Virtual Network Embedding in Hybrid Datacenters with Dynamic Wavelength Grouping

Raksha Srinivas\(^*\), Soumya Hegde\(^†\), Dinil Mon Divakaran\(^‡\), Mohan Gurusamy\(^§\)
\(^*\)Finisar Corporation, Email: raksha.srinivas@finisar.com
\(^†\)Hewlett-Packard, Email: soumya.hegde@hp.com
\(^‡\)Institute for Infocomm Research, Email: divakarand@i2r.a-star.edu.sg
\(^§\)Department of Electrical and Computer Engineering, National University of Singapore, Email: elegm@nus.edu.sg

Abstract—With ever increasing traffic demands, datacenter networks are envisioned to be a hybrid of both optical and electrical networks. In this context, we consider the recently proposed dynamic wavelength grouping (DWG) architecture for the optical network. This architecture can dynamically group wavelengths from different ports onto a single fiber carrying fixed number of wavelength groups. We focus on the joint problem of VM-placement and bandwidth allocation in such a hybrid optical-electrical datacenter network with DWG capability. There are multiple challenges: (i) the number of edge-switches that can be simultaneously reached using optical paths from an edge-switch is limited by cost; and (ii) wavelength-group continuity constraint. Abstracting the requests of tenants as virtual networks, we study the novel problem of embedding virtual networks on this hybrid datacenter, which translates to the joint problem of bandwidth allocation and placement such that the requirements of virtual networks are satisfied. We develop and analyse two algorithms for embedding dynamically arriving virtual network demands on a hybrid datacenter with DWG capability. The performance studies demonstrate the effectiveness of exploiting existing optical paths as well as using electrical links in the face of multiple constraints to accept higher number of requests.

Index Terms—datacenter, bandwidth, optical, embedding

I. INTRODUCTION

Datacenters are expected to see increasing traffic demands in the coming years. The global datacenter traffic growth rate is estimated to be \(\approx 25\%\) per year till 2017; and the annual global datacenter IP traffic is estimated to reach \(7.7\) zettabytes \((10^{21}\) bytes) by the end of 2017 [1]. The annual traffic growth rate between datacenters, between datacenters and end-users, as well as within datacenters, are all predicted to be \(\approx 30\%\). Of all datacenter traffic, \(76\%\) is estimated to remain within [1]. Given these trends, recent works proposing optical switching based on WDM (wavelength division multiplexing) technology for datacenter networks look promising [8], [15], [3]. In comparison to electrical networks, optical networks have huge bandwidth, and also reduce power consumption and cabling complexity. Yet, building an all-optical datacenter network providing simultaneous connectivity between every pair of edge-switches (top-of-rack switches) is prohibitively expensive. Besides, electrical network is good at multiplexing short and bursty traffic. Therefore, hybrid optical-electrical network architectures are better suited for datacenters.

We consider a hybrid architecture similar to Helios [8] and HyPaC [15], but one that uses dynamic wavelength grouping (DWG) capability in optical network [3]. The architecture is illustrated in Fig. 1. The optical switch connects the edge-switches using optical fibers. In this optical network, the traffic from all servers connected to an edge-switch are carried by different wavelengths on a single fiber. These wavelengths are separated into \(k\) groups by the wavelength selective switch (WSS), before they reach the optical switch (in Fig. 1, \(k\) is set to two). The important novelty with the DWG optical network architecture is that the WSS is reconfigurable. The number of wavelengths forming a group can be decided dynamically, thereby being able to adapt (and increase) the capacity of the optical path connecting two edge-switches as and when required. While the DWG architecture gives flexibility in deciding the number of wavelengths that can be assigned to each of the \(k\) groups from an edge-switch, this also introduces the wavelength-group continuity constraint—an optical path between two edge-switches can be established only if the same group of wavelengths is available at both the edge-

Raksha Srinivas, Soumya Hedge and Dinil Mon Divakaran were affiliated with the NUS while carrying out this work.
switches. Besides, the number of edge-switches that can be simultaneously reached using optical paths from an edge-switch is limited by \( k \), the reachability factor. This is where the electrical network becomes useful, providing connectivity between every pair of edge-switches at the same time.

