Virtual Topology Mapping in Elastic Optical Networks

Juzi Zhao, Suresh Subramaniam
Department of Electrical and
Computer Engineering
The George Washington University, Washington, DC 20052
Email: juzizhao@gwu.edu, suresh@gwu.edu

Maïté Brandt-Pearce
Charles L. Brown Department of
Electrical and Computer Engineering
University of Virginia, Charlottesville, Virginia 22904
Email: mb-p@virginia.edu

Abstract—**Virtualization improves the efficiency of networks by allowing multiple virtual networks to share a single physical network's resources. Next-generation optical transport networks are expected to support virtualization by accommodating multiple virtual networks with different topologies and bit rate requirements. Meanwhile, Optical Orthogonal Frequency-Division Multiplexing (OOFDM) is emerging as a viable technique for efficiently using the optical fiber's bandwidth in an elastic manner. OOFDM partitions the fiber's bandwidth into hundreds or even thousands of OFDM subcarriers that may be allocated to services. In this paper, we consider an OOFDM-based optical network and formulate a virtual network mapping problem for both static and dynamic traffic. This problem has several natural applications, such as e-Science, Grid, and cloud computing. The objective for static traffic is to maximize the subcarrier utilization, while minimizing the blocking ratio is the aim for dynamic traffic. Two heuristics are proposed and compared. Simulation results are presented to demonstrate the effectiveness of the proposed approaches.**

Index Terms—**Virtual optical networks, OOFDM, elastic networks, mapping, list scheduling, subcarriers.**

I. INTRODUCTION

Network virtualization serves as an efficient method to circumvent the Internet's rigidity, in which multiple virtual networks with different topologies and requirements are allowed to share a single physical network. This concept was originally applied to the higher protocol layers, but recent efforts have turned their attention to virtualizing the physical layer [1], which has traditionally been based on Wavelength Division Multiplexing (WDM). These Virtual Optical Networks (VON) are expected to handle the exploding traffic demands in carrier networks in the future. The concept of optical network virtualization, and its implication and challenges for optical network elements and transport technologies are presented in [1]. A possible approach for virtualization of classical wavelength-switched optical network by partitioning and aggregation of optical switching nodes and link capacity is also discussed.

Concurrent with the emergence of virtualization of the optical layer, new ways of getting around the rigidity and coarseness of the wavelength spectrum in WDM-based transmission are being identified. Efforts have been underway for some time on optical orthogonal frequency division multiplexing (OOFDM), which has been proposed as a viable technology for optical transmission [2]. OOFDM allows fiber bandwidth to be carved up into subcarriers that have much finer bandwidth granularity than wavelengths. Whereas in WDM the typical wavelength spacing is 25 GHz, wavelength bit-rates are 10, 40, or 100 Gbps, and whole wavelengths are allocated to services, in OOFDM the spacing between subcarriers is only a few GHz and subcarrier bit rates are a few Gbps. An attractive feature of OOFDM is that different bit rates may be achieved by using one of a variety of modulation schemes, and bands of subcarriers may be assigned to a service as needed. In this way, allocated network resources can be matched up with service requirements in a much more flexible manner than in WDM-based networks. Such optical networks have therefore been called *elastic optical networks* [3].

This paper is at the intersection of the above two emerging research areas. One of the challenges of network virtualization is the virtual topology mapping (or embedding) problem, which assigns substrate physical nodes to virtual nodes, and guarantees the bandwidth requirements of the virtual links. We consider the problem of mapping virtual topologies in elastic optical networks. To the best of our knowledge, this is the first paper on this topic, though virtualization in WDM-based network has been an active topic of research in the past couple of years [4], [5], [6]. The problem of mapping virtual topologies in elastic optical networks is different from that in WDM networks because of the multiple subcarrier allocation. Further, the multiple subcarriers (a subcarrier *band*) for a service are typically allocated in a contiguous manner, as blocks allocated to two different services must be separated by a guardband in order to avoid interference [3].

