Comparing Single Layer and Multilayer Approaches to Serve Multicast Requests on Flexgrid Networks

Lluís Gifre*, Marc Ruiz, Adrià Asensio, and Luis Velasco
Optical Communications Group (GCO), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
**e-mail: lgifre@ac.upc.edu*

ABSTRACT

Advanced services require high bitrate multicast connectivity being provided by core transport networks. Multicast services can be supported by a tree connecting the source to every destination of the multicast request. As a result of the bitrate that might be required, providing them directly on the optical layer could bring benefits. Notwithstanding, with the advent of the flexgrid technology, connections with capacities of 400 Gb/s and beyond can be set up that opens opportunities to create virtual topologies on which multicast services can be provided. In this paper, we compare the performance of providing high bitrate multicast services on the single layer and the multilayer approach; heuristic algorithms for the two considered approaches are developed. Exhaustive simulation results carried out on three national core network topologies show that the tree scheme on the multilayer approach outperforms the rest of options.

Keywords: multicast services, flexgrid networks, datacenter interconnection, content distribution.

1. INTRODUCTION

The increasing demand of multimedia, cloud, and other advanced services frequently requires providing multicast services over core transport networks; examples include ultra-high definition television distribution, synchronization of distributed database instances running in distant datacenters (DC), etc. Providing multicast services requires establishing connections from one source to a number of destinations. However, the previously defined service examples require multi Gb/s connections, e.g., uncompressed real time 8k transmission needs 72 Gb/s connections [1].

As a result of the large bitrate required, providing these stringent multicast services over an IP network is not the only solution. Instead, the optical layer seems a better option as long as they provide the required support. In that regard, the flexgrid optical technology allows creating unicast optical connections (*lightpaths*) of unprecedented bitrate, e.g. 400 Gb/s and even higher. Sliceable Bandwidth-variable optical transponders (SBVT), with a given capacity (B) and able to transmit and receive several (f) optical signals of different bitrates can be used to simplify inventory. Hence, multicast services can be provided on the optical layer by creating as many lightpaths as the number of destinations. A bulk path computation algorithm (e.g., see [2]) can be used to compute the optimal Routing and Spectrum Allocation (RSA) for the whole set.

A more natural way of providing multicast services is by creating trees, connecting source to destination nodes in one single connection. As demonstrated in [3], trees (*light-trees*) can be implemented on flexgrid optical networks. Besides, authors in [4] experimentally demonstrate the feasibility of control plane architectures to support light-trees in flexgrid networks (Fig. 1).

Several papers have compared the performance of light-trees against establishing a set of lightpaths. Notwithstanding the feasibility of serving large bitrate multicast requests on the optical layer, implementing light-trees depends on the technology of the optical nodes. In addition, authors in [5] showed that creating transparent light-trees produces poor performance in large networks. Therefore, some works have explored creating virtual topologies to groom multicast demand (Fig. 2). Serving multicast connections using a virtual network originates an asymmetric use of resources when bidirectional virtual links are considered. To deal with such asymmetry, authors in [6] studied the benefits of creating unidirectional lightpaths to support the virtual topology when traffic is significantly asymmetric.

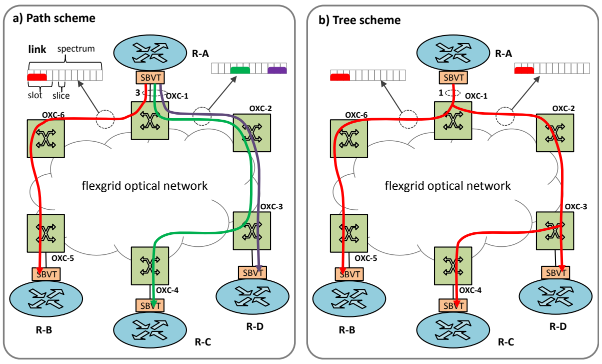

Figure 1. Path and tree schemes under the single layer approach.

