Cost-Efficient Virtual Network Function Graph (vNFG) Provisioning in Multi-Domain Elastic Optical Networks

Yixiang Wang, Ping Lu, Wei Lu, Zuqing Zhu, *Senior Member, IEEE*

***Abstract*—Acting as a promising technology to bring revolutionary changes to how the networks are architected, network function virtualization (NFV) leverages IT virtualization technologies to instantiate various types of virtual network functions (vNFs) flexibly and dynamically on commodity hardware, which can be easily found in a variety of NFV infrastructure points-of-presence, *e.g.*, datacenters (DCs). In this work, we investigate how to cost-effectively provision vNF graphs (vNFGs) with arbitrary topologies in a multi-domain elastic optical network (MD-EON) that consists of two domains, *i.e.*, the private and public ones. We first formulate an integer linear programming (ILP) model with the objective to minimize the total resource cost of vNFG provisioning, and show that it can solve the problem exactly. With the ILP model, we analyze the complexity of the problem and prove that it is an $\mathcal{NP}$-hard one. Then, we leverage the minimum k-cut problem to design two time-efficient heuristics. The results from extensive simulations verify the performance of our proposed algorithms, and indicate that they can balance the IT and spectrum resource usages intelligently according to the resource price setting in multi-domain environment.**

***Index Terms*—Network function virtualization (NFV), virtual network function graph (vNFG), multi-domain, elastic optical networks (EONs).**

I. Introduction

RECENTLY, network function virtualization (NFV) enjoys increasing popularity because the deployment of new network services can be greatly expedited with it [1, 2]. Specifically, NFV aims to migrate network functions from expensive special-purpose hardware to software-defined elements by leveraging IT resource virtualization, *i.e.*, processing traffic with virtual network functions (vNFs) [3–8]. Meanwhile, bandwidth-intensive emerging network services, such as video streaming, social TV, *etc*, are developing rapidly in the Internet. Hence, how to deploy these services in the Internet cost-effectively has become a hot research topic. As these services usually use various network functions to process data traffic, *e.g.*, video data needs compression and transcoding while voice data requires noise suppression and sampling, realizing them with vNFs in datacenters (DCs) can improve the adaptivity and efficiency of service provisioning [6, 7].

Note that, each network service may require a set of vNFs and the connectivity among them can formulate an arbitrary graph [3]. For example, to realize the NFV-based network defense system in [9] or to achieve the multipath routing based load balancing in [10], vNFs should be grouped into vNF graphs (vNFGs) instead of simple service chains. This makes deploying network services equivalent to provisioning vNFGs in an inter-DC network, which could be challenging when the constraints on IT and bandwidth resources both need to be addressed [6, 7]. Previously, people have considered the problem of vNF placement in [4, 5], while the studies in [11–14] have investigated how to deploy vNFGs in the forms of chain or tree. However, to the best of our knowledge, the problem of provisioning vNFGs with arbitrary topologies in inter-DC networks under IT and bandwidth resource constraints has not been fully explored yet. Meanwhile, the capacity and flexibility of physical infrastructure can affect the performance of vNFG provisioning in inter-DC networks significantly [12, 14]. This is because the dynamics of bandwidth-intensive network services would make the traffic flowing through vNFs in a vNFG exhibit high peak throughput and high burstiness [15]. Consequently, agile bandwidth management in the optical layer of inter-DC networks would be necessary for provisioning vNFGs efficiently, which can be realized by leveraging the technical advances on flexible-grid elastic optical networks (EONs) [16].

In a practical scenario, the service provider (SP) of vNFGs might have to utilize resources from a multi-domain environment [17, 18]. It is known that multi-domain EONs (MD-EONs) can solve the inter-operability issues resulting from network elements owned by different vendors, provide enhanced network scalability and extended service reach, and handle the situation where the optical switches are geographically-distributed and/or operated by different carriers [18, 19]. More importantly, an SP can build a multi-DCs system connected with an EON as its private domain, and when the computing/storage capacity in the private domain becomes insufficient, it can rent DCs and EON infrastructure from external public networks as a quick and elastic solution. This actually mimics the classic hybrid cloud architecture [20]. Therefore, it would be relevant to study how to provision vNFGs cost-effectively in MD-EONs, which, to the best of our knowledge, has not been addressed in literature before.

Fig. 1 provides an illustrative example on vNFG provisioning in an MD-EON. As shown in Fig. 1(a), the inter-DC network is built over an MD-EON that consists of two domains, *i.e.*, the private and public ones. To provision the network services that request for the vNFGs in Fig. 1(b), the SP needs to instantiate the vNFs in the DCs and then establishes necessary lightpaths to connect them accordingly. Then,

Y. Wang, P. Lu, W. Lu and Z. Zhu are with the School of Information Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, China (Email: zqzhu@ieee.org).

Manuscript received on December 2, 2016.

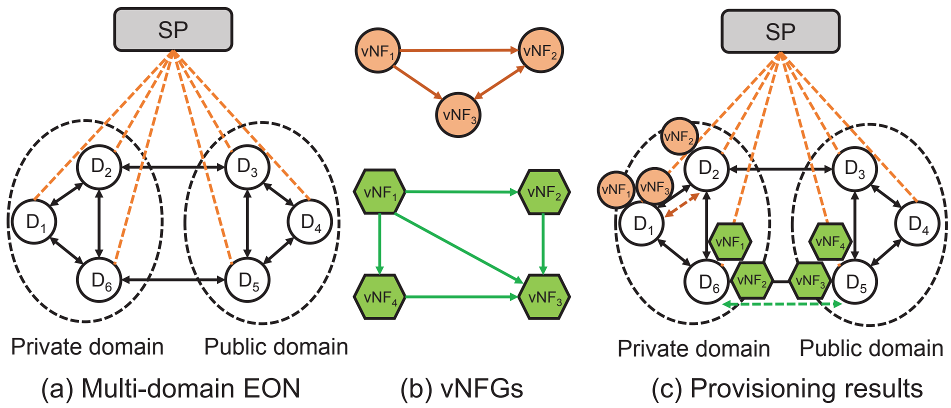


Fig. 1. Example on provisioning vNFGs in an MD-EON.

it can obtain the provisioning results as illustrated in Fig. 1(c). Note that, the IT and spectrum resources in the private domain are usually cheaper, and thus for improving cost-effectiveness, the SP should only turn to the public domain when the private one does not have sufficient resources to provision the vNFGs. Although the problem looks similar to the multi-domain virtual network embedding (VNE) problem that has already been studied in [17], they are fundamentally different. Basically, in multi-domain VNE, the virtual networks' topologies would not change during the embedding. While in our problem, the actual substrate topology used to support a vNFG can change with the vNF placement. For instance, in Fig. 1(c), for the green vNFG, after we placing *vNFs* 1 and 2 on *DC* D_6 and *vNFs* 3 and 4 on *DC* D_5, the vNFG with a mesh topology is carried by a substrate chain. More specifically, the connections for *vNF* 1→*vNF* 2 and *vNF* 4→*vNF* 3 are supported with intra-DC communications within *DC* D_6 and *DC* D_5, respectively, while the rest of the connections are merged and supported with the inter-domain link between *DCs* D_6 and D_5. Hence, our problem is more sophisticated than multi-domain VNE.

This paper studies how to provision vNFGs for network services cost-effectively in MD-EONs. We first formulate an integer linear programming (ILP) model with the objective to minimize the total resource cost of vNFG provisioning, and show that it solves the problem exactly. With the ILP, we prove the problem's $\mathcal{NP}$-hardness. Then, we leverage the minimum k-cut problem [21] to design two time-efficient heuristics. The results from extensive simulations verify the performance of our proposed algorithms. In summary, the major contributions of this work are as follows.

- We formulate an ILP model to solve the problem of vNFG provisioning in MD-EONs exactly for minimizing the total resource cost.
- We analyze the complexity of the problem and prove that it is an $\mathcal{NP}$-hard one.
- We propose two time-efficient heuristics by leveraging the minimum k-cut problem and conduct simulations for offline and online vNFG provisioning to verify that the proposed heuristics can provide near-optimal results.

The rest of this paper is organized as follows. We provide a survey on the related work in Section II. Section III introduces the network model, formulates the ILP model, and analyzes the problem's complexity. The heuristics are proposed in Section IV, and the algorithms' performance is evaluated in Section V. Finally, Section VI summarizes the paper.

II. Related Work

Previously, people have studied the service provisioning schemes in MD-EONs from different perspectives [18, 19, 22–25]. Zhu *et al.* [19] considered how to realize service provisioning with energy-aware regenerator allocation in MD-EONs, and they also tried to address the security issues due to cross-domain attacks in MD-EONs and designed differential routing and spectrum assignment (RSA) schemes for intra- and inter-domain requests in [22]. For system implementations, the studies in [18, 23, 24] leveraged the idea of software-defined networking (SDN) to realize the network control and management frameworks for MD-EONs. The authors of [25] listed the challenges brought by integrating NFV and SDN in multi-layer and multi-domain network environments. However, these studies did not address how to deploy vNFGs with arbitrary topologies in MD-EONs.

