Control Plane Innovations to Realize Dynamic Formulation of Multicast Sessions in Inter-DC Software-Defined Elastic Optical Networks

Menglu Zeng[1], Yan Li[2], Wenjian Fang[1], Wei Lu[1], Xiahe Liu[1], Hang Yu[2], Zuqing Zhu[1*]

[1] *University of Science and Technology of China, Hefei, Anhui 230027, China*

[2] *Shenzhen University, Shenzhen, Guangdong 518060, China*

Abstract

It is known that to support the applications such as datacenter backup and migration, multicast should be supported efficiently in inter-datacenter (inter-DC) networks to carry the corresponding point-to-multiple-point communications. Moreover, due to the traffic dynamics in inter-DC networks, we might have to consider the case that the multicast members can join or leave a multicast session dynamically. Therefore, in this work, we try to leverage control plane innovations to realize dynamic formulation of multicast sessions in inter-DC software-defined elastic optical networks (SD-EONs), which are equipped with multicast-incapable bandwidth-variable wavelength selective switches (MI-BV-WSS'). Here, one key issue to address is that the continuous changing of multicast group members can degrade the optimality of a multicast-tree. Hence, we propose to rearrange the multicast-trees adaptively to reduce their spectrum usage. Meanwhile, we try to minimize the frequency of rearrangements to avoid unnecessary operation complexity. Based on these considerations, we propose several multicast-tree rearrangement algorithms for updating multicast sessions dynamically with lightpath reroutings in inter-DC SD-EONs. Both partial and full multicast-tree rearrangements are studied. Simulation results indicate that the proposed algorithms can rearrange the multicast-trees intelligently such

*Corresponding author, email: zqzhu@ieee.org.

that the blocking probability can be reduced effectively with the least lightpath reroutings. Next, based on these theoretical investigations, we consider how to implement the proposed algorithms in the control plane of an inter-DC SD-EON. We extend the OpenFlow (OF) protocol to support the dynamic formulation of multicast sessions and also design the functional models in the control plane elements to realize multicast-tree rearrangements. Experiment results verify the effectiveness of our proposed algorithms and system design.

Keywords: Multicast, elastic optical networks (EONs), software-defined networking (SDN), inter-datacenter network (Inter-DC Network)

---

1. Introduction

Recently, with the rise of inter-datacenter (inter-DC) networks, there have been increasing demands for bandwidth-intensive applications such as datacenter backup and migration, cloud computing, and video streaming. For instance, according to the latest statistics in Cisco's report [1], by 2019, the global Internet traffic will surpass 2.0 ZettaBytes, 65% of which would come from video-related applications. It is known that most of these applications are now leveraging the multi-DC cloud systems to achieve cost-effective and elastic services, and need to transfer large amounts of data among geographically distributed end-systems [2]. Therefore, to support these services in inter-DC networks, multicast should be supported efficiently to carry the corresponding point-to-multiple-point communications [3, 4]. Meanwhile, with the tremendous bandwidth in optical fibers, optical networks provide a viable infrastructure for delivering high-throughput traffics. More importantly, the newly-developed elastic optical networks (EONs) can make the resource management in the optical layer more adaptive and application-aware than the traditional fixed-grid wavelength division multiplexing (WDM) networks [5, 6]. Specifically, with bandwidth-variable transponders (BV-Ts) and switches (BV-WSS'), EONs provision bandwidth by grooming the capacities of several narrow-band frequency slots (FS'). Hence, EONs can achieve agile spectrum management and provision requests with various bandwidth requirements more efficiently. These advantages match with the requirements of inter-DC networks well, and thus EON has been considered as a promising physical infrastructure to carry future inter-DC networks [2, 7, 8].

The problem of multicast in EONs was first studied in [9], where the authors proposed two heuristics to solve the routing and spectrum assignment (RSA) for all-optical multicast. Later on, we formulated integer linear programming (ILP) models and developed heuristics with better performance in [10, 11]. The problem of provisioning multiple static multicast sessions in WDM network was studied in [12]. In this work, we also try to accommodate multiple multicast sessions in an EON but in a dynamic manner. In [13], we considered the scenario where one multicast session can be provisioned with multiple sub-trees to adapt to the physical-layer impairments. However, since this work focuses on the dynamic formulation of multicast sessions in EONs, we still assume that each multicast session only uses one multicast-tree. Note that, with minor modifications, the scheme proposed in this work can be extended to support the scenario discussed in [13]. Walkowiak *et al.* [14] considered a joint optimization of multicast and unicast flows in EONs, with the focus only on offline optimization. Nevertheless, the studies mentioned above assumed that all the BV-WSS' in EONs are multicast-capable (MC) (*i.e.*, supporting light-splitting). Note that, MC optical switches usually have complicated structures [15] and can be prohibitively expensive. Therefore, it would not be economical to build an optical network solely with them. To address this issue, people have investigated how to realize multicast in optical networks built with multicast-incapable (MI) switches [16, 17]. Specifically, they leveraged multiple unicast lightpaths to set up the logic light-tree for a multicast request (*i.e.*, overlay multicast).

Note that, previous studies on multicast in EONs only considered the static formulation of multicast sessions, *i.e.*, the multicast group members stay unchanged for the whole life-time of a multicast session. However, in order to support the emerging applications such as grid computing, dynamic data backup /synchronization and high-definition teleconferencing, we might have to consider the case that the multicast members can join or leave a multicast session dynamically. To the best of our knowledge, dynamic formulation of multicast sessions in EONs has not been studied so far.

In order to achieve dynamic formulation of multicast sessions efficiently in inter-DC EONs, we need to know the global resource utilization and the information of all the in-service requests in the network. This brings new challenges to the network control and management (NC&M). Fortunately, it is known that software-defined networking (SDN) with OpenFlow (OF) [18] can improve optical networks' programmability by decoupling their control and data planes and leveraging centralized NC&M [19, 20, 21]. The combination of SDN and EON leads to software-defined EONs (SD-EONs) and can further improve the flexibility of EONs [19, 20, 21, 22, 23, 24]. Note that, SD-EONs can be realized by leveraging OpenFlow switches (*e.g.*, Pica 8 [25]) together with flexible-grid reconfigurable add/drop multiplexers (ROADMs) [26]. More promisingly, leading vendors of optical network equipment such as Alcatel-Lucent have already utilized the concept of SD-EON to offer agile optical networking solutions, which make SDN architecture work with flexible-grid ROADMs [27]. Therefore, SD-EONs provide new opportunities on provisioning the services efficiently for dynamic multicast sessions. In line with this, we can leverage the centralized NC&M in SD-EONs to utilize the network resources (*e.g.*, optical spectra, optical-to-electrical-to-optical (O/E/O) converters, *etc*) cost-effectively. For instance, under the assumptions that all the BV-WSS' in SD-EONs are MC and multicast sessions are formulated statically, the authors of [4] have already studied the control plane operations for setting up multicast sessions, and we have also demonstrated the fragmentation-aware service provisioning for advance reservation multicast [28].

In this paper, we try to leverage control plane innovations to realize dynamic formulation of multicast sessions in inter-DC SD-EONs that are equipped with MI BV-WSS'. Here, one key issue to consider is that the continuous changing of multicast group members causes the degradation of the multicast-tree. Therefore, we propose to rearrange the multicast-trees adaptively to reduce their spectrum usages. Meanwhile, we try to minimize the frequency of rearrangements to avoid unnecessary operation complexity. Based on these considerations, we propose several multicast-tree rearrangement algorithms for updating multicast sessions dynamically with lightpath reroutings in inter-DC SD-EONs. Both partial and full multicast-tree rearrangements are addressed. Simulation results indicate that the proposed algorithms can rearrange the multicast-trees intelligently such that the blocking probability can be reduced effectively with the least lightpath reroutings. Next, based on these theoretical investigations, we consider how to implement the proposed algorithms in the control plane of an inter-DC SD-EON. We extend the OF protocol to support the dynamic formulation of multicast sessions and also design the functional models in the control plane elements to realize multicast-tree rearrangements. Experiment results verify the effectiveness of our proposed algorithms and system design.

The rest of the paper is organized as follows. We formulate the problem of dynamic formulation of multicast sessions in inter-DC EONs in Section 2. Section 3 discusses the proposed multicast-tree rearrangement algorithms. The numerical simulation results are presented in Section 4 for performance evaluation. We describe the system design for realizing dynamic formulation of multicast sessions in an inter-DC SD-EON in Section 5 and show the experimental results in Section 6. Finally, Section 7 summarizes the paper.