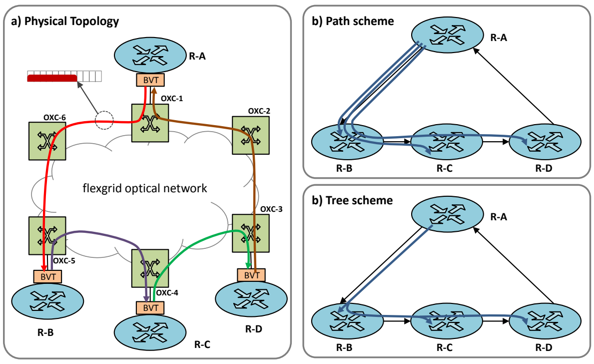

Figure 2. Path and tree schemes under the multilayer approach.

In this paper we compare the performance of path and tree schemes to serve large bitrate multicast requests under two different network approaches: *i*) single layer, where each multicast request is served either by a number of lightpaths or by a single light-tree in the flexgrid layer, and *ii*) multilayer, where each multicast request is served either by a number of paths or by a single tree in a virtual topology supported by lightpaths.

2. HEURISTIC ALGORITHMS

Multicast Provisioning Single Layer tree (McP-SL-tree)

Table 1 shows the main algorithm for solving the McP-SL-tree problem, which consists in a routing phase that builds a tree followed by a spectrum allocation phase. Although routing and spectrum allocation are independent, the routing phase tries to guarantee the availability of at least one end-to-end slot for the request.

Before starting the routing phase, the set of all possible slots with enough bandwidth to allow serving the requested bitrate is pre-computed (line 2 in Table 1). Next, an iterative procedure that finds and merges routes from source to destinations into a tree under construction is executed (lines 3-12). The procedure runs until all the destination nodes are added to the tree (line 3). At each iteration, all remaining destinations are evaluated as candidates to be inserted into the tree. Once some parameters are initialized (line 4), the route with the minimum cost for each remaining destination is found (lines 5-6). Among all destinations, the one with the highest minimum cost is selected (lines 7-8). The solution is infeasible if no route is available for any destination (line 9). On the contrary, the tree is updated adding the new links in the selected route (line 10). The set of slots is updated eliminating those slots that cannot be selected after adding the new route (line 11). Finally, since a route can cover more than one remaining destination, the set *T* must be also updated at the end of each iteration (line 12). Once the tree has been built, an available slot is chosen from the set of end-to-end available slots along every tree edge (line 13), and the light-tree is eventually returned.

Table 1. McP-SL-tree algorithm.

```
INPUT <o, T, b>, S, R, <ch,ct,cs>
OUTPUT solution
 1: solution←Ø
 2: slots←computeSlots(S,b)
 3: while T≠ Ø do
 4:     maxCost←-∞; $r_{sel}$←Ø
 5:     for each t ∈ T do
 6:         <route,cost> ←
                R-SL($R^O$(o,t),S,slots,T,solution,<ch,ct,cs>)
 7:         if cost>maxCost then
 8:             maxCost←cost; $r_{sel}$←route
 9:     if $r_{sel}$=Ø return infeasible
10:     solution←updateTree(solution,$r_{sel}$)
11:     slots←updateSlots(slots,solution)
12:     T←updateT(T,solution)
13: solution←SA(slots)
14: return solution
```

Table 2. R-SL algorithm.

```
INPUT $R^O$, S, slots, T, solution, <ch,ct,cs>
OUTPUT route, cost
 1: route←Ø; cost←∞
 2: for each r ∈ $R^O$ do
 3:     nslots←compAvailableSlots(r,slots,S,solution)
 4:     if nslots=0 then continue
 5:     nhops←compHopsNotInTree(r,solution)
 6:     ndests←compIntermDestinations(r,T)
 7:     costIte←ch*nhops-ct*ndests-cs*nslots
 8:     if cost<costIte then
 9:         route←r
10:         cost←costIte
11: return route, cost
```

The success of the algorithm in Table 1 strongly depends on the algorithm R-SL that finds the candidate routes to be added to the tree. Table 2 shows the details of such algorithm, which receives, in addition to other data and parameters previously described, a set R^O of pre-computed routes from the source to the destination being evaluated over the optical layer. After initializing the output variables (line 1), the cost of each route is evaluated aiming at finding that with min cost, taking into account the already built tree and slot availability.