In [3], the white paper of NFV suggests that network services can be realized by deploying individual vNFs or vNFGs in substrate networks. However, most of previous studies on NFV-related service provisioning only considered individual vNFs or vNF chains [4, 5, 11, 12], which are relatively simple topologies for vNFGs. In [4, 5], the resource allocation schemes for placing individual vNFs were investigated. Mehraghdam *et al.* [11] considered the formulation of vNF chains in an operator's network, while the vNF chaining schemes in packet/optical DCs were discussed in [12]. The tree-type vNFGs were considered in [13, 14] for realizing multicast services in software-defined packet networks and inter-DC EONs, respectively. Since this work studies how to provision vNFGs with arbitrary topologies in MD-EONs, our network model of vNFGs is more generic than those used in the aforementioned studies.

Note that, as vNFs usually run as the applications in ensembles of virtual machines (VMs) [5], vNF provisioning and VM placement are generally related. Previously, Duong-Ba *et al.* [26] considered the VM management (*i.e.*, placement and migration) in DCs with the objective of minimizing energy consumption and cross-network traffic. In [27], the authors proposed a cost-aware two-phase meta-heuristic to minimize the cost of VM placement in geographically-distributed DCs. Nevertheless, we can see that in [26, 27], the VMs are usually independent and do not have specified relations as described by vNFGs. Hence, despite of the similarity, VM placement and vNFG provisioning have different constraints.

Lastly, as we have explained before, vNFG provisioning is fundamentally different from the VNE problems that have been investigated in [28–35]. The authors of [31] studied an energy-efficient VNE model, while the energy-efficient VNE schemes in IP over optical networks have been considered in [35]. However, since the network model of vNFG provisioning is different and we consider a multi-domain environment, the VNE algorithms proposed in these existing studies cannot be applied to solve our problem. For instance, in VNE, the virtual networks' topologies are determined before the embedding, while in vNFG provisioning, the actual topologies of vNFGs are only finalized after they have been provisioned.

III. Problem Formulation

In this section, we describe the network model of vNFG provisioning in MD-EONs, formulate an ILP model to solve the problem exactly, and provide formal analysis on the problem's complexity.

A. Network Model

We model the MD-EON as a directed graph $G(D, E)$, where D is the DC set, and E is the set of fiber links that connect the DCs. There are F frequency slots (FS') on each fiber link $e \in E$ [36], and the unit cost of FS usage on link e is β_e. Each DC $d \in D$ can belong to the private or public domain in the MD-EON, and the SP uses the multi-dimensional IT resources on it (*i.e.,* CPU cycles, memory and storage) to instantiate vNFs. We denote the capacity of IT resources on DC d as $C_d =< C_d^c, C_d^m, C_d^s >$, where C_d^c, C_d^m and C_d^s represent the DC's capacities of CPU cycle, memory and storage, respectively. We assume that $\mathcal{M}$ is the set of vNFs that the SP can instantiate in the MD-EON to accomplish all the network services requested by end users. v_d^m denotes the cost of the IT resource usage for instantiating a type $m \in \mathcal{M}$ vNF in DC d. The costs of IT and spectrum resources in the private domain are lower than those in the public domain.

A network service may use different vNFs to process the data from end users and these vNFs can formulate a vNFG as examples shown in Fig. 1(b). Each vNFG can be denoted as a directed graph $G^r = \{U, \mathbf{A}\}$, where U is the set of requested vNFs and $\mathbf{A} = [a_{k,n}]$ $(k, n \in [1, |U|])$ is the traffic matrix for these vNFs. Here, $a_{k,n} = \lceil \frac{b_{k,n}}{B_w} \rceil$ represents the bandwidth requirement in FS' for the traffic from the k-th vNF to the n-th vNF in the vNFG, where $b_{k,n}$ is the actual bit-rate requirement and B_w represents the capacity of an FS. To generalize the vNFG model, we assume that multiple type m vNFs can appear in a vNFG and use N^m to denote the number of type m vNFs that are needed by the vNFG, and thus the total number of requested vNFs can be calculated as

$$|U| = \sum_{m \in \mathcal{M}} N^m. \quad (1)$$

B. ILP Formulation

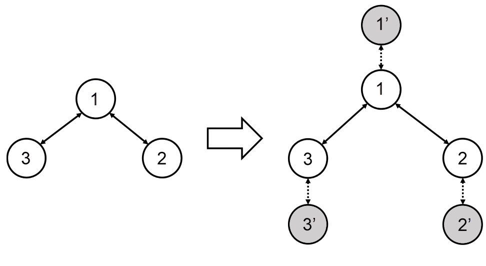
To provision a vNFG in the MD-EON, we need to accomplish two tasks. One is to instantiate vNFs in the DCs, and the other is to set up lightpaths by allocating FS' on fiber links to satisfy the bandwidth requirements among the vNFs. In the process, the resource usages in the DCs and on the links should not exceed their capacities, and the lightpaths should also satisfy the spectrum contiguous and continuity constraints. For each DC pair, we precalculate K shortest paths with the Yen's algorithm [37] and get all the possible RSA solutions on each path based on the current network status. The RSA solutions on all the precalculated paths are included in $\mathcal{S}$ to build the RSA solution set in the MD-EON [38], which are used as the MILP's input. In the following, we formulate an ILP model to solve the problem of cost-effective vNFG provisioning in the MD-EON exactly. We denote the ILP model as vNFG-ILP.

Notations:

- $G(D, E)$: the substrate topology of the MD-EON.
- F: the number of FS' on each fiber link e.
- β_e: the unit cost of FS usage on link e.
- $\mathcal{M}$: the set of vNF types that the MD-EON can support.
- C_d: the capacity of multi-dimensional IT resources on DC $d \in D$, *i.e.*, $C_d =< C_d^c, C_d^m, C_d^s >$.
- c_m^c: the CPU cycle consumption of a type m vNF.
- c_m^m: the memory consumption of a type m vNF.
- c_m^s: the storage consumption of a type m vNF.
- v_d^m: the cost of the IT resource usage for instantiating a type m vNF in DC d.
- G^r: the vNFG request, *i.e.*, $G^r = \{U, \mathbf{A}\}$.
- $\mathcal{S}$: the set of precalculated RSA solutions in $G(D, E)$.
- s^*: the number of FS' used on each link in solution $s \in \tilde{\mathcal{S}}$.
- $\tilde{\mathcal{S}}$: the extended RSA solution set based on S.
- $\tilde{\mathcal{S}}_{d_1,d_2}$: the extended RSA solution set for lightpaths from d_1 to d_2, $d_1, d_2 \in D$.
- $I_{k,m}$: the indicator that equals 1 if the k-th vNF in vNFG G^r belongs to type m, and 0 otherwise.
- $z_{e,s}$: the indicator that equals 1 if solution $s \in \tilde{\mathcal{S}}$ uses FS' on link e, and 0 otherwise.
- $z_{f,s}^e$: the indicator that equals 1 if solution $s \in \tilde{\mathcal{S}}$ on link e includes the f-th FS, and 0 otherwise.
- $l_{e,f}$: the indicator that equals 1 if the f-th FS has not been used on link e, and 0 otherwise.

Variables:

- n_e: the integer variable that indicates the number of used FS' on link e.
- h_d^m: the integer variable that indicates the number of type m vNFs instantiated in DC d.
- $y_{k,d}$: the boolean variable that equals 1 if the k-th vNF in vNFG request G^r is instantiated in DC d, and 0 otherwise.
- $f_{k,n,s}$: the boolean variable that equals 1 if for vNFG request G^r, the lightpath from the k-th vNF to the n-th vNF uses RSA solution $s \in \tilde{\mathcal{S}}$, and 0 otherwise.



(a) Original graph (b) Extended graph

Fig. 2. Example on constructing the extended RSA solution set.