2. Problem Formulation

2.1. Network Model

We model the topology of the inter-DC EON as $G(V, E)$, where V is the set of DCs and E is the fiber link set. Since we consider an MI-EON, all the DCs in V are equipped with MI switches, *i.e.*, none of them is capable of performing light-splitting. There are F FS' on each link in E. Note that, we assume that the SD-EON is equipped with the sliceable BV-Ts discussed in [29]. Specifically, as long as there are enough spectrum resources on a fiber, the BV-Ts connecting to it can always be sliced and tuned to support the requested optical transmission. Hence, this model merges the constraints on spectra and BV-Ts into one, and we do not need to consider the BV-T constraint explicitly. Each FS has a bandwidth of 12.5 GHz and can provide a capacity of $C_{FS} = 12.5$ Gb/s if its modulation format is BPSK [5]. In this work, we consider distance-adaptive modulation selection [10], and assume that there are four feasible modulation formats, *i.e.*, BPSK, QPSK, 8QAM and 16QAM, in the EON. Here, we consider the scenario where there is a set of multicast sessions to be served. For each multicast session $MR(s, D, C)$, where s is the source node and D is the set of destinations, we use the OL-M-SFMOR scheme in [16] to determine the routing, modulation and spectrum assignment (RMSA) for it and try to satisfy the capacity requirement C of each destination in D. Note that as we consider dynamic multicast sessions, D can change over time and the RMSA needs to be adjusted accordingly. Here, we define the modulation-level as $m = 1$, 2, 3 and 4 for BPSK, QPSK, 8QAM and 16QAM, respectively. Hence, the number of spectrally-contiguous FS' that need to be assigned for MR is [10]

$$n = \lceil \frac{C}{m \cdot C_{FS}} \rceil + N_{gb}, \tag{1}$$

where N_{gb} is the guard-band FS'. Note that similar to our work in [16], we still assume that the spectrum assignment and modulation selection of a multicast-branch can be changed by a relay node on the multicast-tree. Meanwhile, the maximum transmission reaches of BPSK, QPSK, 8QAM and 16QAM signals are set as 5000 km, 2500 km, 1250 km and 625 km, respectively, with the same assumptions as those in [10].

2.2. Design Considerations

With a multicast session $MR(s, D, C)$, we define its member nodes as the source s and all the destinations in D. In order to cover all the member nodes, we need to formulate a logic tree $\mathcal{T}$ and set up a set of unicast lightpaths $\mathcal{P}$ to support it. According to OL-M-SFMOR, each lightpath $p \in \mathcal{P}$ can only start and end at the member nodes of MR. Hence, redundant BV-Ts can be avoided. Note that the signal is transmitted all-optically end-to-end on each lightpath $p \in \mathcal{P}$ and the RMSAs of different lightpaths are independent.

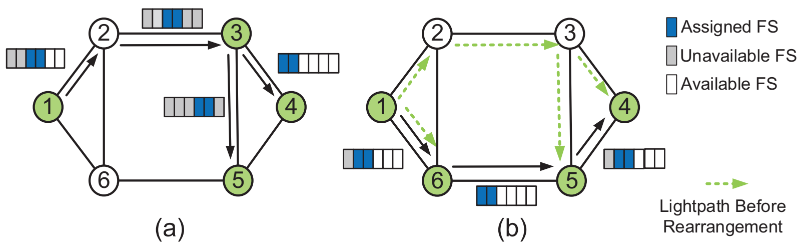


Figure 1: Example on the multicast-tree of a dynamic multicast session losing the optimal structure over time, (a) original multicast-tree, (b) the optimal multicast-tree for the new multicast group.

When determining the RMSA of a lightpath $p \in \mathcal{P}$, we use distance-adaptive modulation selection and follow the spectrum contiguous, non-overlapping and continuity constraints.

As D can change on-the-fly, we have to address the issue that its multicast-tree loses the optimal structure due to the operation principle of OL-M-SFMOR. For instance, Fig. 1 shows an illustrative example. We first have a session with $s = 1$ and $D = \{3, 4, 5\}$, and the session is set up as shown in Fig. 1(a). *DC* 3 sends the traffic flow to the local port and also forwards it to *DCs* 4 and 5 through all-optical paths, respectively. Then, *DC* 6 tries to join the multicast session, while the multicast service to *DC* 3 is expiring. Without the tree rearrangement, we cannot remove *DC* 3 from the multicast-tree because it still has downstream nodes. Consequently, the spectrum assignment of the multicast-tree represented by the dashed lines in Fig. 1(b) is sub-optimal and certain spectra are wasted. On the other hand, we can rearrange the multicast-tree to the one with the solid lines in Fig. 1(b), and apparently, after we updating the dynamic multicast sessions adaptively, the efficiency of the spectrum usage gets improved significantly.

Algorithm 1: Dynamic Formulation of Multicast Sessions in MI-EONs

```
while MI-EON is operational do
    if multicast session MR(s, D, C) first appears then
        use OL-M-SFMOR to build a logic tree 𝒯 that covers {s, D};
        set up the lightpaths 𝒫 to support 𝒯;
    else
        if a new member v to join MR then
            for each existing member u ∈ {s, D} do
                calculate K shortest paths for u → v;
                get RMSA for each path with FMA;
            end
            select path that has the least fragmentation cost to connect
            v to session MR;
            update 𝒯 to include the new branch;
        end
        if a member v to leave MR then
            if v has downstream member(s) then
                mark v as a non-member in 𝒯;
            else
                remove v and the branch to its upstream member from 𝒯;
            end
        end
        if a service provisioning period is about to end then
            select multicast-trees for rearrangements;
            rearrange the selected multicast-trees;
        end
    end
end
```

3. Dynamic Formulation of Multicast Sessions

3.1. Overall Procedure

Algorithm 1 shows the overall procedure for dynamic formulation of multicast sessions in MI-EONs. As shown in *Lines* 2-4, when a multicast session $MR(s, D, C)$ first appears, we leverage OL-M-SFMOR to set up the unicast lightpaths $\mathcal{P}$ that formulate a logic tree $\mathcal{T}$ to cover all the member nodes. Meanwhile, the RMSA of each lightpath $p \in \mathcal{P}$ is also determined. *Lines* 6-13 explain the operations for adding a new member in session MR. Specifically, when a new member v tries to join the session, we leverage the reverse-anycast scheme to establish the transmission for it, *i.e.*, we can use any existing member in $\{s, D\}$ to relay the multicast transmission to v with a new lightpath. For each existing member in $\{s, D\}$, we calculate K shortest paths to v and determine the corresponding RMSAs with the fragmentation and misalignment-aware spectrum assignment (FMA) in [30]. Then, we select the path that has the least fragmentation cost to connect v to session MR.

Then, if a member $v \in D$ needs to leave MR, *Line* 15 checks whether v has any downstream member(s). If yes, *Line* 16 marks v as a non-member relay node (NM-RN) in $\mathcal{T}$. Otherwise, *Line* 18 removes v and the branch to its upstream member from $\mathcal{T}$ and releases the associated spectrum resources. For the multicast-tree rearrangement, we need to balance the tradeoff between structure optimality and rearrangement frequency. *Lines* 21-24 show the related operations. Basically, the MI-EON performs the multicast-tree rearrangement when each service provisioning period is about to end. We first choose the multicast sessions to rearrange with a tree selection strategy and then rearrange the selected sessions. We propose two tree selection strategies and consider both full and partial rearrangements of selected multicast-trees.

3.2. Tree Selection Strategies

When the EON is operational, there will be multiple active multicast sessions in it. In order to reduce operation complexity, we develop two kinds of tree selection strategies that can intelligently select the most "critical" sessions (*i.e.*, the multicast sessions that are off the optimal RMSAs the most) to rearrange.

3.2.1. $\mathcal{D}$-value based Tree Selection (DTS)

Definition 1. *We define the $\mathcal{D}$-**value** of a multicast-tree $\mathcal{T}$ as the hop-count of its longest source-destination branch,* i.e.*, the multicast-tree's depth. Specifically, we have*

$$\mathcal{D}(\mathcal{T}) = \max(hops(s \rightarrow d),\ \forall d \in D), \tag{2}$$

where $hops(\cdot)$ returns the hop-count of a path.

With the example in Fig. 1, we find that for a multicast session $MR(s, D, C)$, if the $\mathcal{D}$-value of its multicast-tree $\mathcal{T}$ is abnormally large, the possibility that $\mathcal{T}$ is sub-optimal is relatively high. Then, when we need to select multicast sessions to rearrange, we calculate $\mathcal{D}$-values of all the multicast-trees and get the average value $\overline{\mathcal{D}}$. If the $\mathcal{D}$-value of a multicast-tree is larger than $\overline{\mathcal{D}}$, the $\mathcal{D}$-value based tree selection strategy (DTS) chooses to rearrange it. Note that, the $\mathcal{D}$-value of a multicast-tree can have a large absolute value for a few reasons, *e.g.*, the network has a relatively large topology and/or the locations of the source and destinations are dispersed in the network. This is actually the reason why we do not design DTS to use an arbitrary and fixed threshold on $\mathcal{D}$-values. Nevertheless, by comparing $\mathcal{D}$-values with their average value $\overline{\mathcal{D}}$, the effects of these topology-related disturbances can be minimized. Then, with this design, we can select the real sub-optimal multicast-trees, *e.g.*, those contain many NM-RNs, such as DC 3 in Fig. 1(b), and/or those use non-optimal relay nodes on some of their branches. The time complexity of DTS to choose the most "critical" multicast-trees is $O(|\mathcal{T}|)$, where $|\mathcal{T}|$ is the number of in-service multicast sessions in the network.