Before computing the cost of a candidate route, the number of slots in common with the current tree is computed (lines 3-4). If some slot in common is found, the number of hops in the route not used in the current tree is computed (line 5). Next, the number of intermediate destinations not included in the current tree that are intermediate nodes in the route to the end destination is obtained (line 6). These variables are used to compute the cost of the route (line 7). If cost coefficients are non-negative, the cost of the route is minimized when: *i*) the route adds few new links to the existing tree; *ii*) the number of intermediate destinations increases; and *iii*) the number of available slots increases. It is worth mentioning that the weight of each cost coefficient will affect the selected route and, consequently, the quality of the obtained light-tree.

Multicast Provisioning Multilayer tree (McP-ML-tree)

Table 3 presents the McP-ML-tree main procedure. In contrast to the set of pre-computed routes R^O used in the McP-ML-tree algorithm, the set R^V stores pre-computed routes between *o* and *T* over the virtual topology. To compute R^V, two kind of virtual links are considered: *i*) current virtual links with enough residual capacity to serve the request (E^{on}), and *ii*) new virtual links that can be supported by feasible lightpaths, taking into account spectrum and transponders availability (E^{off}). All routes that do not fulfill those conditions are pruned (lines 1-3).

After pre-processing routes, the McP-ML-tree algorithm is called to find a tree on the virtual topology and the set of virtual links in E^{off*} to be set up for the multicast request (line 4). A lightpath with the desired bitrate needs to be found for each of the links in E^{off*} (line 7). To minimize resource utilization when routing several lightpaths, a bulkRSA algorithm can be used (e.g. see [2]). Note that in the pre-processing phase we ensured that each single link in E^{off} can be established. However, when E^{off*} contains several links, not all could be set-up simultaneously with the available capacity, so the request could be rejected (line 8). If every virtual link can be created, G^V is updated and the solution finally returned (lines 9-10).

Table 3. McP-ML-tree main procedure.

```
INPUT $G^O(N,L)$, $G^V(N^R,E)$, <o, T, b>, S, $R^V$, <ch,ct,cv>
OUTPUT solution
1:  $E^{on}$ ← getAvailableVlinks($G^V$,b)
2:  $E^{off}$ ← getNewVlinks($G^O$,$G^V$,S)
3:  $R^V$ ← pruneR($R^V$,$E^{on}$,$E^{off}$)
4:  <solution,$E^{off*}$> ←
        McP-ML-tree(<o,T,b>, $R^V$,$E^{on}$,$E^{off}$,<ch,ct,cv>)
5:  if solution=Ø then return infeasible
6:  if $E^{off*}$≠Ø then
7:      status ← bulkRSA($G^O$,S,$E^{off*}$)
8:      if status=infeasible then return infeasible
9:      $G^V$ ← addVLinks($G^V$,$E^{off*}$)
10: return solution
```

Table 4. R-ML algorithm.

```
INPUT $R^V$, $E^{off}$, T, solution, <ch,ct,cv>
OUTPUT route,cost
1:  route ← Ø; cost ← ∞
2:  for each $r \in R^V$ do
3:      nvlinks ← compNewVlinks(r,$E^{off}$)
4:      nhops ← compHopsNotInTree(r,solution)
5:      ndests ← compIntermDestinations(r,T)
6:      costIte ← $cv \cdot nvlinks + ch \cdot nhops - ct \cdot ndests$
7:      if cost < costIte then
8:          route ← r
9:          cost ← costIte
10: return {route,cost}
```

The McP-ML-tree algorithm is similar to that of the McP-SL-tree in Table 1. Just note that pre-computing and updating available slots is not done in the McP-ML-tree algorithm. Much the same, the R-ML algorithm to compute the cost of adding a route to the tree for the multi-layer approach in Table 4 is similar to the R-SL algorithm in Table 2. In this case, the number of new virtual links to be established needs to be obtained (line 3). As for the single layer case, the cost is computed as the sum of the tree components weighted by their cost coefficients. Note that the higher is the amount of new links to install, the higher is the cost of the route.

3. ILLUSTRATIVE NUMERICAL RESULTS

The performance of path and tree schemes under the two approaches is compared on three realistic national network topologies: the 12-node Deustche Telekom (DT), the 22-node British Telecom (BT), and the 30-node Telefonica (TEL). For these experiments, the optical spectrum width is fixed to 2 THz in each link and all connections use the QPSK modulation format. Over such optical topologies, 6 DCs have been placed in areas with high population density; we consider that connections are requested among those DCs.

