IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 9, 2019, accepted January 21, 2019, date of publication January 31, 2019, date of current version April 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2895316

On Orchestrating Service Function Chains in 5G Mobile Network

DONGCHENG ZHAO[1], JING REN[1], RONGPING LIN[1,2], SHIZHONG XU[1], AND VICTOR CHANG[3]

[1]Key Lab of Optical Fiber Sensing and Communications, Ministry of Education, University of Electronic Science and Technology of China, Chengdu, China
[2]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China
[3]Xi'an Jiaotong -Liverpool University, Suzhou, China

Corresponding authors: Jing Ren (renjing@uestc.edu.cn) and Rongping Lin (linrp@uestc.edu.cn)

This work was supported in part by the Natural Science Foundation of China under Grant 61571098, in part by the 111 Project under Grant B14039, in part by the Open Foundation of State key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, under Grant SKLNST-2016-2-24.

ABSTRACT As two of the key technologies of 5G, fog-based radio access network and network function virtualization have become an important direction for the radio network architecture evolution. Virtual network functions (VNFs) compose the service function chain (SFC) in a particular order, and the mobile network users communicate with each other or service terminals through SFCs. For service providers, it is crucially important for efficient deploying/mapping SFCs into the 5G mobile network since the SFCs deployment problem is an NP-hard problem. In this paper, we propose the efficient SFCs deployment algorithms for solving this challenge with two main design goals: 1) minimizing the cost of link resource, i.e., minimizing the path length of the entire SFC by combining VNFs and mapping temporary links and 2) minimizing the cost of computing resources by using the existing virtual machines as much as possible while performing the same VNF. We model the SFCs deployment problem as an optimization problem by using ILP as well as devise the heuristic algorithms to make the tradeoff between these two conflicting design goals. From simulation results, we can see that the performance of our proposed algorithms is promising in terms of the total SFCs mapping cost, total links mapping cost, and blocking ratio.

INDEX TERMS Service function chain, fog-based radio access network, virtual network function, network function virtualization, deployment, 5G mobile network.

I. INTRODUCTION

Mobile radio traffic has experienced explosive growth over the past decade. In order to meet the explosively increasing demands of users, researchers have launched researches on the fifth-generation (5G) mobile radio networks [1]–[3]. Researchers have made significant progress in 5G network architecture and the Radio Access Network (RAN) [4]–[7].

To meet users' requirements for a more diversity, telecom operators must purchase correspondingly to meet the requirements of users [8]. But, the purchase of more physical equipment will lead to high Operational Expenditures (OPEX) and Capital Expenditures (CAPEX) [9]. With the explosive growth of mobile radio traffics, service providers consider to extend their services by using cloud resources [10], based on the virtualization technology [11]–[14], the researcher has presented the Network Function Virtualization (NFV) [15] which devotes to migrate the packet processing from the hardware middle box to the software middle box running on the hardware. The network function running in software middle box is called as Virtual Network Function (VNF). Multiple VNFs are typically connected in a particular order to compose Service Function Chain (SFC) which provides various network services. With the increasing demand for cloud resources, the centralized cloud computing is facing some challenges (such as the network congestion and the longer latency). To save these challenges, CISCO proposed the distributed fog computing. As a promising technology, the fog computing has received extensive

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

attention [16]–[22], where there are some discussion about deploying 5G/NFV in fog computing [17]–[20], and some researchers have studied the Fog-based Radio Access Network (FRAN) [21], [22].

As one of the key technologies of 5G, NFV has become an important direction for the radio network architecture evolution. For service providers, it is crucially important for efficient deploying/mapping SFCs into 5G mobile network. Now, the VNF placement also has become the hot study topic, but few studies have considered the fog-based radio access network and deploying NFV in 5G, since they only have some discussion of the problem or just present some overall architecture. In this paper, we efficient deploy SFCs into 5G mobile network to minimize the costs of computing resources and link resources. Firstly, the SFCs deployment problem is a NP-hard problem. Secondly, deploying SFCs in the 5G mobile network has a more restrictive location constraint. In addition, the addition of fog computing will bring new challenges to the SFC deployment. In the SFCs deployment, minimizing the deployment cost of the SFC or minimizing the network resource consumption is usually the goal pursued by service providers. While network resources are limited, service providers can maintain service delivery by minimizing the resource consumption and restricting user requests under the limited resources. To deliver this, service providers can jointly optimize allocation of links and compute resources to minimize the total resource consumption. This allows service providers to minimize the deployment cost of the SFC and improve their revenue.

Therefore, to optimize the SFCs deployment problem in 5G mobile network based on the fog radio access network, it is essential to design an effective SFCs deployment algorithm. In this paper, we research the SFCs deployment problem in 5G mobile network based on the fog radio access network. We also propose efficient SFCs deployment algorithms with two main design goals: *i*) minimizing link resource costs, i.e., minimizing the path length of the entire SFC by combining VNFs and mapping temporary links; *ii*) minimizing the cost of computing resources through using the existing virtual machines (VMs) as much as possible while performing the same VNF. The main contributions of this work are as follows:

- To improve the resource utilization of servers, we introduce the strategy of reusing virtual machines, i.e., when mapping the VNFs, we can reuse the existing VMs that perform the same VNF. In order to obtain the optimal path for the entire SFC, we propose two key strategies: the VNFs combination strategy and the temporary link mapping strategy.
- Under certain link constraints to minimize the cost of computing resources as much as possible, i.e., SFC mapping formulation with VMs reusing. VNFs combination and temporary link mapping can be used to minimize the cost of computing resources (SFCM-MC). To move forward, we first model the SFCM-MC formulation and design an algorithm based on the SFCM-MC formulation.
- Under certain computing resources constraints to minimize the cost of link resources, i.e., SFC mapping formulation with VMs reusing. VNFs combination and temporary link mapping can be used to minimize the cost of link resources (SFCM-ML). To make it happen, we first model the SFCM-ML formulation, then design an algorithm based on the SFCM-ML formulation.
- Based on a two-player pure-strategy's Game of Battle of Sex (BoS) model which captures the competition on physical resources between VNF allocation and routing to find a fair solution for the costs of computing resources and link resources, i.e., SFC mapping formulation with VMs reusing. VNFs combination and temporary link mapping can be used to optimize the costs of computing resources and link resources (SFCM-FOCL) fairly. Therefore, we first model the SFCM-FOCL formulation and then design an algorithm based on Game Theory to minimize the total cost.

Organization of this paper is as follows. The related work is discussed in Section II. We model the formulation for the SFCs deployment problem in Section III. In Section IV, we present our heuristic algorithms. In Section V, we evaluate and analyze simulation results. In Section VI, we conclude this paper.

II. RELATED WORK

A. SFCs DEPLOYMENT IN FEDERATED CLOUD

In the traditional mobile networks, Network Function (NF) is implemented as some physical proprietary devices and equipment. In order to solve the problems of high CAPEX and OPEX, the researcher has presented the NFV technology that devotes to migrate the packet processing from the hardware middle box to the software middle box. In NFV, multiple VNFs are typically in a particular order connected to compose service function chain which provides various network services [23]–[25]. With the emerging NFV technology, there are a large number of researches on VNFs or SFCs deployment in the federated cloud [26]–[34].

In order to ensure the reliability and reduce the operation cost, Cohen *et al.* [26] studied the VNF deployment problem, and proposed near optimal approximation algorithms. Although the near optimal approximation algorithms can ensure the reliability and reduce the operation cost, simulation results cannot reflect the deploying success rate of VNFs. In [27], although the authors studied the problem of joint VNF deployment and path selection to better utilize links and servers, it also has room for improvement. To optimize the deployment of VNFs to reduce network operational costs, improve utilization, and without violating SLAs, the authors in [28] proposed a heuristic based on dynamic programming, but the heuristic mainly optimizes the utilization of servers and ignores the cost of the whole path. To minimize the cost of services, the research [29] researched the elastic VNF deployment problem and proposed the SLFL algorithm to optimize operational costs. From the simulation results, the SLFL algorithm can reduce more cost of services than the random algorithm, the SLFL algorithm does not compare to existing algorithms. Thus, it cannot reflect the real performance.

Kim *et al.* [30] presented a SFC deployment algorithm to ensure QoS from the perspective of service providers, the algorithm mainly ensures QoS, but does not compare with the existing algorithms, and it cannot reflect the real performance. In order to address network attacks, Park *et al.* [31] presented QoSE, a security solution that offers adaptive security service based on the NFV. QoSE is also a novel resource optimization algorithm to operate security services efficiently, but it does not consider the deployment cost of VNFs.

The research [32] proposed a system to design and implement NFV-RT, and the system can dynamically allocate resources and provide timing guarantees, but it also ignores the deployment cost of VNFs. Qu *et al.* [33] considered the overall of VNF to minimize the overall latency of VNFs scheduling. Authors also formulated the problem of joint traffic steering and VNF scheduling as a MILP, and presented an efficient heuristic algorithm. The algorithm mainly optimizes the scheduling time, but it does not focus on the performance of other aspects. In [34], to solve the problem of multi-resource packet scheduling, the authors designed a packet scheduling algorithm in space-efficient and low-complexity. The algorithm can schedule VNFs in space-efficient and low-complexity to reduce the queuing time and the processing time, but it ignores the deployment of VNFs.

Moreover, these researches [26]–[34] mentioned above aim at the VNF or SFC placement problem, and the NFVs or SFC placement algorithms are proposed for the virtual network or the federated cloud. However, 5G mobile network is composed by the FRANs and the core network (e.g., the federated cloud). Therefore, these researches are not suitable for 5G network.

B. SFCs DEPLOYMENT IN 5G MOBILE NETWORK

To meet the increasing demands of mobile users, researchers have launched researches on the 5G mobile networks. As one of the key technologies of 5G, NFV has become an important direction for the radio network architecture evolution. As an emerging technology, NFV has received extensive attention from industry, academia, and standardization bodies. For service providers, it's crucial important for efficient deploying/mapping SFCs into 5G mobile network, and some recent works have tried to solve the problem of VNF deployment in 5G network [35]–[43].

Although the algorithms proposed in [35] are relevant for the VNFs or SFC placement problem in 5G mobile network, they only considered the VNFs deployment problem of the radio access network (such as, PGW and SGW), however, they did not consider the VNFs deployment problem of the core network or the datacenter network (such as, WAN optimizers, content filters and firewalls). In contrast, the researchers [36] mainly considered the VNFs placement problem of the core network or the datacenter network, however, the authors ignored the virtualization and placement of the PGW and the SGW. To efficiently determine the key set of physical or logical nodes, the researchers [37] proposed specific algorithms and the first framework for reliability to evaluate the NFV deployment, although the algorithm considered the VNFs placement problem of the complete SFC, the proposed algorithm is for to minimize the total failure removal.