Objective:

The objective is to minimize the total cost of resources used for provisioning the vNFG in the MD-EON, as

$$Minimize \quad \mathcal{C}_t = \mathcal{C}_{NF} + \mathcal{C}_{FS}, \tag{2}$$

where $\mathcal{C}_{NF}$ denotes the total cost of IT resource usage for instantiating vNFs and $\mathcal{C}_{FS}$ is the total cost of FS usage. The total number of type m vNFs instantiated in DC d can be calculated as

$$h_d^m = \sum_{k=1}^{|U|} y_{k,d} \cdot I_{k,m}, \quad \forall m \in \mathcal{M},\ d \in D. \tag{3}$$

Then, the total cost of IT resource usage is

$$\mathcal{C}_{NF} = \sum_{d\in D}\sum_{m\in\mathcal{M}} h_d^m \cdot v_d^m. \tag{4}$$

On the other hand, to calculate the total cost of FS usage, we introduce the concept of extended RSA solution set, which is leveraged to cover the situation in which more than one vNFs in a vNFG are instantiated in the same DC. Fig. 2 explains how to obtain the extended RSA solution set $\tilde{\mathcal{S}}$ with the precalculated RSA solution set $\mathcal{S}$. Specifically, with the original graph in Fig. 2(a), we add a dummy node (*i.e.*, grey nodes in Fig. 2(b)) aside each node and connect them with a dummy bidirectional link (*i.e.*, dotted lines in Fig. 2(b)) on which the number of FS' is $+\infty$ and the unit cost of FS usage is 0. Then, we get an extended graph as shown in Fig. 2(b), and with each dummy node d' and its original node d, we calculate several feasible RSA solutions between them to support the traffic among the vNFs that are instantiated in the same DC d. By adding these new RSA solutions in the original solution set $\mathcal{S}$, we obtain the extended RSA solution set $\tilde{\mathcal{S}}$. Then, the FS usage on each link can be calculated as

$$n_e = \sum_{k,n}\sum_{s\in\tilde{\mathcal{S}}} f_{k,n,s} \cdot z_{e,s} \cdot s^*,\ \forall e \in E. \tag{5}$$

Since each newly-added RSA solution s only includes a dummy link with $z_{e,s} = 0$, the situation in which more than one vNFs in a vNFG are instantiated in the same DC is correctly represented. Finally, we have

$$\mathcal{C}_{FS} = \sum_{e\in E} n_e \cdot \beta_e. \tag{6}$$

Constraints:

1) *IT Resource Constraints:*

$$\sum_{m\in\mathcal{M}} h_d^m \cdot c_m^c \le C_d^c, \quad \forall d \in D, \tag{7}$$

$$\sum_{m\in\mathcal{M}} h_d^m \cdot c_m^m \le C_d^m, \quad \forall d \in D, \tag{8}$$

$$\sum_{m\in\mathcal{M}} h_d^m \cdot c_m^s \le C_d^s, \quad \forall d \in D. \tag{9}$$

Eqs. (7)-(9) ensure that when vNFs are instantiated in a certain DC, the consumption on IT resources, *i.e.*, CPU cycles, memory and storage, should not exceed the corresponding IT resource capacities of that DC.

2) *Spectrum Resource Constraints:*

$$\sum_{s\in\tilde{\mathcal{S}}} f_{k,n,s} \cdot s^* \ge a_{k,n}, \quad \forall k,n \in [1,|U|]. \tag{10}$$

$$\sum_{k,n}\sum_{s\in\tilde{\mathcal{S}}} f_{k,n,s} \cdot z_{f,s}^e \le l_{e,f}, \quad \forall e \in E, f \in F. \tag{11}$$

Eq. (10) ensures that the selected RSA solution can satisfy the spectrum requirement, and Eq. (11) guarantees that each FS f on a link $e \in E$ can only be used once, *i.e.*, satisfying the spectrum non-overlapping constraint.

3) *vNF Provisioning Constraints:*

$$\sum_{d\in D} y_{k,d} = 1, \quad \forall k \in [1,|U|]. \tag{12}$$

Eq. (12) ensures that each of the vNFs in a vNFG is instantiated in one and only one DC.

$$\sum_{s\in\tilde{\mathcal{S}}} f_{k,n,s} = 1, \quad \forall k,n \in [1,|U|]. \tag{13}$$

Eq. (13) ensures that for vNFG G^r, the traffic from the k-th vNF to the n-th vNF should use a feasible solution s in the extended RSA solution set $\tilde{\mathcal{S}}$.

$$\begin{gathered} y_{k,d_1} + y_{n,d_2} - 2 < \sum_{s\in\tilde{\mathcal{S}}_{d_1,d_2}} f_{k,n,s}, \\ \forall k,n \in [1,|U|], \forall d_1,d_2 \in D. \end{gathered} \tag{14}$$

$$\begin{gathered} 2\cdot\left(\sum_{s\in\tilde{\mathcal{S}}_{d_1,d_2}} f_{k,n,s} - 1\right) \le y_{k,d_1} + y_{n,d_2} - 2, \\ \forall k,n \in [1,|U|], \forall d_1,d_2 \in D. \end{gathered} \tag{15}$$

Eqs. (14)-(15) ensure that the results of vNF provisioning and spectrum allocation are matched. Specifically, for vNFG G^r, if the k-th and n-th vNFs are placed on DCs d_1 and d_2, respectively, the traffic from the k-th vNF to the n-th vNF should use a feasible RSA solution s in the extended RSA solution set $\tilde{\mathcal{S}}_{d_1,d_2}$, and *vice versa*.

C. Complexity Analysis

Theorem 1. *The optimization described by the aforementioned vNFG-ILP model for cost-efficient vNFG provisioning in an MD-EON is $\mathcal{NP}$-hard.*

Proof: We prove the $\mathcal{NP}$-hardness of the optimization with restriction, *i.e.*, restricting away certain aspects of the problem until a known $\mathcal{NP}$-hard problem appears [39]. For a certain vNFG to be served, we first relax the restrictions on spectrum resource. Basically, we set the FS' on each fiber link in the MD-EON as infinite and assume that the unit cost of FS usage on each link is zero (*i.e.*, $\beta_e = 0, \forall e \in E$). Hence, the spectrum allocation results become irrelevant and the optimization becomes to instantiate the vNFs of the vNFG in the DCs to minimize the total cost of IT resource usage. Then, if we treat each vNF as an item with a three-dimensional weight (*i.e.*, CPU cycle, memory and storage consumptions) and the cost to instantiate it in a DC is its value, the optimization is transformed into the three-dimensional knapsack problem, which is known to be $\mathcal{NP}$-hard [39]. Therefore, since the special/restricted case of the optimization is the general case of a known $\mathcal{NP}$-hard problem, we prove that the optimization described by vNFG-ILP is $\mathcal{NP}$-hard. ■

IV. Heuristic Algorithms

Since the problem of cost-efficient vNFG provisioning in MD-EONs is $\mathcal{NP}$-hard, we design two time-efficient heuristics in this section by leveraging the minimum k-cut problem discussed in [21]. Before discussing the heuristics, we introduce a concept for assisting the algorithm design.

Definition. *For an arbitrary vNFG G^r, a **vNF-DC mapping scheme** is a feasible partition of its set of requested vNFs U.*

Fig. 3 shows an example on obtaining a vNF-DC mapping scheme for a vNFG $G^r = \{U, \mathbf{A}\}$. First of all, with the traffic matrix $\mathbf{A}$ in Fig. 3(a), we can formulate the original topology of the vNFG as shown in Fig. 3(b), which shows the logic connectivity among the five requested vNFs. Then, if we decide to deploy *vNFs* 1 and 2 on one DC and put each of the remaining vNFs to an individual DC, the original topology gets transformed to the four-node topology in Fig. 3(c) and the corresponding vNF-DC mapping scheme is also shown there. Apparently, the vNF-DC mapping scheme is just a feasible partition of the set of requested vNFs U.

Theorem 2. *For a vNFG with N vNFs, the number of all the possible vNF-DC mapping schemes is at least $2^N - N$.*

Proof: First of all, it is known that the total number of partitions of an N-element set is the Bell number with the formula as [40]

$$\mathcal{B}(N+1) = \sum_{k=0}^{N} \binom{N}{k} \cdot \mathcal{B}(k), \tag{16}$$

which is not an explicit expression. Then, if we define $\mathcal{B}(N,k)$ as the number of partitions when the N vNFs are divided into $k \in [1, N]$ non-empty subsets, we can express $\mathcal{B}(N)$ as

$$\mathcal{B}(N) = \sum_{k=1}^{N} \mathcal{B}(N,k). \tag{17}$$

Then, if we consider to get the set partition with the method that first chooses k "seed" vNFs to form the initial subsets and then inserts the remaining $N-k$ vNFs into the k subsets, the following inequality can be obtained

$$\mathcal{B}(N,k) \geq \binom{N}{k}, \quad \{N, k : N \geq 2,\ k \in [2, N]\}. \tag{18}$$

Then, by combining Eqs. (17)-(18), we have

$$\begin{aligned} \mathcal{B}(N) = \sum_{k=1}^{N} \mathcal{B}(N,k) &\geq 1 + \binom{N}{2} + \ldots + \binom{N}{N} \\ &= \sum_{k=0}^{N} \binom{N}{k} - \binom{N}{1} \\ &= 2^N - N. \end{aligned} \tag{19}$$

Hence, we prove the theorem and this suggests that even without considering the spectrum allocation to connect the deployed vNFs, the possible vNF-DC mapping schemes increase exponentially with the size of the vNFG. ■

A. Iterative Two-Phase Algorithm with Minimum 2-Cut

Basically, according to the network model described in Section III-A, the cost for provisioning a vNFG in an MD-EON consists of two components, *i.e.*, the IT resource cost for instantiating the requested vNFs and the spectrum resource cost for setting up lightpaths among the vNFs. Apparently, the first cost component can be reduced by trying to deploy the requested vNFs in the private domain where the cost of IT resource usage is lower. The second cost component can be reduced with two methods. Firstly, if we can deploy multiple vNFs on a same DC, the traffic among them becomes intra-DC and thus does not consume any spectrum resource on fiber links. Secondly, since the unit cost of FS usage on links in the private domain is lower than that of other links in the MD-EON, the spectrum resource cost can also be reduced by adjusting the vNF-DC mapping scheme to use as few DCs in the public domain as possible. However, according to *Theorem* 2, the number of possible vNF-DC mapping schemes increases exponentially with the number of vNFs in a vNFG, and thus a polynomial-time algorithm should not try to traverse all the vNF-DC mapping schemes.