In this work, we study the problem of bandwidth allocation to VMs (virtual machines) of tenants in a hybrid datacenter with DWG capability. We abstract a request from a tenant as a graph of VM-clusters, called virtual network, and focus on the problem of embedding input virtual networks on a hybrid datacenter network. This can potentially give rise to multiple challenges: (i) the wavelength-group continuity constraint, (ii) the formation of a constrained optical logical topology due to the dynamic creation of optical paths, and (iii) meeting the requirements of input virtual networks. The third challenge translates to the joint problem of placement of VM-clusters of a virtual network and bandwidth allocation for meeting the communication demands between VM-clusters.

Though researchers have recently attempted to solve the problem of bandwidth allocation to tenants in all-electrical datacenter networks, the problem is yet to be addressed in the context of hybrid optical-electrical datacenter network. We describe the problem and the solution approach in Section III. In Sec. IV, we define two different algorithms: (i) ELWA (Existing-Link-Wavelength-Assignment), that exploits existing optical paths to embed edges of arriving virtual networks; and (ii) NLWA (New-Link-Wavelength-Assignment) that always creates new optical paths before exploiting the existing optical paths. To reduce the impact of wavelength-group continuity constraint, both algorithms choose a wavelength with the lowest index among the free wavelengths whenever a new wavelength is needed. We carry out the performance studies in Sec. V. Our results demonstrate, the wavelength-group continuity as well as the constrained optical logical topology can contribute considerably to the rejection of requests. Between the two algorithms, ELWA outperforms NLWA, leading to a significant reduction in the rejection ratio.

II. RELATED WORKS

The importance of allocating network bandwidth to guarantee predictable performance to applications running in datacenters was highlighted in [2]. Recent research works have attempted to solve the problem of bandwidth allocation in all-electrical datacenters, mostly using and assuming advance bandwidth reservation capability [10], [2], [5], [7], [13], [6]. Dynamic allocation of bandwidth on links was proposed in Seawall [14], where bandwidth is allocated proportional to the weights of the competing VMs. SecondNet [10] as well as the integrated resource allocator proposed in [7], both assume as input a matrix specifying bandwidth requirements between every VM-pair of a tenant. However the results in [7] show that, efficient allocation of bandwidth can be done by forming a small number of VM-clusters (four to six). Oktopus [2], on the other hand, introduced two topologies for capturing (as input) bandwidth requirements of VMs of a tenant. One is a star topology resembling a switch connecting VMs; the other, a tree topology abstracting communication demands between VM-clusters. CloudMirror also proposes solution for bandwidth allocation in datacenter network, focussing more on a new abstraction based on clusters of VMs to express the traffic demands of a tenant’s application [12]. Taking cues from these works, we abstract an input request from a tenant as a virtual network as explained in the next section.

The hybrid optical-electrical architecture we consider in this work (refer to Fig. 1) is similar to Helios [8] and HyPaC [8]; the major difference being that in these architectures, the optical network has wavelength groups statically defined. Both these works focus on design and implementation of the architecture. The DWG all-optical network architecture was proposed in [3]. However, as optical paths cannot be created between every pair of edge-switches simultaneously (in the absence of electrical network), the architecture relies on multi-hop communication, routing traffic through a sequence of optical paths with electrical-optical-electrical (o-e-o) conversions at the edge-switches. But o-e-o conversions are undesirable for applications; besides, they also add load to the edge-switches. Hence, we have assumed single-hop communication for the optical network in our hybrid architecture.

In our earlier work [11], we studied the virtual network embedding problem for a different optical networking architecture, wherein fibers carry fixed and same set of wavelengths. The entire set of wavelengths is switched together from an input fiber to an output fiber at an optical switch. More specifically, the architecture in [11] does not have dynamic wavelength grouping capability, cannot support switching of different subsets of wavelengths on a fiber to different output fibers at an optical switch, and does not have wavelength group continuity constraint.

