Joint Spectrum and IT Resource Allocation for Efficient VNF Service Chaining in Inter-Datacenter Elastic Optical Networks

Wenjian Fang, Menglu Zeng, Xiahe Liu, Wei Lu, and Zuqing Zhu, *Senior Member, IEEE*

***Abstract*—We study how to allocate spectrum and IT resources jointly for realizing efficient virtual network function (VNF) service chaining in inter-datacenter elastic optical networks (inter-DC EONs). We first formulate an integer linear programming (ILP) model to solve the problem exactly, and then a longest common subsequence (LCS) based heuristic is proposed. Simulation results indicate that the proposed algorithms can reuse the deployed VNFs efficiently and arrange the spectrum utilization in a much more load-balanced manner.**

***Index Terms*—Network function virtualization (NFV), Service chaining, Datacenter (DC), Elastic optical networks (EONs).**

I. Introduction

NOWADAYS, network function virtualization (NFV) [1] becomes attractive as it can expedite the deployment of new network services. Specifically, with NFV, network operators can realize virtual network functions (VNFs) using generic network resources (*e.g.*, bandwidth, CPU cycles and memory space) to replace the special-purpose network elements that are expensive and difficult to maintain and upgrade. To support data-/bandwidth-intensive applications, VNFs can be deployed on high-performance servers in datacenters (DCs). Then, the deployment of new network services can be easily realized by routing data traffic through a series of VNFs on DC(s) (*i.e.*, service chaining (SC) [1]). Note that SC generally needs to place VNFs sequentially [2].

It is known that optical SC has the advantages of high bandwidth capacity and low power consumption [2, 3], which are especially beneficial for inter-DC networks. Also, we hope to point out that the elasticity of the optical infrastructure can affect the efficiency of SC in inter-DC networks significantly [2]. The reasons are generally two-fold. Firstly, the average data-rate of a traffic flow can change when it goes through an SC. For instance, a VNF can decrease or increase the data-rate, if it instantiates a fire-wall or a video optimizer, respectively [1]. Secondly, due to the dynamic nature of data-/bandwidth-intensive applications, the traffic flowing through VNF-based SCs (VNF-SCs) can exhibit high burstiness [4]. Fortunately, we know that elastic optical networks (EONs) realize agile bandwidth management in the optical layer, and thus traffics with bursty/various bandwidth demands can be provisioned more efficiently [4]. Hence, it would be relevant to study how to realize VNF-SCs efficiently in inter-DC EONs.

We need to jointly allocate the spectrum and IT resources in inter-DC EONs to provision VNF-SCs. This includes both the VNF placement in DCs and the routing and spectrum assignment (RSA) on fibers to connect the clients to their required VNFs. Note that, with the network orchestration that leverages software-defined networking (SDN), joint allocation of optical and cloud resources can be realized efficiently [5]. However, the network planning algorithm for the orchestration is still under explored. Although the problem of VNF placement has already been studied in [2], the VNF placement was not optimized jointly with the spectrum allocation on fibers. Meanwhile, the problem studied in this work is different from the anycast-based RSA [6, 7] for optimizing cloud traffics, which only considers how to assign spectrum resources (*i.e.*, frequency slots (FS')) in inter-DC EONs.

This letter studies how to solve the static network planning that allocates spectrum and IT resources jointly to realize VNF-SCs efficiently in inter-DC EONs. To the best of our knowledge, this is the first attempt to address the problem with a joint and coordinated approach. It is essential to do so because separated approaches would lead to sub-optimal solutions [8]. The rest of the letter is organized as follows. We describe the problem in Section II. Section III presents an integer linear programming (ILP) model to solve it exactly, while a heuristic algorithm is proposed in Section IV. We present the simulation results and related discussions in Section V. Finally, Section VI summarizes the paper.

II. Problem Description

We model the inter-DC EON as a directed graph $G(V, E)$, where V and E represent the sets of nodes and fiber links in it, respectively. Each node $v \in V$ includes a bandwidth-variable optical switch, some of which are locally attached to a DC, and VNFs can be deployed in the DC. A client VNF-SC request is denoted as $R_i(s_i, d_i, T_i, B_i)$, where i is its unique index, and s_i and d_i are the source and destination nodes, respectively. $T_i = \langle t_{i,1}, t_{i,2}, ..., t_{i,J_i}\rangle$ denotes the VNF sequence in the VNF-SC, where $t_{i,j}$ is the type of the j-th VNF and J_i is the total number of VNFs. $B_i = \langle b_{i,0}, b_{i,1}, b_{i,2}, ..., b_{i,J_i}\rangle$ indicates the bandwidth requirements in terms of FS', where $b_{i,0}$ is the initial bandwidth requirement and $b_{i,j}$ is the requirement after steering through *VNF* $t_{i,j}$. Here, we address the practical scenario in which the bandwidth requirement of the VNF-SC can change after steering through a VNF. Note that, since we only address the static network planning problem in this work, all the requests are assumed to be known in advance.

