LETTER

Topological Consistency-Based Virtual Network Embedding in Elastic Optical Networks

Wenting WEI[†], **Kun WANG**[††a)], **Gu BAN**[†], **Keming FENG**[†], **Xuan WANG**[†], ***Nonmembers***, ***and*** **Huaxi GU**[†], ***Member***

SUMMARY Network virtualization is viewed as a promising approach to facilitate the sharing of physical infrastructure among different kinds of users and applications. In this letter, we propose a topological consistency-based virtual network embedding (TC-VNE) over elastic optical networks (EONs). Based on the concept of topological consistency, we propose a new node ranking approach, named Sum-N-Rank, which contributes to the reduction of optical path length between preferred substrate nodes. In the simulation results, we found our work contributes to improve spectral efficiency and balance link load simultaneously without deteriorating blocking probability.

key words: *elastic optical networks, network virtualization, virtual network embedding, topological consistency*

1. Introduction

Recently, the exponentially rising trend of emerging applications has brought a vast amount of data traffic to Internet [1], [2]. To adapt to such numerous traffic, network virtualization has been regarded as an expected solution. Underlying network infrastructure is abstracted to support sharing substrate resources for multiple users. A major challenge is how to allocate resources of one substrate network (SN) to several virtual networks (VNs), termed virtual network embedding (VNE) [3].

As the volume of traffic is forecasted to explode, elastic optical networks (EONs) have been introduced as promising substrate networks due to its large resource capacity and flexibility [4]. By exploiting optical orthogonal frequency division multiplexing technology, EONs can provide fine-grained spectrum resource [5] to achieve more spectrum efficiency.

Some schemes were proposed for VNE over EONs, namely virtual optical network embedding. Authors in [6] proposed a resource and load aware mapping algorithm considering load jointly with spectrum continuity during the node mapping stage, but it does not take the number of hops into consideration in the link mapping. Besides, it may lead to high time and computational complexity due to the usage of ant colony algorithm.

Differently, the authors presented two heuristic algorithms based on layered auxiliary graph [2] to decrease blocking ratio of requests in acceptable time. In their algorithms, substrate nodes are ranked according to a local index without topology knowledge in the node mapping. Unfortunately, virtual nodes might be mapped onto high ranked substrate nodes but are far away from each other. In [7], the authors present a VNE algorithm based on subgraph extraction to speed up the solution time. However, their work focused on a general network, rather than EONs which has the constraints of spectrum non-overlapping, consecutiveness, and continuity in link mapping [8].

In this letter, we propose a VNE scheme over EONs, termed topological consistency VNE (TC-VNE). TC-VNE takes topological consistency into account to decrease the number of hops between selective substrate nodes and extracts a relevant subgraph before the mapping process to cut down the search space. Specifically, when each VN arrives, our proposed algorithm will first extract a relevant subgraph from the SN topology to facilitate embedding process. In the node mapping, Sum-N-Rank algorithm is applied to evaluate the importance of a virtual node in order to achieve high spectral efficiency and a low blocking ratio.

2. Problem Formulation

2.1 Virtual Network Embedding

To solve VNE problem over EONs, the SN is modeled as an undirected graph $G^s(V^s, E^s)$, where V^s and E^s are the set of substrate nodes and substrate fiber links (SFL) respectively. Each node $v^s \in V^s$ has a certain amount of computing resources c^s. For each SFL $e^s \in E^s$, the entire spectrum domain is divided into a list of equal-sized frequency slots (FSs). Similarly, a virtual network is denoted by an undirected graph $G^r(V^r, E^r)$. Each $v^r \in V^r$ has a certain computing resource requirement c^r and each $e^r \in E^r$ has spectrum resource requirement n^r.

After the model construction, we can further describe the VNE problem more properly. When a virtual network request (VNR) arrives, the VNE algorithm will perform two operations:

(1) To map virtual nodes onto the substrate nodes with enough computing resources.
(2) To select substrate light paths and allocate adequate FSs to support virtual links and satisfy their bandwidth

Manuscript received January 4, 2019.
Manuscript publicized March 1, 2019.

[†]The authors are with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an, China.
[††]The author is with the School of Computer Science and Technology, Xidian University, Xi'an, China.
a) E-mail: kwang@mail.xidian.edu.cn

DOI: 10.1587/transinf.2019EDL8003

requirement.