III. EMBEDDING OF VIRTUAL NETWORKS

We abstract input requests from tenants as virtual networks. A virtual network is a graph with edges and nodes, such that the each node represents a cluster of VMs (VM-cluster). The weight of an edge connecting two nodes gives the bandwidth demand between the two corresponding VM-clusters. This is a simpler interface than mandating tenants to specify demands between every VM-pair; besides, most applications can be viewed as a set of tasks, with each task carried out by a VM-cluster. Fig. 2(b) gives an example of an input virtual network, where the nodes a, b, c and d corresponds to VM-clusters.

**Problem:** Given a hybrid optical-electrical network, the problem is to allocate bandwidth to dynamically arriving virtual networks from tenants. The problem translates to the joint problem of placement of VM-clusters as well as bandwidth allocation for communication between VM-clusters.

We divide the problem into two. The first subproblem is to find the subgraph with the maximum degree of every node limited to \( k \), the reachability factor in the optical network. The second subproblem is to embed the subgraph onto the hybrid
datacenter network. We discuss solutions to these problems in the two sections below.

A. Finding degree-constrained maximum weighted subgraph

No edge-switch can simultaneously connect to more the \( k \) edge-switches using optical paths. Therefore, an input virtual network with maximum degree greater than \( k \) (the reachability factor) can not be embedded on the optical network. Hence, the first task is to find a subgraph of the given input virtual network such that no node has a degree greater than \( k \). The unselected edges can be mapped on to the electrical network. While doing so, our objective is to maximize the bandwidth allocated on the optical network. Let \( \vartheta^r \) denote the set of nodes in an input virtual network \( r \), and \( \xi^r \) the corresponding set of connecting edges. Denote by \( w_{(u,v)} \) the weight of the edge \((u,v)\) such that \( u,v \in \vartheta^r \). The problem can be formulated as:

\[
\begin{align*}
\text{maximize} & \quad \sum_{(u,v) \in \xi^r} x_{(u,v)}w_{(u,v)} \\
\text{subject to} & \quad \sum_{v \in \vartheta^r, v \neq u} x_{(u,v)} \leq k; \quad \forall u \in \vartheta^r
\end{align*}
\]

where \( x \)'s are the binary variables selecting the edges of the virtual network. This is similar to the well-studied problem of finding a degree-constrained maximum weighted connected subgraph of a given graph, and is \( \mathcal{NP} \)-hard [9]. However, as the induced subgraph we obtain need not be connected, the problem can be solved in polynomial time (without the connectivity constraint) [9], [4]. We use a linear programming (LP) solver to obtain the maximum weighted induced subgraph.

B. Embedding a subgraph of a virtual network

We consider the optical network as a graph \( G \), where the nodes correspond to the edge-switches of a datacenter, and a link connecting two nodes is the optical path connecting the corresponding edge-switches using the optical switch (see Fig. 1). The nodes in \( G \) can be partitioned into two sets, \( L \) and \( F \); the subgraph formed of nodes in \( L \) is a logical topology established by optical paths, and \( F \) forms a free set of nodes. Any node in the logical topology would have at least one link incident on it; on the other hand, there is no link incident on any node in \( F \). Observe, logical topology evolves dynamically as and when new requests are accepted. Fig. 2(a) shows a graph representation of an example optical network; here \( L = \{1, 2, 3, 4\} \) and \( F = \{5\} \).

After obtaining the subgraph of a virtual network by solving the first subproblem, we proceed to embed the subgraph on the hybrid datacenter. The edges of a virtual network not selected in the maximum weighted induced subgraph will be mapped to the electrical network, but only if the subgraph was successfully embedded on the hybrid network. For each edge of the subgraph, there are three possible operations to embed it on the network:

1) Use existing optical path.
2) Create a new optical path.
3) Use electrical path.