In [38], to reduce the traffic load of transport network, the authors presented a model to solve the VNFs deployment problem. Liang *et al.* [39] presented an information-centric radio network virtualization architecture for 5G mobile radio networks, and formulated the in-network caching strategy and the network resource allocation as an optimization problem to maximize the utility function of mobile network operations. Martini *et al.* [40] formulated the node selection problem of composing, computing and VNFs to minimize the overall latency. These researches [38]–[40] only proposed models or formulations to solve the VNFs deployment problem, however, they did not present the corresponding heuristic algorithm.

To explore the potential of NFV in enhancing the 5G radio access networks' function and minimize the capital expenditure, the researchers [41] discussed how NFV can address critical design challenges in 5G network through service abstraction and virtualized computing, storage, and network resources. In the research [42], the authors presented a new flexible 5G mobile network architecture, then discussed implementation reference architecture and some cases about the typical 5G network deployment. To address those challenges of 5G networks, the researchers [43] proposed an architecture vision and a two-layer architecture that consists of a RAN and a network cloud. Abdelwahab *et al.* [41], Yang *et al.* [42], and Agyapong *et al.* [43] mainly studied the framework of NFV in 5G mobile network, they also didn't present the corresponding heuristic algorithm. Moreover, these researches [35]–[43] do not consider taking the advantage of the fog radio access networks to deploy SFC in 5G mobile network. Although these researches [18]–[20] discuss about the fog-based radio access network and deploy NFV in 5G, they only do some discussion of the problem or put forward some overall architecture.

At present, the research on NFV in 5G mobile network has just begun, therefore, it is necessary for us to propose the efficient formulations and heuristic algorithms for placing/deploying SFCs into 5G mobile network.

III. PROBLEM DEFINITION AND FORMULATION

A. PROBLEM STATEMENT

In this paper, we research the SFCs deployment problem in 5G mobile network. We consider a scenario in which there are dynamically arrived SFCs need to be placed/deployed. Specifically, given a SFC request and the locations of the mobile network user and the service terminal, and

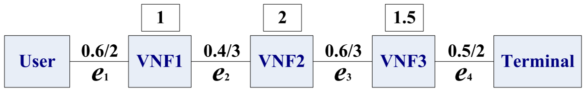

FIGURE 1. A SFC request.

given 5G mobile network composed by multiple fog radio access networks interconnected by a core network and there are multiple datacenters interconnect to the core network, the problem is how to efficiently deploy/map the dynamically arrived SFCs, such that the total SFCs mapping cost, the total VNFs mapping cost, the total links mapping cost and the blocking ratio are minimized, while satisfying all placement constraints.

B. SFC REQUEST

A SFC request can be modeled as an undirected weighted graph $G_V = (N_V, E_V)$, and $N_V = \{VNF_1, VNF_2, \ldots, VNF_n\}$ represents the set of VNFs, and n denotes the number of VNFs.$E_V = \{e_1, e_2, \ldots, e_{|E_V|}\}$ denotes the set of virtual links in the SFC request, $|E_V|$ represents the number of virtual links. We use $PC = (C_N, C_E, C_D, L_N, L_U, L_T)$ to represent the placement constraints, where $C_N = \{\varepsilon(VNF_1), \varepsilon(VNF_2), \ldots, \varepsilon(VNF_n)\}$ denotes the set of computing resource demands of all VNFs, the computing resources represent the overall demand of the server resources including CPU, memory, storage, cache and I/O resources etc. $C_E = \{x_1, x_2, \ldots, x_{|E_V|}\}$ represents the set of the requirements for bandwidth resources of all virtual links, and x_i indicates the requirements for bandwidth resources of the virtual link e_i, $e_i \in E_V$. We define C_D as the tolerable transmission delay of substrate paths for hosting the virtual links. We use L_N to indicate the location constraints of VNFs. L_U and L_T respectively represent the locations of the mobile network user and the service terminal. Fig.1 shows an example of the SFC placement request. In the example, the numbers above the VNFs denote the computing resource requirements, and numbers above virtual links denote the requirements for the bandwidth resource and the tolerable transmission delay. In this paper, we respectively use the VNFs VNF_1 and VNF_2 to represent the virtual Serving Gateway and the virtual Packet Data Network Gateway, which should be deployed into the corresponding fog radio access network, and other VNFs belong to the cloud network.

C. SUBSTRATE NETWORK

In this paper, the substrate network is 5G network, since we study the SFCs deployment problem in 5G mobile network based on the fog radio access network (FRANs). The 5G mobile network is composed by multiple fog radio access networks, the core network and multiple datacenters, Fig.2 shows an example of the substrate network. Similarly, a substrate network can be modeled as an undirected weighted graph $G^S = (N^S, E^S)$, and N^S represents the set of physical servers and routers, E^S denotes the set of substrate links. We use $SC = (C^E, C^N, L^N)$ to represent the constraints of the

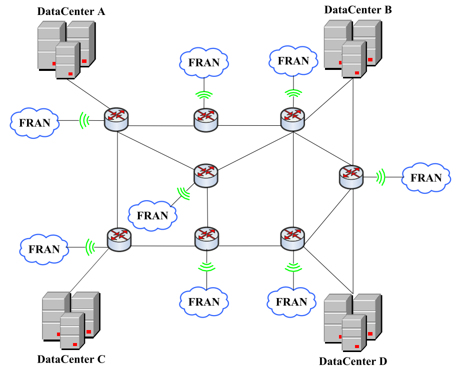

FIGURE 2. An example of substrate network.

substrate network resource, where C^E denotes the attributes of substrate links (such as the delay $d(e_s)$, the bandwidth capacity $c(e_s)$ and the per unit cost of link resources $p(e_s)$). C^N denotes the attributes of the substrate servers and routers (such as the capacity of server resources $c(n_s)$, the per unit cost of server resource $p(n_s)$ and node type that is a server or a router). We use L^N to denote the locations of physical servers.

D. STRATEGIES FOR SFC DEPLOYMENT

In the mapping process for a SFC request, in order to efficient mapping/deploying the SFC request into 5G mobile network, we introduce three efficient strategies to minimize the costs of computing resources and link resources and thus improve the acceptance ratio of the SFC requests.

1) VMs REUSING STRATEGY

Mehraghdam *et al.* [24] presented the conception of sharing and reusing network functions. In this paper, to improve the resource utilization of servers, we introduce the VMs reusing strategy, i.e., when map the VNFs, we can reuse the existing VMs for running the same VNF. The VMs reusing strategy not only can improve the resource utilization of servers, but also can improve the acceptance ratio of the SFC requests when the server resources are limited. To improve the server's resource utilization and minimize the cost of computing resources, when we map a VNF into a server, we first try to map the VNF into an existing VM that running the same VNF on the server; when there isn't an available existing VM, we try to map the VNF into a new VM on the server. When the VNF reuses an existing VM, in the original/intrinsic service time of the existing VM, we have not to consider the cost of computing resources, i.e., the cost of computing resources is zero, but when the service time of the VNF exceeds the original service time of the existing VM, we have to compute the cost of computing resources in the extra time, and we use T_i^p to present the running time for the i-th VNF that needs to pay. When the VNF uses a new VM, we have to compute the cost of computing resources in the whole service time of the VNF. The calculation process of the cost of computing resources of the i-th VNF can be described as follows:

$$Cost(VNF_i \rightarrow n_k) = p(n_k) \times \varepsilon(VNF_i) \times T_i^p,$$

$$T_i^p = \begin{cases} T_i, & when\ \pi_i = 1, \forall VNF_i \in N_V \\ \max\{T_i - T_o, 0\}, & when\ \pi_i > 1, \forall VNF_i \in N_V, \end{cases}$$

where T_i^p represents the running time for the i-th VNF that needs to pay, T_i denotes the service time of the i-th VNF, T_o represents the original service time of the existing VM that is reused, n_k denotes a server, and π_i represents the number of VNFs hosting on the VM which hosting the i-th VNF; when $\pi_i = 1$ denotes that the VM is now only used by one VNF, when $\pi_i > 1$ denotes that the VM is now reused by multiple VNFs. Because in practice, these VMs for hosting VNFs are usually not fully loaded, so these VMs have some idle periods, and resources are idle in these idle periods, the VMs reusing strategy provides performance isolation and achieves shared and reused VMs running the same VNF by using different time periods to process different VNF packets. But reusing the existing VM will have a certain impact on the performance of other VNFs, so we have to limit the number of VNFs hosting on the existing VM. In practice, the service provider can set the number of the reused VM according to the realistic load of each VM in the network. We define δ as the maximum number of VNFs hosting on a VM, i.e.:

$$\pi_i \leq \delta, \quad \forall VNF_i \in N_V.$$

2) VNFs COMBINATION STRATEGY

When we map the i-th VNF, we map the i-th VNF into the server hosting the (i-1)-th VNF as much as possible if the server has enough available resources, we call the strategy as the VNFs combination strategy. In this paper, we do not merge the VNFs of the SFC request before mapping, rather than in the mapping process, consider to map the i-th VNF into the same server hosting the (i-1)-th VNF when there are enough available resources. In our VNFs combination strategy, the i-th VNF only can be mapped into the server hosting the (i-1)-th VNF, but not can be mapped into other servers hosting other VNFs to avoid the Ping-pong routing problem. An example of VNFs combination strategy is shown in Fig.3. In the example, the 1st VNF is mapped into the physical node B, and the 2nd VNF is mapped into the physical node F. When we map the 3rd VNF, we map it into the physical node F if the physical node F has enough available resources, i.e., we allow to combine the 2nd VNF and the 3rd VNF together, this does not need consume the link resources due to the 2nd VNF communicate with the 3rd VNF in the internal of physical node F. But we cannot map the 3rd VNF into the physical node B, because this will result in a Ping-pong routing that is not expected.