Fig. 1 shows an intuitive example of provisioning VNF-SCs in an inter-DC EON. The requests are $R_1(1, 6, \langle \textit{VNF1}, \textit{VNF2}\rangle, \langle 2, 1, 2\rangle)$ and $R_2(1, 4, \langle \textit{VNF1}, \textit{VNF2}\rangle, \langle 2, 1, 2\rangle)$. Fig. 1(a) shows a provisioning scheme, where R_1 takes the path 1→2→4→6 and deploys *VNF1* and *VNF2* on *Nodes* 2 and 4, respectively, and R_2 goes through 1→2→4 and reuses the VNFs that have been deployed in the network.

W. Fang, M. Zeng, X. Liu, W. Lu and Z. Zhu are with the School of Information Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, P. R. China (email: zqzhu@ieee.org).

Manuscript received November 25, 2015.

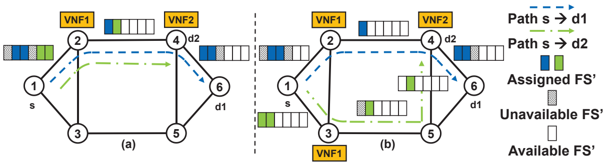


Fig. 1. Example of provisioning VNF-SCs in inter-DC EONs.

For this scheme, we deploy two VNFs and the network's maximum FS' index (MFSI) is 6. Another scheme is illustrated in Fig. 1(b), where R_2 takes the path 1→3→5→4, only reuses the deployed *VNF2*, and instantiates a new *VNF1* on *Node* 3. Hence, there are three deployed VNFs and the MFSI is 4. To this end, we can see that to realize VNF-SCs in an inter-DC EON efficiently, it is essential to allocate spectrum and IT resources jointly, since the VNF placement and RSA are correlated and we need to balance the utilizations of spectrum and IT resources carefully.

III. ILP Formulation

This section formulates an ILP model to solve the problem of provisioning VNF-SCs in inter-DC EONs exactly[1].

Notations:

- $\{R_i(s_i, d_i, T_i, B_i)\}$: set of requests.
- $V_{i,j}$: set of feasible nodes for deploying *VNF* $t_{i,j}$.
- T: set of all the possible VNF types.
- F: number of FS' on each fiber link $e \in E$.
- $P_{u,v}$: path set that contains K shortest routing paths from u to v, where $u, v \in V$ and $|P_{u,v}| = K$.
- $G_{i,j,p}$: set of available FS-blocks that each contains $b_{i,j}$ FS' on path p to connect the VNFs on nodes v_j and v_{j+1}[2], where $p \in P_{v_j,v_{j+1}}$, $v_j \in V_{i,j}$, $v_{j+1} \in V_{i,j+1}, j \in [0, J_i]$.
- $L_{i,v_j,v_{j+1}}$: set of RSA solutions for $v_j \to v_{j+1}$, in which each element l is a tuple $\langle p, g \rangle$, *i.e.*, a path $p \in P_{v_j,v_{j+1}}$ and an available FS-block $g \in G_{i,j,p}$ on it, where $v_j \in V_{i,j}$, $v_{j+1} \in V_{i,j+1}, j \in [0, J_i]$.
- L_i: set of RSA solutions for connecting the VNFs of R_i, *i.e.*, $L_i = \bigcup_{j \in [1, J_i)} L_{i,v_j,v_{j+1}}$.

Variables:

- x_{i,v_j}: boolean variable that equals 1 if *VNF* $t_{i,j}$ chooses node $v_j \in V_{i,j}$, and 0 otherwise.
- $y_{i,l,j}$: boolean variable that equals 1 if R_i chooses RSA solution l to connect $v_j \to v_{j+1}$, and 0 otherwise.
- $h_{v,t}$: boolean variable that equals 1 if a type $t \in T$ VNF is deployed on node $v \in V$, and 0 otherwise.
- $z_{e,f}$: boolean variable that equals 1 if the f-th FS on link e is used, and 0 otherwise.
- f^{max}: integer variable that indicates the MFSI.