Based on these considerations, we propose an iterative two-phase vNFG provisioning algorithm that applies the minimum 2-cut (M2C) algorithm [21] to the vNFG repeatedly, *i.e.*, iterative two-phase with M2C (iTP-M2C). The two phases are DC selection and vNFG partitioning, which are repeated until the vNFG is successfully provisioned. Specifically, the DC selection chooses an appropriate DC for vNF deployment and then the vNFG partitioning applies M2C to determine the subset of the vNFs to be put in that DC. Apparently, the selected DC(s) should have sufficient and cheap IT resources as well as be connected with fiber link(s) that have enough and

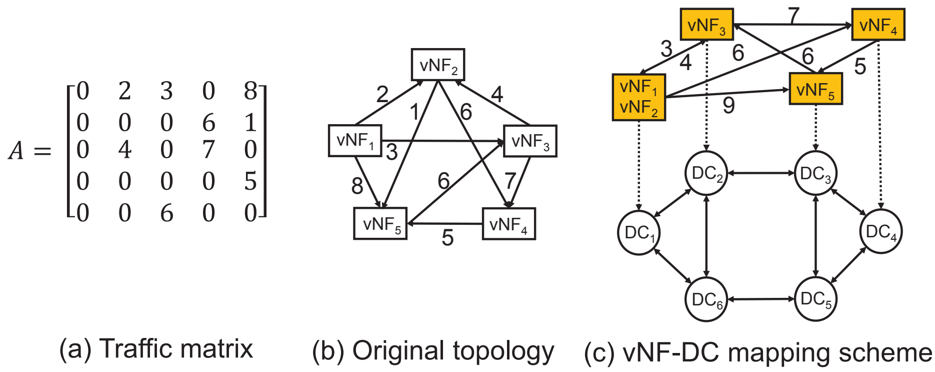


Fig. 3. Example on the vNF-DC mapping scheme.

low-cost spectrum resources. Hence, we define the following cost metric for each DC

$$w_d = \alpha \cdot \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \frac{v_d^m}{h_d^m} + \gamma \cdot \sum_{\{e: e=(d,v) \in E\}} \frac{\beta_e}{S_e}. \tag{20}$$

where α and γ are the coefficients to balance the importance of IT and spectrum resources in the DC selection, v_d^m is the price of the IT resource usage for instantiating a type m vNF in DC d, h_d^m is the number of type m vNFs that can be instantiated in DC d, and for each link e that origins from or ends at DC d, β_e is the unit cost of FS usage on it and S_e denotes the number of available FS' on it.

Basically, after determining the cost metrics for all the DCs in the MD-EON, we choose the DC with the smallest cost metric in the DC selection phase. Then, in the phase of vNFG partitioning, we first apply the M2C algorithm to divide the vNFG into two sub-graphs that require the minimum FS' for communication in between, and then adjust the two sub-graphs with least spectrum requirement increase each time until the vNFs in one of the sub-graphs can be accommodated in the selected DC. The operations in these two phases are repeated until the vNFG is provisioned or blocked.

Algorithm 1 shows the detailed procedure of iTP-M2C. *Line* 1 is for the initialization. The while-loop that covers *Lines* 2-42 applies the aforementioned two phases iteratively until the vNFG is provisioned or blocked. Here, D_{temp} stores the DCs that currently can still be used to instantiate vNFs, and *Lines* 3-5 will block the vNFG request G^r if D_{temp} is empty. Otherwise, *Lines* 7-8 perform the DC selection based on the cost metric defined in Eq. (20). Then, if all the outstanding vNFs in U_{temp} can be accommodated in the selected DC d, *Lines* 10-20 check whether the spectrum resources in the MD-EON are enough to set up all the necessary inter-DC lightpaths. If yes, *Lines* 21-23 mark G^r as provisioned with the selected DCs in D_{used}. Otherwise, we block G^r as shown in *Lines* 15-17. On the other hand, if all the outstanding vNFs in U_{temp} cannot be accommodated in the selected DC d, *Lines* 25-26 apply the M2C algorithm to divide the outstanding vNFs into two subsets (*i.e.*, U_1 and U_2) that have the least traffic in between, and then adjust the two subsets with the least traffic increase in between until the vNFs in one of the subsets (*w.o.l.g.*, we assume that it is U_1) can be accommodated in the selected DC d. Again, *Lines* 27-37 try to set up all the necessary inter-DC lightpaths with the lowest spectrum resource cost. If this can be done, *Lines* 38-39 update the variables and proceed to the next iteration.

Four subroutines contribute to the time complexity of iTP-M2C, *i.e.*, cost metric calculation and DC sorting, M2C operation, subset adjustment, RSA solution selection and spectrum refreshing. The complexity of the first one is $O(|D| \cdot \log(|D|))$. We leverage the K-means clustering discussed in [41] to realize M2C, and its complexity is $O(|U|)$. The complexity of the subset adjustment is also $O(|U|)$. There are K precalculated paths between any two DCs and F FS' on each link, so the complexity of RSA solution selection and spectrum refreshing is $O((|D|-1) \cdot (K \cdot F + E))$ in the worst case. Finally, we would use at most $|D|$ iterations to serve a vNFG request, and thus the overall time complexity of iTP-M2C is $O(|D|^2 \cdot (\log(|D|) + K \cdot F + E) + |D| \cdot |U|)$.

B. Two-Phase Algorithm with Minimum k-Cut

Note that, since iTP-M2C uses a greedy strategy to minimize the total cost of vNFG provisioning in each iteration, it might not be able to obtain the solution that is global optimal. Therefore, we propose another two-phase algorithm that utilizes minimum k-cut (MkC) to search for low-cost vNFG provisioning schemes, which is referred to as two-phase algorithm with MkC (TP-MkC). Specifically, TP-MkC first tries to divide the vNFs in $G^r = \{U, \mathbf{A}\}$ into $k = 1, 2, \cdots, \min(|D|, |U|)$ subsets with the MkC algorithm that ensures the minimum traffic among them. Then, the original vNFG is transformed into $\min(|D|, |U|)$ different virtual networks (VNs), and we try to embed the VNs into the MD-EON with consideration of the DC cost metric defined in Eq. (20) and spectrum resource cost. After obtaining all the feasible embedding schemes, we calculate the total costs of them and then select the one with the lowest cost to provision the vNFG. *Algorithm* 2 provides the detailed procedure of TP-MkC. *Lines* 1-2 are for the initialization. The for-loop covering *Lines* 3-11 divides the vNFs in $G^r = \{U, \mathbf{A}\}$ into $k = 1, 2, \cdots, \min(|D|, |U|)$ subsets with the MkC, constructs the corresponding VNs, and then tries to embed each VN into

Algorithm 1: Iterative Two-Phase Algorithm with M2C

input : vNFG request $G^r = \{U, \mathbf{A}\}$, and MD-EON $G(D, E)$.

$D_{temp} = D$, $D_{used} = \emptyset$, $U_{temp} = U$;
while $U_{temp} \neq \emptyset$ **do**
  if $D_{temp} = \emptyset$ **then**
    mark G^r as blocked;
    return;
  else
    calculate cost metric w_d with Eq. (20) for each DC $d \in D_{temp}$ based on status of $G(D, E)$;
    choose the DC d with the minimum w_d;
    if *vNFs in* U_{temp} *can be deployed in DC* d **then**
      if $D_{used} \neq \emptyset$ **then**
        for *each DC* $d' \in D_{used}$ **do**
          try to find an RSA solution with the lowest spectrum cost to support the traffic among the vNFs in DC d and those in DC d';
          if *the RSA solution can be found* **then**
            update network resource usage;
          else
            mark G^r as blocked;
            return;
          end
        end
      end
      $D_{used} = D_{used} \cup d$;
      mark G^r as provisioned with DCs in D_{used};
      return;
    else
      apply M2C to divide the vNFs in U_{temp} into two subsets U_1 and U_2 with the least traffic between them;
      adjust U_1 and U_2 with the least traffic increase in between such that the vNFs in U_1 can be accommodated in DC d;
      if $D_{used} \neq \emptyset$ **then**
        for *each DC* $d' \in D_{used}$ **do**
          try to find an RSA solution with the lowest spectrum cost to support the traffic among the vNFs in DC d and those in DC d';
          if *the RSA solution can be found* **then**
            update network resource usage;
          else
            mark G^r as blocked;
            return;
          end
        end
      end
      $D_{used} = D_{used} \cup d$, $D_{temp} = D_{temp} \setminus d$;
      $U_{temp} = U_{temp} \setminus U_1$;
    end
  end
end

$G(D, E)$ with node mapping based on DC cost metric $\{w_d\}$ and spectrum allocation based on FS cost for minimizing the total provisioning cost. If a feasible embedding scheme can be obtained for a VN, we store it in the solution set Ω and calculate the corresponding total provisioning cost, as shown in *Lines* 7-10. Finally, after trying all the VNs, we select the embedding scheme with the lowest total cost to provision G^r as illustrated in *Lines* 12-18. The time complexity of TP-MkC is $O(|D| \cdot \log(|D|) + \min(|D|, |U|) \cdot (|U| + |D|^2 \cdot (K \cdot F + E)))$.