For each of these three operations, it is important that the constraints due to the previously embedded edges (topology constraints of the virtual network) are not violated. Observe that by embedding an edge on a physical link, we are also deciding the placement of the corresponding VM-clusters on the end-points of the link (switches). We now describe each operation elaborately.

1) Operation 1: Use existing optical path

Algorithm 1 performs this operation; it finds a link on the logical topology, and embeds an edge of the subgraph \( r \) on the link. Let \( \mathcal{B}^r \) and \( \mathcal{B}^l \) denote the bandwidth matrices of the subgraph \( r \) and the logical topology, respectively (with \( \infty \) denoting bandwidth on unestablished optical links). Denote by \( \mathcal{E}^r \), the set of edges of subgraph \( r \) previously embedded; and let \( \mathcal{V}^r \) be the corresponding set of nodes of \( r \) that have been mapped to edge-switches (in implementation, they are ordered lists for easy access to the corresponding mapped element). We denote by \( \mathcal{E}^l \) and \( \mathcal{V}^l \) the set of links and switches on the logical network, on which the sets of \( \mathcal{E}^r \) and \( \mathcal{V}^r \), respectively, are embedded. For each of the above set, say \( X \), we denote the complement by \( \overline{X} \).

Algorithm 1 basically tries to find an existing optical path to which an edge of the virtual request can be mapped. To find the end-points of the optical path, it first checks if there is any unmapped edge in the virtual network with both the end-nodes mapped (lines 1-12); if the remaining part, it finds one or two edge-switches to identify the optical path. Once the path is identified, the edge can be mapped if there is either sufficient unused bandwidth or free wavelength. For a switch \( s \) in the logical topology, \( \mathcal{W}_s \) is the set of free (used) wavelengths at that switch.

To illustrate, assume the current state of the optical network to be as depicted in Fig. 2(a). At each edge-switch there are four wavelengths: \( \lambda_1, \lambda_2, \lambda_3 \) and \( \lambda_4 \). Take \( k \) to be two. In the current state, some optical paths are already established. Hence, the free wavelengths available at node 1, \( \mathcal{W}_1 = \{\lambda_3, \lambda_4\} \). Assume bandwidth demands are symmetric. Let there be one unit of unused bandwidth available in the optical path between nodes 1 and 2, and none in other two optical paths. Consider an input virtual network shown in Fig. 2(b), with the weight of each edge being one unit of bandwidth. As none of the nodes in this virtual network have been mapped, a least degree node, say \( b \), is selected (in line no. 14). For the optical network, among the least degree nodes in \( L \), tie is broken randomly between 2 and 3, and assume 2 is selected. Since there is a sufficient bandwidth on the optical link \((1, 2)\), the edge \((a, b)\) is mapped on to it, as shown in Fig. 2(c).

The function add_wavelength in the algorithm adds wavelength(s) in the optical path between the two given nodes. The number of wavelengths added is the minimum required to meet the bandwidth specified as the third argument. An important point to note here is that, whenever a new wavelength is added between two nodes, say \( s \) and \( t \), the wavelength chosen is the
Algorithm 1 ExistingOpticalLinkEmbedding(r)
1: if \( \exists (u, v) \in E^r, \exists (u \in V^r \land v \in V^r) \) then
2:  Find \( s, t \in V^l \) corresponding to \( u, v \) respectively
3:  if \( (B^l_{s,t}) \geq (B^r_{u,v}) \) then
4:      update_link((s, t), (s, t))
5:  return \([u, v], (s, t)\]
6: end if
7: if \( W_s \cap W_t \neq \{\} \) then
8:   add_wavelength(s, t, B^r_{s,t})
9:   return \([u, v], (s, t)\]
10: end if
11: return FAIL
12: end if
13: if \( \exists (u, v) \in E^r, \exists (u \in V^r \lor v \in V^r) \) then
14:   Pick least degree node \( u \in V^r \)
15:   Pick least degree node \( s \in V^r \), \( \exists \text{degree}(s) \geq \text{degree}(u) \)
16: else
17:   Let \( u \) be the node \( \exists (u \in V^r \land v \notin V^r) \)
18:   Find \( s \in V^r \) corresponding to \( u \)
19: end if
20: if \( \sum_{v \in \text{neighbours}(s)} (B^l_{s,v}) \geq \sum_{v \in \text{neighbours}(u)} (B^r_{u,v}) \) then
21:   \( t = \max(B^l_{s,v}) \); \( v = \max(B^r_{u,v}) \)
22: if \( (B^l_{s,t}) \geq (B^r_{u,v}) \) then
23:   update_link((s, t), (s, t))
24: return \([u, v], (s, t)\]
25: end if
26: if \( W_s \cap W_t \neq \{\} \) then
27:   add_wavelength(s, t, B^r_{s,t})
28: return \([u, v], (s, t)\]
29: end if
30: end if
31: return FAIL