If both operations are successful, the VNR is accepted, otherwise, the VNR is blocked.

2.2 Optimization Problem Formulation

The objective of the VNE algorithm proposed in this letter is to minimize the total spectrum consumption. The total spectrum consumption can be formulated as the total number of occupied FSs by VNRs over a given period.

$$Minimize \sum_{G_n^r \in G^r} \sum_{e^S \in E^S} (B^{e^s} - \sum_{m=1}^{B^{e^S}} b_m^{e^S}) \tag{1}$$

where B^{e^s} is the total number of FSs on the physical link e^s. An array b^{e^S} containing B^{e^s} bits is to represent FSs. More specially, when $b_m^{e^S} = 1$, the m^{th} slot on link e^s is occupied, otherwise $b_m^{e^S} = 0$.

The node mapping process can be represented as a one to one mapping process, E_N, such that:

$$E_N(v^r) = v^s, \quad v^r \in V^r, \quad v^s \in V^s \tag{2}$$

$$E_N(v^{r,1}) \neq E_N(v^{r,2}), \quad \forall v^{r,1}, v^{r,2} \in V^r \quad and \quad v^{r,1} \neq v^{r,2} \tag{3}$$

$$c^r \leq c^s, \quad \text{if} \quad E_N(v^r) = v^s \tag{4}$$

Equation (3) makes sure that one virtual node is mapped onto only one substance node. Equation (4) ensures each selected substance node has enough available computing resources left.

Link mapping decides how to map a virtual link onto the substrate light path which contains one or several SFLs. We use the notation P^s to denote the set of substrate light paths in G^s and similarly, use an array $b^r[j]$ contains B bits to represent the spectrum allocations. If $b^r[j] = 1$, the j^{th} slot is used, otherwise, $b^r[j] = 0$. E_L is employed to denote the link mapping process:

$$E_L(e^r) = p^s, \ e^r \in E^r, \ p^s \in P^s \tag{5}$$

$$add(b^r) = n^r \tag{6}$$

$$B - add(\bigcup_{e^r \in E^r} \bigcup_{e^s \in E^s} b_e) \geqslant n^r \tag{7}$$

$$add(b^r \cap (\bigcup_{e^r \in E^r} \bigcup_{e^s \in E^s} b_e)) = 0 \tag{8}$$

where $add(\cdot)$ is the sum of all bits in a bit array, $\bigcup$ means bit *OR* for several bit arrays, and $\cap$ means bit *AND* for two-bit arrays.

3. Heuristic Design

In this section, the proposed algorithm named *TC-VNE* is described as Algorithm 1. Inspired by PageRank, we propose a new ranking method to evaluate the significance of a node.

Algorithm 1 Topological Consistency-based Virtual Network Embedding (TC-VNE)

Input: substrate network G^s, virtual network request G^r
Output: node mapping E_N, link mapping E_L.
1: extracts a relevant subgraph (G_k^{sub});
2: calculate Sum-N-Rank (SNR) value for each v^r in G^r;
3: sort all virtual nodes in non-increasing order of SNR and mark the first one as v_{max}^r;
4: **for** *each* v^s *in* G_k^{sub} in non-increasing of degree **do**
5:     **if** *degree*(v^s) $\geqslant$ *degree*(v_{max}^r) **then**
6:         embed v_{max}^r onto v^s;
7:         delete v^s in the current G_k^{sub};
8:         **break**;
9:     **end if**
10: calculate the radiative radius B of v^r and construct G_k^{susub} over G_k^{sub}
11:     **if** there does not exist a ring in G_k^{susub} **then**
12:         exexute step 4 to 12 until there exists a ring in G_k^{susub};
13:     **end if**
14:     **for** each unmapped v^r in descending order of SNR **do**
15:         excute step 4 to 9 to embed v^r onto v^s in G_k^{susub};
16:         **if** two arbitrary v^rs are interconnected **then**
17:             accomplish link mapping from VOL to SFL;
18:             **if** link mapping fails **then**
19:                 excute step 11 to 19 to accomplish link mapping;
20:             **end if**
21:         **end if**
22:         **if** mapping fails **then**
23:             mark G^r as blocked;
24:         **end if**
25:     **end for**
26: **end for**