Objective:

The objective is to minimize the MFSI and deployed VNFs in the network jointly, as

$$Minimize \quad (\alpha \cdot \frac{f^{max}}{F} + \beta \cdot \frac{1}{|V| \cdot |T|} \cdot \sum_{v \in V} \sum_{t \in T} h_{v,t}), \quad (1)$$

where α and β are the factors introduced to adjust the importance of the two terms, and should be set based on the actual situation. Here, the first term reflects the normalized MFSI, and we try to minimize MFSI because a smaller MFSI means that the EON's spectrum utilization is organized in a more load-balanced manner. The second term represents the normalized value of the total number of deployed VNFs, and the efficiency of the IT resource utilization in the DCs can be improved if we reduce it. Note that, for simplicity, we do not restrict the maximum number of VNFs that a DC can support as the IT resources in a DC are usually abundant.

Constraints:

1) *VNF Placement Constraints*:

$$\sum_{v_j \in V_{i,j}} x_{i,v_j} = 1, \ \forall i, \ \forall j \in J_i, \quad (2)$$

$$h_{v_j,t_{i,j}} \geq x_{i,v_j}, \ \forall i, j. \quad (3)$$

Eq. (2) ensures that each VNF in request R_i chooses one and only one DC for deployment. Eq. (3) determines whether a corresponding VNF is deployed on a node.

2) *RSA Related Constraints:*

$$\sum_{l \in L_{i,s_i,v_1}} y_{i,l,0} = \begin{cases} x_{i,v_1}, & v_1 \neq s_i, \\ 0, & v_1 = s_i, \end{cases} \ \forall i, \ \forall v_1 \in V_{i,1}, \quad (4)$$

$$\sum_{l \in L_{i,v_{J_i},d_i}} y_{i,l,J_i} = \begin{cases} x_{i,v_{J_i}}, & v_{J_i} \neq d_i, \\ 0, & v_{J_i} = d_i, \end{cases} \ \forall i, \ \forall v_{J_i} \in V_{i,J_i}, \quad (5)$$

$$\begin{cases} x_{i,v_j} + x_{i,v_{j+1}} - 1 \leq \sum_{l \in L_{i,v_j,v_{j+1}}} y_{i,l,j} \leq 1, & v_j \neq v_{j+1}, \\ \sum_{l \in L_{i,v_j,v_{j+1}}} y_{i,l,j} = 0, & v_j = v_{j+1}, \end{cases} \quad (6)$$
$$\forall i, \ \forall j \in [1, J_i), \{v_j, v_{j+1} : v_j \in V_{i,j}, \ v_{j+1} \in V_{i,j+1}\},$$

$$\sum_i \left(\sum_{\substack{l \in L_{i,s_i,v_1}, \\ l = \langle p,g \rangle, \\ e \in p, f \in g}} y_{i,l,0} + \sum_{\substack{l \in L_i, \\ l = \langle p,g \rangle, \\ e \in p, f \in g}} y_{i,l,j} + \sum_{\substack{l \in L_{i,v_{J_i},d_i}, \\ l = \langle p,g \rangle, \\ e \in p, f \in g}} y_{i,l,J_i} \right) = z_{e,f}$$
$$\forall e \in E, \ \forall f \in F, \quad (7)$$

$$f^{max} \geq f \cdot z_{e,f}, \ \forall e \in E, \ \forall f \in F. \quad (8)$$

Eqs. (4)-(6) ensure that each VNF-SC gets proper RSA schemes to connect the source, intermediate VNFs, and destination correctly. Specifically, on a VNF-SC, one and only one RSA solution should be selected to connect two adjacent VNFs[3] if they are on different nodes, otherwise, no RSA scheme should be used. Eq. (7) ensures that the selected RSA solutions for all the VNF-SCs satisfy the spectrum non-overlapping constraint. Eq. (8) gets the MFSI in the network.

[1] We precalculate all the feasible RSA solutions as the ILP's inputs [9].

[2] Here, for the purpose of generalization, we denote $v_0 = s_i$ and $v_{J_i+1} = d_i$.

[3] Here, for the purpose of generalization, we can treat the source and destination as dummy VNFs, *i.e.*, as $t_{i,0}$ and t_{i,J_i+1}, respectively.

IV. Heuristic Algorithm

In this section, we propose a time-efficient heuristic by leveraging the concept of longest common subsequence.