We consider both static and dynamic versions of the virtual topology mapping problem. In this problem, a virtual topology request includes a set of virtual nodes, the amount of computation requirements at the virtual nodes (e.g., number of virtual machines (VMs) needed), and the bandwidth requirements between the virtual nodes. The virtual topology mapping problem is NP-hard. We present an optimal integer linear programming (ILP) formulation suitable to small networks and two heuristic algorithms, applicable to larger systems, to map the virtual topology requests onto the physical network, and assign VMs and OFDM subcarriers to virtual network requests.

The paper is organized as follows. In Section II, we present some background on virtual topology mapping and OOFDM, and formulate a problem. An ILP formulation for the problem and the proposed heuristics are presented in Section III. Section IV presents and discusses the simulation results. We conclude the paper in Section V.

978-1-4673-3122-7/13/$31.00 ©2013 IEEE

II. MODEL AND PROBLEM STATEMENT

In this section, we provide some background on optical network virtualization and OOFDM, present the system model, and define our problem.

A. Virtual Optical Network Problems

We first provide a classification of virtualization problems. These problems are applicable to WDM- as well as OOFDM-based optical networks. In general, these problems can be classified into Slice Provisioning problems or Virtual Network (VN) Mapping problems. In Slice Provisioning, each request requires a slice of the entire physical network including (sub)wavelengths on fibers, ports of optical devices, and/or optical-electrical-optical (OEO) converters at regenerator nodes [1]. The objective can be maximizing the acceptance ratio or maximizing revenue (each VON request has to pay the price of the resources). This model can further be classified in two ways:

1. Slice Provisioning: specific model

The slice request includes specific (sub)wavelengths of each physical link, specific ports of each optical device (e.g., optical cross-connetcs (OXC)), and the number of OEO circuits in each regenerator.

2. Slice Provisioning: flexible model

The slice request includes the bit rate requirement on each physical link, the number of ports at each optical device, and the reachability of each pair of nodes. The provider assigns (sub)wavelengths on each link, the ports on each node and the OEO converters on each regenerator node to the request.

In VN Mapping, each request is a virtual network topology including virtual nodes and/or virtual links. The provider assigns physical nodes to virtual nodes, and allocates (sub)wavelengths on each link to the virtual links. In both versions, the request may also include Bit Error Rate (BER) or survivability requirements.

1. VN mapping: pipe model

The VN request includes a topology consisting of virtual nodes and virtual links, each virtual node has a set of candidate physical nodes that it could be assigned to (based, for example, on proximity) and possibly a computing resources requirement [7]. Virtual links have a bit-rate requirement.

2. VN mapping: hose model ([8])

The VN request consists of virtual nodes with computing resources requirement, and the aggregate incoming/outgoing traffic from/to the other virtual nodes.

We consider the VN mapping using the pipe model in this paper.

B. OOFDM

OOFDM is a technology providing flexible subcarrier assignments and more efficient bandwidth utilization. The optical spectrum of each fiber (e.g., C-band) is divided into subcarriers, which are orthogonal to each other, and a guardband consisting of multiple subcarriers is placed between two adjacent subcarrier bands assigned to different connections. Guardbands are used for avoiding interference, and their width is in general a function of the maximum number of filters on the path and the filter characteristics (such as filter bandwidth and order) [9].

Different modulation levels can be adopted by different subcarriers, and a connection can be assigned multiple subcarriers depending on its bit rate requirement. However, all the subcarriers assigned to a single lightpath use the same modulation level. The highest modulation level of a path depends on the length of that path due to physical layer impairments [3]. Regenerators in the network can extend the reach of a lightpath.

The subcarriers assigned to a connection may be contiguous or non-contiguous. Contiguous assignment can take advantage of overlap between adjacent subcarriers due to their orthogonality, but is obviously more restrictive because of the contiguity restriction. Non-contiguous assignment is more flexible but may increase total guardband overhead. There may be subcarrier converters [10] in the network; at a subcarrier converter, a whole band of subcarriers assigned to a connection on the incoming link can be converted to another band of subcarriers on the outgoing link.