3) TEMPORARY LINK MAPPING STRATEGY

When we map the i-th VNF, meanwhile, we need map the i-th link e_i connecting the i-th VNF and the (i-1)-th VNF to guarantee the optimal mapping solution of the i-th VNF. The traditional method is committed to such a goal to find the local optimal mapping solution of the i-th VNF, but this can't guarantee the optimal path for the entire SFC. To guarantee the optimal path for the entire SFC and improve the acceptance ratio of the SFC request, we generate a temporary link te_i to connect the i-th VNF and the service terminal, the bandwidth requirement of the temporary link is equal to the bandwidth requirement of the (i+1)-th link, and map the temporary link when we map the i-th VNF and the i-th link, and find an optimal mapping solution for the i-th VNF by using the temporary link mapping strategy. In the temporary link mapping strategy, the temporary link does not need to consume the actual link resources, it is only used to constraint that the i-th VNF does not to deviate too far from the service terminal, and to guarantee that the physical node hosting the i-th VNF has enough link resources to map the next link, thereby to improve the acceptance ratio of the SFC request. The cost of link resources of the i-th link can be calculated as follows:

$$Cost(p_{e_i}) = (\sum_{e_s \in p_{e_i}} p(e_s)x_i + \sum_{e_s \in p_{te_i}} p(e_s)x_{i+1}) \times T_i,$$

where e_i denotes the i-th link, p_{ei} denotes the mapping path for the i-th link, e_s represents a physical link, te_i denotes the temporary link,p_{tei} denotes the mapping path for the temporary link.

For example, in Fig.4, we give an example for mapping a SFC without considering the VNFs combination strategy. In the figure, the red short dashed line represents the mapping path for the virtual link in the SFC request, the blue dotted line denotes the mapping path for the temporary link. The Fig.4 (a) gives a possible mapping result for the traditional method, we can see that the mapping solution of the 1st VNF is local optimal when we do not consider the path from the 1st VNF to the service terminal, and the mapping solution of the 2nd VNF is local optimal when we do not consider the path from the 2nd VNF to the service terminal, but it results that the 2nd VNF deviates too far from the service terminal. The Fig.4 (b) and (c) give possible mapping results for the our method, from the mapping results, we can see that temporary link mapping strategy can bring that each VNF is

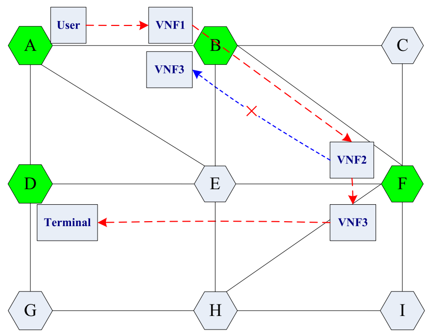

FIGURE 3. An example of VNFs combination.

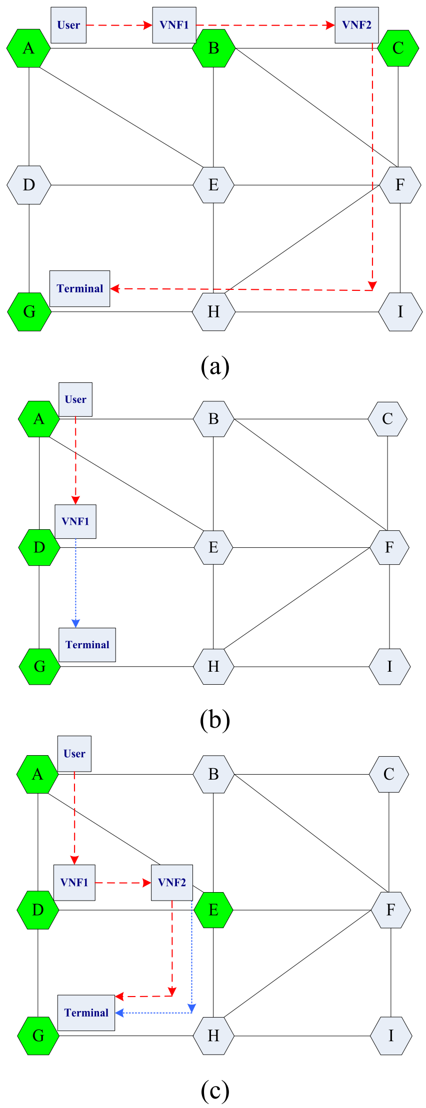

FIGURE 4. An example of temporary link mapping strategy.

not far away from the service terminal, and results a shorter path for the entire SFC than the traditional method does.

E. INTEGER LINEAR PROGRAMMING FORMULATIONS

In the placement process of SFC, we first place and allocate resources for VNFs of the SFC request, and then map and allocate bandwidth resources for virtual links of the SFC request. The SFC mapping procedure can be formulated as follows.

1) VNFs MAPPING

The placement process of VNFs can be formulated as:

$$
\begin{aligned}
M_N : (N_V, C_N) &\xrightarrow{M_N} (N^{S1}, C^{N1}),\\
M(VNF_i) &\in N^{S1}, \quad \forall VNF_i \in N_V,\\
R(M(VNF_i)) &\geq \varepsilon(VNF_i), \quad \forall VNF_i \in N_V,\ when\ \pi_i = 1,\\
\varepsilon(VM_i) &\geq \varepsilon(VNF_i), \quad \forall VNF_i \in N_V,\ when\ \pi_i > 1,\\
Z^y_{VNF_i} &\in \{0, 1\}, \quad \forall VNF_i \in N_V,\ \forall y \in \{0, 1, \ldots, Y\},\\
L(M(VNF_i)) &\in \{0, 1, 2, \ldots, Y\}, \quad \forall VNF_i \in N_V,\\
Z^{L(M(VNF_i))}_{VNF_i} &= 1, \quad \forall VNF_i \in N_V,
\end{aligned}
$$

where $N^{S1} \subset N^S$, C^{N1} denotes the server resources allocated to VNFs of the SFC request, $M_N = \{M(VNF_1), M(VNF_2), \ldots, M(VNF_n)\}$represents the mapping records of VNFs in the SFC requests. $M(VNF_i)$ denotes a server hosting VNFVNF_i, $R(M(VNF_i))$ represents the server's available resources. VM_i represents the existing VM for hosting the VNF VNF_i, and $\varepsilon(VM_i)$ denotes the computing resource of the existing VM. $y \in \{0, 1, 2, \ldots, Y\}$ denotes the network area number, $L(M(VNF_i))$ represents the network area number that the server $M(VNF_i)$ located, and a server only can belong to a network area, Z^y_{VNFi} =1 represents VNF_i can be placed in the network area, and Z^y_{VNFi}=0 represents VNF_i cannot be placed in the network area. $Z^{L(M(VNF_i))}_{VNF_i} = 1$ denotes that the server $M(VNF_i)$ meets the location constraint of the VNF_i. In 5G mobile network, SGW and PGW belong to the fog radio access network functions, they are usually only deployed in the fog radio access network, and Firewall, IDS and Proxy belong to the core or datacenter network functions, they are usually only deployed in the core or datacenter network.

2) SFC LINK MAPPING

The mapping for SFC links can be formulated as:

$$
\begin{aligned}
M_E : (E_V, C_E) &\xrightarrow{M_E} (P^1, C^{E1}),\\
M(e_i) &= p_{e_i}, \quad \forall e_i \in E_V, \exists p_{e_i} \in P^1,\\
B(p_{e_i}) &= \min_{e_s \in p_{e_i}} \{b(e_s)\} \geq x_i, \quad \forall p_{e_i} \in P^1,\\
D(p_{e_i}) &= \sum_{e_s \in p_{e_i}} d(e_s) \leq C_D, \quad \forall p_{e_i} \in P^1,
\end{aligned}
$$

where $M_E = \{M(e_1), M(e_2), \ldots, M(e_{|Ev|})\}$ denotes the mapping records of virtual links of the SFC request. $P^1 \subset P$, where P indicates the set of substrate paths. C^{E1} indicates the allocated bandwidth resources. $M(e_i)$ represents a substrate path for hosting virtual link e_i. $B(p_{ei})$ indicates the available bandwidth resources of substrate path p_{ei}, $D(p_{ei})$ denotes the delay of p_{ei}.

Accordingly, the problem of deploying SFCs in 5G mobile network, such that *i*) the cost of link resource is minimized; *ii*) the cost of computing resources is minimized, can be formulated the linear programming (1) as follows.

The first objective aims at minimizing the cost of computing resources as much as possible. The second objective aims at minimizing the cost of link resource, i.e., shortening the path for the entire SFC as much as possible. At the same time, the constraints in linear programming (1) are used to ensure the following conditions:

Constraint 1 ensures that the number of VNFs hosting on a VM don't exceed the number specified by the service providers.

Constraint 2 gives it has to pay for running the VNFs.

Constraints 3 and 4 ensure that the server being used satisfies the computing resource demands of VNFs.

Constraint 5 ensures that the physical link being used satisfies the bandwidth resource requirements of virtual link.

Constraint 6 ensures that the physical link being used satisfies the delay constraint of virtual link.

Constraints 7, 8 and 9 ensure that the server being used satisfies the location constraint of VNFs.

$$
\begin{aligned}
&\min \sum_{VNF_i \in N_V} P(M(VNF_i))\varepsilon(VNF_i)T_i^p \\
&\min \sum_{e_i \in E_V} \Big(\sum_{e_s \in p_{e_i}} P(e_s)x_i + \sum_{e_s \in p_{te_i}} p(e_s)x_{i+1} \Big) T_i \\
&s.t.\ \pi_i \le \delta, \quad \forall VNF_i \in N_V \\
&\quad T_i^p = \begin{cases} T_i, & \text{when } \pi_i = 1, \forall VNF_i \in N_V \\ \max\{T_i - T_o, 0\}, & \text{when } \pi_i > 1, \forall VNF_i \in N_V \end{cases} \\
&\quad R(M(VNF_i)) \ge \varepsilon(VNF_i), \quad \forall VNF_i \in N_V, \text{when } \pi_i = 1 \\
&\quad \varepsilon(VM_i) \ge \varepsilon(VNF_i), \quad \forall VNF_i \in N_V, \text{ when } \pi_i > 1 \\
&\quad B(p_{e_i}) = \min_{e_s \in p_{e_i}} \{b(e_s)\} \ge x_i, \quad \forall e_i \in E_V \\
&\quad D(p_{e_i}) = \min_{e_s \in p_{e_i}} \{d(e_s)\} \le C_D, \forall e_i \in E_V \\
&\quad Z_{VNF_i}^y \in \{0, 1\}, \quad \forall VNF_i \in N_V, \forall y \in \{0, 1, \ldots, Y\} \\
&\quad L(M(VNF_i)) \in \{0, 1, 2, \ldots, Y\}, \quad \forall VNF_i \in N_V \\
&\quad Z_{VNF_i}^{L(M(VNF_i))} = 1, \quad \forall VNF_i \in N_V
\end{aligned} \tag{1}
$$

We propose three solutions to solve the multi-objectives problem (1). Minimizing the cost of computing resources needs to reuse these existing VMs, but these existing VMs are usually not on the shortest path that will increase the probability of having a longer path for the entire SFC. On the contrary, shortening the path for the entire SFC usually needs to abandon the use of these existing VMs. So the two objectives of minimizing the cost of computing resources and minimizing the cost of link resource are two conflicting goals. For three different scenarios, we propose three different solutions. The first solution is proposed for minimizing the cost of computing resources. The second solution is presented for shortening the path for the entire SFC. The third solution is proposed for VNF allocation and routing to find a fair solution for the costs of computing resources and link resources by using Game of Battle of Sex (BoS) model.