Algorithm 2: Two-Phase Algorithm with MkC

input : vNFG request $G^r = \{U, \mathbf{A}\}$, and MD-EON $G(D, E)$.

1 $\mathbf{U}_{temp} = \emptyset$, $\Omega = \emptyset$;
2 calculate cost metric w_d with Eq. (20) for each DC $d \in D$ based on status of $G(D, E)$;
3 **for** $k = 1$ *to* $\min(|D|, |U|)$ **do**
4   apply MkC to divide the vNFs in U into k subsets with the least traffic among them;
5   store the k subsets in $\mathbf{U}_{temp}$ and construct a VN based on them;
6   try to map the VN into $G(D, E)$ with node mapping based on $\{w_d\}$ and spectrum allocation based on FS cost;
7   **if** *the VN can be embedded successfully* **then**
8   store the feasible embedding scheme in Ω;
9   calculate total resource cost of the scheme;
10   **end**
11 **end**
12 **if** $\Omega \neq \emptyset$ **then**
13   select the embedding scheme with the lowest total cost to provision G^r;
14   mark G^r as provisioned;
15   update network resource usage;
16 **else**
17   mark G^r as blocked;
18 **end**

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed algorithms, *i.e.*, the vNFG-ILP, iTP-M2C, and TP-MkC with the six-node topology in Fig. 1(a) and the NSFNET topology in Fig. 4. Note that, in addition to these two, there are also other practical multi-domain topologies that can be leveraged to model the MD-EON [42]. Our simulations consider both offline and online scenarios. Here, the offline scenario means that all the vNFGs are known in advance. While for the online scenario, the vNFGs can come and leave on-the-fly following the Poisson traffic model, *i.e.*, the average arrival rate is λ vNFGs per time-unit and the average holding time of each vNFG is $\frac{1}{\mu}$ time-units. Hence, the traffic load is $\frac{\lambda}{\mu}$ in Erlangs.

A. Performance Metrics

We first explain the performance metrics that are used in the simulations. The cost of the IT and spectrum resources

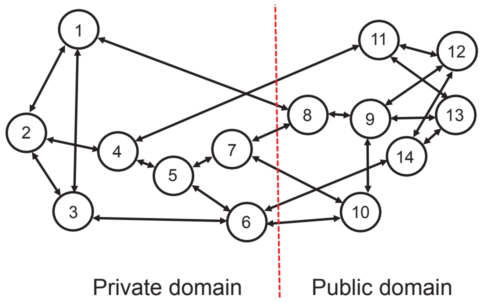


Fig. 4. MD-EON derived from the NSFNET topology.

used for provisioning vNFGs is an important metric to evaluate whether an algorithm can achieve high cost-efficiency. Therefore, we obtain the average cost as the total resource cost averaged by the number of vNFGs. We also consider the blocking probability of vNFGs in the online scenario. Note that, to evaluate the time efficiency of the algorithms, we present the results on the total running time for the offline scenario as well. Moreover, we design a benchmark algorithm with the straightforward approach, *i.e.*, it selects the DC with the most available IT resources to instantiate vNFs and adopts the same method as iTP-M2C to support the traffic among vNFs provisioned in different DCs iteratively. The benchmark algorithm is named as the first-fit algorithm (FF). To ensure sufficient statistic accuracy, we obtain each data point by averaging the results from 100 independent simulations.

B. Evaluations for Offline Scenario

We consider three types of vNFs. A *vNF* 1 needs 1 unit of CPU cycles, 3.75 units of memory and 4 units of storage, a *vNF* 2 needs 2 units of CPU cycles, 7.5 units of memory and 32 units of storage, and a *vNF* 3 needs 4 units of CPU cycles, 15 units of memory and 80 units of storage [43]. The costs of the IT resource usage for instantiating a *vNF* 1, a *vNF* 2 and a *vNF* 3 are 0.067, 0.133, and 0.266 units, respectively in the private domain, and the costs are doubled in the public domain [43]. There are three types of fiber links in the MD-EON, *i.e.*, the intra-links in the private domain, the inter-links in between the private and public domains, and the intra-links in the public domain, the unit costs of FS usage on them are set as 0.005, 0.01, and 0.02 units, respectively [43].

Firstly, we fix the scale of vNFG requests and change the number of requests in each group to observe the performance of the algorithms. Here for each vNFG, we assume that it randomly requires $[2, 3]$ instances of *vNF* 1, $[0, 2]$ instances of *vNF* 2, and $[0, 1]$ instances of *vNF* 3. Table I shows the simulation results with the six-node topology. As expected, vNFG-ILP provides the lowest average cost all the time, which verifies that it can get the optimal provisioning schemes for the vNFGs. In terms of the average cost, vNFG-ILP is followed by TP-MkC. The fact that the average costs from TP-MkC are the same as or only slightly higher than those from vNFG-ILP verifies its effectiveness on providing near-optimal solutions. The performance of iTP-M2C is worse than that of TP-MkC since it uses a greedy strategy to minimize the total cost of vNFG provisioning in each iteration. Among all the algorithms, FF performs the worst in terms of the average cost. Meanwhile, although vNFG-ILP can obtain the optimal solutions, its running time is also the longest and thus it can easily become intractable for large-scale problems. The running time of TP-MkC is much shorter than vNFG-ILP, followed by iTP-M2C and FF.

We then perform simulations with different vNFG sizes (*i.e.*, number of vNFs in each vNFG) and fix the number of vNFG requests in each group as 10. Table II shows the results on the average cost and running time. As we can see that vNFG-ILP always outperforms the other three algorithms in terms of the average cost, but its running time increases rapidly with the size of the vNFGs and it cannot finish the problem-solving within a reasonable time period, when the size of the vNFGs is larger than 12. TP-MkC still provides lower average cost than iTP-M2C and FF, and FF still performs the worst in terms of the average cost. The running time of these four algorithms also follows a similar trend as that in Table I. Note that, although the average running time per vNFG of TP-MkC is longer than that of iTP-M2C and FF, it can finish the computation within 0.5 second for provisioning a vNFG whose size is 12, which is still reasonably good for practical network operation.

C. Evaluations for Online Scenario

For the online scenario, the vNFG requests can be blocked due to the insufficiency of spectrum resources on fiber links, multi-dimensional IT resources in DCs, or both. This time, we only consider the NSFNET topology. On each DC, the CPU, memory and storage resources range randomly within $[50, 120]$ units, $[150, 400]$ units, and $[500, 1500]$ units, respectively. Each fiber link carries 1000 FS'. For each vNFG, we assume that it randomly requires $[2, 4]$ instances of *vNF* 1, $[0, 4]$ instances of *vNF* 2, and $[0, 2]$ instances of *vNF* 3. The rest of the simulation parameters are the same as those in the offline scenario. Each simulation runs for $10,000$ time units.

1) Basic Performance Comparison: Fig. 5 shows the simulation results on blocking probability. We observe that the blocking performance of TP-MkC is the best, followed by iTP-M2C, while FF performs the worst. This is because FF does not consider the traffic demands among the vNFs when instantiating them, which would cause unnecessary and unbalanced spectrum utilization in the MD-EON and eventually lead to the blocking of vNFGs. On the other hand, since iTP-M2C and TP-MkC properly address the traffic demands when designing the vNF-DC mapping schemes, they achieve better utilization of the spectrum resources.

Fig. 6 plots the results on average total resource cost. It is interesting to notice that FF also provides the highest average resource cost, even though its blocking performance is the worst. This further confirms the cost-efficiency of iTP-M2C and TP-MkC. When comparing the results from iTP-M2C and TP-MkC, we can see that the average resource cost from TP-MkC is lower than that from iTP-M2C only when the traffic load is lower than 60 Erlangs. This is because since the blocking probability of iTP-M2C is higher than that of TP-MkC, TP-MkC would serve more requests using resources at

TABLE I
RESULTS WITH FIXED VNFG SCALE UNDER SIX-NODE TOPOLOGY FOR OFFLINE SCENARIO.