Definition 1. *For a VNF-SC* $\varphi = \langle \text{VNF}(a_1), \dots, \text{VNF}(a_m) \rangle$, *a VNF sequence* $\hat{\varphi} = \langle \text{VNF}(c_1), \dots, \text{VNF}(c_k) \rangle$ *is its* ***subsequence*** *if there exists a strictly increasing index sequence* $\langle i_1, \dots, i_k \rangle$ *such that* $\text{VNF}(a_{i_j}) = \text{VNF}(c_j)$, $\forall j \in [1, k]$. *Here,* $\text{VNF}(a_{i_j}) = \text{VNF}(c_j)$ *means that the VNFs are of the same type. Then, for two VNF-SCs* φ *and* φ'*, a VNF sequence* $\hat{\varphi}$ *is a* ***common subsequence*** *of them if* $\hat{\varphi}$ *is a subsequence of both* φ *and* φ'*. Consequently, for any two VNF-SCs* φ *and* φ'*, we can obtain their* ***longest common subsequence (LCS)*** *as* $LCS(\varphi, \varphi').seq$ *with a length of* $LCS(\varphi, \varphi').len$.

Note that, LCS can be calculated in polynomial time with dynamic programming [10]. If we assume that φ is the VNF-SC of a new request R_i and φ' is the VNF sequence that has already been deployed on a routing path for $s_i \to d_i$, $LCS(\varphi, \varphi').seq$ can quantify the matching degree of φ and φ'. For example, for $\varphi = \langle \textit{VNF1}, \textit{VNF6}, \textit{VNF8} \rangle$ and $\varphi' = \langle \textit{VNF1}, \textit{VNF5}, \textit{VNF8}, \textit{VNF9} \rangle$, we have $LCS(\varphi, \varphi').seq = \langle \textit{VNF1}, \textit{VNF8} \rangle$, and $LCS(\varphi, \varphi').len = 2$.

Algorithm 1 shows the proposed LCS based algorithm (LBA) for provisioning VNF-SCs in inter-DC EONs, which considers the LCS' of VNF-SCs together with the spectrum utilization in the network jointly. For each request R_i, *Line* 2 obtains its VNF-SC φ. *Lines* 3-6 obtain the sequences of deployed VNFs ${\varphi_k}'$ along the path $p_k \in P_{s_i,d_i}$, and calculate $LCS(\varphi, {\varphi_k}')$, where P_{s_i,d_i} is the pre-calculated K shortest routing paths for $s_i \to d_i$. Next, *Line* 7 chooses the path $p_{k_m} \in P_{s_i,d_i}$ with the maximum LCS length $LCS(\varphi, {\varphi_{k_m}}').len$. If $LCS(\varphi, {\varphi_{k_m}}').len > 0$, *Line* 9 reuses certain VNFs of ${\varphi_{k_m}}'$, which have already been deployed on path p_{k_m}. After this step, for the remaining VNFs in φ that are not instantiated, *Lines* 10-12 deploy them with the **nearby principle**. Specifically, we first check the bandwidth requirements in between *VNFs* $t_{i,j}$ and $t_{i,j+1}$ and *VNFs* $t_{i,j+1}$ and $t_{i,j+2}$, *i.e.*, $b_{i,j}$ and $b_{i,j+1}$, respectively. Then, to save spectra, if $b_{i,j} \geq b_{i,j+1}$, we deploy *VNFs* $t_{i,j}$ and $t_{i,j+1}$ on the same DC, otherwise, *VNFs* $t_{i,j+1}$ and $t_{i,j+2}$ are put on the same DC.

If $LCS(\varphi, {\varphi_{k_m}}').len = 0$, *i.e.*, no deployed and reuseable VNF can be found on all the pre-calculated paths, *Line* 14 deploys VNFs with the **spectrum-saving principle**. Specifically, we find the minimum bandwidth requirement $b_{i,j} \in B_i$, deploy VNF sequence $\langle t_{i,1}, \dots, t_{i,j} \rangle$ on the DC that is the nearest to s_i, and put $\langle t_{i,j+1}, \dots, t_{i,J_i} \rangle$ on the DC that is the nearest to d_i. Note that, if a DC is locally attached to s_i or d_i, it is the nearest DC to them, otherwise, the nearest DC should be the one on the node that has the smallest hop-count to s_i or d_i. Hence, we ensure that the traffic transmitted in between DCs is minimized. Finally, *Line* 16, connects s_i, the deployed VNFs, and d_i with the RSA schemes that induce the smallest MFSI increase. The time complexity of LBA is $O(|I| \cdot |V|^2)$.