(1) ***MC: Minimizing the cost of Computing resources***

In this solution, we define $LCost_i^{max}$ as the maximum mapping cost for each virtual link in the SFC that can be tolerated by the service provider. The $LCost_i^{max}$ is given by the service provider, and it is usually less than the charges of the service provider for each virtual link in the SFC. The optimization model aims at reducing the cost of computing resources which can be formulated by using the linear programming (2).

$$
\begin{aligned}
&\min \sum_{VNF_i \in N_V} P(M(VNF_i))\varepsilon(VNF_i)T_i^p \\
&s.t.\ \Big(\sum_{e_s \in p_{e_i}} P(e_s)x_i + \sum_{e_s \in p_{te_i}} p(e_s)x_{i+1}\Big)T_i \le LCost_i^{\max}, \quad \forall e_i \in E_V \\
&\quad \pi_i \le \delta, \forall VNF_i \in N_V \\
&\quad T_i^p = \begin{cases} T_i, & \text{when } \pi_i = 1, \forall VNF_i \in N_V \\ \max\{T_i - T_o, 0\}, & \text{when } \pi_i > 1, \forall VNF_i \in N_V \end{cases} \\
&\quad R(M(VNF_i)) \ge \varepsilon(VNF_i), \ \forall VNF_i \in N_V, \text{when } \pi_i = 1 \\
&\quad \varepsilon(VM_i) \ge \varepsilon(VNF_i), \quad \forall VNF_i \in N_V, \text{when } \pi_i > 1 \\
&\quad B(p_{e_i}) = \min_{e_s \in p_{e_i}} \{b(e_s)\} \ge x_i, \quad \forall e_i \in E_V \\
&\quad D(p_{e_i}) = \min_{e_s \in p_{e_i}} \{d(e_s)\} \le C_D, \quad \forall e_i \in E_V \\
&\quad Z_{VNF_i}^y \in \{0, 1\}, \quad \forall VNF_i \in N_V, \ \forall y \in \{0, 1, \ldots, Y\} \\
&\quad L(M(VNF_i)) \in \{0, 1, 2, \ldots, Y\}, \quad \forall VNF_i \in N_V \\
&\quad Z_{VNF_i}^{L(M(VNF_i))} = 1, \quad \forall VNF_i \in N_V
\end{aligned} \tag{2}
$$

(2) ***ML: Minimizing the cost of Link resources***

$$
\begin{aligned}
&\min \sum_{e_i \in E_V} \Big(\sum_{e_s \in p_{e_i}} P(e_s)x_i + \sum_{e_s \in p_{te_i}} p(e_s)x_{i+1} \Big) T_i \\
&s.t.\ P(M(VNF_i))\varepsilon(VNF_i)T_i^p \le SCost_i^{\max}, \quad \forall VNF_i \in N_V \\
&\quad \pi_i \le \delta, \quad \forall VNF_i \in N_V \\
&\quad T_i^p = \begin{cases} T_i, & \text{when } \pi_i = 1, \forall VNF_i \in N_V \\ \max\{T_i - T_o, 0\}, & \text{when } \pi_i > 1, \forall VNF_i \in N_V \end{cases} \\
&\quad R(M(VNF_i)) \ge \varepsilon(VNF_i), \quad \forall VNF_i \in N_V, \text{when } \pi_i = 1 \\
&\quad \varepsilon(VM_i) \ge \varepsilon(VNF_i), \quad \forall VNF_i \in N_V, \text{when } \pi_i > 1 \\
&\quad B(p_{e_i}) = \min_{e_s \in p_{e_i}} \{b(e_s)\} \ge x_i, \quad \forall e_i \in E_V \\
&\quad D(p_{e_i}) = \min_{e_s \in p_{e_i}} \{d(e_s)\} \le C_D, \quad \forall e_i \in E_V \\
&\quad Z_{VNF_i}^y \in \{0, 1\}, \quad \forall VNF_i \in N_V, \ \forall y \in \{0, 1, \ldots, Y\} \\
&\quad L(M(VNF_i)) \in \{0, 1, 2, \ldots, Y\}, \quad \forall VNF_i \in N_V \\
&\quad Z_{VNF_i}^{L(M(VNF_i))} = 1, \quad \forall VNF_i \in N_V
\end{aligned} \tag{3}
$$

In this solution, we define $SCost_i^{max}$ as the maximum mapping cost for each VNF in the SFC that can be tolerated by the service provider. The $SCost_i^{max}$ is given by the service provider, and it is usually less than the charges of the service provider for each VNF in the SFC. The optimization model aims at shortening the path for the entire SFC which be formulated as the following linear programming (3).

(3) ***FOCL: Fair Optimizing the costs of Computing resources and Link resources by using BoS model***

BoS model depicts such a game situation: in the game, the two players have some common interests, but with different results of the common interests and have relative conflicting preferences. For example, the couple prefers to see their respective favorite programs, but the couple would prefer to watch the same TV program together, and do not want to separate to see their respective TV programs. Similarly, we can model the costs of computing resources and link

TABLE 1. The strategies of this game.

1-P \ 2-P	1-S	2-S
1-S	$SCost_i^{\max}-Cost(M^E(VNF_i))$, $LCost_i^{\max}-Cost(p_{e_i}^E)$	0,0
2-S	0,0	$SCost_i^{\max}-Cost(M^N(VNF_i))$, $LCost_i^{\max}-Cost(p_{e_i}^N)$

resources as the two players of the BoS model. This model is based on two strategies: *i*) reusing an existing VM when map a VNF, *ii*) using a new VM when map a VNF. The two strategies are given and known to the players. An existing VM is reused to map the VNF, it can reduce the cost of computing resources, but it may lead to a longer path for the SFC. A new VM is used to map the VNF, it is more likely to find an optimal path for the SFC, but it may lead to a higher cost of computing resources. So the mapping process of each VNF is a game process, the two players of the costs of computing resources and link resources carry on the game to decide whether to reuse an existing VM. We refer the strategy of reusing an existing VM as the first strategy (i.e., 1-S), the strategy of using a new VM as the second strategy (i.e., 2-S), the cost of computing resources as the first player (i.e., 1-P) and the cost of link resources as the second player (i.e., 2-P). The strategies of this game are expressed as in Table 1.

$SCost_i^{max}$–$Cost(M^E(VNF_i))$: the revenue of computing resources when using an existing VM to map VNF_i;

$LCost_i^{max}$–$Cost(p_{ei}^E)$: the revenue of link resources when using an existing VM to map VNF_i;

$SCost_i^{max}$–$Cost(M^N(VNF_i))$: the revenue of computing resources when using a new VM to map VNF_i;

$LCost_i^{max}$–$Cost(p_{ei}^N)$: the revenue of link resources when using a new VM to map VNF_i;

$M^E(VNF_i)$: the mapping solution of VNF_i when using an existing VM;

$M^N(VNF_i)$: the mapping solution of VNF_i when using a new VM;

p_{ei}^E: the mapping path of link e_i when using an existing VM;

p_{ei}^N: the mapping path of link e_i when using a new VM;

p_{tei}^E: the mapping path of temporary link te_i when using an existing VM;

p_{tei}^N: the mapping path of temporary link te_i when using a new VM;

$Cost(M^E(VNF_i))$, $Cost(M^N(VNF_i))$, $Cost(p_{ei}^E)$ and $Cost(p_{ei}^N)$ can be computed as follow:

$$Cost\left(M^{(\bullet)}(VNF_i)\right) = P(M^{(\bullet)}(VNF_i))\varepsilon(VNF_i)T_i^p,$$
$$Cost(p_{ei}^{(\bullet)}) = (\sum_{e_s\in p_{e_i}^{(\bullet)}} p(e_s)x_i + \sum_{e_s\in p_{te_i}^{(\bullet)}} p(e_s)x_{i+1}) \times T_i.$$

In BoS model, there are two pure-strategy Nash equilibrium points, i.e., ($SCost_i^{max}$–$Cost(M^E(VNF_i))$, $LCost_i^{max}$–$Cost(p_{ei}^E)$) and ($SCost_i^{max}$–$Cost(M^N(VNF_i))$, $LCost_i^{max}$–$Cost(p_{ei}^N)$). To obtain the optimal mapping solution for VNF_i, we select the pure-strategy Nash equilibrium point with the highest overall revenue as the focus equilibrium point. The focus equilibrium point is our mapping solution for VNF_i. The mapping process of each VNF is a repeated game process. The optimization model which fair optimizing the costs of computing resources and link resources can be formulated through the following linear programming (4).

$$\begin{aligned}
&\sum_{VNF_i\in N_V} \max\{(SCost_i^{\max} - P(M^E(VNF_i))\varepsilon(VNF_i)T_i^p)\\
&\quad + (LCost_i^{\max} - (\sum_{e_s\in p_{ei}^E} p(e_s)x_i\\
&\quad + \sum_{e_s\in p_{te_i}^E} p(e_s)x_{i+1})T_i),\\
&\quad \times (SCost_i^{\max} - P(M^N(VNF_i))\varepsilon(VNF_i)T_i)\\
&\quad + (LCost_i^{\max} - (\sum_{e_s\in p_{ei}^N} p(e_s)x_i + \sum_{e_s\in p_{te_i}^N} p(e_s)x_{i+1})T_i\}\\
&\quad s.t.\ \pi_i \le \delta, \quad \forall VNF_i \in N_V\\
&\qquad T_i^p = \begin{cases} T_i, & when\ \pi_i = 1, \\ & \forall VNF_i \in N_V \\ \max\{T_i - T_o, 0\}, & when\ \pi_i > 1, \\ & \forall VNF_i \in N_V \end{cases}\\
&\qquad R(M(VNF_i)) \ge \varepsilon(VNF_i),\\
&\qquad \forall VNF_i \in N_V, when\ \pi_i = 1\\
&\qquad \varepsilon(VM_i) \ge \varepsilon(VNF_i),\\
&\qquad \forall VNF_i \in N_V, when\ \pi_i > 1\\
&\qquad B(p_{e_i}) = \min_{e_s\in p_{e_i}} \{b(e_s)\} \ge x_i, \quad \forall e_i \in E_V\\
&\qquad D(p_{e_i}) = \min_{e_s\in p_{e_i}} \{d(e_s)\} \le C_D, \quad \forall e_i \in E_V\\
&\qquad Z_{VNF_i}^y \in \{0,1\}, \quad \forall VNF_i \in N_V, \forall y \in \{0,1,\dots,Y\}\\
&\qquad L(M(VNF_i)) \in \{0,1,2,\dots,Y\}, \quad \forall VNF_i \in N_V\\
&\qquad Z_{VNF_i}^{L(M(VNF_i))} = 1, \quad \forall VNF_i \in N_V \qquad (4)
\end{aligned}$$

In summary, the *MC* formulation, the *ML* formulation and the *FOCL* formulation respectively describe formulations for three different scenarios. When service providers are more concerned about the cost of computing resources, the *MC* formulation is used to achieve their target. When service providers are more concerned about the cost of link resources, the *ML* formulation is used to deliver. When service providers also are concerned about the cost of computing resources and the cost of link resources, the *FOCL*formulation is used to achieve their goal.