C. Network Model and Notation

The physical network includes a set of physical nodes, each with h virtual machines (VMs), and a set of fibers, each with multiple subcarriers. Each subcarrier is C GHz. Guardband G is a fixed integer number of subcarriers. The shortest path $p_{s,d}$ (based on distance) for each pair of nodes (s, d) is precomputed, and depending on its length each path has a highest modulation level that can be used.

Virtual topology (VT) request i includes a set of virtual nodes and a set of virtual links.[1] For virtual link k^i, there is a bit rate requirement Λ_k^i. Virtual node j^i has a candidate set (N_j^i) of physical nodes to map to, and a VM requirement m_j^i. Determined by the candidate nodes, virtual link k^i has a candidate path set P_k^i. Path $p \in P_k^i$ has a subcarrier requirement $b_k(p)$, which is obtained from its highest modulation level and Λ_k^i.

D. Problem Statement

We consider the VN mapping problem using the pipe model. Within this context, there are several variants based on whether traffic can be split among multiple paths or not, whether the subcarrier bands should be contiguous or not, etc. We assume that the network is all-optical (i.e., no regenerators), a single path is used for each virtual link, and subcarriers for each virtual link are contiguous.

Consider a VT request. For each of its virtual nodes, a physical node that has sufficient VMs available must be assigned from the set of candidate physical nodes. At the same time, a physical path p should be assigned to each virtual link, and a band of contiguous subcarriers on each physical link of the path should be allocated to the virtual link to satisfy the virtual link's bit rate requirement. We allow different virtual

---

[1] We use the letter i for VT index, j for virtual node index, k for virtual link index, n for physical node index, p for physical path index.

nodes of a request to be mapped to the same physical node in case they have common candidate physical nodes; in this case, there is no need to map the virtual link between these virtual nodes (and no subcarriers are allocated).

For the static traffic case, a set of VT requests is given, and the objective is to minimize the maximum used subcarrier index on any link. For the dynamic traffic case, the number of subcarriers per link is fixed and VT requests arrive and depart in a random manner. The objective is to minimize the *request blocking* ratio. All virtual links are assumed to be bidirectional and use the same path and subcarriers for both directions. Since the bit rate (bandwidth) requirements of each VT could be different, in addition to the request blocking ratio, we consider *bandwidth blocking* ratio, defined as: Bandwidth blocking = $(\sum_{i' \in I'} B_{i'})/(\sum_{i \in I} B_i)$, where B_i denote the aggregate bit rate requirement of request i (the sum of the bit rate requirements over all virtual links in request i).

III. ALGORITHMS

In this section we introduce our algorithms for VT mapping. We first present an ILP formulation for static requests that can be used to solve small problem instances. We then present two heuristics for static requests, and later adapt them to dynamic requests. Recall that for static traffic we are given a set of VT requests which never depart, and VT requests arrive and depart in a random manner for dynamic traffic. Each VT request has bit-rate requirements on its virtual links, VM requirements at the virtual nodes, and there is a constraint on the set of physical nodes a virtual node can be mapped to.

A. ILP Formulation

Input parameters used in the ILP formulation are listed below: $S(p)$ denotes source of path p; $D(p)$ denotes destination of path p; $\gamma_{p,l} = 1$ if path p uses link l; $S(k^i)$ denotes the source virtual node for virtual link k of request i; $D(k^i)$ denotes the destination virtual node for virtual link k of request i; q denotes a dummy path with $S(q) = D(q)$.