one with the smallest index in the set \( W_s \cap W_t \). This is to reduce the effect of wavelength-group continuity constraint. If a path does not already exist, the function add_wavelength will also configure the optical path before adding wavelength(s).

2) Operation 2: Create a new optical path

A new optical path between two edge-switches can be created in two ways as described below.

Operation 2a: Select any two nodes from the free set of nodes \( F \), and establish an optical path with the required number of wavelengths starting with the smallest index.

Operation 2b: Find an appropriate node from \( L \) that is part of the logical topology such that the degree of the node is less than \( k \). Select another node, either from \( L \) or from \( F \); and create an optical path between the two nodes. Again, the new optical path will consist of the (required number) of wavelengths with smallest indices. Algorithm 2 finds an unmapped edge with maximum weight from the given subgraph, such that at least one incident node is already mapped (to a switch in \( L \)). The algorithm then searches for another edge-switch to which an optical path should be created. For the example request in Fig. 2(b), no existing optical paths can be used for embedding the unmapped edges; so we proceed to create a new optical path. At this time, \( a \) is already mapped. We now have, \( E^r = \{(a,b), (a,c)\} \), \( V^r = \{a,b\} \), \( E^l = \{(1,2)\} \), \( V^l = \{1,2\} \), \( E^F = \{(a,b), (a,d)\} \).

Both the remaining edges demand a unit of bandwidth each. Assume, after breaking tie randomly \((a, c)\) is selected for mapping. Algorithm 2 gives priority in selecting two nodes in \( L \) to create a new link (than using a node from \( F \)); and hence chooses node 3, and creates a new link (1,3) to map \((a, c)\) as illustrated in Fig. 2(d). Observe, the new link was created with wavelength \( \lambda_4 \), the next free wavelength with the lowest index commonly available at nodes 1 and 3. Going further, the only edge remaining is \((a, d)\). Since \( a \) is mapped on node 1, the natural choice is node 4; but the free wavelength available at node 1 is \( \lambda_3 \) while that available at node 4 is \( \lambda_4 \). This wavelength-continuity constraint blocks creation of another optical path between nodes 1 and 4. Hence edge \((a, d)\) has to be mapped to \((1,5)\) after the new link is created.

Algorithm 2 NewOpticalLinkEmbedding(r)
1: Find maximum weighted edge \((u, v) \in E^r, \exists u \in V^r \)
2: Let \( s \in V^l \) be the node on which \( u \) is mapped
3: if degree(s) \( \geq k \) then
4:   return FAIL
5: end if
6: if \( v \in V^r \) then
7:   Let \( t \in V^l \) be the node on which \( v \) is mapped
8: else
9:   if \( \exists t \in L, \exists \text{degree}(t) < k \) then
10:      Let \( t \) be the other edge-switch
11: else
12:      Choose a node randomly from \( F \) as \( t \)
13: end if
14: end if
15: add_wavelength(s, t, B^l_{s,t}); update all sets
16: return \([u, v], (s, t)\]