IV. HEURISTIC ALGORITHM

Based on the SFCM-FOCL formulation, we propose the SFCs mapping algorithm with VMs reusing, VNFs combination and temporary link mapping can be used to fairly optimize the costs of computing resources and link resources, i.e., SFCM-FOCL algorithm can online map dynamic arriving SFC requests into the substrate network. Without losing generality, the SFC requests can be derived following a Poisson process in the paper. In the SFCM-FOCL algorithm, we first store all arrived SFC requests into a queue, represented as *ArrivedSFC*. We denote the set of expired SFC requests as*ExpiredSFC*. Each SFC request of the *ArrivedSFC* queue is mapped one by one. We define SFC_{blo} as the set of blocked requests due to the lack of resources. The proposed SFCM-FOCL algorithm is as shown in *Algorithm*1.

Algorithm 1 SFCs Mapping Algorithm With VMs Reusing, VNFs Combination and Temporary Link Mapping for Fair Optimizing the Costs of Computing Resources and Link Resources (SFCM-FOCL)

Input: 1. Substrate network$G^S = (N^S, E^S)$ and resource constraints $SC = (C^E, C^N, L^N)$;
2. SFC requests queue *ArrivedSFC*.
Output: Mapping cost M^{ttotal}_{cost} and the set of blocked SFCs,SFC_{blo}.
Initialization: **let**M^{ttotal}_{cost} =0 and $SFC_{blo} = \emptyset$;
while $ArrivedSFC \neq \emptyset$, **do**
  Updating *ExpiredSFC* and substrate network resources according to *ExpiredSFC*, then let $ExpiredSFC = \emptyset$;
  Call SFCM procedure for mapping the first SFC request SFC_1 in *ArrivedSFC*;
  if found a mapping solution$M$ for SFC_1, **then**
    $M^{ttotal}_{cost} = M^{ttotal}_{cost} + M_{SFC}$, updating substrate network;
  else
    $SFC_{blo} = SFC_{blo} \cup \{SFC_1\}$;
  end if
  $ArrivedSFC = ArrivedSFC \backslash \{SFC_1\}$;
end while
return M^{ttotal}_{cost} and SFC_{blo}.

The SFCM procedure is used to map a SFC request, as shown in *Procedure*1. It is responsible for finding the mapping solution for each VNF in the SFC request as well finds the path for the SFC request and allocates resources for each VNF and each virtual link. *Procedure* 1 finds the mapping solutions for a VNF by using an existing VM and a new VM, then determines the final mapping solution of the VNF by using the BoS model. *Procedure* 1 finds the mapping solution with the minimum cost as the final mapping solution, presented as M. The solution M includes the mapping cost M_{SFC}, the mapping records of the VNFs M_N and the mapping records of the links M_E.

Procedure 1 A SFC Mapping (SFCM)

Input: 1. Substrate network$G^S = (N^S, E^S)$ and resource constraints$SC = (C^E, C^N, L^N)$;
2. A SFC request $G_V = (N_V, E_V)$ and placement constraints $PC = (C_N, C_E, C_D, L_N, L_U, L_T)$.
Output: Mapping solution M.
Store all of the available substrate servers in U^S;
for each VNF $VNF_i \in N_V$, **do**
  for each $n_k \in U^S$, **do**
  if $ZL(n_k)$ VNF_i ==1and n_k is not used by other VNFs or satisfy the VNFs combination strategy, **then**
    Try to place VNF_i into a new VM of server n_k, calculate $CostVNF^N(VNF_i \rightarrow n_k)$ according to Equation (5);
    Find minimal cost paths p^N_{ei} and p^N_{tei} by using the Dijkstra algorithm, compute and record $Cost(p^N_{ei})$ according to Equation (6); compute and record the total mapping cost $TCostVNF^N$ $(VNF_i \rightarrow n_k)$ according to Equation (9);
end if
end for
  for each $n_j \in U^S$, **do**
  if $ZL(n_j)$ VNF_i ==1 and n_j is not used by other VNFs or satisfy the VNFs combination strategy, **then**
    Try to place VNF_i into an existing VM of server n_j, calculate and record$CostVNF^E(VNF_i \rightarrow n_j)$ according to Equation (7);
    Find the minimal cost paths p^E_{ei} and p^E_{tei} by using the Dijkstra algorithm, compute and record Cost(pE ei) according to Equation (8); compute and record the total mapping cost $TCostVNF^E(VNF_i \rightarrow n_j)$ by using Equation (10);
end if
  end for
  Find the mapping solution of using a new VM for VNF_i with the minimal total mapping cost $TCostVNF^N(VNF_i \rightarrow n_k)$, and find the mapping solution of using an existing VM for VNF_i with the minimal total mapping cost$TCostVNF^E(VNF_i \rightarrow n_j)$; find the mapping solution for VNF_i with the minimal mapping cost$TCostVNF(VNF_i \rightarrow n_m)$ according to Equation (11), and stored the mapping solutions of VNF_i and SFC link e_i in M;
end for
updating the final mapping cost into M;
return M.

When we use a new VM of the server n_k to map the VNF_i, the mapping cost of the VNF_i can be calculated according to Equation (5), and the mapping cost of the e_i can be computed according to Equation (6).

$$CostVNF^N(VNF_i \rightarrow n_k) = P(n_k)\varepsilon(VNF_i)T_i \quad (5)$$

$$Cost(p^N_{e_i}) = (\sum_{e_s \in p^N_{e_i}} p(e_s)x_i + \sum_{e_s \in p^N_{te_i}} p(e_s)x_{i+1}) \times T_i \quad (6)$$

When we use an existing VM of the server n_j to map the VNF_i, the mapping cost of the VNF_i can be computed according to Equation (7), and the mapping cost of the e_i can be calculated in Equation (8).

$$CostVNF^E(VNF_i \rightarrow n_j) = P(n_j)\varepsilon(VNF_i)\max\{T_i - T_o, 0\} \quad (7)$$

$$Cost(p^E_{e_i}) = (\sum_{e_s \in p^E_{e_i}} p(e_s)x_i + \sum_{e_s \in p^E_{te_i}} p(e_s)x_{i+1}) \times T_i \quad (8)$$

When we use a new VM to map the VNF_i, the minimal total mapping cost can be calculated as in Equation (9).

$$TCostVNF^N(VNF_i \rightarrow n_k) = CostVNF^N(VNF_i \rightarrow n_k) + Cost(p^N_{e_i}) \quad (9)$$

When we use an existing VM to map the VNF_i, the minimal total mapping cost can be defined in Equation (10).

$$TCostVNF^E(VNF_i \rightarrow n_j) = CostVNF^E(VNF_i \rightarrow n_j) + Cost(p^E_{e_i}) \quad (10)$$

The minimal total mapping cost for the VNF_i can be calculated as in Equation (11).

$$TCostVNF(VNF_i \rightarrow n_m) = \min\{TCostVNF^N(VNF_i \rightarrow n_k), TCostVNF^E(VNF_i \rightarrow n_j)\} \quad (11)$$

In the following, we simply describe the SFCM-MC algorithm and the SFCM-ML algorithm.

3) SFCM-MC ALGORITHM

SFC mapping algorithm with VMs reusing, VNFs combination and temporary link mappingfor minimizing the cost of computing resources. In the SFCM-MC algorithm, for mapping the VNF_i, we only find paths for the link e_i and temporary link te_i, don't consider the cost of link resources; we find the mapping solution of the VNF_i with the minimal cost of computing resources as the final mapping solution for the VNF_i.

4) SFCM-ML ALGORITHM

SFC mapping algorithm with VMs reusing, VNFs combination and temporary link mapping for minimizing the cost of link resources. In the SFCM-ML algorithm, when mapping the VNF_i, we only find the mapping solution of the VNF_i with the minimal cost of the link e_i and temporary link te_i as the final mapping solution for the VNF_i, and don't consider the cost of computing resources for the VNF_i.

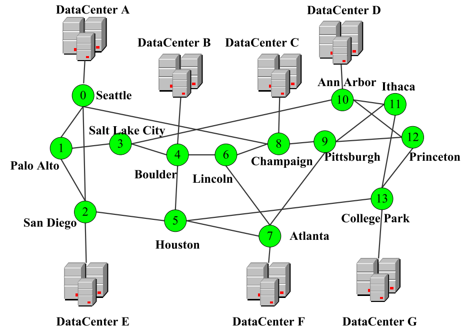

FIGURE 5. The overall topology used by our simulation.

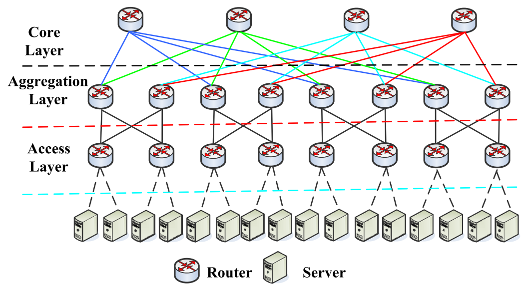

FIGURE 6. The fat tree topology.

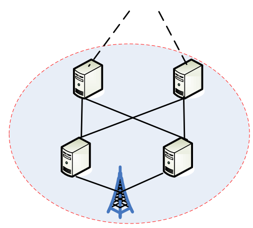

FIGURE 7. The fog radio access network topology.