A dynamic network environment was simulated where incoming multicast requests arrive to the system following a Poisson process and are sequentially served without prior knowledge of future incoming requests. The number of nodes in each multicast request was selected in the range [3, 6], and the DCs are chosen using a uniform distribution among all DCs. The bitrate was fixed for every request to 100 Gb/s. The time between two consecutive request arrivals and the holding time of an accepted request were randomly generated following an exponential distribution. Different values of offered network load, computed as the total bitrate of the connections being served, were considered by changing the arrival rate.

The algorithms described in Section 2 were used to solve the McP problem for the tree scheme. For comparison purposes, the path scheme was also considered in the experiments. A multicast request involving n destinations is divided into n unicast requests from the source to each single destination. Note that, for the sake of a fair comparison, a multicast request is rejected if one or more unicast requests cannot be served. To solve the McP-SL-path problem, a dynamic RSA algorithm based on the bulk RSA proposed in [2] was implemented. Note that this algorithm was used for finding lightpaths for new virtual links in the McP-ML-tree procedure in Table 3. As for the McP-ML-path problem, a reduced variation of the bulk RSA algorithm without spectrum assignment was adapted to find minimum cost paths over the virtual topology. In the SL approach, we assumed that $f = 4$, $B = 400$ Gb/s SBVTs are equipped in the DC routers, whereas in the ML approach, to take advantage of grooming, the virtual topology is supported by 400 Gb/s lightpaths and therefore, 400 Gb/s BVTs are equipped.

Figure 3 plots the blocking probability (Pb) as a function of the offered load for the schemes and approaches considered on the three national topologies. Loads are normalized with respect to the load that causes a $Pb = 1\%$ under the McP-ML- tree scheme. For this set of experiments, we assumed that enough transponders are equipped at each node to ensure that blocking is as a result of the lack of spectrum resources.

The McP-ML-tree scheme provides the best performance on every topology as a result of its efficient resource usage that combines low resource consumption of the tree scheme and high efficient capacity utilization of the ML approach. In contrast, the McP-SL-tree scheme does not achieve good performance in every topology; whilst it performs similarly than the McP-ML-tree on the DT topology, it shows a noticeable worse performance in the TEL network, even worse than that of the McP-ML-path scheme. The rationale behind this is related to the spectrum continuity constraint and the size of the network. In fact, finding a light-tree under the spectrum continuity constraint poses a challenge in large networks, where the number of hops increases.

As for the paths schemes, the McP-SL-path performs the worst in every network topology, and although the path scheme performs better under the ML approach, it is far from that of the McP-ML-tree.

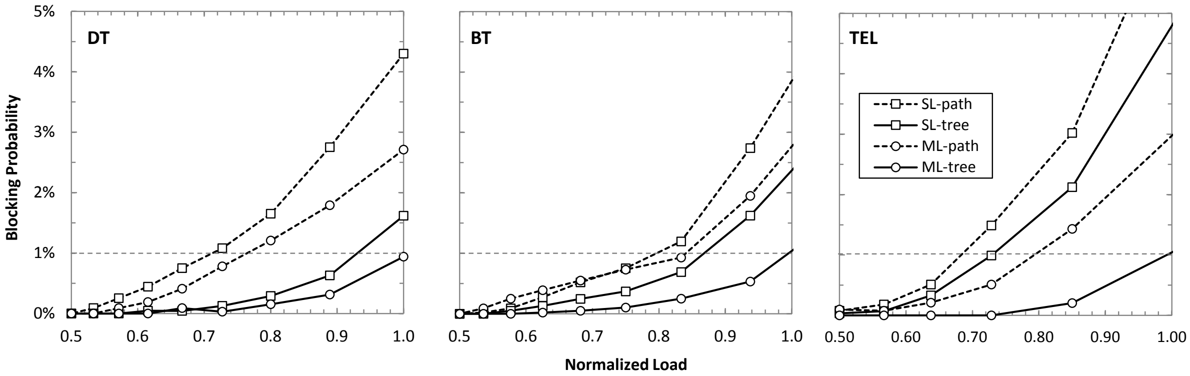

Figure 3. Blocking probability vs. normalized load.