Number of vNFGs	Average cost per vNFG (units)				Average running time per vNFG (seconds)			
	vNFG-ILP	iTP-M2C	TP-MkC	FF	vNFG-ILP	iTP-M2C	TP-MkC	FF
10	0.455	0.544	0.473	0.638	1.107e+2	1.138e-3	4.953e-2	4.732e-4
20	0.625	0.652	0.629	0.672	93.922	1.071e-3	4.592e-2	4.673e-4
30	0.606	0.654	0.606	0.681	2.366e+2	8.897e-4	5.350e-2	4.533e-4
40	0.594	0.645	0.598	0.674	3.303e+2	9.720e-4	4.942e-2	4.512e-4
50	0.621	0.649	0.621	0.664	2.788e+2	8.295e-4	4.854e-2	4.488e-4
60	0.559	0.597	0.562	0.647	2.954e+2	9.503e-4	5.065e-2	4.186e-4
70	0.554	0.616	0.559	0.652	4.422e+2	1.097e-3	5.292e-2	4.467e-4

TABLE II
RESULTS WITH FIXED NUMBER OF VNFGS UNDER SIX-NODE TOPOLOGY FOR OFFLINE SCENARIO.

Size of vNFGs	Average cost per vNFG (units)				Average running time per vNFG (seconds)			
	vNFG-ILP	iTP-M2C	TP-MkC	FF	vNFG-ILP	iTP-M2C	TP-MkC	FF
4	0.524	0.606	0.560	0.680	96.285	9.947e-4	4.819e-2	4.672e-4
8	1.270	1.436	1.345	1.579	3.519e+3	4.742e-3	0.218	8.405e-4
12	1.963	2.106	2.066	2.452	2.970e+4	3.132e-2	0.412	2.131e-3

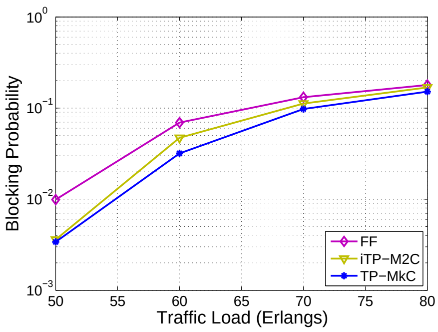


Fig. 5. Blocking probability for online scenario.

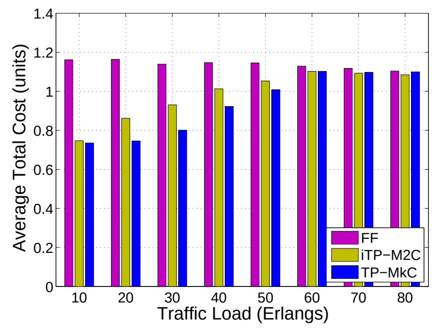


Fig. 6. Average total resource cost for online scenario.

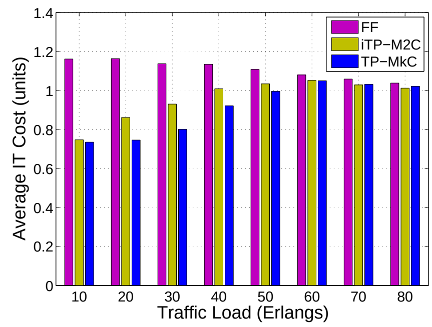


Fig. 7. Average IT cost for online scenario.

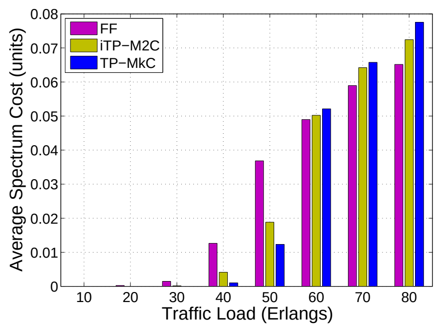


Fig. 8. Average spectrum cost for online scenario.

high price when the MD-EON is relatively crowded. We then investigate the cost components for the vNFG provisioning, and Figs. 7 and 8 show the results on average IT and spectrum costs, respectively. It can be seen that FF always provides the highest average IT cost. For the average spectrum cost, FF provides the highest results when the traffic load is less than 50 Erlangs and then its results are the lowest among the three algorithms. Again, these results confirm that FF cannot balance the IT and spectrum usages well. The average IT cost from iTP-M2C is higher than that from TP-MkC when the traffic load is less than 60 Erlangs. The average spectrum cost from TP-MkC is the lowest when the traffic load is less than 50 Erlangs, but it increases much faster with the traffic load than that from iTP-M2C and becomes higher eventually. Hence, we can conclude that TP-MkC outperforms iTP-M2C when the traffic load is relatively low, and for the high traffic load cases, we should use iTP-M2C instead.

2) Distribution of Resource Utilization in the MD-EON: Table III shows the distributions of resource utilization in the MD-EON. Basically, for the IT resources (*i.e.*, CPU cycles, memory, and storage), we show the percentages of resource utilization in the private and public domains, while for the spectrum resources, we provide the percentages of FS' usage on the three types of links in the MD-EON, *i.e.*, the intra-links in the private domain (Private), the inter-links in between the

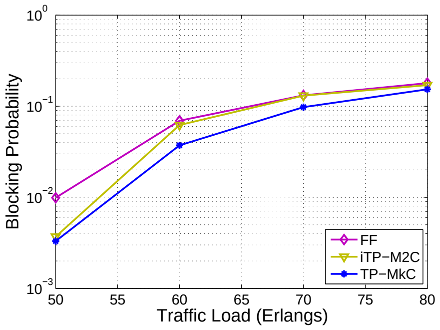


Fig. 9. Blocking probability for online scenario (high IT price setting).

private and public domains (Inter), and the intra-link in the public domain (Public). It can be seen that when the traffic load is low, iTP-M2C and TP-MkC try to use the network resources in the private domain. Only when the traffic load increases and the private domain can not serve the requests , they consider to use the network resources out of the private domain, because iTP-M2C and TP-MkC arrange the IT and spectrum resource utilizations in a more intelligent way to better balance the costs of IT and spectrum consumptions. However, FF uses almost equal resources in the private and public domains for the reason that it only considers the quantity of resource in each DC. These results verify that our algorithms can properly address different resource prices in the MD-EON to achieve cost-efficient vNFG provisioning.

3) Impact of Resource Price Setting: Finally, we run simulations with different resource price settings to further confirm the robustness of our proposed algorithms. Firstly, we keep the price setting on spectrum resources unchanged but increase the prices of the IT resources in the public domain to four times of those in the private domain. This simulation scenario is referred to as the "high IT price setting". Fig. 9 shows the results on blocking probability, which exhibit the similar trend as that in Fig. 5. Table IV presents the results on the average total resource cost, average IT cost and average spectrum cost. We observe that because the price of IT resources in the public domain is higher, the average IT cost increases a lot for all the algorithms. However, the general trends among the algorithms for the costs are still the same as those in Figs. 6-8. Table V shows the distributions of resource utilization in the MD-EON. It can be seen clearly that due to the fact that the price of IT resources in the public domain is higher, iTP-M2C and TP-MkC tend to use more IT resources in the private domain, when being compared with the results in Table III. Nevertheless, without this type of intelligence, FF does not change the distribution of resource utilization.

Secondly, we keep the price setting on IT resources unchanged but increase the price difference on the spectrum resources, *i.e.*, setting the prices of per FS usages on the intra-links in the private domain, the inter-links in between the private and public domains, and the intra-link in the public domain as 0.005, 0.1, and 0.2 units, respectively. This simulation scenario is referred to as the "high spectrum price setting". The results on blocking probability in Fig. 10 still show the similar trend as that in Fig. 5. Table VI illustrates the results on the average total resource cost, average IT cost and average bandwidth cost, which still exhibit the similar trends as those in Figs. 6-8. Finally, the results on the distributions of resource utilization in the MD-EON in Table VII also show that iTP-M2C and TP-MkC can adjust the resource utilizations intelligently based on the resource prices.

Note that, in addition to the aforementioned scenarios, we also simulate other scenarios to further verify the performance of iTP-M2C and TP-MkC, *i.e.*, changing the vNFGs with random topologies to pure vNF chains or vNF trees and changing the MD-EON's topology to the practical ones in [42]. Since the simulation results still exhibit the similar trends as discussed above, we omit them due to the page limit.

TABLE IV
AVERAGE COSTS FOR ONLINE SCENARIO (HIGH IT PRICE SETTING) (UNITS).