V. SIMULATION RESULTS

A. SIMULATION ENVIRONMENT

In the simulations, to cope with the tremendous growth of the mobile data traffic, we consider using the resources of cloud computing to meet mobile user requirements, therefore, the 5G mobile network is composed by a core network, multiple fog radio access networks and multiple datacenters. As shown in Fig.5, the US-wide NSF network [44], [45] has been used as the core network, and seven datacenters are connected to the core network. In addition, each city in the US-wide NSF network connects a fog radio access network. Each datacenter in Fig.6 is a fat tree topology [46]. Fig.7 shows the fog radio access network.

In our simulations, we use two scenarios: the scenario of limited resource capacity and the scenario of unlimited resource capacity. In the scene of limited resource capacity, the computing resource capacity of the physical server obeys the uniform distribution U(15, 20), the bandwidth capacity of the physical link in the datacenter directly connecting a server is 30Gbps, the bandwidth capacity of the server-to-server physical link in fog radio access networks is 30Gbps and the bandwidth capacities of other physical links are 120Gbps. Without losing generality, we make assumption that: *i*) The per unit cost of computing resource follows the uniform distribution U(0.5, 1.5) and the per unit cost of bandwidth resources is both 1 unit; *ii*) The transmission delay of each core network link is 1 unit; *iii*) Other links has no transmission delay.

Additionally, we assume that these SFC requests dynamically arrive on the basis of the Poisson process, the number of VNFs of the SFC request is varied among 5, 6, 7 and 8, the computing resource requirement of each VNF in each SFC request also follows the uniform distribution U(5, 10) and the bandwidth resource requirement of each link in each SFC request also follows the uniform distribution U(5, 10). The first VNF and the second VNF in each SFC request are respectively the SGW and the PGW, and they can only be placed in the user located fog radio access network. Other VNFs including firewalls, WAN optimizers, proxies, content filters, intrusion prevention systems and intrusion detection systems should be placed in datacenters. Each virtual link's the transmission delay constraint is 4 time units.

Until now, SFC placement algorithm in 5G mobile network has very little presence. Hence, we use the heuristic SFC placement algorithm proposed in [47], i.e., SAMA. The SAMA algorithm is proposed for the virtual network or the federated cloud, to suitable for 5G mobile network, we extend and modify the SAMA algorithm so that it can map SFC request in 5G mobile network.

B. PERFORMANCE METRICS

In our simulations, the performance of our proposed algorithms can be measured by using the following metrics. We evaluate the total SFCs mapping cost, total VNFs mapping cost and total links mapping cost in the scene of unlimited resource capacity. In addition, the blocking ratio is evaluated in the scenario of limited resource capacity.

1) THE TOTAL SFCs MAPPING COST

can be computed as in Equation (12). It describes the total cost of mapping all SFC requests by using substrate network resources.

$$M_{cost}^{total} = \sum_{|ArrivedSFC|} M_{SFC}, \tag{12}$$

where M_{SFC} denotes the cost of mapping a SFC request by using substrate network resources.

2) THE TOTAL VNFs MAPPING COST

describes the total cost for mapping VNFs of all SFC requests by using substrate network resources.

$$M_{cost}^{VNF} = \sum_{|ArrivedSFC|} M_{VNF}, \tag{13}$$

where M_{VNF} represents the cost of mapping VNFs of a SFC request by using substrate network resources.

3) THE TOTAL LINKS MAPPING COST

describes the total cost of mapping links of the SFC requests by using substrate network resources.

$$M_{cost}^{Link} = \sum_{|ArrivedSFC|} M_{Link} \tag{14}$$

where M_{Link} represents the cost of mapping links of a SFC request by using substrate network resources. Note that, the total SFC mapping cost is the sum of the total VNFs mapping cost and the total links mapping cost.

4) THE BLOCKING RATIO

is the ratio of the number of blocked SFC requests to the number of total arrived SFC requests. And, it can be calculated as in Equation (15).

$$P_b = \frac{|SFC_{blo}|}{|ArrivedSFC|}, \tag{15}$$

where $|ArrivedSFC|$ and $|SFC_{blo}|$ respectively denote the numbers of total arrived SFC requests and blocked SFC requests.

C. SIMULATION RESULTS AND ANALYSIS

Fig.8 compares the total SFCs mapping costs of the SFCM-FOCL algorithm, SFCM-ML algorithm, SFCM-MC algorithm and SAMA algorithm, where the number of VNFs in the SFC request (i.e., n) is varied among 5, 6, 7 and 8. From Fig.8, we can see that the total SFC mapping costs of our three algorithms are lower than that of the SAMA algorithm. This is because that our algorithms use the VMs reusing strategy, VNFs combination strategy and temporary link mapping strategy. The VMs reusing strategy can reduce the VNFs mapping cost, and the VNFs combination strategy and temporary link mapping strategy can shorten the path for the SFC to reduce the links mapping cost. Therefore, our algorithms can get the lower mapping costs than the SAMA algorithm does. In addition, the total SFC mapping cost of the SFCM-FOCL algorithm is lower than that of our other two algorithms. This is because the SFCM-FOCL algorithm optimizes costs of computing resources and link resources, so the total SFC mapping cost is lower.

Fig.9 shows the total VNFs mapping costs of our three algorithms and the SAMA algorithm, when the number of VNFs is varied among 5, 6, 7 and 8. From Fig.9, we can see that the total VNFs mapping costs of our three algorithms are lower than that of the SAMA algorithm. This is because our algorithms use the VMs reusing strategy, and the VMs reusing strategy can reduce the VNFs mapping cost. Therefore, the total VNFs mapping costs of our algorithms are lower than that of the SAMA algorithm in mobile 5G network. In addition, the SFCM-MC algorithm minimizes the cost of computing resources, so the total VNFs mapping cost of the SFCM-MC algorithm is the lowest. The SFCM-FOCL algorithm optimizes the costs of computing resources and link resources and the SFCM-ML algorithm

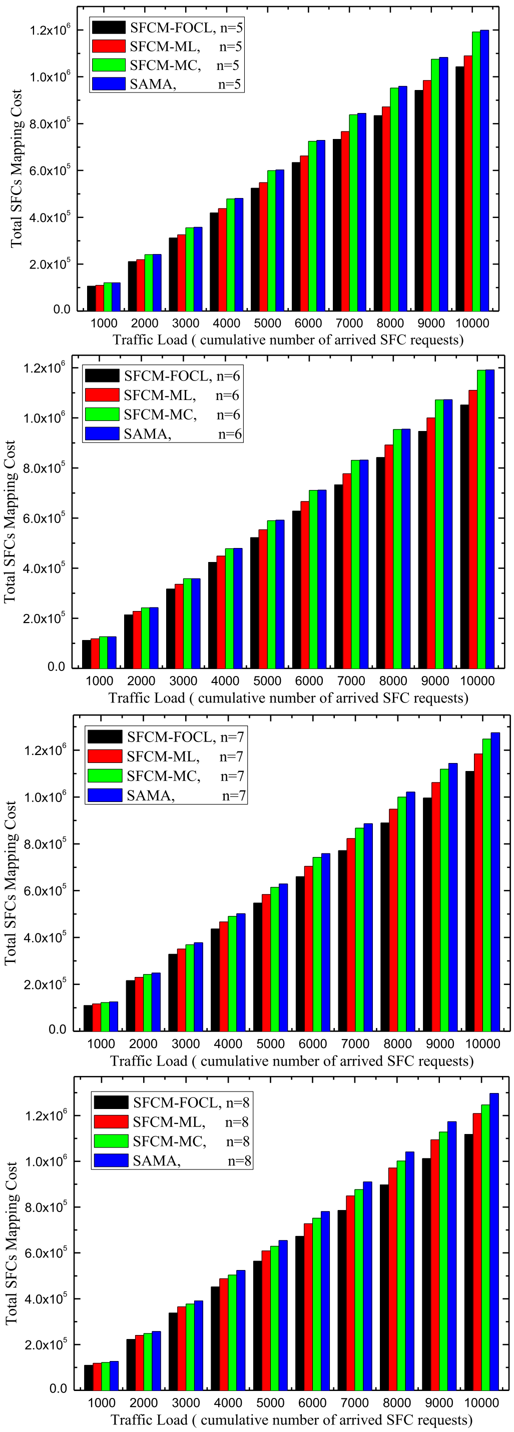

FIGURE 8. The total SFCs mapping cost.

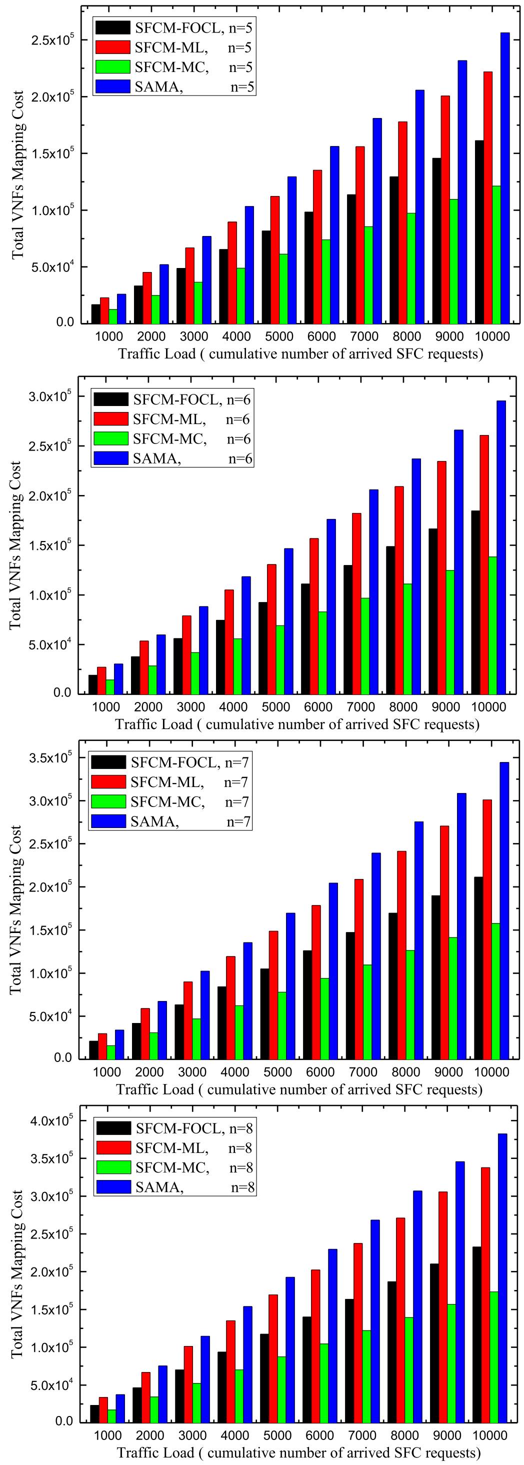

FIGURE 9. The total VNFs mapping cost.

only minimizes the cost of link resources, so the total VNFs mapping cost of the SFCM-FOCL algorithm is lower than that of the SFCM-ML algorithm.

