Virtual Network Mapping for Multicast Traffic over Elastic Optical Networks

Xiujiao Gao[1], Zilong Ye[1], Weida Zhong[1], Chunming Qiao[1], Xiaojun Cao[2], Hanjia Zhao[3], Hongfang Yu[3], Vishal Anand[4]
[1]Dept. of Computer Science & Engineering, SUNY-Buffalo, Buffalo, NY USA 14260
[2]Dept. of Computer Science & Engineering, Georgia State University, Atlanta, GA USA 30302
[3]School of Communication and Information Engineering, UESTC, Chengdu, Sichuan, China 611731
[4]Dept. of Computer Science, the College at Brockport, SUNY, Brockport, NY USA 14420
{xiujiaog,zilongye,weidazho,qiao}@buffalo.edu, cao@cs.gsu.edu,{zhjpaul,yuhf2004}@gmail.com, vanand@brockport.edu

Abstract—**Network Function Virtualization (NFV) allows multiple Virtual Networks (VNs) to share the underlying physical infrastructure via VN mapping, thus improving the utilization of physical resources. In this paper, for the first time, we study the multicast service-oriented VN mapping that can support big data applications over Elastic Optical Networks (EONs). We first formulate the problem of minimizing the spectrum consumption in multicast service-oriented VN using a Mixed Integer Linear Programming (MILP) model in order to achieve the lower bound on the spectrum consumption. We then propose an efficient heuristic algorithm, called Integrated Genetic and Simulated Annealing (IGSA) algorithm to address the problem with low computational complexity. By encoding node mapping, multicast tree construction, link mapping and spectrum requirements in the same gene and auto-adjusted evolution, and utilizing simulated annealing to find the fittest multicast requests mapping order, IGSA can perform joint optimization for all the multicast requests in a global way. Through extensive simulations, we demonstrate that IGSA outperforms the other heuristic solutions in terms of spectrum consumption, blocking probability and normalized throughput, while achieving close to minimum spectrum consumption with a much lower time complexity than MILP.**

Keywords—Multicast; Virtual Network Mapping; Elastic Optical Networks (EONs); Mixed Integer Linear Programming (MILP); Integrated Genetic and Simulated Annealing

I. INTRODUCTION

Network Function Virtualization (NFV) along with virtualization technologies can provide network services without deploying dedicated appliances. This is accomplished by implementing network functions in software that can be run using standardized high volume servers/switches/storage. A set of network services, sometimes referred to as Service Function Chains [1], can be provided by a *Virtual Network* (VN) consisting of virtual nodes and virtual links. Such a VN is then mapped to the *Substrate Network* (SN) by mapping each virtual node to a physical node (servers/switches/storage), and mapping each virtual link to a physical path (and allocating the necessary bandwidth) in the SN. With NFV, multiple diverse VNs can coexist on a common SN to share the physical resources.

One of the major challenges is how to efficiently map the virtual nodes and virtual links of VNs onto the shared SN, which is known as VN mapping problem [2]. Although many schemes have been proposed to map VNs for unicast service [2-4], few focused on designing efficient strategies to accommodate VNs with multicast service traffic [5]. In fact, many big data applications, distributed file systems (e.g., Map Reduce), point-to-multipoint real-time and interactive applications (e.g., video-conferencing and IPTV) use multicast communications in order to improve the utilization of the physical resources. Unlike the unicast service where data packets are transmitted between a single sender and a single receiver, multicast service requires that the same data packet flows to a selected group of destinations, which can share the data transmission along the common links. For example, to deliver the same data packet to multiple multicast receivers, the data packet may just traverse the common links only once. This can dramatically improve the network utilization efficiency by sharing bandwidth along the common links. To maximize such sharing, the existing unicast VN mapping schemes cannot be directly applied and efficient multicast VN mapping approaches are needed.

The problem of mapping multicast service-oriented VNs has a few unique aspects, e.g., i) the physical multicasting source and destination nodes are not fixed; ii) the same multicasting streams over different virtual links may go through the same substrate link (e.g., fiber); iii) multicast VN mapping allows different multicast tree (VN topology level) design (i.e., determining which destination nodes can be relay nodes to other destinations nodes) in order to maximize network resource utilization. Recently, some multicast tree routing issues were investigated for networks with fixed multicasting source and destination nodes [6-13]. The authors in [6] studied the online routing of bandwidth-guaranteed multicasts in traditional IP networks which can be applied into provisioning bandwidth-guaranteed virtual private network (VPN) services under the "hose" service model. The study in [7] dealt with the multicasting problem using the notion of virtual forces over mobile ad-hoc networks. The authors in [8] studied the joint optimization problem of multicast routing and sparse splitting in WDM networks wherein some switches are incapable of splitting light. The work in [9] focused on supporting multicast routing from electronic layers in Multicast Incapable (MI) WDM networks. The work in [11-13] studied the multicast resource allocation problem over Elastic Optical Networks (EONs) when source and destination nodes are given. However, these works do not fully consider multicast routing together with virtual node mapping and multicast tree design, which may limit the bandwidth sharing on links thus decreasing the utilization of physical resources.