Objective: Minimize $\max_s s f_s$

Variables:

a)

$$f_s = \begin{cases} 1, & \text{if subcarrier index } s \text{ is used on some physical link} \\ 0, & \text{otherwise} \end{cases}$$

b)

$$x_{ijn} = \begin{cases} 1, & \text{if virtual node } j^i \text{ uses physical node } n \\ 0, & \text{otherwise} \end{cases}$$

c)

$$y_{ikp} = \begin{cases} 1, & \text{if virtual link } k^i \text{ uses path } p \\ 0, & \text{otherwise} \end{cases}$$

d)

$$z_{ikpc} = \begin{cases} 1, & \text{if virtual link } k^i \text{ uses path } p \\ & \text{and the starting subcarrier index is } c \\ 0, & \text{otherwise} \end{cases}$$

Constraints:

a) Each virtual node is assigned to one physical node.

$$\textstyle\sum_{n \in N_j^i} x_{ijn} = 1 \text{ for all } i, j$$

b) Each virtual link is assigned to one physical path.

$$\textstyle\sum_{p \in P_k^i} y_{ikp} = 1 \text{ for all } i, k$$

c) Each physical node's VM capacity cannot be exceeded.

$$\textstyle\sum_{i,j} (x_{ijn} \cdot m_j^i) \leq h, \text{ for all } n$$

d) If the source and destination of a virtual link are assigned to the same physical node, the virtual link is assigned to dummy path q (which uses no subcarriers).

$$y_{ikq} = [\textstyle\sum_{n \in N^i_{S(k^i)}} (x_{iS(k^i)n} \cdot n) = \textstyle\sum_{n \in N^i_{D(k^i)}} (x_{iD(k^i)n} \cdot n)]$$

(Note that n is *not* a decision variable, and so the constraint is linear.)

e) If virtual link k is assigned to physical path p ($p \neq q$), the source and destination of the two must match, i.e.,

if $y_{ikp} = 1$, then either $x_{iS(k^i)S(p)} = 1$, $x_{iD(k^i)D(p)} = 1$ or $x_{iS(k^i)D(p)} = 1$, $x_{iD(k^i)S(p)} = 1$.

f) If virtual link k uses path p, then it must be assigned subcarriers on that path, i.e.,

$$\textstyle\sum_c z_{ikpc} = y_{ikp}$$

g) All assigned subcarriers for non-dummy paths are marked as used, i.e.,

$c_{ikpc} = 1$ ($p \neq q$), the corresponding $f_s = 1$

$$z_{ikpc} \leq f_s \text{ for } c \leq s < c + b_k(p) + G$$

h) Each subcarrier on a physical link can be used by at most one virtual link.

$$\textstyle\sum_{i,k,p,c \leq s < c + b_k(p) + G} z_{ikpc} \gamma_{p,l} \leq 1 \text{ for all } l,\ s.$$

B. Static Requests

Before describing the algorithms, we present three functions (that are used by the heuristics) that check if certain VT assignments can be made.

i) *TrivialRequestCheck()*: This function checks whether any VT can be mapped to a single physical node with no influence on other virtual node mappings in other VTs (i.e., the candidate node sets of the other virtual nodes won't be affected by this mapping). In this case, it is best to assign the whole VT to that physical node (since it will not use any subcarrier).

ii) *CheckNode()*: If a virtual node has only one candidate physical node (taking into account the number of currently available VMs at physical nodes), then we assign it to that physical node. Since each time a virtual node mapping may affect candidate nodes for other virtual nodes (due to the VM limit), this procedure is repeated for at most $I \cdot N_l$ times (where I is the number of VT requests and N_l is the maximum number of virtual nodes in a VT). In addition, if a virtual link's two terminal virtual nodes are mapped to the same physical node, we mark the virtual link as an assigned virtual link (a dummy physical path).

iii) *CandidateCheck()*: A candidate physical node $n \in N_j^i$ for virtual node j^i will be considered for node mapping if it can satisfy the following condition: after mapping node j^i to n, all other unmapped virtual nodes (belonging to the same or other VTs) still have at least one potential candidate physical node by considering the number of free VMs on each physical node.

The following two algorithms are proposed.