3.2.2. $\mathcal{Q}$-value based Tree Selection (QTS)

DTS only considers the longest branch of a multicast-tree, but does not address its overall structure or the spectrum assignments on the links. Hence, we introduce the definition below.

Definition 2. *We define the* $\mathcal{Q}$***-value*** *of a multicast-tree* $\mathcal{T}$ *for a multicast session* $MR(s, D, C)$ *as*

$$\mathcal{Q}(\mathcal{T}) = \frac{hops(\mathcal{T}^*) \cdot hidx(\mathcal{T}^*)}{hops(\mathcal{T}) \cdot hidx(\mathcal{T})}, \tag{3}$$

where $\mathcal{T}^*$ *is the approximate optimal multicast-tree to carry the multicast session based on the current network status,* $hops(\cdot)$ *returns the total hop-count of a multicast-tree, and* $hidx(\cdot)$ *supplies the highest index of the used FS' of a multicast-tree. Here,* $\mathcal{T}^*$ *can be obtained by using a clean-slate approach,* i.e.*, calculating the multicast-tree with OL-M-SFMOR in [16] to cover* $\{s, D\}$ *as* $MR(s, D, C)$ *just comes in as a new session.*

Obviously, if the RMSA of a multicast-tree $\mathcal{T}$ approaches to that of $\mathcal{T}^*$, its $\mathcal{Q}$-value increases. Hence, the $\mathcal{Q}$-value based tree selection (QTS) takes a preset lower-bound on $\mathcal{Q}$-value as $\mathcal{Q}_{lb}$, and will select a multicast-tree for rearrangement if its $\mathcal{Q}$-value is smaller than $\mathcal{Q}_{lb}$. The complexity for calculate $\mathcal{T}^*$ using OL-M-SFMOR in [16] is $O(|V|^3)$, where $|V|$ is the number of the nodes in the topology. Then, the time complexity of QTS to choose the most "critical" multicast-trees is $O(|\mathcal{T}| \cdot |V|^3)$.

3.3. Tree Rearrangement with Lightpath Rerouting

For tree rearrangements, we have to minimize traffic disruptions and this can be done by leveraging the "make-before-break" scenario [30], *i.e.*, install new paths before tearing down old paths. Here, we consider two scenarios, *i.e.*, full and partial rearrangements of selected multicast-trees.

3.3.1. Full Rearrangement

Full rearrangement is straightforward. Basically, when a multicast session is selected, we recalculate the approximate optimal multicast-tree with OL-M-SFMOR and re-establish the session with it in the clean-slate manner.

3.3.2. Partial Rearrangement

Partial rearrangement tries to further balance the tradeoff between structure optimality and operation complexity by only modifying certain parts of

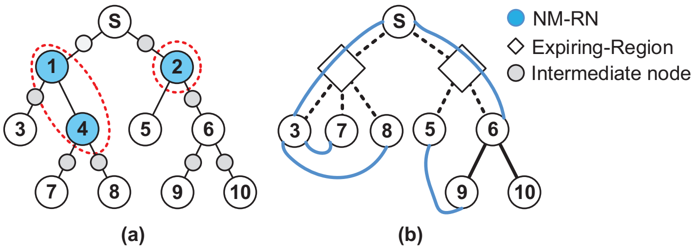


Figure 2: Example for the node-compression approach, (a) finding expiring-regions in $\mathcal{T}$, and (b) removing expiring-regions and reconnecting downstream members.

a multicast-tree. As shown in *Line* 16 of *Algorithm* 1, when a member node has left the multicast session, it cannot be removed immediately if it still has downstream member(s). Instead, we just mark it as an NM-RN. These NM-RNs cost redundant spectrum usages on links and consume unnecessary O/E/O converters, and thus should be minimized in the tree rearrangements. In order to achieve this, we introduce a node-compression approach. Specifically, with a multicast-tree $\mathcal{T}$, we first abstract all the transparent lightpaths on it as directed virtual links and obtain a virtual tree $\mathcal{T}'$. Then, if multiple NM-RNs are adjacent on $\mathcal{T}'$, we compress them as an expiring-region. Fig. 2 shows an illustrative example of the node-compression. The partial rearrangements first reconnects all the downstream members of expiring regions in $\mathcal{T}$, and then removes the expiring regions.

After handling the expiring-regions, we try to address the sub-optimal lightpaths in $\mathcal{T}$. Specifically, for a destination $d \in D$, we define its cost as

$$Cost(d) = hops(s \to d) \cdot hidx(v \to d), \tag{4}$$

where $s \to d$ refers to the lightpath from s to d, $hops(\cdot)$ returns the hop-count of a lightpath, v is the upstream relay node of d, $v \to d$ is the transparent lightpath from v to d, and $hidx(\cdot)$ supplies the highest index of the used FS' of a transparent lightpath. The cost jointly considers the hop-count of the source-destination branch of d and the spectrum assignment for d. Then, we define the

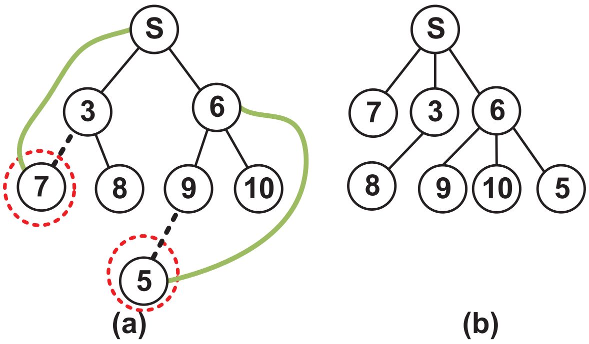


Figure 3: Lightpath rerouting for costly destinations, (a) finding costly destinations and reconnecting the destinations, and (b) new multicast-tree.

cost threshold as

$$Cost_{th} = \frac{hidx(\mathcal{T})}{|D|} \cdot \sum_{u \in D} hops(s \to u), \tag{5}$$

where $|D|$ represents the number of destinations. $Cost_{th}$ presents the average cost of destinations in $\mathcal{T}$, where $\frac{\sum\limits_{u \in D} hops(s \to u)}{|D|}$ represents the average hop-count from source to destinations. If the cost of a destination is higher than $Cost_{th}$, we will reroute its transparent lightpath with the procedure in *Algorithm* 2. Specifically, we still use the reverse-anycast scheme as shown in *Lines* 6-12. Fig. 3 gives an example on the lightpath rerouting for costly destinations. Finally, *Algorithm* 3 shows the overall procedure of partial rearrangement for a multicast session $MR(s, D, C)$.

4. Numerical Simulations

In this section, we use simulations to evaluate the proposed algorithms for dynamic formulation of multicast sessions in inter-DC EONs. We consider two physical topologies, *i.e.*, the 14-node NSFNET topology and the 28-node US Backbone topology [10]. The EON is deployed in the C-band and hence, each fiber link can accommodate $F = 358$ FS' (*i.e.*, each FS' bandwidth is 12.5 GHz),

Algorithm 2: Lightpath Rerouting for Costly Destinations

calculate $Cost_{th}$ with Eq. (5);
for *each destination* $d \in D$ **do**
  calculate $Cost(d)$ with Eq. (4);
  if $Cost(d) > Cost_{th}$ **then**
    obtain the upstream relay node v of d;
    for *each existing member* $u \in \{s, D\} \setminus \{d\}$ **do**
      calculate K shortest paths for $u \rightarrow d$;
      get RMSA for each path with FMA;
    end
    select path that has the least fragmentation cost to reconnect d to session MR;
    tear down transparent lightpath $v \rightarrow d$;
    update $\mathcal{T}$ to include the new branch;
  end
end