Driven by the recent innovations in Orthogonal Frequency Division Multiplexing (OFDM) technology, the finer-grained subcarriers (i.e., 12.5GHz) were proposed for EONs which is called OFDM-based EONs [14]. OFDM-based EONs can support multiple distance-adaptive modulation formats and the OFDM transponder can assign continuous subcarriers to serve multicast requests with different bandwidth demand. Through efficient and flexible modulation format selection and spectrum allocation, OFDM-based EONs are promising candidates for the next generation optical networks of 100G, 400G and beyond. Hence, in this work, we consider OFDM-based EONs as the substrate network to support multicast service-oriented VNs. The spectrum continuity and spectrum conflict constraints in OFDM-based EONs have to be considered when performing multicast VN mapping, and this introduces additional challenges. However, the existing work on multicast VN mapping either investigated the problem subject to delay and delay variation constraints [5] (enabling multiple description coding based video applications [15]) over general physical network, or focused on mapping multicast VNs with reliability constraints onto a wireless mesh network [16].

In this paper, we study the multicast service-oriented VN mapping problem over OFDM-based EONs with the objective of minimizing the spectrum consumption. Specifically, we consider the use of multiple distance adaptive modulation formats and Multicast Capable Optical Cross-connects (MC OXCs) with limited multicast ability. We propose a Mixed Integer Linear Programming (MILP) model to obtain the lower bound of the spectrum consumption. To solve the problem more efficiently, we design an Integrated Genetic and Simulated Annealing (IGSA) algorithm which simultaneously optimizes node mapping, multicast tree construction, routing, and spectrum assignment for multiple multicast requests. A genetic algorithm is used to encode node mapping, multicast tree construction, and subcarriers requirement. In addition, we use simulated annealing to find the optimal multicast requests mapping order in terms of fitness value. Combining genetic and simulated annealing algorithm, IGSA can efficiently produce the fittest individual (i.e., solution to the problem) which is close to the optimal solution. We evaluate our solutions in terms of complexity, spectrum consumption, blocking probability and normalized throughput.

The reminder of this paper is organized as follows. We describe the problem in Section II and present the MILP model in Section III. Then we propose two heuristic algorithms Greedy and IGSA algorithm in Section IV. The performance evaluation is presented in Section V and finally we conclude the paper in Section VI.

II. MULTICAST SERVICE-ORIENTED VN MAPPING PROBLEM OVER OFDM-BASED EONS

An OFDM-based EON can be modeled as a graph $G = (V, E)$, where V is the set of physical nodes and E is the set of physical fiber links. Each physical node is equipped with C units of computation resources while each fiber link has B subcarriers (the size of each Subcarrier Frequency Slot (SFS) is 12.5 *GHz*). The OFDM-based EONs uses Multicast Capable Optical Cross-connect (MC-OXCs) with limited multicast ability, which can support four types of modulation formats: BPSK, QPSK, 8QAM and 16QAM. The corresponding capacity of one SFS is $C^{SFS} = k$*12.5 *Gb/s*, where k=1,2,3,4 with transmission reaches of 10,000 *km*, 5,000 *km*, 2,500 *km*, 1,250 *km* for the above four modulation formats, respectively. We assume that only BPSK can be used when the distance between the source and destination nodes is more than 10,000 *km*.

For a given multicast VN request $MR_i = (s_i, D_i, b_i), i \in R$, s_i is the virtual source node, D_i ($|D_i|$>1) is the virtual destination node set, and b_i is the requested bandwidth in the multicast group. We assume a given node $v \in \{s_i, D_i\}$, requires $c(v)$ computing resources and can only be mapped onto a subset of physical nodes denoted by $S(v)$.

For each multicast request, we need to find the virtual node mapping, construct a multicast light-tree (i.e., decide which nodes are "spilt and copy" nodes in the optical domain), and determine the routing, modulation selection and spectrum allocation of the light-tree over OFDM-based EONs such that the total spectrum consumption is minimized. In addition to the spectrum continuity and spectrum conflict constraints in the process of Routing and Spectrum Allocation (RSA) in EONs, we have to consider the limited number of "split and copy" on each node for the same multicast traffic, which is referred to as the *fan out* limitation. For example, Fig.1 (a) shows a multicast request, where the number in the rectangle next to each node is the required computing resource, and the bandwidth of the multicast request b is 2 subcarriers in QPSK format. As shown in Fig.1 (b), s_1 , d_{11}, d_{12} and d_{13} can be mapped onto node A, F, C and D, respectively. Node F is the mapping node for d_{11} which is the "split and copy" node. The virtual link s_1-d_{11} is routed on links A-B and B-F; s_1-d_{12} is routed on links A-B, B-F and F-C and s_1-d_{13} is routed on links A-B, B-F and F-D. Bandwidth resources on links A-B and B-F can be shared by three multicast destinations. Assume the number of subcarriers on each fiber link is B =10 as shown in Fig.1 (c), where the gray blocks represent the occupied subcarriers and the blank ones are the available subcarriers, and consecutive subcarrier 5 and 6 can be selected to carry the multicast traffic on the mapped physical links AB, BF, FC and FD.