1) First Fit (FF): This is a simple greedy algorithm. VTs are considered one by one (from lowest index) and unmapped virtual nodes of each VT are mapped one by one. For each virtual node j^i, the first fit physical node (the one with smallest index) is selected from the candidate set, as long as the physical node has enough available VMs to satisfy the virtual node's VM requirement m_j^i. Each time a virtual node is mapped, the function CheckNode() is called. After all the virtual nodes are mapped, subcarriers are allocated to unassigned virtual links. Since the node mapping is already done, each virtual link's path (say p) is fixed (recall that the shortest path is used for each pair of physical nodes), and the number of subcarriers $b_k(p)$ is also known. Then, the first available band of $b_k(p)$ subcarriers is allocated to virtual link k^i.

2) Link List (LL): The idea behind this algorithm is to map the VTs one by one in decreasing order of their VM and bit-rate requirements because these require the most resources. Within each VT, virtual links are mapped one by one, starting from the "most-constrained" virtual link.

We will use the following notations in the algorithm. For VT i, denote the sum of the VM requirements of unmapped virtual nodes as M_i; the sum of the bit rate requirements of unassigned virtual links as B_i. Further, let the number of used VMs on physical node n be c_n, and denote the two ends of a generic virtual link as t_1 and t_2.

A VT list is created first. For each VT, a virtual link list is created based on the relative weight of a virtual link k^i's bit rate requirement Λ_k^i. When considering the current virtual link in the list, the candidate nodes of virtual node t_1 and t_2 are checked by the function CandidateCheck() if they are not mapped yet. Suppose the virtual link k is mapped to a path p ending with candidate nodes (or mapped physical nodes) n_1 and n_2 for virtual nodes t_1 and t_2 respectively. Let the resulting maximum subcarrier index on any link be s_n after first-fit subcarrier assignment. Assign a cost to the path as $\max(c_{n_1} + m_{t_1}^i, c_{n_2} + m_{t_2}^i, s_n)$ (if either t_1 or t_2 are already mapped, set the corresponding cost to 0). Finally, virtual link k^i is mapped to the candidate path with minimum cost, and t_1, t_2 are mapped to the corresponding end nodes of the selected path. The pseudo-code of the algorithm is shown as Algortihm 1.

C. Dynamic Requests

For dynamic traffic, since VTs arrive (and depart) one by one, VTs are mapped in the order they arrive. No remapping of currently mapped VTs is done. The same heuristics for static traffic are adapted by removing the TrivialRequestCheck() function. Further, the CheckNode() and CandidateCheck() functions consider only the virtual nodes in the current VT.

Algorithm 1 Link List

```
TrivialRequestCheck() and CheckNode()
for VT $i = 1, 2, \ldots, I$ do
    Find $M_i$, $B_i$, and assign VT $i$ a cost $\max(\frac{M_i}{\sum_i M_i}, \frac{B_i}{\sum_i B_i})$
end for
Create $LIST$ in decreasing order of the cost
for each VT $i$ in $LIST$ do
    for unassigned virtual link $k^i$ do
        Assign it a cost $\frac{\Lambda_k^i}{B_i}$
    end for
    Create virtual link $LIST(i)$ in decreasing order of the cost
    for each virtual link $k^i$ in $LIST(i)$ do
        1. CandidateCheck() for $t_1$ and $t_2$
        2. Assign a cost to each candidate path
        3. Assign link $k^i$ to the path with minimum cost
        4. Map $t_1$ and $t_2$
        CheckNode()
    end for
end for
```