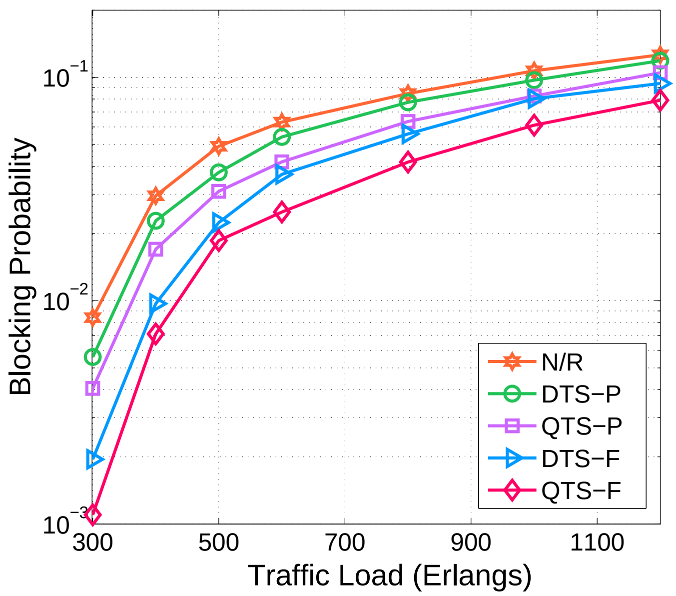
Algorithm 3: Partial Multicast-Tree Rearrangement

perform node-compression to find the expiring-regions;
reconnect downstream members of expiring-regions in $\mathcal{T}$;
remove the expiring-regions;
apply *Algorithm* 2 to reroute lightpaths for costly destinations;

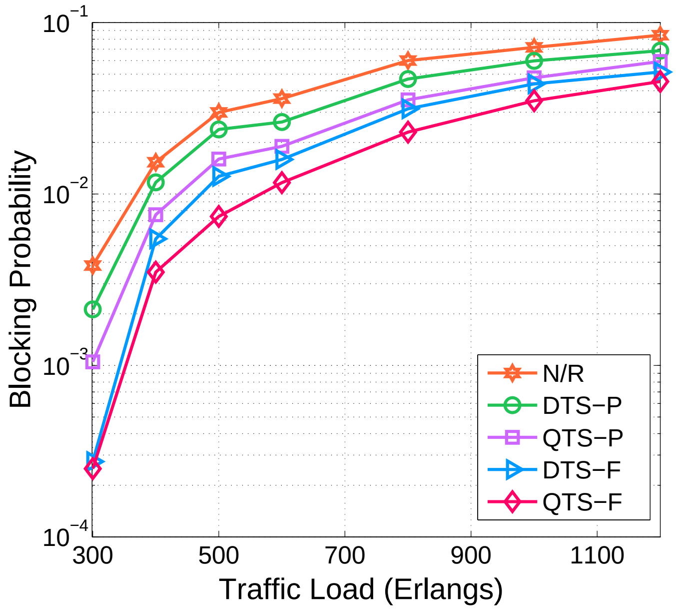
which correspond to 4.475 THz bandwidth [10]. The multicast requests are generated using the Poisson traffic model, *i.e.*, requests come in according to the Poisson process with an average arrival rate of λ and their holding time follows a negative exponential distribution with an average of $\frac{1}{\mu}$. Hence, we can quantify the traffic load with $\frac{\lambda}{\mu}$ in Erlangs. Each request carries the information on source and destination node(s), the multicast session to join, and required capacity in Gb/s. The capacity requirements of the multicast sessions are uniformly distributed within $[50, 400]$ Gb/s.

We first compare the performance of the algorithms for dynamic formulation of multicast sessions with different tree rearrangement schemes. Fig. 4 shows the results on blocking probability. In the plots, the curves for "N/R" imply the scheme that we do not apply tree rearrangement, *i.e.*, *Lines* 21-24 in *Algorithm* 1 are not executed. For the algorithms that use DTS as the tree selection strategy, we consider both full rearrangement (DTS-F) and partial rearrangement (DTS-P). Note that here the "full" rearrangement refers to the operation on a selected multicast-tree but not all the multicast-trees in the network, *i.e.*, when a multicast-tree is selected, we recalculate the approximate optimal tree structure and re-establish the session with it. Similarly, QTS-F and QTS-P denote the algorithms that use QTS with full and partial rearrangements, respectively. Specifically, for QTS-F, we have $\mathcal{Q}_{lb} = 0.7$, *i.e.*, a multicast-tree will be selected for rearrangement if its $\mathcal{Q}$-value is less than 0.7, and QTS-P also uses $\mathcal{Q}_{lb} = 0.7$. As expected, tree rearrangement can effectively improve the blocking performance, as the algorithms with it provide lower blocking probabilities than the N/R scheme in both topologies. In general, compared with partial rearrangement, full rearrangement can improve the blocking performance further. We also observe that when the rearrangement scheme is the same, the QTS-based algorithm outperforms the DTS-based one in terms of blocking probability. This is because QTS considers both the overall tree structure and spectrum utilization in the tree selection strategy.

Table 1 summarizes the results on the average lightpath reroutings per service provisioning period for the dynamic formulation of multicast sessions. It



(a) NSFNET topology.



(b) US Backbone topology.

Figure 4: Results on blocking probability.

Table 1: Average lightpath reroutings per service provisioning period

Traffic Load (Erlangs)	NSFNET				US Backbone			
	DTS-F	DTS-P	QTS-F	QTS-P	DTS-F	DTS-P	QTS-F	QTS-P
400	71	11	15	6	55	12	15	6
600	94	16	21	8	90	20	22	11
800	118	20	25	11	121	26	28	15
1000	131	24	30	13	143	31	32	18

is exciting to notice that the QTS-based algorithm also invokes significantly less number of lightpath reroutings than the DTS-based one, when the rearrangement scheme is the same. This observation suggests that the QTS-based algorithms can provide lower blocking probability with a smaller number of lightpath reroutings (*i.e.*, $\frac{1}{4} \sim \frac{1}{3}$ of those performed in DTS-based algorithms). This verifies that QTS can intelligently select the most "critical" multicast-trees to rearrange. When comparing the results from QTS-F and QTS-P, we can see that QTS-P can further reduce the number of lightpath reroutings significantly. Basically, for the simulations in both topologies, QTS-P only uses 37% $\sim$ 56% number of lightpath reroutings, when being compared with QTS-F. Note that, we use the same $\mathcal{Q}_{lb}$ in QTS-P, which means that in the simulations, a multicast-tree has the same probability to be selected for rearrangement.

We then run more simulations to investigate the performance of QTS-P further. Basically, we notice that for QTS-P, there is a tradeoff between the blocking performance and operation complexity, which can be adjusted by varying $\mathcal{Q}_{lb}$. Here, we name the algorithms as QTS-P-$\mathcal{Q}_{lb}$. For example, QTS-P-0.9 means that the QTS-P algorithm adopts $\mathcal{Q}_{lb} = 0.9$. Note that if we have $\mathcal{Q}_{lb} = 0$, then QTS-P becomes "N/R". Here, due to the page limit, we only show the results from the simulations that use the US Backbone topology, but we do confirm that the results from the NSFNET topology follow the similar trends. Fig. 5 illustrates the effect of $\mathcal{Q}_{lb}$ on the blocking performance of

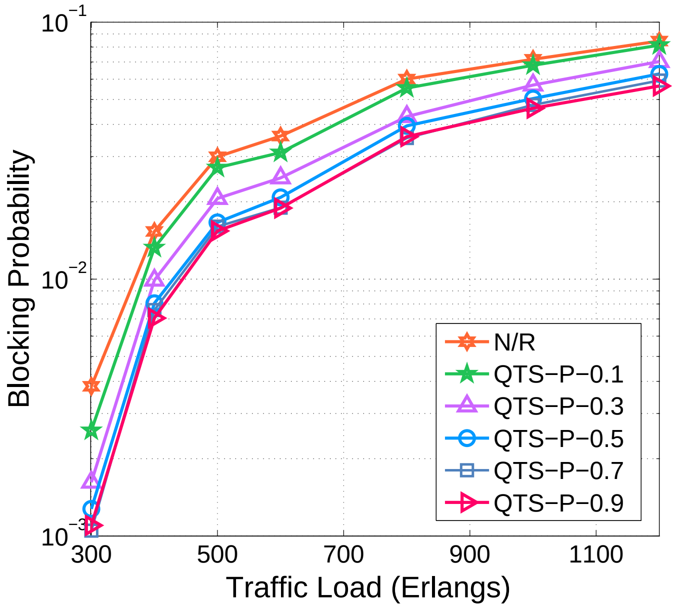


Figure 5: Results on blocking probability of QTS-P (US Backbone).

QTS-P. Apparently, a larger $\mathcal{Q}_{lb}$ generally leads to lower blocking probability in the network. This can be explained as follows. When $\mathcal{Q}_{lb}$ is larger, the restriction on the optimality of multicast-trees becomes stricter according to Eq. (3). Consequently, QTS-P using high $\mathcal{Q}_{lb}$ considers more multicast-trees as sub-optimal and selects them to rearrange. Therefore, QTS-P invokes more lightpath reroutings and provides better blocking performance. However, the operation complexity also increases when the network needs to accomplish more lightpath reroutings. Moreover, it is interesting to notice that the blocking performance of QTS-P does not increase evenly with $\mathcal{Q}_{lb}$. Specifically, we observe that the blocking probability only reduces slightly if we increase $\mathcal{Q}_{lb}$ over 0.5. Therefore, we can draw the conclusion that when $\mathcal{Q}_{lb} > 0.5$, the optimization margin that QTS-P can obtain by increasing $\mathcal{Q}_{lb}$ is very limited.