III. MILP FORMULATION FOR MULTICAST SERVICE-ORIENTED VN MAPPING PROBLEM OVER OFDM-BSAED EONS

In this section, we develop a Mixed Integer Linear Programming (MILP) model to mathematically formulate the

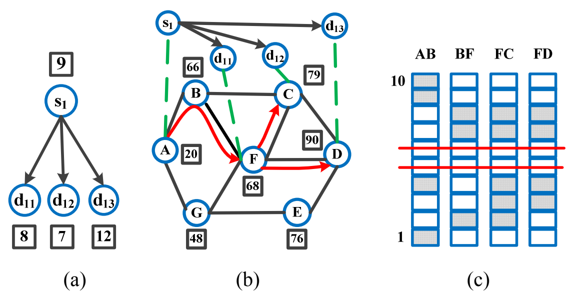


Fig.1 An example of multicast service-oriented VN mapping

multicast service-oriented VN mapping over OFDM-based EONs.

Notations

$G = (V, E)$: graph representing the physical substrate network
$v \in V$: a physical node
$C(v)$: computing capacity of node v
$RC(v)$: residual capacity of node v
N_v : fan-out of node v
$mn \in E$: physical fiber link between m and n
$d(mn)$: distance of fiber link mn
B: the number of subcarriers on each link
B_{GB}: guard band between spectrum allocations for different requests
$MR_i = (s_i, D_i, b_i)$: a multicast request i, s_i is the source node, D_i ($|D_i|>1$) is the destination node set, b_i is the request bandwidth
$V_i = \{s_i\} \cup D_i$: the node set for request MR_i
d_{ij} : the j^{th} destination node in D_i
$c(v_i)$: the computation resource requirement of node $v_i \in V_i$
$S(v_i)$: the set of candidate physical mapping nodes of virtual node $v_i \in V_i$
E' : set of links from virtual source to its physical candidate mapping nodes and links from destination's candidate physical mapping nodes to corresponding destination
$E_{AG} = E \cup E'$
$V_{AG} = V \cup (\sum_i V_i)$: the set of nodes from all requests and physical network
$G_{AG} = (V_{AG}, E_{AG})$: the augmented graph

Variables

$$\sigma_{mn,i} = \begin{cases} 1, \textit{if } MR_i \textit{ uses link } mn \in E_{AG} \\ 0, \textit{otherwise} \end{cases}$$

$$\delta_{mn,ij} = \begin{cases} 1, \textit{if the traffic flow to node } d_{ij} \in D_i \\ \quad \textit{goes through link } mn \in V_{AG} \\ 0, \textit{otherwise} \end{cases}$$

$$o_{ij} = \begin{cases} 1, \textit{if the starting spectrum slot for } MR_i \\ \quad \textit{is smaller than that of } MR_i \\ 0, \textit{otherwise} \end{cases}$$

$$c_{ij} = \begin{cases} 1, \textit{if requests } MR_i \textit{ and } MR_j \textit{ use common link}(s) \\ 0, \textit{otherwise} \end{cases}$$

d_i : the maximal path length in the light-tree of MR_i
n_i : the number of subcarriers MR_i requires
$ss_i(es_i)$: an integer variable indicating subcarrier allocation starting (ending) slot for request MR_i

Objective

The objective is to minimize the total spectrum allocation of all multicast requests across all the fiber links over EONs.

$$\text{Minimize} \sum_{i \in R} \sum_{mn \in E} \sigma_{mn,i} * n_i \quad (1)$$

Constraints

One on one node mapping

$$\sum_{n \in S(s_i)} \sigma_{s_i n,i} = 1, \forall i \in R \quad (2)$$

$$\sum_{m \in S(d_{ij})} \sigma_{m\, d_{ij},i} = 1, \forall d_{ij} \in D_i, \forall i \in R \quad (3)$$

$$\sum_{m \in V_i} \sigma_{mn,i} \le 1, \forall i \in R, \forall n \in V \quad (4)$$

Equation (2) and (3) ensure that each virtual node is mapped to at most one physical node and Equation (4) ensures that multiple virtual nodes from the same request cannot be mapped to the same physical node.

Node capacity constraint

$$\sum_{i \in R} \sigma_{s_i v,i} * c(s_i) + \sum_{i \in R} \sum_{d_{ij} \in D_i} \sigma_{v\, d_{ij},i} * c(d_{ij}) \le C(v), \forall v \in V \quad (5)$$

Equation (5) specifies that the total occupied computing resources on a physical node cannot exceed its capacity $C(v)$.

Fan Out Limitation

$$\sum_{n \in V} \sigma_{vn,i} \le N_v, \forall i \in R, \forall v \in V \quad (6)$$

Equation (6) guarantees that the number of "split and copy" on each physical node cannot exceed its fan out limitation N_v for the same multicast request.