Fig. 2 provides an illustrative example for LBA. For VNF-SC φ, we first check the $K = 2$ pre-calculated paths for $s \to d$ and get a VNF sequence for each of them, *i.e.*, ${\varphi_1}'$ and ${\varphi_2}'$ in Fig. 2(a). It can be seen that related to the requested VNF-

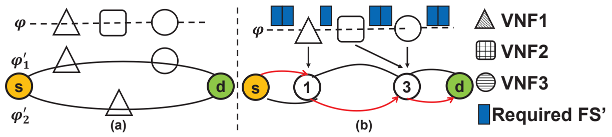


Fig. 2. An example of LBA.

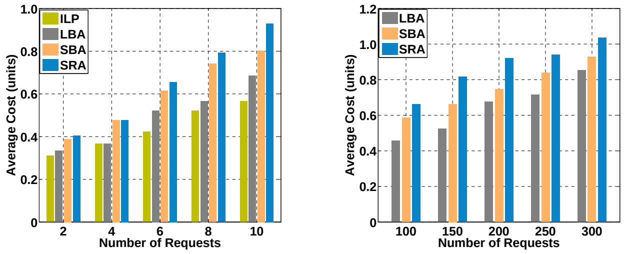
SC φ, ${\varphi_1}'$ has a longer LCS length than ${\varphi_2}'$, and thus LBA chooses to reuse the VNFs on it for realizing the first and third VNFs of φ. Then for the second VNF that is not instantiated, we find that it increases the bandwidth requirement. Hence, the nearby principle makes LBA deploy a new VNF for it on the same DC that carries the third VNF, as shown in Fig. 2(b). Finally, we connect s, the three deployed VNFs, and d with the RSA schemes on the $K = 2$ pre-calculated paths between each node pair, which cause the smallest MFSI increase.

Algorithm 1: LCS Based Algorithm (LBA)

1 **for** *each VNF-SC request* $R_i(s_i, d_i, T_i, B_i)$ **do**
2   get VNF-SC as $\varphi = T_i$;
3   **for** $k = 1$ *to* K **do**
4   get VNFs already deployed on $p_k \in P_{s_i,d_i}$ as sequence ${\varphi_k}'$;
5   calculate $LCS(\varphi, {\varphi_k}')$;
6   **end**
7   find $k_m = \arg \max_{k \in [1,K]} LCS(\varphi, {\varphi_k}').len$;
8   **if** $LCS(\varphi, {\varphi_{k_m}}').len > 0$ **then**
9   reuse VNFs of $LCS(\varphi, {\varphi_{k_m}}').seq$ on the corresponding DCs on p_{k_m};
10   **if** *required VNFs are not all instantiated* **then**
11   deploy remaining VNFs with **nearby principle**;
12   **end**
13   **else**
14   deploy VNFs with **spectrum-saving principle**;
15   **end**
16   connect s_i, deployed VNFs, and d_i with the RSA schemes that cause the smallest MFSI increase;
17 **end**

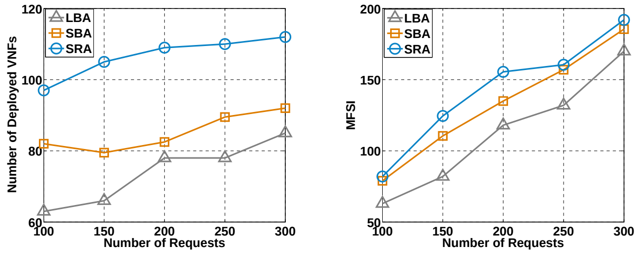
V. Performance Evaluation

We evaluate the performance of the proposed algorithms with two topologies, *i.e.*, the six-node topology in Fig. 1 and the 28-node US Backbone topology in [11]. In the simulations, the number of nodes that have local DCs is $\frac{1}{2}|V|$ and these nodes are selected from each topology randomly. For the six-node topology, there are $|T| = 3$ types of VNFs, each R_i asks for at most 2 VNFs, and each fiber accommodates $F = 10$ FS'. While for the US Backbone topology, we assume that there are $|T| = 8$ types of VNFs, each request R_i asks for at most 3 VNFs, and $F = 358$ FS'. To generalize the simulation scenarios, we choose the value of B_i randomly for each request, *i.e.*, with the averages as 1.5 FS' and 7



(a) Six-node topology (b) US Backbone topology

Fig. 3. Average cost for serving VNF-SCs.



(a) Number of deployed VNFs (b) MFSI

Fig. 4. Simulation results with US Backbone topology.