Fig.10 compares the total links mapping costs of our three algorithms and the SAMA algorithm. Fig.10 shows that the total links mapping costs of the SFCM-ML algorithm and the SFCM-FOCL algorithm are lower than the total links mapping cost of the SAMA algorithm. This is because the SFCM-ML algorithm and the SFCM-FOCL algorithm use the temporary link mapping strategy, which can effectively shorten paths for SFC requests. Therefore, the SFCM-ML algorithm and the SFCM-FOCL algorithm can achieve lower total links mapping costs than

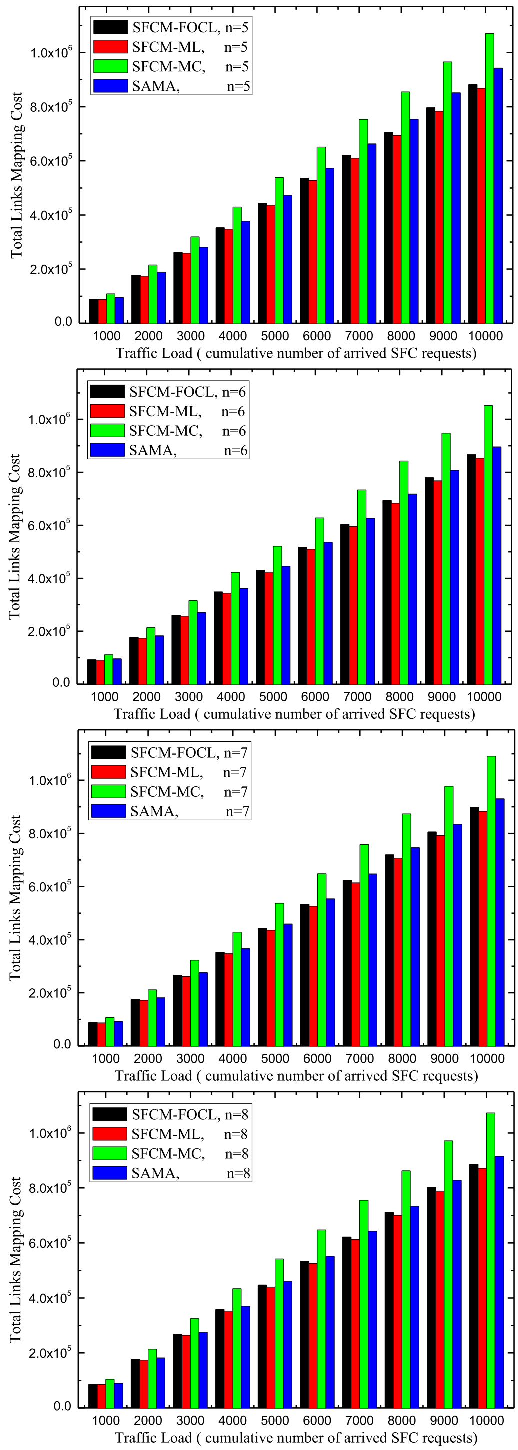

FIGURE 10. The total links mapping cost.

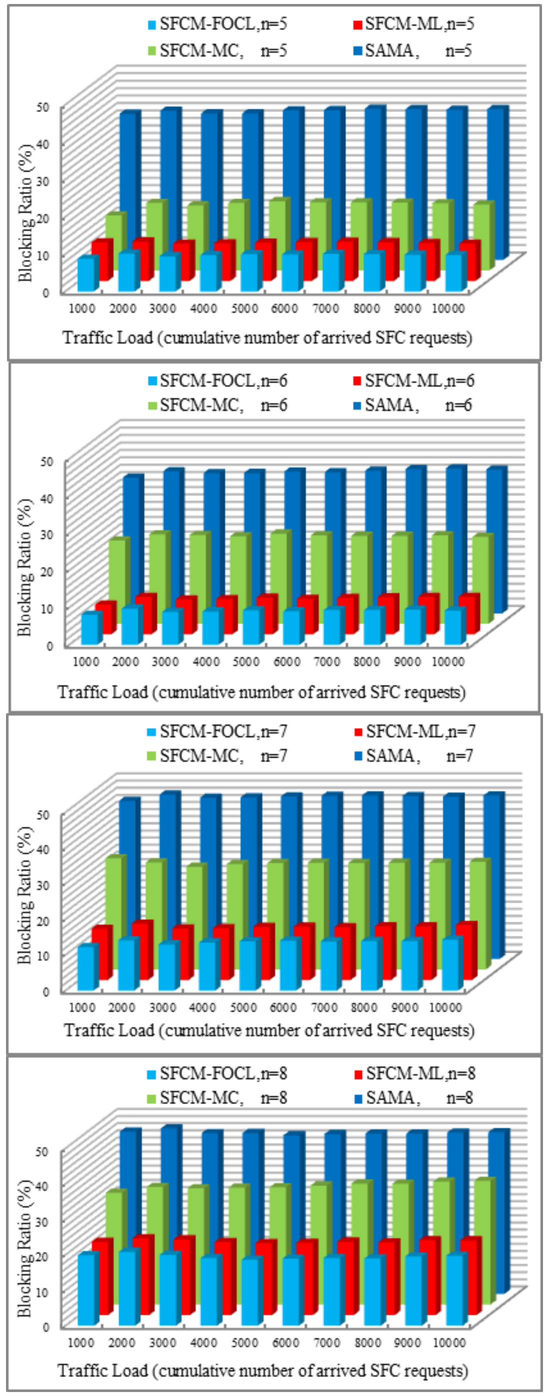

FIGURE 11. The blocking ratio.

the SAMA algorithm does. In our algorithms, due to the SFCM-ML algorithm minimizes the cost of link resources, so its the total links mapping cost is the lowest; and due to the SFCM-MC algorithm only minimizes the cost of computing resources, so the total links mapping cost of the SFCM-MC algorithm is higher than the total links mapping cost of the SAMA algorithm.

Fig.11 illustrates the blocking ratios of our three algorithms and the SAMA algorithm, under various VNFs numbers. From Fig.11, we see that the blocking ratios of our algorithms are much lower than that of the SAMA algorithm. This is because that the VMs reusing strategy, VNFs combination strategy and temporary link mapping strategy have a positive impact on the mapping success of entire SFC, and thus decreases the resource consumption, that results in a lower blocking ratio. With the adoption of the tempo-

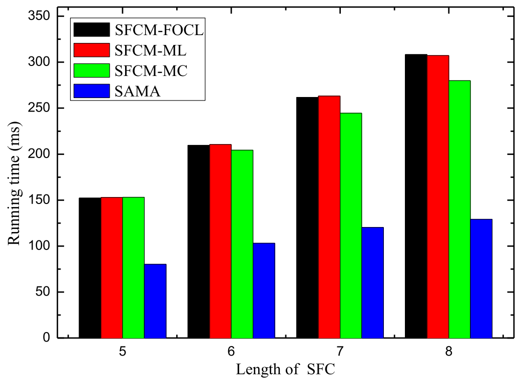

FIGURE 12. The running time.

rary link mapping strategy, our algorithms can effectively reduce the mapping failure. That is because the SFCM-FOCL algorithm optimizes the utilization of computing resources and the utilization of link resources fairly. Additionally, the SFCM-FOCL algorithm has a lower blocking ratio than the SFCM-MC algorithm and the SFCM-ML algorithm does.

Fig.12 describes the running times of our three algorithms and the SAMA algorithm, under various VNFs numbers. From Fig.12, we see that our three algorithms have a similar running time due to our three algorithms use the same strategies and only the optimization objectives are different. Due to the SAMA algorithm does not adopt the temporary link mapping strategy, the running time of the SAMA algorithm is about half that of our three algorithms.

VI. CONCLUSION

In this paper, we study the problem of online placement for SFC requests in 5G mobile network based on the fog radio access network, and model the problem as a set of efficient formulations. The SFCM-MC formulation is proposed for minimizing the cost of computing resources as much as possible; to minimize the cost of link resources, we give the SFCM-ML formulation; then follow by a two-player pure-strategy's Game of Battle of Sex (BoS) model which captures the competition on physical resources between VNF allocation and routing to find a fair solution for the costs of computing resources and link resources, i.e., SFCM-FOCL model; then we design a set of heuristic algorithms based on the three formulations. We conduct detailed simulations for performance measurement and identify all the experimental work of our proposed algorithms can be satisfied under the computer-simulated testing environments for 5G networks. From simulation results, we can see that the performance of our approach has a better status than the benchmark algorithm, SAMA, for mapping the SFC requests in terms of the total SFCs mapping cost, total VNFs mapping cost, total links mapping cost, and blocking ratio. Our future work will include performance measurement of our proposed algorithms with the 5G networks in outdoor and also design to location privacy to improve the security and privacy of our algorithms and performance in the real world scenario.