Flow Conservation Constraint

$$\sum_{n \in V} \delta_{s_i n,ij} - \sum_{m \in V} \delta_{m s_i,ij} = 1, \forall j \in \{1...|D_i|\}, \forall i \in R \quad (7)$$

$$\sum_{n \in V} \delta_{d_{ij} n,ij} - \sum_{m \in V} \delta_{m d_{ij},ij} = -1, \forall j \in \{1...|D_i|\}, \forall i \in R \quad (8)$$

$$\sum_{m \in V \cup V_i} \delta_{mn,ij} - \sum_{r \in V \cup V_i} \delta_{\mathrm{nr},ij} = 0, \\ \forall n \in V, \forall j \in \{1...|D_i|\}, \forall i \in R \quad (9)$$

Equations (7)-(9) ensure that traffic from s_i to the destination is routed on exactly one path.

Multicast Tree Construction

$$\sigma_{mn,i} \ge \delta_{mn,ij}, \forall j \in \{1...|D_i|\}, \forall i \in R, \forall mn \in E_{AG} \quad (10)$$

Equation (10) ensures that all the links used by MR_i are considered at most once if it is used by any source and destination node pair in MR_i to construct the multicast tree.

Modulation Selection

$$d_i = \max_{j \in \{1...|D_i|\}} \sum_{mn \in E} \delta_{mn,ij} * d(mn), \forall i \in R \quad (11)$$

$$n_i = \left\lceil \frac{b_i}{C_i^{SFS}} \right\rceil + B_{GB} \quad (12)$$

Equation (11) calculates the distance of the longest path in the multicast light-tree. Equation (12) is used to calculate the number of subcarriers that MR_i requires, where C_i^{SFS} is determined by the reach distance d_i as described in Section II.

Spectrum Allocation Constraint

$$es_i - ss_i + 1 = n_i, \forall i \in R \tag{13}$$

$$es_i, ss_i \in (0, B], \forall i \in R \tag{14}$$

$$c_{ij} \geq \sigma_{mn,i} + \sigma_{mn,j} - 1, \forall i, j \in R, i \neq j, \forall mn \in E \tag{15}$$

$$o_{ij} + o_{ji} = 1, \forall i, j \in R, i \neq j \tag{16}$$

$$es_j - ss_i + 1 \leq B(1 + o_{ij} - c_{ij}), \forall i, j \in R, i \neq j \tag{17}$$

$$es_i - ss_j + 1 \leq B(2 - o_{ij} - c_{ij}), \forall i, j \in R, i \neq j \tag{18}$$

Equation (13) shows that continuous subcarriers are allocated to the light-tree. Equation (14) is the spectrum capacity constraint. Equation (15)-(18) ensures that the spectrum conflict and continuity constraints are satisfied.

The above MILP model can obtain optimal results in terms of minimizing the total spectrum allocation. However, due to the computational complexity, the MILP model is impractical to solve for a large-scale physical network. Hence, in the next section, we propose efficient heuristics.

IV. Heuristics for Multicast Service-oriented VN Mapping Problem over OFDM-based EONs

In this section, we propose two heuristic algorithms which are the Greedy algorithm and the Integrated Genetic and Simulated Annealing (IGSA) algorithm to efficiently solve the problem of multicast VN mapping over OFDM-based EONs.

A. The Greedy Algorithm

To efficiently solve the multiple multicast VN mapping, the Greedy algorithm sorts all multicast requests in decreasing order of their bandwidth requirements, and then greedily maps them. For each unmapped multicast request, the Greedy algorithm first maps the source node and then greedily maps the destination nodes and constructs the light-tree using the link mapping first concept in [17]. After virtual node mapping, multicast tree construction and virtual link mapping are determined, the most efficient modulation level that can satisfy the transmission reach of the longest path in the optical multicast tree is employed. Then the number of subcarriers required is calculated using Equation (12). Finally, the Greedy algorithm uses the first-fit approach to allocate subcarriers while considering the spectrum continuity and spectrum conflict constraints. The detailed steps of the Greedy algorithm are shown in **Algorithm 1**, where $(d_{sp}(vm), m, SP(vm)) \in Vector(v), \forall v, m \in V$; $SP(vm)$ is the shortest path from node v to node m, $\forall v, m \in V$; $d_{sp}(vm)$ is the corresponding distance; T=($root$, Dst, U, E_T) is the multicast tree over the physical network; $root$ is the physical node used to map the source; Dst is the set of physical nodes used to map all the multicast destinations; U is the set of non-destination nodes in the multicast tree and E_T is the link set for the multicast tree.

B. The Integrated Genetic and Simulated Annealing (IGSA) Algorithm

Although the Greedy algorithm has a much lower computational complexity compared to the MILP, it maps multiple multicast requests one by one sequentially by solving the node mapping, multicast tree construction, link mapping and spectrum allocation in separated steps. In this subsection, we propose an Integrated Genetic and Simulated Annealing (IGSA) algorithm that jointly optimizes node mapping, multicast tree construction, routing and spectrum allocation for all the multicast requests. In the following subsections, we present the encoding mechanism, the fitness function, the evolution process, and the convergence condition for IGSA.