Definition 1: The Sum-N-Rank:

$$SNR(v^{r,k}) = LI(v^{r,k}) + \sum_{\forall v^r \in V^r, v^{r,k} \neq v^{r,j}} \frac{LI(v^{r,j})}{hop(v^{r,k}, v^{r,j})} \tag{9}$$

where for a VN, $LI(v^{r,k}) = c_v^{r,k} \cdot d_v^{r,k}$ represents the local information of its k^{th} virtual node ($k = 1, 2, 3, \cdots, |G^r|$) and $|G^r|$ refers to the total number of its nodes. $d_v^{r,k}$ is the degree of node $v^{r,k}$. $\sum_{\forall v^r \in V^r, v^{r,k} \neq v^{r,j}} \frac{LI(v^{r,j})}{hop(v^{r,k}, v^{r,j})}$ is indicated as the influence of node $v^{r,j}$ to $v^{r,k}$. Function $hop(v^{r,k}, v^{r,j})$ is used to calculate the number of hops between $v^{r,k}$ and $v^{r,j}$.

For a given substrate network G^s, TC-VNE algorithm firstly extracts a relevant subgraph (G_k^{sub}) before mapping process. specifically, the scheme eliminates the substrate node N^s that are not able to satisfy the minimum computing resources request and the SFL L^s that can not satisfy the maximal bandwidth requirement for an arrived VNR. Next, the algorithm can set up the adjacency matrixes M_s and M_v, which are used to formulate the k^{th} substrate subgraph (G_k^{sub}) and VN G^r respectively.

$$M_{s,ij} = \begin{cases} m_{s,ij} & (v^{s,i}, v^{s,j}) \in E^s \\ 0 & (v^{s,i}, v^{s,j}) \notin E^s \end{cases} \tag{10}$$

$$M_{v,ij} = \begin{cases} m_{v,kl} & (v^{r,k}, v^{r,l}) \in E^r \\ 0 & (v^{r,k}, v^{r,l}) \notin E^r \end{cases} \tag{11}$$

where $m_{s,ij}$ means the number of available frequency slots on the SFL that interconnects $N^{S,i}$ and $N^{S,i}$, the frequency slots requirement of a virtual link which interconnects $v^{r,k}$ to $v^{r,l}$ is denoted as $m_{v,kl}$. Then, The vector V_s that denotes the degrees of nodes in the k^{th} subgraph and V_r represents the degrees of nodes in the VN can be given in accordance with M_s and M_v.

$$V_s = [V_{s1}, V_{s2}, \cdots, V_{sn}] \tag{12}$$

$$V_r = [V_{r1}, V_{r2}, \cdots, V_{rn}] \tag{13}$$

Go through adjacency matrix $M_{r,ij}$ to find the VN ($v^{r,max}$) which has the maximal SNR value and get the degree of $v^{r,max}$ from V_r. Search for the substrate node having a degree d_s which is larger than $d_{r,max}$ from the V_s and add it to the set S. Finally, we can get the set S and then sort the nodes in S in ascending order according to their degrees. After SNs that satisfy the node mapping constraints are successfully added to S, we can determine the central SN n_s.

In accordance with the set S, we can choose SNs within as central SN successively. For the first node to be mapped, the algorithm will check whether the constructed virtual network G^r has rings or not. If yes, we search for subgraphs with n_{s1}, which is the first element in S, as the first central node. If not, we delete n_{s1} and regenerate the new set $S' = S - \{n_{s1}\}$.

Definition 2: The optimal relevance p(s)

$$p(s) = \frac{d_s}{\sum_{v^{s,j}} hop(n_s, m_s)} \quad s = 1, 2, \cdots, |G^s| \tag{14}$$

where d_s represents the degree of a SN, m_s is the set of VNs which were mapped successfully. $s = 1, 2, \cdots, |G^s|$ denotes the total number of SNs in the substrate network. And the function $\sum hop(\cdot)$ is the number of hops between an alternative SN n_s and m_s.

The optimal relevance is calculated in the circumstance that two SNs in S have the same degree value. Then, we use the SN with the minimal $p(s)$ as the central node.