Table 2 summarizes the results on average lightpath reroutings per service provisioning period for QTS-P using different $\mathcal{Q}_{th}$. As expected, a larger $\mathcal{Q}_{lb}$ leads to more lightpath reroutings. We can also see that the number of lightpath

Table 2: Average lightpath reroutings per service provisioning period (US Backbone)

Traffic Load (Erlangs)	QTS-P				
	$\mathcal{Q}_{lb} = 0.1$	$\mathcal{Q}_{lb} = 0.3$	$\mathcal{Q}_{lb} = 0.5$	$\mathcal{Q}_{lb} = 0.7$	$\mathcal{Q}_{lb} = 0.9$
400	0.5	2.3	4.2	6.4	9.4
600	0.8	3.5	6.8	10.8	16.1
800	0.8	4.3	9.0	14.6	22.0
1000	0.9	4.9	10.7	18.0	27.7

reroutings keeps increasing when $\mathcal{Q}_{lb} \geq 0.5$. However, as Fig. 5 shows, the blocking probability reduction becomes very limited when $\mathcal{Q}_{lb} \geq 0.5$. Hence, for the joint consideration of blocking probability and operation complexity, we think that $\mathcal{Q}_{lb} = 0.5$ is the proper value to be used in QTS-P.

Fig. 6 shows the results on the average number of O/E/O converters per multicast destination. It can be seen that compared with N/R, QTS-P can achieve around 29% reduction on the O/E/O converter usage on average. We also observe that DTS-based strategies use less O/E/O converters than QTS-based ones, which is because DTS-based strategies focus on the long source-destination branches of multicast-trees which may contains NM-RNs.

5. Control Plane System Design

In this section, we describe the system design for realizing dynamic formulation of multicast sessions in an inter-DC SD-EON.

5.1. Network Architecture

Fig. 7 shows the architecture of inter-DC SD-EON. The network consists of two separated planes, *i.e.*, the data and control planes. The data plane consists of several geographically distributed DCs, each of which attaches to a multicast-incapable BV-WSS (MI-BV-WSS), which can be used to set up multicast sessions for inter-DC data transmissions. The control plane consists

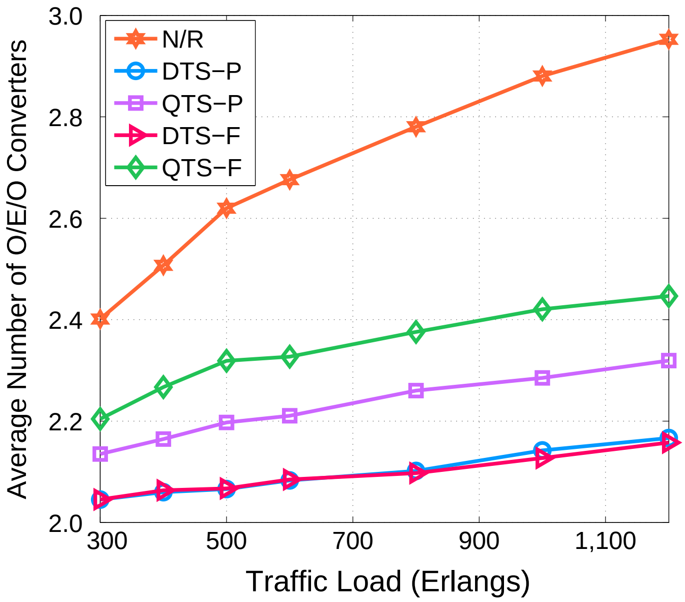


Figure 6: Average number of O/E/O converters per multicast destination (US Backbone).

of several OpenFlow agents (OF-AGs) and an OpenFlow controller (OF-C) [19, 20]. Each OF-AG is attached to an MI-BV-WSS to manage the network element according to the instructions from OF-C.

5.2. System Functional Design

The functional modules inside the OF-AG and OF-C are shown in the Fig. 8. In OF-C, the **network abstraction module (NAM)** communicates with the OF-AGs to abstract the data plane information (*e.g.*, network topology) and sends the information to the **traffic engineering database (TED)**. TED stores the information on the spectrum utilization and lightpaths and assists the **resources computing and allocation module (RCAM)** to provision new multicast sessions or reconfigure existing ones. The tree rearrangement algorithm runs in the **multicast session reconfiguration module (MRM)**. MRM gets the information of in-service sessions from the **multicast session management module (MMM)**. The provisioning and reconfiguration strategies

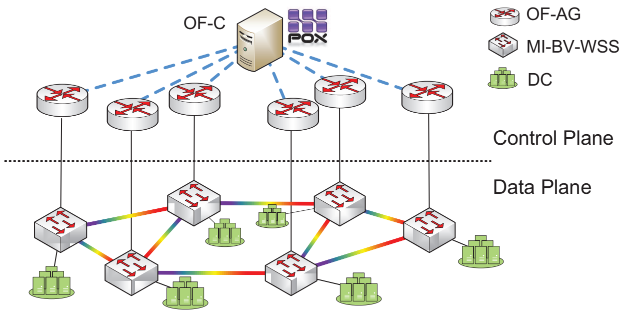


Figure 7: Network architecture of an inter-DC SD-EON.

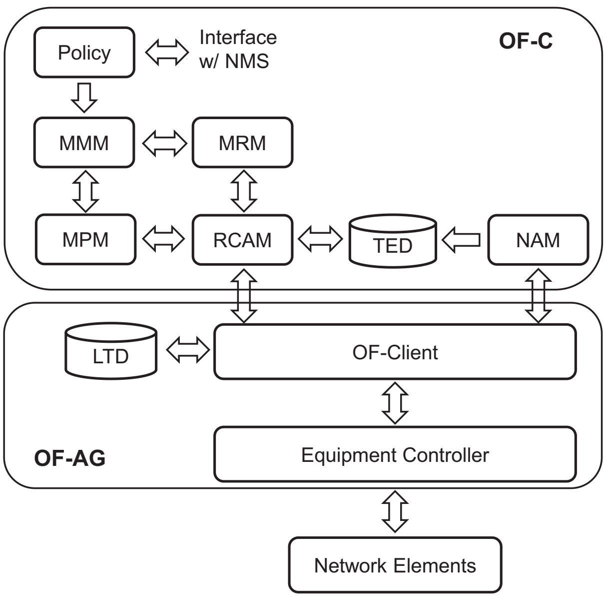


Figure 8: Functional modules in OF-C and OF-AG to enable dynamic formulation of multicast sessions.

are defined by **network management system (NMS)** and configured with the **policy module (Policy)**. According to the outputs of **multicast session provisioning module (MPM)** or MRM, RCAM computes the resource allocations, and then it encodes the provisioning or reconfiguration schemes in OF-messages and sends them to related OF-AGs. The OF client in an OF-AG communicates with OF-C using a multicast-enabled extended OF protocol, and the flow entries that are used to configure the data plane network elements are stored in the **local traffic database (LTD)** [21].

5.3. OF Extension for Dynamic Formulation of Multicast Sessions

According to the working principle of OF, the inter-DC SD-EON identifies each lightpath as an optical flow with the flow-entry that consists of matching fields, actions and related counters [18]. We implement a control plane system for the inter-DC SD-EON based on OF v1.0 [18] since it is a stable version and widely supported by various OF systems. We propose an extension of the matching fields to support dynamic formulation of multicast sessions. Specifically, to identify a multicast session, the matching fields are **Multicast_Group_Address**, **Starting_Frequency** and **Number_of_Frequency _Slots**. Note that, we use multicast group addresses to identify a set of destinations in the network topology, and the mapping between them is pre-determined. By doing so, we do not need to encode a long list of destination addresses in OF messages and achieve good scalability. Specifically, in $G(V, E)$, we assign a unique multicast group address to each combination of multiple nodes, and thus we need to assign $2^{|V|} - |V| - 1$ multicast group addresses in total. We also add a "Re-Flag" field in the related OF messages to indicate whether a message is for provisioning a new session or reconfiguring an existing one. Meanwhile, to let an OF-AG know whether an OF message is for installing a new connection or tearing down an existing one, we include a "Command" field. For the actions, we create **SET_MULTICAST_GROUP_ADDRESS** as a new action type, which is used to set new multicast group address when a multicast session's destinations have been changed.