1) Genetic Encoding

We encode each *gene* as the provisioning for a single multicast request. An *individual* composed by a set of different *genes* represents the provisioning for all the multicast requests and a *population* is a set of individuals. Specifically, we encode each $Gene_i$ as {$NodeM_i$, $SteinerT_i$, n_i} for multicast request MR_i, where $NodeM_i$ is the node mapping index defined in Equation (19); $SteinerT_i$ is multicast tree index formulated in Equation (20) which is associated with an alternative Steiner Tree (ST) and n_i is the number of subcarriers MR_i requires.

Algorithm 1: Greedy Algorithm

Pre-calculation: For each physical node v, calculate shortest paths to all other physical nodes and store corresponding results in $Vector(v)$.

1: Sort all the virtual requests decreasingly according to bandwidth requirement b_i and calculate n_i assuming it uses the lowest modulation level in Equation (12);

2: For each MR_i , set $T = \Phi$, map the source node s_i by choosing the available candidate physical node v' with largest available subcarriers on its outgoing links as the mapping node for source node s_i , let $root = v'$ in T ;

3: Map destination nodes;

1) Find a physical node $m \in V$ and $m \notin \{root\} \cup Dst$ that is the candidate mapping node for some destination node $d_{ij} \in D_i$ where $d_{sp}(vm)$ is the smallest $\forall u \in \{root\} \cup Dst \cup U$, and make sure the spectrum capacity constraint on $SP(vm)$ and fan-out limitation constraint on node u can be satisfied;
2) If it fails to find such a node described in 1), block the current multicast request and go to **2;**
3) If there is more than one destination node that uses physical node $m \in V$ and $m \notin \{root\} \cup Dst$ as the candidate mapping node, select node d_{ij} with the least physical candidate mapping nodes to map first;
4) Map node d_{ij} onto node m, remove node d_{ij} from D_i;
5) Copy all the links along $SP(vm)$ into set E_T, node m into Dst and all the intermediate nodes along $SP(vm)$ into node set U ;

4: Go to **3** if $D_i \neq \Phi$;

5: Map multicast request i according to T by recalculating n_i and allocate spectrum resources using the first-fit approach;

6: Update network status and go to **2** to process the next request if it exists, otherwise end the process.

$$NodeM_i \in \left\{ \begin{array}{l} [1, NM_1],\ when\ i = 1 \\ [\sum_{k=1}^{k=i-1} NM_k + 1, \sum_{k=1}^{k=i} NM_k],\ when\ 1 < i \le R \end{array} \right\} \quad (19)$$

$$SteinerT_i \in [(NodeM_i - 1)\cdot N_T + 1, NodeM_i \cdot N_T], \forall i \in R \quad (20)$$

In Equation (19-20), NM_i is the number of combinations of node mappings for MR_i, and N_T is the number of alternative STs for each node mapping. For each MR_i, we randomly select an index $NodeM_i$ to present its node mapping, and then we randomly pick an index $SteinerT_i$ to present its multicast light-tree construction. Afterwards, we use Equation (11) and (12) to calculate n_i. We repeat this process for each multicast request to obtain an individual. A different node mapping and ST for each request is randomly selected (i.e., new gene is generated) to generate more individuals and those different individuals are grouped together to form a population of size *P*. The corresponding gene is marked as blocked when any physical mapping node (link) does not have sufficient computing (spectrum) resources for the request or cannot satisfy the fan-out limitation.

2) Fitness Function

We evaluate each individual by the fitness function in Equation (21), where F_{total_s} is the total allocated subcarriers for all the requests served; *H* is a large positive constant to suppress the fitness value if there is blocking and F_{bb} is the total bandwidth demand from the source to all its destinations of all the blocked requests. The fitness function maximizes accepted multicast requests due to a relatively large suppression factor *H*, and also tries to minimize the spectrum allocation for the non-blocked multicast requests. Note that a different order of mapping all multicast requests in an individual will affect its fitness value due to the spectrum fragmentations generated during spectrum allocation process (IGSA simply uses the first-fit approach to allocate spectrum resources for each request). In order to find an optimized request mapping order for the multiple multicast requests in each individual in terms of fitness value, we design a Simulated Annealing Algorithm (SAA) as shown in **Algorithm 2**. A configuration *C* is defined as an order of genes (requests) in which the node resource and spectrum allocation are addressed and the energy function *E*(*C*) is defined the same as the fitness function in Equation (21). Temperature *T* is a global time varying parameter and how this temperature is varied over time is the annealing schedule. The probability P_r defined in Equation (22) will decrease when *T* becomes smaller.