Definition 3: The radiative radius C of a subgraph:

$$C = Max\{hop(v^{r,max}, v^{r,k})\},\ \forall v^{r,k} \in G^r, v^{r,k} \neq v^{r,max} \tag{15}$$

where $v^{r,max}$ shows the chosen VN ans $v^{r,k}$ means other left VNs in the G^r. After we get the value of C, and we utilize C as radius and n_s as the central node to construct sub-graphs over (G_k^{sub}). Finally, we can accomplish the VNR.

Definition 4: The occupied frequency slot ratio:

$$F = \frac{\sum_{i=1}^{\overline{m_{s,ij}}} \sum_{j=1}^{l} b_{e,ij}}{\sum_{i=1}^{q} \sum_{j=1}^{B} b_{e,ij}} \tag{16}$$

where $\overline{m_{s,ij}}$ means the number of occupied frequency slots on the SFL that interconnects SN i and SN j, l means the average number of occupied frequency slots on each SFL, q is the total number of SFLs in the substrate network.

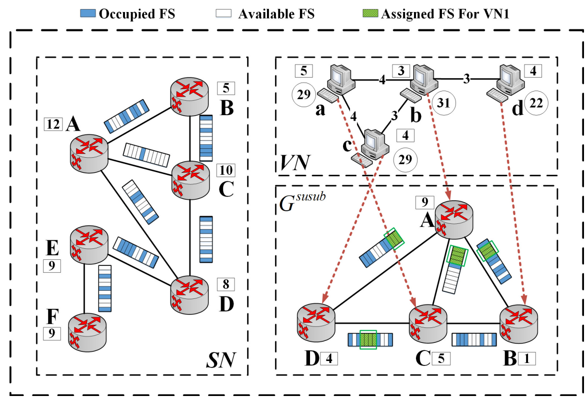


Fig. 1 An instance for TC-VNE

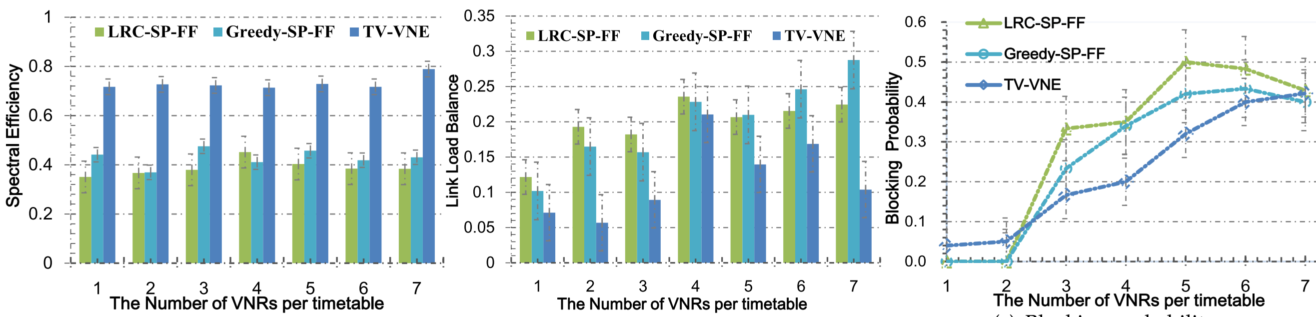
We define F to measure the usage of frequency slots on each SFL, which can intuitively reflect the potential capability of accepting the upcoming VNRs of the substrate network. And a smaller F represents a greater potential capability.

As shown in Fig. 1, we provide an example to illustrate our algorithm in the mapping procedure. For starters, we calculated SNR of each node in the VN, which is shown in the box next to the node. In this example, $v^r_{\max}$ is virtual node b. Taken that the node F's and E's degrees are smaller than that of $v^r_{\max}$, node F and E are deleted to construct G^{susub}. As suggested by our algorithm, the virtual node b is embedded onto the substrate node A, which is the first substrate node to satisfy the computation resource request of node b. Then to map the virtual node with the second large Sum-N-Rank (node a), substrate nodes that are connected to node A directly, i. e., node B, C, and D, are desirable. Specifically, taking into account the available bandwidth between node A and the node to be selected, node C is preferred to accommodate node a, meanwhile due to its capability to satisfy virtual node a. Thereafter, we would do the same job for the node c and d, which will finally be embedded to the node D and B respectively. In a nutshell, red arrows from VN to G^{susub} is an illustration of embedding VN to SN, where G^{susub} is the sub-graph generated from SN. In such a heuristic algorithm, average-hops are decreased as much as possible, so that spectral efficiency can be improved and blocking rate can be reduced.