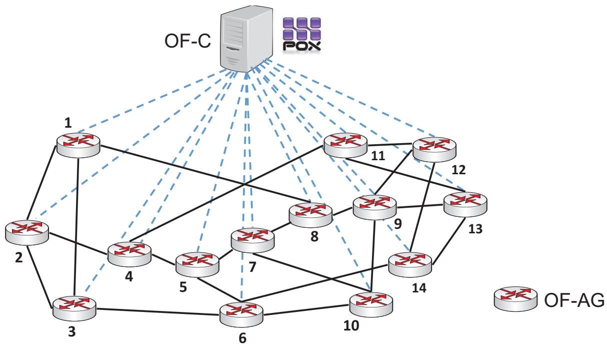


Figure 9: Experimental setup with the NSFNET topology.

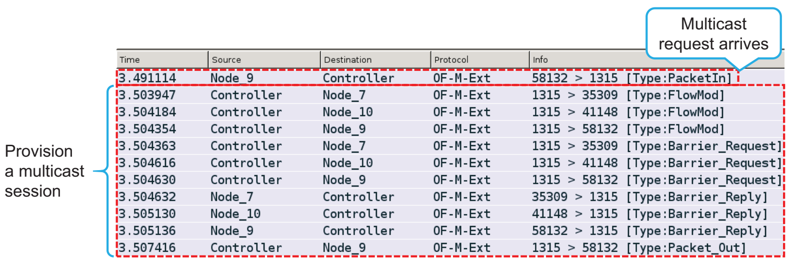
6. Experimental Demonstrations

We implement the aforementioned design in a control plane testbed for an inter-DC SD-EON, which is built with high performance servers (ThinkServer RD530). We have 14 OF-AGs that each runs on an independent server, and they are connected according to the scheme in Fig. 9 to mimic the NSFNET topology. Each OF-AG is programmed based on Open-vSwitch [31] running on Linux. The OF-C is implemented with the POX platform [32] and runs on another independent server that is directly connected to all the OF-AGs. Similar to our previous work in [20, 21], we only focus on the control plane implementation for the dynamic provisioning and reconfiguration of multicast sessions, and the network elements in the data plane (*e.g.*, MI-BV-WSS') are software-emulated.

We first show the procedure for provisioning a new multicast session dynamically in the testbed. Fig. 10 shows the *Packet_In* message for setting up the new multicast session, which shows that the source is *DC* 9, the multicast group address is 7, *i.e.*, corresponding to the destination set *DCs* {7,10}. Fig. 11 shows the Wireshark capture for the OF messages used to provision the multicast session. To establish the logic multicast-tree for the session, OF-C configures two lightpaths, as $9 \rightarrow 10$ and $10 \rightarrow 7$. Figs. 12-14 show the *Flow_Mod* message received by different nodes on the logic multicast-tree. Specifically, the OF-AG

```
▿ Header
    version: 1
    Type: PacketIn (10)
    length: 108
    xid: 0
  Buffer_id: 688
  Total_len: 90
  In_port: 65534
  Reason: No_match (0)
▿ Match
    Source: DC_9 (9)
    Type: Multicast (1)
    Tree_Id: 8
    Capacity_Requirement_GHz: 25
    Multicast_Group_Address: 7
    Expiring_Time: 2
```

Figure 10: *Packet_In* message for setting up a multicast session.



Time	Source	Destination	Protocol	Info
3.491114	Node_9	Controller	OF-M-Ext	58132 > 1315 [Type:PacketIn]
3.503947	Controller	Node_7	OF-M-Ext	1315 > 35309 [Type:FlowMod]
3.504184	Controller	Node_10	OF-M-Ext	1315 > 41148 [Type:FlowMod]
3.504354	Controller	Node_9	OF-M-Ext	1315 > 58132 [Type:FlowMod]
3.504363	Controller	Node_7	OF-M-Ext	1315 > 35309 [Type:Barrier_Request]
3.504616	Controller	Node_10	OF-M-Ext	1315 > 41148 [Type:Barrier_Request]
3.504630	Controller	Node_9	OF-M-Ext	1315 > 58132 [Type:Barrier_Request]
3.504632	Node_7	Controller	OF-M-Ext	35309 > 1315 [Type:Barrier_Reply]
3.505130	Node_10	Controller	OF-M-Ext	41148 > 1315 [Type:Barrier_Reply]
3.505136	Node_9	Controller	OF-M-Ext	58132 > 1315 [Type:Barrier_Reply]
3.507416	Controller	Node_9	OF-M-Ext	1315 > 58132 [Type:Packet_Out]

Figure 11: Wireshark capture for the OF messages involved in provisioning a session.

```
▿ Header
    version: 1
    Type: FlowMod (14)
    length: 112
    xid: 75
▿ Match
    Inport: 65534
    Match_Type
    Multicast_Group_Address: 7
    Starting_Frequency: 0
    Number_of_Frequency_Slots: 0
    Re_Flag: Dynamic_Multicast_Service (0)
    Command: Set_Up_New_Connection (0)
    IdleTimeout: 2
    HardTimeout: 0
    Priority: 32768
    BufferId: 4294967295
    OutPort: Any (65535)
    Flags: 0
▿ Actions
    action_type: SET_STARTING_FS (13)
    Starting_FS: 0
    action_type: SET_NUMBER_OF_FS (14)
    Number_of_FS: 1
    action_type: OUTPUT (0)
    Output_Port: 1
```

Multicast group address

Figure 12: *Flow_Mod* messages received on the source for setting up a logic multicast-tree.

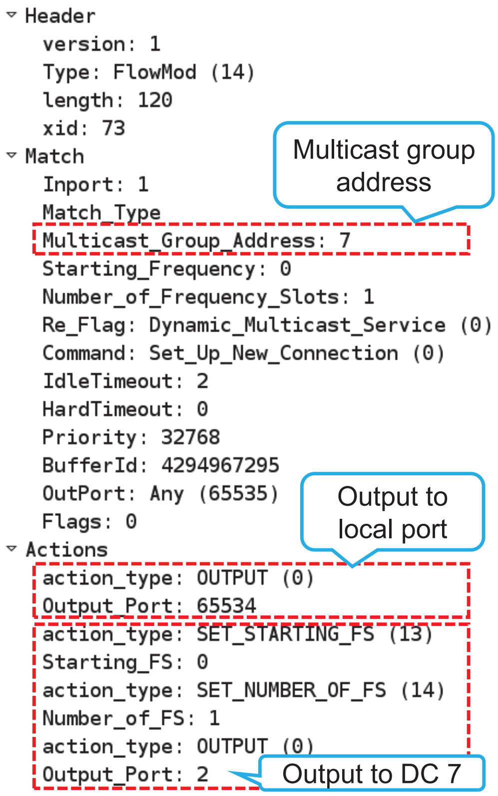


Figure 13: *Flow_Mod* messages received on a destination with a downstream member for setting up a logic multicast-tree.

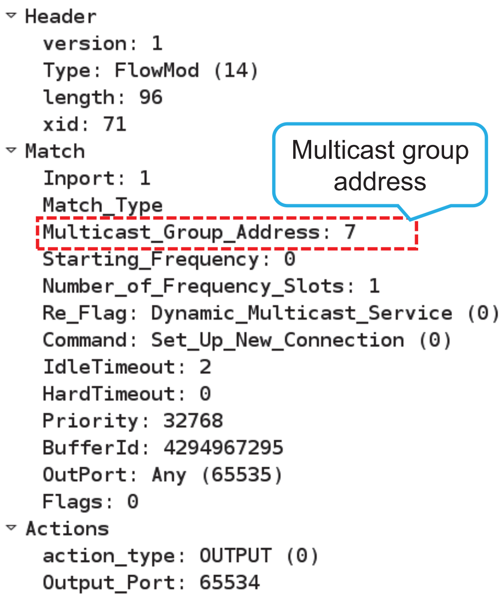
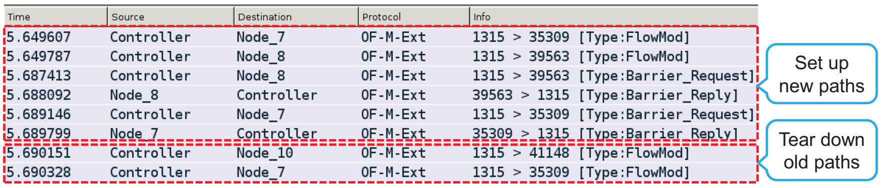


Figure 14: *Flow_Mod* messages received on a destination without a downstream member for setting up a logic multicast-tree.