$$F = F_{total\_s} + H * F_{bb} \quad (21)$$

$$P_r = e^{-(E(N)-E(\mathrm{C}))/\mathrm{T}} \quad (22)$$

3) Design of IGSA

At the beginning of IGSA, the first generation *G* of size *P* is initialized randomly, then *G* goes into the evolution phase which mainly includes selection, crossover and mutation operations. Specifically, a fixed number of individuals denoted as G_S are randomly selected from the population *G*, then we apply the tournament selection within individual set G_S and the winner of each tournament (i.e., the fittest one in the competing group) is selected to evolve to the crossover phase. We randomly pair all the winners as parents to do multipoint gene level crossover to get offspring. For each parent pair, we randomly choose |R|*p_c (where p_c is the cross rate) number of genes to swap. We then select *P* fittest individuals from the parents' generation population and their offspring pools to keep the population size constant. The chosen *P* fittest individuals then go into the mutation phase. In the mutation phase, a number of genes are randomly selected by mutation ratio |R|*p_m (where p_m is the mutation rate) for each offspring. For each chosen gene, the node mapping index and ST index are mutated by the mutation probability which is usually a very small number between 0.001 and 0.1 according to the size of population and length of individual. To improve IGSA's performance, we adopt an adaptive strategy [18] to dynamically adjust the crossover rate p_c and mutation rate p_m based on the individuals' fitness as shown in Equation (23) and (24):

$$p_c = \left\{ \begin{array}{l} \alpha_c \dfrac{F_{p1p2} - F_{\min}}{F_{mean} - F_{\min}} + P_{c0}, F_{p1p2} \le F_{mean} \\ \beta_c,\ otherwise \end{array} \right\} \quad (23)$$

$$p_m = \left\{ \begin{array}{l} \alpha_m \dfrac{F_p - F_{\min}}{F_{mean} - F_{\min}} + P_{m0}, F_p \le F_{mean} \\ \beta_m,\ otherwise \end{array} \right\} \quad (24)$$

where F_p is the fitness of individual *p*; $F_{\min}$ is the smallest fitness value in the population; F_{mean} is the average fitness value and F_{p1p2} are the average fitness value of individual p_1, p_2. $\alpha_c, \beta_c, \alpha_m, \beta_m \in [0,1]$ are randomly generated constant coefficients, and P_{c0} and P_{m0} are default rates for the fittest individuals in the population.

The mutated child who has decreased fitness value will replace the individual with highest fitness value to keep population size unchanged. It then goes to the next evolution stage with this new generation. After the convergence, it will map all multicast requests according to the genetic encoding of the best fitness individual in the last generation and its corresponding configuration C. The detailed procedure of IGSA is shown in **Algorithm 3**.

Algorithm 2: Simulated Annealing Algorithm (SAA)

1: Initialize configuration *C* randomly, set temperature *T*, and calculate *E*(*C*);

2: Generate a new configuration *N* by randomly swapping two neighboring genes and calculate *E*(*N*);

3: If *E*(*N*) < *E*(*C*), go to **5** ; otherwise go to **4**;

4: If $P_r > Rand\,[0,1]$, go to **5**; otherwise go to **6**;

5: Let *C* = *N*;

6: Calculate $T = \gamma * T, where\ 0 < \gamma < 1$;

7: If T=0, go to **8**, otherwise go to **2**;

8: Store configuration C and set *E*(*C*) as the fitness value and terminate the SAA process.

4) Convergence Condition

To evaluate IGSA's convergence performance, we modify the degree of diversity [19] as in Equation (25),

$$D_P = \frac{2}{P(P-1)} \sum_{p1=1}^{P-1} \sum_{p2=p1+1}^{P} \frac{|D_F(p1, p2)|}{F_{\max}} \tag{25}$$

where $|D_F(p_1, p_2)|$ is the absolute value of fitness difference of individual p_1 and p_2; and F_{max} is the maximal fitness value in the generation. If D_p is lower than a certain threshold for 5 generations or more [20], we say the algorithm has converged. We stop IGSA when it converges or the number of iteration reaches a preset threshold.

V. PERFORMANCE EVALUATION

A. Experiment Setting

We evaluate the proposed MILP model and heuristic algorithms on a 14-node, 22-link NSF network. By default, each physical node can provide 10,000 units computing resource, while each physical link has a spectrum capacity of 4.75 *THz* with a subcarrier of 12.5 *GHz* (i.e., 358 subcarriers). Note that the fan out of the MC-OXC on each physical node is randomly selected between 2 and 4 [21]. The guard band for each request is 1 subcarrier [11].

For each multicast request, the source node and destination node set are randomly generated with the number of destinations uniformly distributed between [2, 8]. The computing demand of each node is less than 100 units with a uniform distribution. The set of candidate mapping nodes $S(v)$ (where $|S(v)|$ is within [3,14] following a uniform distribution) for each node v in multicast request are randomly selected from the physical node set. The bandwidth demand of each multicast request is uniformly distributed within 10-100 *Gb/s*. Table I lists other parameter settings. For the purpose of comparison, we modified and implemented the naturally-inspired (NI) algorithm proposed in [12].