IV. SIMULATION RESULTS

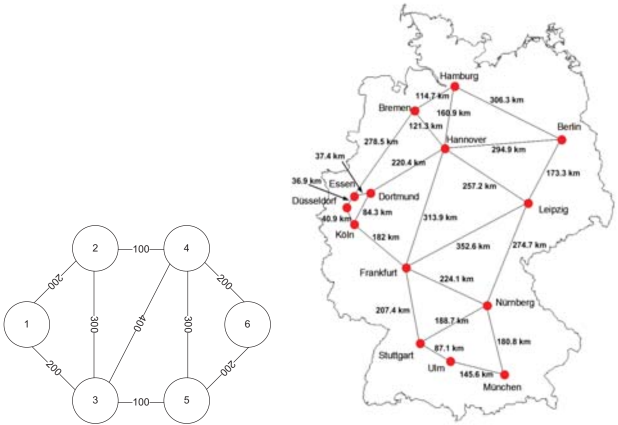
We present results for two network topologies, a small 6-node network and the larger Deutsche Telekom (DT) network, shown in Fig. 1.

We use 6 modulation levels for the subcarriers: BPSK, QPSK, 8QAM, 16QAM, 32AQM, 64QAM, and adopt the half distance law used in [11] for the optical reach. According to that, BPSK, QPSK, 8QAM, 16QAM, 32QAM, 64QAM can be used for paths up to 3000 km, 1500 km, 750 km, 375 km, 187.5 km, and 93.75 km, respectively. With each subcarrier being 5 GHz, the data rate for BPSK is 2.5 Gbps per subcarrier [11], and the data rates for other modulation schemes increase by factors of 2, 4, 8, 16, and 32. We assume that each physical node has 100 VMs (as modern-day servers allow tens of VMs per host [7]) and there are 800 subcarriers on each link (as C-band is 4000 GHz). In addition, the guardband is assumed to be $G = 2$ subcarriers.

VT requests are generated as follows. Each VT in the small network is a complete graph with 3 or 4 virtual nodes (selected randomly). In the DT network, each VT has between 3 and 10 virtual nodes, and a virtual link exists between a pair of nodes with probability 0.5 (we discard any topologies that are not connected). Each virtual node has between 1 and 3 physical node candidates (candidate nodes are all adjacent to each other), and requires a random number of VMs between 1 and 4. Each virtual link requires a bit-rate that is uniformly distributed over the range $0 - 2.5L$ Gbps, where L is a data-rate load parameter.

A. Static Requests

We first present results for the static traffic case. Due to the high complexity of the ILP, we are only able to obtain results



(a) 6-node network (b) DT

Fig. 1. 6-node network and 14-node DT. The number on each link corresponds to the number of distance in km.

for the small network. Sample results for 20 VT requests for various load values are shown in Table I.

TABLE I

NUMBER OF SUBCARRIERS FOR ILP

20 VTs	First Fit	Link List	ILP
L=5	63	47	41
L=10	79	54	44
L=15	95	62	51
L=20	112	68	58
L=25	128	73	64
L=30	143	87	71
L=35	159	92	78
L=40	176	102	86

From these results, we observe that the LL algorithm performs much better than FF, suggesting that the simple greedy algorithm can be vastly improved upon with a clever heuristic. The LL results are quite close to those achieved by the ILP (no more than 25% in the cases considered), though there is some room for improvement. For the DT network, results for the two heuristics are presented in Table II. These results confirm our earlier observations.

TABLE II

NUMBER OF SUBCARRIERS FOR STATIC TRAFFIC

20 VTs	First Fit	Link List	30 VTs	First Fit	Link List
L=5	167	143	**L=5**	194	162
L=10	213	172	**L=10**	244	214
L=15	254	208	**L=15**	293	240
L=20	291	239	**L=20**	341	280
L=25	334	275	**L=25**	392	324
L=30	379	313	**L=30**	439	360
L=35	420	326	**L=35**	492	405
L=40	468	375	**L=40**	543	432
L=45	513	390	**L=45**	594	505
L=50	557	454	**L=50**	646	523
L=55	602	485	**L=55**	699	571
L=60	652	489	**L=60**	755	608

B. Dynamic Case

For dynamic traffic, VT requests are assumed to arrive to the network according to a Poisson process. For each data point in the graphs, we simulated 10000 to 100000 VT request arrivals. Each request has a holding time of 1 unit. The VT request arrival rate per unit time is variable, but intended to produce blocking rates between around 10^{-4} and 10^{-1}. Figs. 2 and 3 show the request blocking and bandwidth blocking versus the virtual link bit rate load L for the two algorithms with arrival rates of 20 and 30. The results show that for both performance metrics, LL is much better than FF.