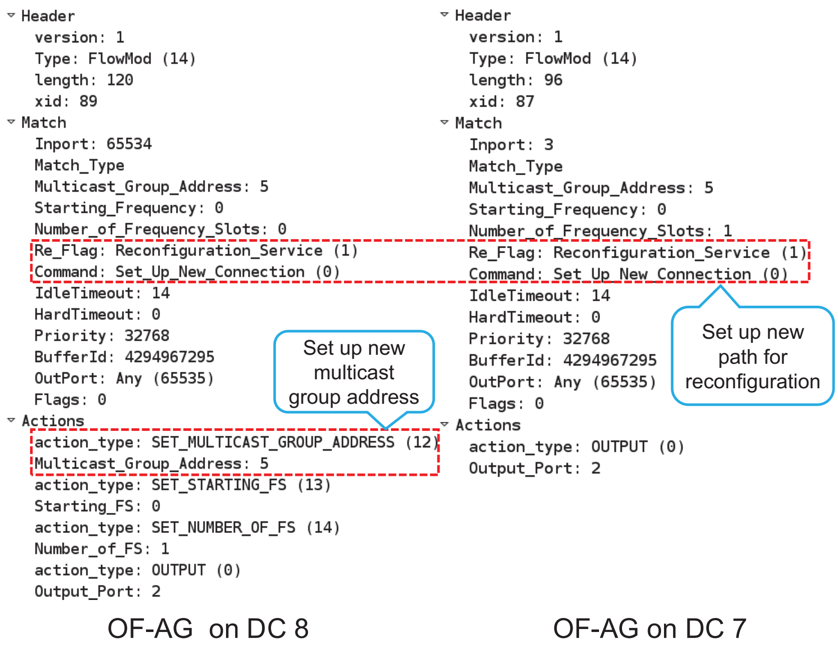
Time	Source	Destination	Protocol	Info
5.649607	Controller	Node_7	OF-M-Ext	1315 > 35309 [Type:FlowMod]
5.649787	Controller	Node_8	OF-M-Ext	1315 > 39563 [Type:FlowMod]
5.687413	Controller	Node_8	OF-M-Ext	1315 > 39563 [Type:Barrier_Request]
5.688092	Node_8	Controller	OF-M-Ext	39563 > 1315 [Type:Barrier_Reply]
5.689146	Controller	Node_7	OF-M-Ext	1315 > 35309 [Type:Barrier_Request]
5.689799	Node_7	Controller	OF-M-Ext	35309 > 1315 [Type:Barrier_Reply]
5.690151	Controller	Node_10	OF-M-Ext	1315 > 41148 [Type:FlowMod]
5.690328	Controller	Node_7	OF-M-Ext	1315 > 35309 [Type:FlowMod]

Figure 15: Wireshark Capture for the OF messages involved in reconfiguration a session.

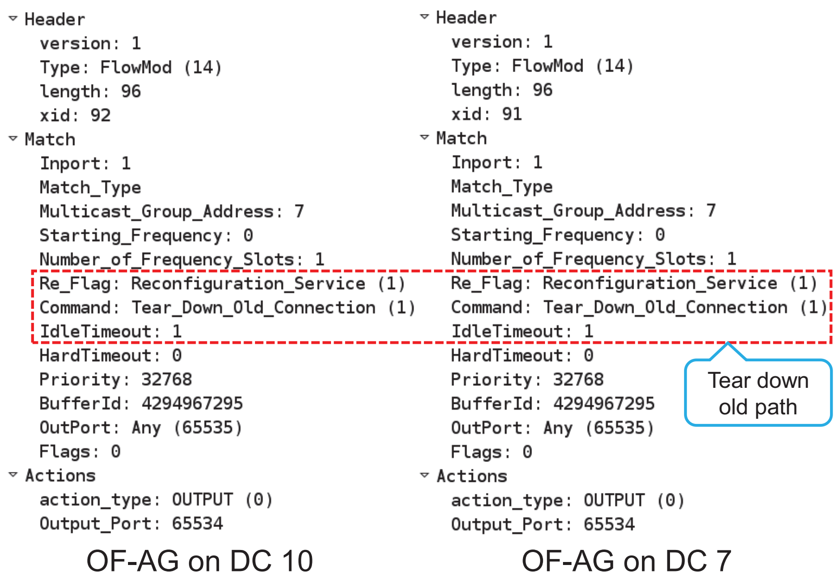
on *DC* 10 has 2 forwarding actions, and needs to output the flow to both the local port and *DC* 7.

We then consider the case that tree rearrangement is invoked, and conduct an online reconfiguration experiments. Here, we assume that due to the dynamic join-ins of multicast group members, the multicast-tree mentioned above becomes $\{9 \rightarrow 10, 10 \rightarrow 7, 7 \rightarrow 5, 9 \rightarrow 8\}$, on which *DCs* 7 and 10 have already left the session and become NM-RNs. Apparently, the multicast-tree is sub-optimal and we can calculate the optimal one as $\{9 \rightarrow 8, 8 \rightarrow 7 \rightarrow 5\}$. Here, we implement a full rearrangement and the Wireshark capture for the OF messages used to reconfigure the multicast session is illustrated in Fig. 15. Basically, we need to set up new connection of $8 \rightarrow 7 \rightarrow 5$ and tear down the original ones of $\{9 \rightarrow 10, 10 \rightarrow 7\}$. The details on the *Flow_Mod* messages used in the multicast-tree rearrangement are shown in Fig. 16. Here, we can see that the *Flow_Mod* message received on the OF-AG on *DC* 8 includes a **SET_MULTICAST_GROUP_ADDRESS** action to change the multicast group address of the session, as the multicast group has been changed.

Finally, we perform dynamic networking experiments to test the system's performance under the situation that the multicast sessions can come, change and leave on-the-fly. Here, we test two algorithms, *i.e.*, N/R and QTS-P-0.7, and Fig. 17 shows the experimental results on blocking probability. For each traffic load, we test around 20000 dynamic requests from the OF-AGs in the testbed, and we invoke a rearrangement every time when 100 requests have expired in the SD-EON. As expected, QTS-P-0.7 provides significantly better blocking



(a) Set up new connection of $8 \rightarrow 7 \rightarrow 5$.



(b) Tear down old connections of $\{9 \rightarrow 10, 10 \rightarrow 7\}$.

Figure 16: *Flow_Mod* messages used to rearrange a multicast session.

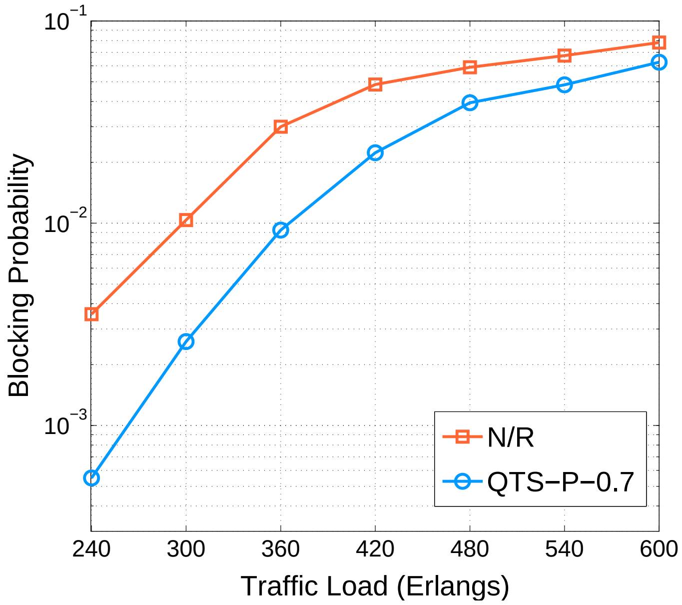


Figure 17: Experimental results on blocking probability.

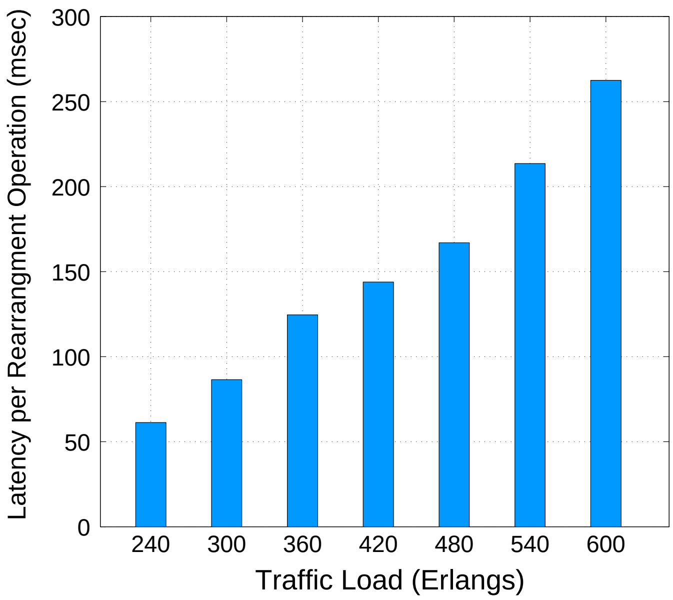


Figure 18: Experimental results on latency per rearrangement operation.

performance than N/R, which verifies that the control plane system operates correctly to facilitate dynamic formulation of multicast sessions in an inter-DC SD-EON. Fig. 18 shows the average latency per reconfiguration operations of QTS-P-0.7. When the traffic load is 600 Erlangs, the latency is 263 milliseconds. The number of lightpaths rearrangement increases with the traffic load, thus the latency also increases.