FS' in the six-node and US Backbone topologies, respectively. Eq. (1) uses $\alpha = \beta$, which means that the spectrum and IT resource utilizations are equally important. We also design two benchmark algorithms. The first one is the shortest-path and batch VNF deployment algorithm (SBA). For each request, SBA first selects the DC on the shortest path from s_i to d_i such that it can reuse the most existing VNFs (if there is no DC on the shortest path, SBA will try the second shortest path and so on so forth), then reuses/instantiates the VNFs accordingly, and finally connects the VNFs with the RSA scheme of LBA. The second one is the shortest-path and random VNF deployment algorithm (SRA), which is similar to SBA, with the exception that it randomly selects the DCs on the selected path to reuse/instantiate the VNFs.

Fig. 3(a) shows the results on the average costs, which are calculated with Eq. (1), from the algorithms for provisioning VNF-SCs in the six-node topology. The ILP achieves the lowest costs and it is followed by LBA. The costs from SBA and SRA are much higher. The results for the US Backbone topology are shown in Fig. 3(b), which indicate that the costs from LBA are 8%-22% and 17%-35% less than those from SBA and SRA, respectively. Fig. 4 shows the results on the number of deployed VNFs and MFSI in the US Backbone topology, which also verify that compared with SBA and SRA, LBA can reduce both of them in the network planning.

Table I shows the running time of the algorithms. We find that for the simulations with the six-node topology, the ILP's running time is the longest due to its high complexity, while LBA, SBA and SRA run much faster. Because LBA needs to spend time on calculating LCS', it consumes more running time than SBA and SRA. Then, for the US Backbone topology, the ILP cannot solve the problem within 10 hours, while the running time of LBA, SBA and SRA is still relatively short. Note that, as the topology is larger and the number of requests increases, LBA needs to spend more time on calculating LCS' and thus there is noticeable increase on its running time. Because LBA spends more time per request on checking the existing VNFs if their number is larger, its running time increases with the number of requests. Meanwhile, since SBA and SRA process each request almost independently, their running time does not increase with the number of requests.

VI. Conclusion

We studied how to allocate spectrum and IT resources jointly for realizing efficient VNF-SC in inter-DC EONs. Both an ILP model and an LCS-based heuristic were proposed. Simulation results showed that the algorithms could reuse VNFs and reduce MFSI in the network efficiently.

TABLE I
Average Running Time per Request (seconds)

	Six-node topology			US Backbone topology		
# of Requests	2	6	10	100	200	300
ILP	2.0	3.3	20.9	–	–	–
LBA	0.004	0.016	0.017	0.057	0.128	0.163
SBA	0.004	0.003	0.002	0.005	0.005	0.004
SRA	0.002	0.001	0.001	0.005	0.005	0.004

Acknowledgments

This work was supported in part by the NSFC Project 61371117, the Fundamental Research Funds for Central Universities (WK2100060010), Natural Science Research Project for Universities in Anhui (KJ2014ZD38), and the Strategic Priority Research Program of the CAS (XDA06011202).

References

[1] "Network functions virtualization (NFV)," Jan. 2012. [Online]. Available: https://portal.etsi.org/portal/server.pt/community/NFV/367

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

[3] M. Zeng, W. Fang, and Z. Zhu, "Orchestrating tree-type VNF forwarding graphs in inter-DC elastic optical networks," *J. Lightw. Technol., in Press*, pp. 1–12, 2016.

[4] P. Lu *et al.*, "Highly efficient data migration and backup for big data applications in elastic optical inter-data-center networks," *IEEE Netw.*, vol. 29, pp. 36–42, Sept. 2015.

[5] 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.

[6] M. Klinkowski and K. Walkowiak, "On the advantages of elastic optical networks for provisioning of cloud computing traffic," *IEEE Netw.*, vol. 27, pp. 44–51, Nov. 2013.

[7] L. Zhang and Z. Zhu, "Spectrum-efficient anycast in elastic optical inter-datacenter networks," *Opt. Switch. Netw.*, vol. 14, pp. 250–259, Aug. 2014.

[8] 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.

[9] M. Zhang, C. You, and Z. Zhu, "On the parallelization of spectrum defragmentation reconfigurations in elastic optical networks," *IEEE/ACM Trans. Netw., in Press*, pp. 1–15, 2016.

[10] T. Cormen, C. Lesiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. MIT Press, 2009.

[11] L. Gong *et al.*, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," *J. Opt. Commun. Netw*, vol. 5, pp. 836–847, Aug. 2013.