Erlangs		20	50	80
Total cost	FF	2.01	1.91	1.82
	iTP-M2C	0.93	1.62	1.73
	TP-MkC	0.76	1.58	1.74
IT cost	FF	2.01	1.87	1.75
	iTP-M2C	0.93	1.60	1.65
	TP-MkC	0.76	1.52	1.66
Spectrum cost	FF	0	0.04	0.07
	iTP-M2C	0	0.02	0.08
	TP-MkC	0	0.06	0.08

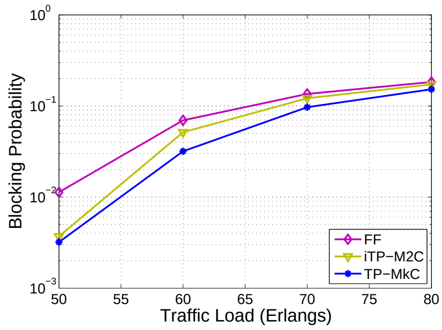


Fig. 10. Blocking probability for online scenario (high spectrum price setting).

VI. CONCLUSIONS

In this paper, we studied how to provision vNFGs cost-effectively in an MD-EON that consists of two domains,

TABLE VI
AVERAGE COSTS FOR ONLINE SCENARIO (HIGH SPECTRUM PRICE SETTING) (UNITS).

Erlangs		20	50	80
Total cost	FF	1.17	1.42	1.58
	iTP-M2C	0.86	1.18	1.60
	TP-MkC	0.75	1.04	1.55
IT cost	FF	1.16	1.11	1.04
	iTP-M2C	0.86	1.04	1.01
	TP-MkC	0.75	1.00	1.02
Spectrum cost	FF	0.01	0.31	0.54
	iTP-M2C	0	0.14	0.59
	TP-MkC	0	0.04	0.53

TABLE III
DISTRIBUTION OF RESOURCE UTILIZATION IN THE MD-EON FOR ONLINE SCENARIO.

Erlangs		20			50			80		
Algorithms		FF	iTP-M2C	TP-MkC	FF	iTP-M2C	TP-MkC	FF	iTP-M2C	TP-MkC
CPU cycles	Total	1.11e+5	1.11e+5	1.11e+5	1.08e+5	1.09e+5	1.09e+5	8.39e+4	8.59e+4	8.81e+4
	Private	42.8%	83.6%	99.3%	47.7%	58.4%	63.6%	47.7%	52.8%	52.3%
	Public	57.2%	16.4%	0.7%	52.3%	41.6%	36.4%	52.3%	47.2%	47.7%
Memory	Total	4.17e+5	4.17e+5	4.17e+5	4.05e+5	4.09e+5	4.10e+5	3.15e+5	3.22e+5	3.30e+5
	Private	42.8%	83.6%	99.3%	47.7%	58.4%	63.6%	47.7%	52.8%	52.3%
	Public	57.2%	16.4%	0.7%	52.3%	41.6%	36.4%	52.3%	47.2%	47.7%
Storage	Total	1.58e+6	1.58e+6	1.58e+6	1.53e+6	1.55e+6	1.55e+6	1.16e+6	1.19e+6	1.23e+6
	Private	42.9%	83.6%	99.2%	47.9%	58.1%	61.4%	48.1%	51.9%	50.1%
	Public	57.1%	16.4%	0.8%	52.1%	41.9%	38.6%	51.9%	48.1%	49.9%
Spectrum	Total	157	0	0	3.20e+4	2.17e+4	1.43e+4	4.64e+4	5.65e+4	6.18e+4
	Private	17.8%	—	—	38.2%	55.3%	55.6%	38.7%	44.4%	43.6%
	Inter	17.8%	—	—	28.6%	30.8%	30.4%	26.8%	26.8%	28.1%
	Public	64.4%	—	—	33.2%	13.9%	14%	34.5%	28.8%	28.3%

TABLE V
DISTRIBUTION OF RESOURCE UTILIZATION IN THE MD-EON FOR ONLINE SCENARIO (HIGH IT PRICE SETTING).

Erlangs		20			50			80		
Algorithms		FF	iTP-M2C	TP-MkC	FF	iTP-M2C	TP-MkC	FF	iTP-M2C	TP-MkC
CPU	Total	1.11e+5	1.11e+5	1.11e+5	1.08e+5	1.09e+5	1.09e+5	8.39e+4	8.56e+4	8.77e+4
	Private	42.8%	91.3%	99.4%	47.7%	60.5%	64.1%	47.7%	53.2%	53.2%
	Public	57.2%	8.7%	0.6%	52.3%	39.5%	35.9%	52.3%	46.8%	46.8%
Memory	Total	4.17e+5	4.17e+5	4.17e+5	4.05e+5	4.09e+5	4.10e+5	3.15e+5	3.21e+5	3.29e+5
	Private	42.8%	91.3%	99.4%	47.7%	60.5%	64.1%	47.7%	53.2%	53.2%
	Public	57.2%	8.7%	0.6%	52.3%	39.5%	35.9%	52.3%	46.8%	46.8%
Storage	Total	1.58e+6	1.58e+6	1.58e+6	1.53e+6	1.55e+6	1.55e+6	1.16e+6	1.19e+6	1.22e+6
	Private	42.9%	91.3%	99.3%	47.9%	60.2%	62.5%	48.1%	52.2%	51%
	Public	57.1%	8.7%	0.7%	52.1%	39.8%	37.5%	51.9%	47.8%	49%
Spectrum	Total	157	75	766	3.20e+4	3.33e+4	5.31e+4	4.64e+4	5.72e+4	6.57e+4
	Private	17.8%	66.7%	53.1%	38.2%	55.6%	40.2%	38.7%	43.6%	41.5%
	Inter	17.8%	33.3%	46.9%	28.6%	29.7%	28.7%	26.8%	27.2%	26.6%
	Public	64.4%	0	0	33.2%	14.7%	31.1%	34.5%	29.2%	31.9%

TABLE VII
DISTRIBUTION OF RESOURCE UTILIZATION IN THE MD-EON FOR ONLINE SCENARIO (HIGH SPECTRUM PRICE SETTING).

Erlangs		20			50			80		
Algorithms		FF	iTP-M2C	TP-MkC	FF	iTP-M2C	TP-MkC	FF	iTP-M2C	TP-MkC
CPU	Total	1.11e+5	1.11e+5	1.11e+5	1.08e+5	1.09e+5	1.09e+5	8.35e+4	8.52e+4	8.81e+4
	Private	42.8%	84.4%	99.3 %	48.1%	58.1%	63.1%	47.8%	52.7%	52.2%
	Public	57.2%	15.6%	0.7%	51.9%	41.9%	36.9%	52.2%	47.3%	47.8%
Memory	Total	4.17e+5	4.17e+5	4.17e+5	4.05e+5	4.09e+5	4.10e+5	3.13e+5	3.19e+5	3.30e+5
	Private	42.8%	84.4%	99.3%	48.1%	58.1%	63.1%	47.8%	52.7%	52.2%
	Public	57.2%	15.6%	0.7%	51.9%	41.9%	36.9%	52.2%	47.3%	47.8%
Storage	Total	1.58e+6	1.58e+6	1.58e+6	1.53e+6	1.55e+6	1.55e+6	1.16e+6	1.18e+6	1.23e+6
	Private	42.9%	84.4%	99.2%	48.3%	57.9%	61%	48.1%	51.7%	49.9%
	Public	57.1%	15.6%	0.8%	51.7%	42.1%	39%	51.9%	48.3%	50.1%
Spectrum	Total	157	0	0	3.20e+4	2.48e+4	7.99e+3	4.55e+4	5.63e+4	5.44e+4
	Private	17.8%	—	—	38.7%	58.9%	65.1%	39.5%	44.4%	47.6%
	Inter	17.8%	—	—	28.2%	28.2%	22.6%	26%	27.1%	25.5%
	Public	64.4%	—	—	33.1%	12.9%	12.3%	34.5%	28.5%	26.9%

i.e., the private and public ones. Because the network model and optimization objective of our problem are different from those of existing studies on VNE and vNF chain provisioning, the algorithms designed in them cannot be applied. We first formulated an ILP model with the objective to minimize the total resource cost of vNFG provisioning, and showed that it can solve the problem exactly. With the ILP model, we analyzed the complexity of the problem and proved its $\mathcal{NP}$-hardness. Then, we leveraged the minimum k-cut problem to design two time-efficient heuristics. The results from extensive simulations verified that our proposed algorithms outperformed an existing benchmark in terms of blocking probability and average total cost in different simulation scenarios. More specifically, our algorithms arranged the IT and spectrum resources intelligently to balance their usages according to the price setting in the MD-EON.

ACKNOWLEDGMENTS

This work was supported in part by the NSFC Project 61371117, Natural Science Research Project for Universities in Anhui (KJ2014ZD38), and the SPR Program of CAS (XDA06011202).

REFERENCES

[1] T. Anderson, L. Peterson, S. Shenker, and J. Turner, “Overcoming the Internet impasse through virtualization,” *IEEE Comput.*, vol. 38, pp. 34–41, Apr. 2005.