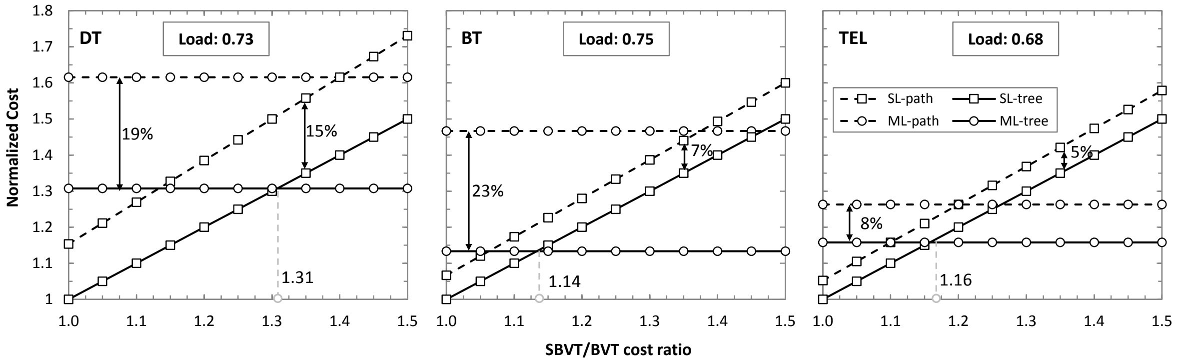

Figure 4. Transponder cost analysis.

Let us now study the cost of the schemes related to transponders. Recall that SBVTs are installed in DC routers under the SL approach, whereas BVTs are equipped under the ML approach. Aiming at comparing transponder cost for a given multicast traffic load under the different schemes, we accounted for the number of transponders to ensure $Pb \leq 1\%$. Figure 4 shows total transponders cost for several SBVT/BVT cost ratios for the highest normalized loads unleashing $Pb \leq 1\%$ for each considered network topology. Note that since schemes under the ML approach use BVTs its cost does not depend on SBVT/BVT ratio.

It is clear, in view of Fig. 4, that the tree scheme brings cost savings compared to the path one since the number of transponders need to be installed in the DC routers is lower. Cost savings of the tree scheme range between 5% and 15% and between 8% and 23% under the SL and the ML approach, respectively. Comparing the tree scheme under the both approaches, the SL approach present cost savings as long as the cost of a SBVT is not higher than 30% that of an BVT for the DT topology and not higher than 14% – 16% for the BT and TEL networks. Above those ratios, the ML approach provides costs savings with respect to the SL approach.

4. CONCLUSIONS

Two approaches for multicast services were compared; the single later approach, serving traffic directly on the optical layer can take advantage of SBVTs to reduce transponder count, whereas the multilayer one serving traffic on a virtual network topology created on top of the optical network just require simpler BVTs. Results validated the ML-tree scheme as the best option to serve a mix of multicast and unicast traffic.

ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement n° 317999 IDEALIST and from the Spanish MINECO SYNERGY project (TEC2014-59995-R).

REFERENCES

[1] S. Namiki *et al.*, “Ultrahigh-definition video transmission and extremely green optical networks for future”, *IEEE Sel. Top. in Q. Elect.*, vol. 17, pp. 446-457, 2011.
[2] L. Velasco *et al.*, “Solving routing and spectrum allocation related optimization problems: From off-line to in-operation flexgrid network planning,” *IEEE/OSA J. of Lightwave Tech.*, vol. 32, pp. 2780-2795, 2014.
[3] N. Sambo *et al.*, “Demonstration of data and control plane for optical multicast at 100 and 200 Gb/s with and without frequency conversion,” *IEEE/OSA J. Opt. Commun. Netw.*, vol. 5, pp. 667-676, 2013.
[4] Ll. Gifre *et al.*, “Experimental assessment of ABNO-driven multicast connectivity in flexgrid networks,” (Invited Paper) *IEEE/OSA Journal of Lightwave Technology (JLT)*, vol. 33, pp. 1549-1556, 2015.
[5] M. Ruiz and L. Velasco, “Performance evaluation of light-tree schemes in flexgrid optical networks,” *IEEE Communications Letters*, vol. 18, pp. 1731-1734, 2014.
[6] S. Woodward *et al.*, “Asymmetric optical connections for improved network efficiency,” *IEEE/OSA J. of Optical Comm. and Netw.*, vol. 5, pp. 1195-1201, 2013.