7. Conclusion

In this paper, we considered the dynamic formulation of multicast sessions in inter-DC EONs built with MI-BV-WSS'. As the changing of multicast group members can degrade the optimality of multicast-trees, we proposed selective multicast rearrangement schemes for efficient service provisioning. Specifically, we tried to rearrange the multicast-trees adaptively to reduce their spectrum usages. Meanwhile, we aimed at minimizing the frequency of rearrangements to avoid unnecessary operation complexity. Based on these considerations, we designed several multicast-tree rearrangement algorithms for updating multicast sessions dynamically with lightpath reroutings. Both partial and full multicast-tree rearrangements were considered. Simulation results indicated that the proposed algorithms could rearrange the multicast-trees intelligently such that the blocking probability could be reduced effectively with the least number of lightpath reroutings. Among all the proposed algorithms, the QTS-based algorithm achieved the best tradeoff between the blocking performance and the operation complexity due to lightpath reroutings. Based on the theoretical investigations, we investigated how to implement the proposed algorithms in the control plane of an inter-DC SD-EON. We extended the OF protocol to support the dynamic formulation of multicast sessions and also designed the functional models in the control plane elements to realize multicast-tree rearrangements. Experiment results verified the effectiveness of our proposed algorithms and system design.

Acknowledgments

This work was supported in part by the NSFC Project 61371117, the Fundamental Research Funds for the Central Universities (WK2100060010), Natural Science Research Project for Universities in Anhui (KJ2014ZD38), and the Strategic Priority Research Program of the Chinese Academy of Sciences (XDA06011202). The authors also would like to point out that the second author, Dr. Yan Li, is the co-corresponding author of this paper.

References

[1] Cisco visual networking index: Forecast and methodology, 2014 - 2019, Tech. rep. (May. 2015).
URL http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.html/

[2] P. Lu, L. Zhang, X. Liu, J. Yao, Z. Zhu, Highly-efficient data migration and backup for big data applications in elastic optical inter-datacenter networks, IEEE Netw. 29 (5) (2015) 36–42.

[3] J. Yao, P. Lu, L. Gong, Z. Zhu, On fast and coordinated data backup in geo-distributed optical inter-datacenter networks, J. Lightw. Technol. 33 (14) (2015) 3005–3015.

[4] L. Gifre, F. Paolucci, O. Gonzalez de Dios, L. Velasco, L. Contreras, F. Cugini, P. Castoldi, V. Lopez, Experimental assessment of ABNO-driven multicast connectivity in flexgrid networks, J. Lightw. Technol. 33 (8) (2015) 1549–1556.

[5] L. Gong, X. Zhou, W. Lu, Z. Zhu, A two-population based evolutionary approach for optimizing routing, modulation and spectrum assignments (RMSA) in O-OFDM networks, IEEE Commun. Lett. 16 (9) (2012) 1520–1523.

[6] Z. Zhu, W. Lu, L. Zhang, N. Ansari, Dynamic service provisioning in elastic optical networks with hybrid single-/multi-path routing, J. Lightw. Technol. 31 (1) (2013) 15–22.

[7] M. Klinkowski, K. Walkowiak, On the advantages of elastic optical networks for provisioning of cloud computing traffic, IEEE Netw. 27 (6) (2013) 44–51.

[8] W. Fang, M. Lu, X. Liu, L. Gong, Z. Zhu, Joint defragmentation of optical spectrum and IT resources in elastic optical datacenter interconnections, J. Opt. Commun. Netw. 7 (4) (2015) 314–324.

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

[10] L. Gong, X. Zhou, X. Liu, W. Zhao, W. Lu, Z. Zhu, Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks, J. Opt. Commun. Netw. 5 (8) (2013) 836–847.

[11] X. Liu, L. Gong, Z. Zhu, Design integrated RSA for multicast in elastic optical networks with a layered approache, in: Proc. of GLOBECOM, 2013, pp. 2346–2351.

[12] D. Le, F. Zhou, M. Molnar, Light-hierarchy for provisioning multiple multicast requests in sparse splitting WDM networks, in: Proc. of ICNC, 2015, pp. 847–852.

[13] L. Yang, L. Gong, F. Zhou, B. Cousin, M. Molnar, Z. Zhu, Leveraging light-forest with rateless network coding to design efficient all-optical multicast schemes for elastic optical networks, J. Lightw. Technol. 33 (18) (2015) 3945–3955.

[14] K. Walkowiak, R. Goscien, M. Wozniak, M. Klinkowski, Joint optimization of multicast and unicast flows in elastic optical networks, in: Proc. of ICC, 2015, pp. 5186–5191.

[15] W. Hu, Q. Zeng, Multicasting optical cross connects employing splitter-and-delivery switch, IEEE Photon. Technol Lett. 10 (7) (1998) 970–972.

[16] X. Liu, L. Gong, Z. Zhu, On the spectrum-efficient overlay multicast in elastic optical networks built with multicast-incapable switches, IEEE Commun. Lett. 17 (9) (2013) 1860–1863.

[17] A. Gadkar, T. Entel, J. Plante, V. Vokkarane, Slotted advance reservation for multicast-incapable optical wavelength division multiplexing networks, J. Opt. Commun. Netw. 6 (3) (2014) 340–354.

[18] OpenFlow.
URL http://www.openflow.org/

[19] L. Liu, R. Muñoz, R. Casellas, T. Tsuritani, R. Martínez, I. Morita, OpenSlice: an OpenFlow-based control plane for spectrum sliced elastic optical path networks, Opt. Express 21 (4) (2013) 4194–4204.

[20] Z. Zhu, X. Chen, C. Chen, S. Ma, M. Zhang, L. Liu, S. Yoo, OpenFlow-assisted online defragmentation in single-/multi-domain software-defined elastic optical networks, J. Opt. Commun. Netw. 7 (1) (2015) A7–A15.

[21] Z. Zhu, C. Chen, S. Ma, L. Liu, X. Feng, S. Yoo, Demonstration of cooperative resource allocation in an OpenFlow-controlled multidomain and multinational SD-EON testbed, J. Lightw. Technol. 33 (8) (2015) 1508–1514.

[22] R. Casellas, R. Martínez, R. Muñoz, R. Vilalta, L. Liu, T. Tsuritani, I. Morita, Control and management of flexi-grid optical networks with an integrated stateful path computation element and OpenFlow controller, J. Opt. Commun. Netw. 5 (10) (2013) A57–A65.

[23] L. Liu, Y. Yin, M. Xia, M. Shirazipour, Z. Zhu, R. Proietti, Q. Xu, S. Dahlfort, S. Yoo, Software-defined fragmentation-aware elastic optical networks enabled by OpenFlow, in: Proc. of ECOC, 2013, pp. 1–3.

[24] S. Ma, C. Chen, S. Li, M. Zhang, S. Li, Y. Shao, Z. Zhu, L. Liu, S. Yoo, Demonstration of online spectrum defragmentation enabled by OpenFlow in software-defined elastic optical networks, in: Proc. of OFC, 2014, pp. 1–3.

[25] Pica 8.
URL http://www.pica8.com/

[26] L. Liu, D. Zhang, T. Tsuritani, R. Vilalta, R. Casellas, L. Hong, I. Morita, H. Guo, J. Wu, R. Martinez, R. Munoz, Field trial of an OpenFlow-based unified control plane for multilayer multigranularity optical switching networks, J. Lightw. Technol. 31 (4) (2013) 506–514.

[27] Bringing wavelength agility and efficiency to software-defined networks, Tech. rep. (Feb. 2015).
URL http://resources.alcatel-lucent.com/?cid=184883/

[28] S. Li, W. Lu, X. Liu, Z. Zhu, Fragmentation-aware service provisioning for advance reservation multicast in SD-EONs, Opt. Express 23 (20) (2015) 25804–25813.

[29] N. Sambo, et al., Next generation sliceable bandwidth variable transponders, IEEE Commun. Mag. 53 (2) (2015) 163–171.

[30] M. Zhang, C. You, H. Jiang, Z. Zhu, Dynamic and adaptive bandwidth defragmentation in spectrum-sliced elastic optical networks with time-varying traffic, J. Lightw. Technol. 32 (2014) 450–460.

[31] Open vSwitch.
URL http://www.openvswitch.org/

[32] POX Wiki.
URL http://openflow.stanford.edu/display/ONL/POX+Wiki