[2] L. Gong, H. Jiang, Y. Wang, and Z. Zhu, “Novel location-constrained virtual network embedding (LC-VNE) algorithms towards integrated node and link mapping,” *IEEE/ACM Trans. Netw.*, vol. 24, pp. 3648–3661, Dec. 2016.

[3] “Network functions virtualisation (NFV),” Jan. 2012. [Online]. Available: https://portal.etsi.org/portal/server.pt/community/NFV/367

[4] H. Moens and F. De Turck, “VNF-P: A model for efficient placement of virtualized network functions,” in *Proc. of CNSM 2014*, pp. 418–423, Nov. 2014.

[5] S. Clayman *et al.*, “The dynamic placement of virtual network functions,” in *Proc. of NOMS 2014*, pp. 1–9, May 2014.

[6] M. Zeng, W. Fang, J. Rodrigues, and Z. Zhu, “Orchestrating multicast-oriented NFV trees in inter-DC elastic optical networks,” in *Proc. of ICC 2016*, pp. 1–6, Jun. 2016.

[7] W. Fang *et al.*, “Joint spectrum and IT resource allocation for efficient vNF service chaining in inter-datacenter elastic optical networks,” *IEEE Commun. Lett.*, vol. 20, pp. 1539–1542, Aug. 2016.

[8] Q. Sun, P. Lu, W. Lu, and Z. Zhu, “Forecast-assisted NFV service chain deployment based on affiliation-aware vNF placement,” in *Proc. of GLOBECOM 2016*, pp. 1–6, Dec. 2016.

[9] A. Jakaria *et al.*, “VFence: A defense against distributed denial of service attacks using network function virtualization,” in *Proc. of COMPSAC 2016*, vol. 2, pp. 431–436, Jun. 2016.

[10] T. Pham and L. Pham, “Load balancing using multipath routing in network functions virtualization,” in *Proc. of RIVF 2016*, pp. 85–90, Nov. 2016.

[11] S. Mehraghdam, M. Keller, and H. Karl, “Specifying and placing chains of virtual network functions,” in *Proc. of CloudNet 2014*, pp. 7–13, Oct. 2014.

[12] M. Xia *et al.*, “Network function placement for NFV chaining in packet/optical datacenters,” *J. Lightw. Technol.*, vol. 33, pp. 1565–1570, Apr. 2015.

[13] S. Zhang, Q. Zhang, H. Bannazadeh, and A. Leon-Garcia, “Routing algorithms for network function virtualization enabled multicast topology on SDN,” *IEEE Trans. Netw. Serv. Manag.*, vol. 12, pp. 580–594, Dec. 2015.

[14] M. Zeng, W. Fang, and Z. Zhu, “Orchestrating tree-type VNF forwarding graphs in inter-DC elastic optical networks,” *J. Lightw. Technol.*, vol. 34, pp. 3330–3341, Jul. 2016.

[15] P. Lu *et al.*, “Highly-efficient data migration and backup for big data applications in elastic optical inter-datacenter networks,” *IEEE Netw.*, vol. 29, pp. 36–42, Sep./Oct. 2015.

[16] Z. Zhu, W. Lu, L. Zhang, and N. Ansari, “Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing,” *J. Lightw. Technol.*, vol. 31, pp. 15–22, Jan. 2013.

[17] S. Hong *et al.*, “Virtual optical network embedding in multi-domain optical networks,” in *Proc. of GLOBECOM 2014*, pp. 2042–2047, Dec. 2014.

[18] Z. Zhu *et al.*, “Demonstration of cooperative resource allocation in an OpenFlow-controlled multidomain and multinational SD-EON testbed,” *J. Lightw. Technol.*, vol. 33, pp. 1508–1514, Apr. 2015.

[19] J. Zhu *et al.*, “Service provisioning with energy-aware regenerator allocation in multi-domain EONs,” in *Proc. of GLOBECOM 2015*, pp. 1–6, Dec. 2015.

[20] P. Lu, Q. Sun, K. Wu, and Z. Zhu, “Distributed online hybrid cloud management for profit-driven multimedia cloud computing,” *IEEE Trans. Multimedia*, vol. 17, pp. 1297–1308, Aug. 2015.

[21] C. Lim, S. Bohacek, J. Hespanha, and K. Obraczka, “Hierarchical max-flow routing,” in *Proc. of GLOBECOM 2005*, pp. 545–550, Dec. 2005.

[22] J. Zhu, B. Zhao, W. Lu, and Z. Zhu, “Attack-aware service provisioning to enhance physical-layer security in multi-domain EONs,” *J. Lightw. Technol.*, vol. 34, pp. 2645–2655, Jun. 2016.

[23] R. Casellas *et al.*, “Control and management of flexi-grid optical networks with an integrated stateful path computation element and OpenFlow controller,” *J. Opt. Commun. Netw.*, vol. 5, pp. A57–A65, Oct. 2013.

[24] Z. Zhu *et al.*, “OpenFlow-assisted online defragmentation in single-/multi-domain software-defined elastic optical networks,” *J. Opt. Commun. Netw.*, vol. 7, pp. A7–A15, Jan. 2015.

[25] P. Iovanna *et al.*, “Main challenges on WAN due to NFV and SDN: multi-layer and multi-domain network virtualization and routing,” in *Proc. of ONDM 2015*, pp. 74–79, May 2015.

[26] T. Duong-Ba, T. Nguyen, B. Bose, and T. Tran, “Joint virtual machine placement and migration scheme for datacenters,” in *Proc. of GLOBECOM 2014*, pp. 2320–2325, Dec. 2014.

[27] K. Chen, Y. Xu, K. Xi, and H. Chao, “Intelligent virtual machine placement for cost efficiency in geo-distributed cloud systems,” in *Proc. of ICC 2013*, pp. 3498–3503, Jun. 2013.

[28] G. Long, Y. Wen, Z. Zhu, and T. Lee, “Toward profit-seeking virtual network embedding algorithm via global resource capacity,” in *Proc. of INFOCOM 2014*, pp. 1–9, Apr. 2014.

[29] X. Cheng *et al.*, “Virtual network embedding through topology-aware node ranking,” *SIGCOMM Comput. Commun. Rev.*, vol. 41, pp. 39–47, Apr. 2011.

[30] M. Chowdhury, M. Rahman, and R. Boutaba, “ViNEYard: Virtual network embedding algorithms with coordinated node and link mapping,” *IEEE/ACM Trans. Netw.*, vol. 20, pp. 206–219, Jan. 2012.

[31] J. Botero *et al.*, “Energy efficient virtual network embedding,” *IEEE Commun. Lett.*, vol. 16, pp. 756–759, May 2012.

[32] L. Gong, W. Zhao, Y. Wen, and Z. Zhu, “Dynamic transparent virtual network embedding over elastic optical infrastructures,” in *Proc. of ICC 2013*, pp. 3466–3470, Jun. 2013.

[33] L. Gong and Z. Zhu, “Virtual optical network embedding (VONE) over elastic optical networks,” *J. Lightw. Technol.*, vol. 32, pp. 450–460, Feb. 2014.

[34] H. Jiang, Y. Wang, L. Gong, and Z. Zhu, “Availability-aware survivable virtual network embedding (A-SVNE) in optical datacenter networks,” *J. Opt. Commun. Netw.*, vol. 7, pp. 1160–1171, Dec. 2015.

[35] L. Nonde, T. El-Gorashi, and J. Elmirghani, “Energy efficient virtual network embedding for cloud networks,” *J. Lightw. Technol.*, vol. 33, pp. 1828–1849, May 2015.

[36] F. Ji *et al.*, “Dynamic p-cycle protection in spectrum-sliced elastic optical networks,” *J. Lightw. Technol.*, vol. 32, pp. 1190–1199, Mar. 2014.

[37] J. Yen, “Finding the K shortest loopless paths in a network,” *Manag. Sci.*, vol. 17, pp. 712–716, Jul. 1971.

[38] W. Fang *et al.*, “Joint defragmentation of optical spectrum and it resources in elastic optical datacenter interconnections,” *J. Opt. Commun. Netw.*, vol. 7, pp. 314–324, Apr. 2015.

[39] M. Garey and D. Johnson, *Computers and Intractability: a Guide to the Theory of NP-Completeness*. W. H. Freeman & Co. New York, 1979.

[40] “Partition of a set.” [Online]. Available: https://en.wikipedia.org/wiki/Partition_of_a_set#Counting_partitions

[41] “K-means clustering.” [Online]. Available: https://en.wikipedia.org/wiki/K-means_clustering

[42] B. Jaumard and T. Do, “Resilient distributed design of very large multi-domain optical networks,” in *Proc. of Networks 2014*, pp. 1–7, Sep. 2014.

[43] “Amazon EC2 pricing.” [Online]. Available: http://aws.amazon.com/ec2/pricing/