Ph.D. degree in communications engineering at the Beijing University of Posts and Telecommunications, China. Her current research interests include network slicing, network architecture, and complex network theory.

XIANGMING WEN received the B.E., M.S., and Ph.D. degrees from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, all in electrical engineering. He is currently the Vice President of BUPT, where he is also a Professor with the Communication Network Center and the Director of the Beijing Key Laboratory of Network System Architecture and Convergence. He is the Vice Director of the Organization Committee of the China Telecommunication Association. In the last five years, he has been the author of over 100 papers published. He is the Principle Investigator of over 18 projects, including the National Key Project of Hi-Tech Research and Development Program of China (863 Program) and the National Natural Science Foundation of China. His current research interests include broadband mobile communication theory, multimedia communications, and information processing.

LUHAN WANG received the B.S. degree in communication engineering from Shandong University, Jinan, China, and the Ph.D. degree from the Beijing University of Posts and Telecommunications in 2017. He joined the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, in 2017. His current research interests include network architecture, network function virtualization, and software-defined networks.

ZHAOMING LU received the Ph.D. degree from the Beijing University of Posts and Telecommunications in 2012. He joined the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, in 2012. His research interests include open wireless networks, QoE management in wireless networks, software-defined wireless networks, and cross-layer design for mobile video applications.

YIDI SHEN received the B.S. degree in communication engineering from Jilin University, Changchun, China. She is currently pursuing the M.S. degree in communications engineering at the Beijing University of Posts and Telecommunications, China. Her current research interests include network slicing and complex network theory.