4. Performance Evaluation

The performance of the TC-VNE is compared with two existing algorithms, named LRC-SP-FF [6] and Greedy-SP-FF [2] respectively. Table 1 shows the simulation parameters. We adopted the NSFNET topology with 14 nodes and 21 links. Each VNR has an exponentially distributed lifetime and is served in a first-come-first-served manner. The number of its nodes obeys uniform distribution and the virtual link between nodes is randomly created with a probability 0.5.

Figure 2 shows the embedding results with confi-



(a) Spectral Efficiency (b) Link Load Balance (c) Blocking probability

Fig. 2 Performance comparison

Table 1 Parameters used in simulation

Parameters	Value
Computing capacity in substrate node	[100, 200]
Bandwidth capacity in substrate link	80
Number of virtual nodes in each VN	[2, 5]
Computing request in virtual node	[1, 3]
Bandwidth request in virtual link	[1, 5]

dence level 95% in TC-VNE, Greedy-SP-FF, and LRC-SP-FF. Figure 2 (a) illustrates that performance comparison on spectral efficiency, which is defined as the ratio of spectrum request to the consumed FSs. It shows that TC-VNE has a clear advantage than others. This results from its consideration of topological consistency so that the consumption of FSs is cut down due to fewer hops. Figure 2 (b) shows that TC-VNE has the best performance in load balance. This is because the TC-VNE can find a layered substrate network mostly meets the demand of the upcoming virtual network, so as to avoid the occurrence of multi-hop during the link mapping process. In terms of blocking probability, we can see that TC-VNE has slightly better performance under heavy load in Fig. 2 (c).

5. Conclusion

In this letter, we proposed a new virtual optical network embedding algorithm over EONs based on topological consistency. It can be verified from the simulation results that our algorithm achieves better performance than the benchmark algorithms in terms of spectral efficiency and load balance on the links.

Acknowledgments

This work was supported by the National Science Foundation of China under Grants 61634004, and Grant 61472300, the Fundamental Research Funds for the Central Universities Grant No. JB180309 and No. JB170107.

References

[1] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka, "Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies," IEEE Commun. Mag., vol.47, no.11, pp.66–73, 2009.

[2] L. Gong and Z. Zhu, " Virtual Optical Network Embedding (VONE) Over Elastic Optical Networks," Journal of Lightwave Technology, vol.32, no.3, pp.450–460, 2014.

[3] A. Fischer, J.F. Botero, M.T. Beck, H. de Meer, and X. Hesselbach, "Virtual network embedding: A survey," IEEE Commun. Surv. Tutorials, vol.15, no.4, pp.1888–1906, 2013.

[4] B.C. Chatterjee, N. Sarma, and E. Oki, "Routing and spectrum allocation in elastic optical networks: A tutorial," IEEE Commun. Surv. Tutorials, vol.17, no.3, pp.1776–1800, 2015.

[5] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Elastic bandwidth allocation in flexible OFDM based optical networks," IEEE/OSA J. Lightw. Technol., vol.29, no.9, pp.1354–1366, 2011.

[6] R. Yang, K. Wang, B. Wang, W. Wei, X. Liu, Y. Guo, and H. Gu, "Resource and load aware mapping algorithm for elastic optical network," IEICE Electron. Express, vol.13, no.14, p.20160590, 2016.

[7] A. Blenk, P. Kalmbach, J. Zerwas, M. Jarschel, S. Schmid, and W. Kellerer, NeuroViNE: A Neural Preprocessor for Your Virtual Network Embedding Algorithm, in Proc. IEEE INFOCOM, pp.405–413, 2018.

[8] B. Chen, J. Zhang, W. Xie, J.P. Jue, Y. Zhao, S. Huang, and W. Gu, Minimum-cost survivable virtual optical network mapping in flexible bandwidth optical networks, in Proc. IEEE Global Telecommun. Conf., pp.2023–2028, 2014.