3) Operation 3: Use electrical path

The third option is to use an electrical path with sufficient bandwidth. This possibility is explored only after one or more edges of the input subgraph are already mapped on to the
optical network. Hence, due to the previously mapped edges, there is restriction on the set of edge-switches that can be selected here; and this also limits the number of possible electrical paths. Therefore we proceed as in Operation 2, except that, instead of creating an optical path, we check if the aggregate of all paths between the two selected switches have bandwidth not less than the weight of the edge. If successful, the set of selected paths identified as a logical link is selected for embedding the edge. We assume multipath routing takes cares of splitting and load balancing the traffic onto the different electrical paths. To ensure high optical utilization, we restrict the number of edges of a given subgraph that can be mapped to electrical network by \( m \), defined as a step function of the instantaneous acceptance ratio (defined in Sec. V-A).

IV. ALGORITHMS FOR EMBEDDING VIRTUAL NETWORKS

We develop two algorithms that use the three operations defined above to embed subgraph of a virtual network, but each algorithm follows a different order. However, both the algorithms solve the first subproblem of finding the degree-constrained maximum weighted subgraph of an input virtual network using the same approach (LP solver). Similarly, in both the algorithms, the edges not selected in the maximum weighted induced subgraph are mapped onto the electrical network (using Operation 3 described above). Lastly but most importantly, recall (from the descriptions of Operations in the previous section), we choose a wavelength with the lowest index among the free wavelengths when needed to reduce the impact of wavelength-group continuity constraint. The need for a new wavelength arises whenever a wavelength group is created with a new optical path or an existing wavelength group is expanded with the provisioning of additional capacity. The algorithms are defined below.

A. ELWA: Existing-Link-Wavelength-Assignment

This algorithm first explores existing optical paths to embed an edge of the subgraph, before creating new optical paths. To be specific, the ELWA first explores Operation 1, followed by Operation 2 and then Operation 3. Within Operation 2, ELWA will try to create a link involving a link in \( L \) (Operation 2b) before proceeding to create a link between two nodes in \( F \) (Operation 2a). Finally, if not successful yet, the ELWA will explore electrical paths for embedding the edge (Operation 3).

B. NLWA: New-Link-Wavelength-Assignment

NLFA always attempts to create a new optical link first, before exploring existing optical links, to embed an edge of a given subgraph. Therefore, the order is Operation 2 (Operation 2a, and then Operation 2b), followed by Operation 1, and finally Operation 3. Mapping to electrical paths is given the least priority.

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the two bandwidth allocation algorithms, namely, ELWA and NLWA.

A. Experimental setup

We simulated a hybrid datacenter network supporting DWG (see Fig. 1) with 100 edge-switches. Each edge-switch had 32 ports connected to the electrical network, and another 32 ports connected to the optical switch using a single fiber. The 32 ports corresponds to 32 unique wavelengths. The bandwidth per port (wavelength) was 1 Gbps. The reachability factor, \( k \), was set to four. This means, the maximum number of groups that can be formed at any edge-switch was limited to four; and each group can carry a minimum of one wavelength to a maximum of 32 wavelengths. Hence the minimum and maximum optical capacity between two edge-switches (carried by an optical path) is 1 Gbps and 32 Gbps, respectively.

As mentioned earlier, \( m \) is a (step) function of the instantaneous acceptance ratio; and it limits the number of edges of a subgraph that can be mapped to electrical network. For our experiments here, \( m \) is seven for acceptance ratio below or equal to 0.75, and then the function decreases by 1 at steps of 0.05, reaching a lower limit of 2 for acceptance ratio between 0.95 and one. That is to say, when the acceptance ratio increases, lesser number of edges can be mapped to electrical network.