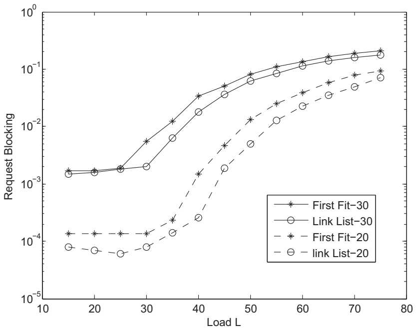


Fig. 2. Request Blocking vs. Load L for arrival rates 20 and 30.

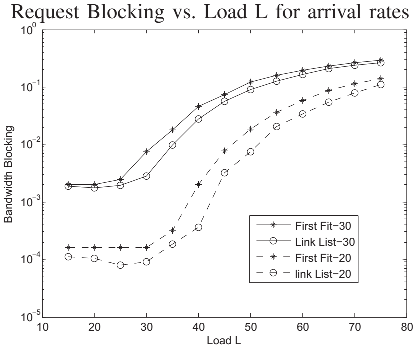


Fig. 3. Bandwidth Blocking vs. Load L for arrival rates 20 and 30.

V. CONCLUSIONS

In this paper, we investigated the virtual network mapping and subcarrier allocation problem for both static and dynamic traffic in elastic optical networks. We proposed two heuristics based on list scheduling, and simulation results showed the virtual link list scheduling approach achieves better performance than a simple first fit approach.

ACKNOWLEDGMENT

This work was supported in part by NSF grants CNS-0915795 and CNS-0916890.

REFERENCES

[1] R. Nejabati et al., “Optical network virtualization,” in *Optical Network Design and Modeling (ONDM)*, Feb. 2011.

[2] I. B. Djordjevic and B. Vasic, “Orthogonal frequency division multiplexing for high-speed optical transmission,” *Optics Express*, vol. 14, no. 9, pp. 3767–3775, 2006.

[3] M. Jinno et al., “Distance-adaptive spectrum resource allocation in spectrum-sliced elastic optical path network,” *IEEE Commun. Mag.*, vol. 48, no. 8, pp. 138 – 145, Aug. 2010.

[4] K. Shiomoto et al., “Network virtualization in high-speed huge-bandwidth optical circuit switching network,” in *INFOCOM, IEEE*, April 2008.

[5] X. Liu et al., “Application-specific resource provisioning for wide-area distributed computing,” *Network, IEEE*, vol. 24, no. 4, pp. 25 – 34, 2010.

[6] S. Peng et al., “Performance modelling and analysis of dynamic virtual optical network composition,” in *Optical Network Design and Modeling (ONDM)*, April 2012.

[7] M. Yu et al., “Rethinking virtual network embedding: substrate support for path splitting and migration,” in *ACM SIGCOMM*, April 2008.

[8] N. G. Duffield et al., “A flexible model for resource management in virtual private networks,” in *ACM SIGCOMM*, Aug. 1999.

[9] B. Kozicki et al., “Filtering characteristics of highly-spectrum efficient spectrum-sliced elastic optical path (SLICE) network,” in *Optical Fiber Communication (OFC)*, March 2009.

[10] S. Blouza et al., “Multi-band OFDM networking concepts,” in *Conference on telecommunications*, April 2011.

[11] K. Christodoulopoulos et al., “Elastic bandwidth allocation in flexible OFDM-based optical networks,” *IEEE/OSA J. Lightwave Technol.*, vol. 29, no. 9, pp. 1354 – 1366, May 2011.