Algorithm 3: Integrated Genetic and Simulated Annealing (IGSA) Algorithm

1: Initialize the first generation G with population size P;

2: For each individual, call SAA to get the fitness value;

3: Selection and crossover to get offspring;

4: For each child, call SAA to calculate the fitness value;

5: Select P fittest individuals from parents and children, and then the chosen children go to the mutation phase;

6: Mutation operation and for each mutated child, call SAA to obtain the fitness value;

7: Use mutated child with decreased fitness value to replace the individual with highest fitness value until all the satisfied individuals are replaced while keeping population size P constant to get new generation G';

8: If it converges or reaches a preset threshold of iteration number, go to **9**; otherwise go to **3** with G';

9: Provide mapping according to the fittest individual in G' and its requests mapping order configuration C, terminate the IGSA process.

All simulations are run on a computer with 2.5 GHz Intel Core i5-3210 CPU and 12 GB RAM. For the MILP, the simulation will be terminated if the optimal solution is obtained or the running time of 5 hours is reached. Each statistic result is the average result of 20 simulations.

B. Performance Analysis

As shown in Fig.2 (a), we first evaluate the spectrum consumption of different schemes when the number of multicast requests increases. We can see that the MILP performs the best and provides the lower bound. It can also be observed that the spectrum consumption of IGSA is close to the lower bound obtained by the MILP formulation, consuming 31%-56% (36%-64%) less spectrum resources than that of Greedy (NI) due to the joint optimization of virtual node mapping, light-tree construction, and routing and spectrum allocation for all the multicast requests. In Table II, we list the computational time of different schemes when the number of multicast requests varies. We can see that IGSA's time complexity is smaller than that of NI and much smaller than that of the MILP (which cannot find the optimal solution in a reasonable time when the problem become large). Similarly, from Fig. 2 (b) we can see that IGSA converges much faster than NI (with number of multicast requests as 20) due to the dynamically updated crossover rate p_c and mutation rate p_m (the curves are crossed because of using different ordinates for different fitness functions). Hence, IGSA can also be applied to the case with dynamic requests due to the low time complexity, particularly when the real time service is sensitive to the provisioning time.

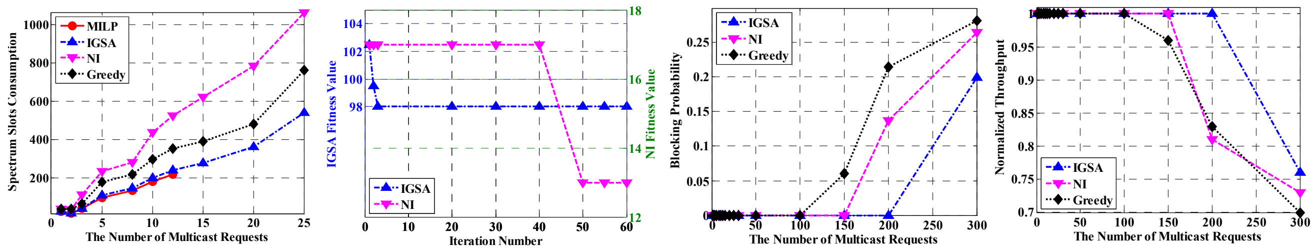
We also evaluate the blocking probability (i.e., the number of blocked multicast requests over the total number of multicast requests) and the normalized throughput (i.e., the sum of allocated bandwidth over the total sum of requested bandwidth) when the network resources are insufficient for different number of multicast requests. From Fig.2 (c) we observe that the first blocking occurs much later when

TABLE I. Parameter Setting for IGSA and SAA

Parameter	Value	Parameter	Value
Population Size	50	Iteration Limitation of IGSA	1200
Number of STs	3	Threshold of DP	0.1
Tournament Size	0.3*P	Suppression factor H	1000
P_{c0}	0.2	Initial temperature T	6
P_{m0}	0.05	γ	0.95

TABLE II. Computation time (seconds) of different solutions

	3	5	10	12	20	100
MILP	1004.61	1011.56	1255.18	1853.33	*	*
IGSA	1.25	1.92	3.12	3.41	4.96	15.92
Greedy	0.04	0.08	0.12	0.14	0.23	1.14
NI	3.86	4.31	4.73	5.35	5.90	16.41



(a) Spectrum consumption (b) Convergence speed (c) Blocking probability (d) Normalized throughput

Fig.2 Results for Multicast VN mapping over OFDM-based EONs

using IGSA compared to Greedy and NI algorithms. More specifically, IGSA does not block any request until it provisions around 200 requests while the Greedy and NI algorithms start to block requests after provisioning around 100 and 150, respectively. The blocking performance improvement of IGSA over Greedy (NI) is at least 30% (25%). In Fig. 2 (d), IGSA's normalized throughput is 4%-24 (9%-21%) higher than that of Greedy (NI). This is because IGSA does not try to minimize the blocking probability by accepting small bandwidth requests. Instead, IGSA also accommodates large bandwidth requests to increase the normalized throughput.

VI. CONCLUSION

In this paper, we have investigated the multicast service-oriented VN mapping problem over OFDM-based EONs. We have proposed a Mixed Integer Linear Programming (MILP) model that can achieve the lower bound of spectrum consumption. In addition, we have proposed an efficient heuristic, namely Integrated Genetic and Simulated Annealing (IGSA) algorithm, which can jointly optimize the process of virtual node mapping, light-tree construction, routing, and spectrum allocation for all the multicast requests globally. Simulation results have shown that the proposed IGSA outperforms existing heuristic schemes in terms of spectrum consumption, blocking probability and normalized throughput. Our study shows that IGSA can achieve the outcomes that are close to the optimal values obtained by the MILP with a much lower time complexity.