REFERENCES

[1] H.-C. Hsieh, J.-L. Chen, and A. Benslimane, "5G virtualized multi-access edge computing platform for IoT applications," *J. Netw. Comput. Appl.*, vol. 115, pp. 94–102, Aug. 2018.
[2] O. Galinina, H. Tabassum, K. Mikhaylov, S. Andreev, E. Hossain, and Y. Koucheryavy, "On feasibility of 5G-grade dedicated RF charging technology for wireless-powered wearables," *IEEE Wireless Commun.*, vol. 23, no. 2, pp. 28–37, Apr. 2016.
[3] O. Holland and M. Dohler, "Geolocation-based architecture for heterogeneous spectrum usage in 5G," in *Proc. IEEE Globecom*, Dec. 2015, pp. 1–6.
[4] C.-L. I, C. Rowell, S. Han, Z. Xu, G. Li, and Z. Pan, "Toward green and soft: A 5G perspective," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 66–73, Feb. 2014.
[5] *5G: Rethink Mobile Communications for 2020+*, FuTURE Forum 5G SIG, 2014.
[6] K. Sundaresan, M. Y. Arslan, S. Singh, S. Rangarajan, and S. V. Krishnamurthy, "FluidNet: A flexible cloud-based radio access network for small cells," *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 915–928, Apr. 2016.
[7] T. Han and N. Ansari, "A traffic load balancing framework for software-defined radio access networks powered by hybrid energy sources," *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, no. 2, pp. 1038–1051, Apr. 2016.
[8] G. Sun, Y. Li, D. Liao, and V. Chang, "Service function chain orchestration across multiple domains: A full mesh aggregation approach," *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 3, pp. 1175–1191, Sep. 2018.
[9] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, "Cloud radio access network (C-RAN): A primer," *IEEE Netw.*, vol. 29, no. 1, pp. 35–41, Jan. 2015.
[10] T. Taleb, "Toward carrier cloud: Potential, challenges, and solutions," *IEEE Wireless Commun.*, vol. 21, no. 3, pp. 80–91, Jun. 2014.
[11] G. Sun, D. Liao, D. Zhao, Z. Sun, and V. Chang, "Towards provisioning hybrid virtual networks in federated cloud data centers," *Future Gener. Comput. Syst.*, vol. 87, pp. 457–469, Oct. 2018.
[12] G. Sun, V. Chang, G. Yang, and D. Liao, "The cost-efficient deployment of replica servers in virtual content distribution networks for data fusion," *Inf. Sci.*, vol. 432, pp. 495–515, Mar. 2017.
[13] G. Sun, D. Liao, S. Bu, H. Yu, Z. Sun, and V. Chang, "The efficient framework and algorithm for provisioning evolving VDC in federated data centers," *Future Gener. Comput. Syst.*, vol. 73, pp. 79–89, Aug. 2017.
[14] G. Sun, D. Liao, V. Anand, D. Zhao, and H. Yu, "A new technique for efficient live migration of multiple virtual machines," *Future Gener. Comput. Syst.*, vol. 55, pp. 74–86, Feb. 2016.
[15] A. F. R. Trajano and M. P. Fernandez, "Two-phase load balancing of in-memory key-value storages using network functions virtualization (NFV)," *J. Netw. Comput. Appl.*, vol. 66, pp. 1–13, Jul. 2016.
[16] P. Hu, H. Ning, T. Qiu, Y. Zhang, and X. Lou, "Fog computing based face identification and resolution scheme in Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 1910–1920, Aug. 2017.
[17] F. Jalali, K. Hinton, R. Ayre, T. Alpcan, and R. S. Tucker, "Fog computing may help to save energy in cloud computing," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1728–1739, May 2016.
[18] F. van Lingen *et al.*, "The unavoidable convergence of NFV, 5G, and fog: A model-driven approach to bridge cloud and edge," *IEEE Commun. Mag.*, vol. 55, no. 8, pp. 28–35, Aug. 2017.
[19] R. Vilalta, A. Mayoral, R. Casellas, R. Martínez, and R. Muñoz, "Experimental demonstration of distributed multi-tenant cloud/fog and heterogeneous SDN/NFV orchestration for 5G services," in *Proc. Eur. Conf. Netw. Commun.*, 2016, pp. 52–56.
[20] R. Vilalta, A. Mayoral, R. Casellas, R. Martínez, and R. Muñoz, "SDN/NFV orchestration of multi-technology and multi-domain networks in cloud/fog architectures for 5G services," in *Proc. Optoelectron. Commun. Conf.*, 2016, pp. 1–3.
[21] Y.-J. Ku *et al.*, "5G radio access network design with the fog paradigm: Confluence of communications and computing," *IEEE Commun. Mag.*, vol. 55, no. 4, pp. 46–52, Apr. 2017.
[22] K. Liang, L. Zhao, X. Chu, and H.-H. Chen, "An integrated architecture for software defined and virtualized radio access networks with fog computing," *IEEE Netw.*, vol. 31, no. 1, pp. 80–87, Jan./Feb. 2017.
[23] G. Sun, G. Zhu, D. Liao, H. Yu, X. Du, and M. Guizani, "Cost-efficient service function chain orchestration for low-latency applications in NFV networks," *IEEE Syst. J.*, to be published. doi: 10.1109/JSYST.2018.2879883.
[24] S. Mehraghdam, M. Keller, and H. Karl, "Specifying and placing chains of virtual network functions," in *Proc. IEEE 3rd Int. Conf. Cloud Netw. (CloudNet)*, Oct. 2014, pp. 7–13.

[25] G. Sun, Y. Li, H. Yu, A. V. Vasilakos, X. Du, and M. Guizani, "Energy-efficient and traffic-aware service function chaining orchestration in multi-domain networks," *Future Gener. Comput. Syst.*, vol. 91, pp. 347–360, Feb. 2019.

[26] R. Cohen, L. Lewin-Eytan, J. S. Naor, and D. Raz, "Near optimal placement of virtual network functions," in *Proc. IEEE INFOCOM*, Apr./May 2015, pp. 1346–1354.

[27] T.-W. Kuo, B.-H. Liou, K. C.-J. Lin, and M.-J. Tsai, "Deploying chains of virtual network functions: On the relation between link and server usage," in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–9.

[28] M. F. Bari, S. Chowdhury, R. Ahmed, and R. Boutaba. (Mar. 2015). "On orchestrating virtual network functions in NFV." [Online]. https://arxiv.org/abs/1503.06377

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

[30] T. Kim, S. Kim, K. Lee, and S. Park, "A QoS assured network service chaining algorithm in network function virtualization architecture," in *Proc. IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput. (CCGrid)*, May 2015, pp. 1221–1224.

[31] T. Park, Y. Kim, J. Park, H. Suh, B. Hong, and S. Shin, "QoSE: Quality of Security a network security framework with distributed NFV," in *Proc. IEEE ICC*, May 2016, pp. 1–6.

[32] Y. Li, L. T. X. Phan, and B. T. Loo, "Network functions virtualization with soft real-time guarantees," in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–9.

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

[34] X. Li and C. Qian, "Low-complexity multi-resource packet scheduling for network function virtualization," in *Proc. IEEE INFOCOM*, Apr./May 2015, pp. 1400–1408.

[35] T. Taleb, M. Bagaa, and A. Ksentini, "User mobility-aware virtual network function placement for virtual 5G network infrastructure," in *Proc. IEEE ICC*, Jun. 2015, pp. 3879–3884.

[36] T. Taleb, A. Ksentini, M. Chen, and R. Jantti, "Coping with emerging mobile social media applications through dynamic service function chaining," *IEEE Trans. Wireless Commun.*, vol. 15, no. 4, pp. 2859–2871, Apr. 2016.

[37] J. Liu, Z. Jiang, N. Kato, O. Akashi, and A. Takahara, "Reliability evaluation for NFV deployment of future mobile broadband networks," *IEEE Wireless Commun.*, vol. 23, no. 3, pp. 90–96, Jun. 2016.

[38] A. Basta, W. Kellerer, M. Hoffmann, H. J. Morper, and K. Hoffmann, "Applying NFV and SDN to LTE mobile core gateways, the functions placement problem," in *Proc. ACM Workshop Things Cellular, Oper., Appl., Challenges*, 2014, pp. 33–38.

[39] C. Liang, F. R. Yu, and X. Zhang, "Information-centric network function virtualization over 5G mobile wireless networks," *IEEE Netw.*, vol. 29, no. 3, pp. 68–74, May 2015.

[40] B. Martini, F. Paganelli, P. Cappanera, S. Turchi, and P. Castoldi, "Latency-aware composition of virtual functions in 5G," in *Proc. IEEE Conf. Netw. Softwarization (NetSoft)*, Apr. 2015, pp. 1–6.

[41] S. Abdelwahab, B. Hamdaoui, M. Guizani, and T. Znati, "Network function virtualization in 5G," *IEEE Commun. Mag.*, vol. 54, no. 4, pp. 84–91, Apr. 2016.

[42] F. Yang, H. Wang, C. Mei, J. Zhang, and M. Wang, "A flexible three clouds 5G mobile network architecture based on NFV & SDN," *China Commun.*, vol. 12, pp. 121–131, Dec. 2015.

[43] P. K. Agyapong, M. Iwamura, D. Staehle, W. Kiess, and A. Benjebbour, "Design considerations for a 5G network architecture," *IEEE Commun. Mag.*, vol. 52, no. 11, pp. 65–75, Nov. 2014.

[44] G. Sun, V. Anand, D. Liao, C. Lu, X. Zhang, and N.-H. Bao, "Power-efficient provisioning for online virtual network requests in cloud-based data centers," *IEEE Syst. J.*, vol. 9, no. 2, pp. 427–441, Jun. 2015.

[45] G. Sun, H. Yu, V. Anand, and L. Li, "A cost efficient framework and algorithm for embedding dynamic virtual network requests," *Future Gener. Comput. Syst.*, vol. 29, no. 5, pp. 1265–1277, 2013.

[46] G. Sun, D. Liao, D. Zhao, Z. Xu, and H. Yu, "Live migration for multiple correlated virtual machines in cloud-based data centers," *IEEE Trans. Services Comput.*, vol. 11, no. 2, pp. 279–291, Mar./Apr. 2018.

[47] C. Pham, N. H. Tran, S. Ren, W. Saad, and C. S. Hong, "Traffic-aware and Energy-efficient vNF placement for service chaining: Joint sampling and matching approach," *IEEE Trans. Services Comput.*, to be published.

DONGCHENG ZHAO is currently pursuing the Ph.D. degree in communication and information system with the University of Electronic Science and Technology of China. His research interests include network function virtualization, cloud computing, fog computing, and 5G mobile networks.

JING REN received the B.E. and Ph.D. degrees in communication engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2007 and 2015, respectively, where she is currently a Researcher. Her research interests include network architecture and protocol design, information-centric networking, and software-defined networking.

RONGPING LIN received the Ph.D. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, in 2013. From 2013 to 2014, he was a Senior Research Assistant with the Electronic Engineering Department, City University of Hong Kong. He is currently an Associate Professor with the School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, China. His research interests include optical networks, SDN networks, and network optimization.

SHIZHONG XU received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 1994, 1997, and 2000, respectively, where he is currently a Professor. His research interests include broadband networks, all-optical networks, and next-generation networks.

VICTOR CHANG received the PGCert (Higher Education, Fellow) and Ph.D. (CS) degrees, in 2012 and 2013, respectively. He was a Senior Lecturer with lLeeds Beckett University, U.K., for 3.5 years. He is currently an Associate Professor with Xi'an Jiaotong-Liverpool University, Suzhou, China. He received the European Award on Cloud Migration in 2011, the Best Paper Awards in 2012 and 2015, and numerous awards, since 2012. He is a Leading Expert on big data/cloud/security, a Visiting Scholar/Ph.D. Examiner at several universities, the Editor-in-Chief of IJOCI and OJBD, an Editor of FGCS, the Founding Chair of two international workshops, and the Founding Conference Chair of the IoTBD 2016 and COMPLEXIS 2016.