B. Input scenarios

Inputs to the bandwidth allocation algorithms are a set of dynamically arriving virtual networks. The bandwidth demands between node-pairs of a virtual network were drawn from the Exponential distribution with mean 200 Mbps. The virtual networks are generally small in size, as the node in a virtual network corresponds to a clusters of VMs. We highlight, previous works have either considered small number of VM-clusters as input [2], [12], or found grouping VMs into small number of VM-clusters suffices for efficient bandwidth allocation [7]. We consider three different scenarios, each having one particular topology for the set of virtual networks.

1) Star: The star topology consists of a master node and several slave nodes, with a master-slave communication pattern. The number of nodes in a virtual network is randomly and uniformly chosen between five and nine.

2) Tree: A three-level tree topology is considered here. Given that the root node is at level one, the number of nodes at level two is fixed as three. The total number of nodes is randomly chosen from the range \([5–10]\); hence the number of nodes in level three of any virtual network with this topology is between one and six. This topology allows oversubscription—the bandwidth demand of a link between two node in level one and two is half the sum of the bandwidth demands of links connecting that particular node in level two and the nodes of level three.

3) Random: In this scenario, random connected graphs are generated, with the number of nodes selected randomly from the range \([5–10]\). The maximum number of edges is limited to 20.

While most applications can be abstracted by star and tree topologies (mapreduce, communicating scientific tasks,
path(s) can not be established due to wavelength-group continuity constraint. Analyzing further, we observed that more than half the total capacity of the established optical paths was not utilized. Hence we rule out the first reason. To analyse the affect of wavelength-group continuity constraint, we recorded the number of instances where requests were rejected due to this constraint. Fig. 3(b) gives the rejection percentage due to the wavelength-group continuity constraint for both the algorithms. We see that these rejections increase for both allocation mechanisms with increasing number of virtual networks. As the number of inputs increases, the gap between NLWA and ELWA widens, with NLWA rejecting more the double the number of requests than ELWA, due to wavelength-group continuity constraint. From these we can also deduce that requests were also rejected due the constraints posed by the dynamic formation of optical topology. NLWA led to a more constrained topology formation than ELWA, as the latter creates new links only if the existing links can not serve a new request.

2) Scenario 2 - Tree topology: As seen in Fig. 4(a), ELWA brings down the rejection percentage for virtual networks with tree topology. While the mean rejection percentage due to NLWA was ≈ 6%, with ELWA it was reduced by more than 70%. The rejections with NLWA was more than 11% when number of input virtual networks reached 150. As with the star topology, here too, we see that the rejections due to wavelength-group continuity constraint is considerably higher due to NLWA than ELWA (refer Fig. 4(b)).

3) Scenario 3 - Random graph topology: We plot the rejection percentage for virtual networks with random graph topology in Fig. 5(a). The average rejection due to ELWA is ≈ 40% lower than NLWA. Fig. 5(b) plots the utilization of the created optical paths. As seen, the utilization is not very high. Though not plotted, we observed that in this scenario the rejections due to wavelength-group continuity was negligible. This essentially means that, even though the established optical network had sufficient bandwidth, it had to reject requests due to the constraints posed by its logical topology. The optical logical topology formed dynamically is too constrained that it had to reject input virtual networks. This is the only contributing factor in this scenario because the input virtual networks here had more complicated (constrained) topologies than star and tree topologies.

VI. CONCLUSIONS

In this paper, we studied the joint problem of bandwidth allocation and placement of VM-clusters in a hybrid optical-electrical datacenter network. We developed two different algorithms that differed in the order in which they explored the solution space (operations). Performance studies showed that, requests can get rejected due to the constrained optical logical topology as well as the wavelength-group continuity constraint. Between the two algorithms developed here, ELWA that exploits existing optical paths for embedding edges of requests, reduced the rejection ratio significantly in comparison to NLWA.
VII. ACKNOWLEDGEMENT

This work was supported by Singapore Ministry of Education Academic Research Fund Tier 2 Grant No. MOE2013- T2-2-135, NUS WBS No. R-263-000-B11-112.

REFERENCES