REFERENCES

[1] Network function virtualization white paper,"http://portal.etsi.org/NFV/NFV White Paper.pdf".

[2] Z. Cai, F. Liu, N. Xiao, Q. Liu, and Z. Wang, "Virtual Infrastructure embedding for evolving networks," in Proc of GLOBECOM, pp.1–5, 2010.

[3] M. Yu, Y. Yi, J. Rexford and M. Chiang, "Rethinking Virtual Infrastructure Embedding: Substrate Support for Path Splitting and Migration," SIGCOMM Computer Communications Review, vol.38, no.2, pp.17-29, 2008.

[4] N.M. M. K. Chowdhury, M. R. Rahman, and R. Boutaba, "Virtual Infrastructure embedding with coordinated node and link mapping," in Proc. of IEEE INFOCOM, pp.783-791, 2009.

[5] M. Zhang, C. Wu, M. Jiang, and Q. Yang, "Mapping multicast service-oriented Virtual Infrastructures with delay and delay variation constraints," in Proc. of GLOBECOM, pp.1–5, 2010.

[6] M. Kodialam, TV Lakshman, and S. Sengupta, "Online multicast routing with bandwidth guarantees: a new approach using multicast network flow," IEEE/ACM Transactions on Networking, vol.11, no.4, pp.676-686, Aug. 2003.

[7] M. Parameswaran, V. Rastogi, and C. Hota, "A virtual-force based multicast routing algorithm for mobile ad-hoc networks," the Fifth International Conference on Ubiquitous and Future Networks (ICUFN), 2013.

[8] X. Zhang, J.Y. Wei, C. Qiao, "Constrained multicast routing in WDM networks with sparse light splitting," Journal of Lightwave Technology, vol.18, no.12, pp.1917-1927, Dec. 2000.

[9] A. Gadkar, J. Plante, and V. Vokkarane, "Multicast overlay for high-bandwidth applications over WDM networks," [J]. Journal of Optical Communications and Networking,vol.4, pp.571–585, Aug. 2012.

[10] A. Ding, and G. Poo, "A survey of optical multicast over WDM networks," Computer Communications, vol.26, no.2, pp.193-200, 2003.

[11] L. Gong, X. Zhou, X. Liu, W. Zhao, W.Lu, and Z. Zhu, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," [J].Journal of Optical Communications and Networking, vol.5, no.8, pp.836-847, Aug. 2013.

[12] A. Patel, P.N. Ji, J.P. Jue, and T. Wang, "Light-tree establishment for optical multicasting in Flexible Optical WDM (FWDM) networks," Asia Communications and Photonics Conference (ACP), pp.1-3, Nov. 2012.

[13] Q. Wang and L. Chen, "Performance analysis of multicast traffic over spectrum elastic optical networks," in Proc. of OFC, pp. 1–3, 2012.

[14] O. Gerstel, M. Jinno, A. Lord, and SJ B. Yoo, "Elastic optical networking: a new dawn for the optical layer?," IEEE Communications Magazine, vol.50, no.2, pp. S12–S20, Apr. 2012.

[15] Y. Miao, Q. Yang, C. Wu, M. Jiang, and J. Chen, "Multicast Virtual Infrastructure mapping for supporting multiple description coding-based video applications," Computer Networks, vol.57, no.4, pp.990-1002, Mar. 2013.

[16] P. Lv, Z. Cai, Jia Xu, and M. Xu, "Multicast Service-Oriented Virtual Infrastructure Embedding in Wireless Mesh Networks," IEEE Communications Letters , vol.16, no.3, pp.375-377, Mar. 2012.

[17] Z. Ye, A. N. Patel, P. N. Ji, C. Qiao and T. Wang, "Virtual Infrastructure Embedding over Software-Defined Flex-grid Optical Networks," GLOBECOM workshop on SDN on Optics, 2013.

[18] M. Srinivas, and L. M. Patnaik, "Adaptive probabilities of crossover and mutation in genetic algorithms," IEEE Transactions On Systems, Man, and Cybernetics Society, vol.24, pp.656- 667, Apr. 1994.

[19] L. Gong, X. Zhou, W. Lu, and Z. Zhu, "A two-population based evolutionary approach for optimizing routing, modulation and spectrum assignments (RMSA) in O-OFDM networks," IEEE Communications Letters, vol.16, no.9, pp.1520–1523, Sept. 2012.

[20] J. Koza, "Genetic Programming: On the Programming of Computers by Means of Natural Selection," Cambridge, MA: MIT Press, 1992.

[21] G.N. Rouskas, "Optical layer multicast: rationale, building blocks, and challenges," Network IEEE, vol.17, no.1, pp.60-65, Jan./Feb., 2003