Hypo-Steiner heuristic for multicast routing in all-optical WDM mesh networks

Fen Zhou, Miklós Molnár, Bernard Cousin

Noname manuscript No.
(will be inserted by the editor)

Hypo-Steiner Heuristic for Multicast Routing in All-Optical WDM Mesh Networks

Fen Zhou · Miklós Molnár · Bernard Cousin

Received: date / Accepted: date

Abstract In sparse light splitting all-optical WDM networks, the more destinations a light-tree can accommodate, the fewer light-trees and wavelengths a multicast session will require. In this article, a Hypo-Steiner Light-tree algorithm (HSLT) is proposed to construct a HSLT light-tree to include as many destinations as possible. The upper bound cost of the light-trees built by HSLT is given as $N(N-1)/2$, where N is the number of nodes in the network. The analytical model proves that, under the same condition, more destinations could be held in a HSLT light-tree than a Member-Only [1] light-tree. Extensive simulations not only validate the proof but also show that the proposed heuristic outperforms the existing multicast routing algorithms by a large margin in terms of link stress, throughput, and efficiency of wavelength usage.

Keywords All-optical WDM Networks · Multicast · Routing and Wavelength Assignment (RWA) · Sparse Light Splitting · Light-tree · Hypo-Steiner Heuristic

1 Introduction

With the increasing popularity of bandwidth-driven and time sensitive applications (like Video-Conference, HTDV and Video On Demand), all-optical WDM networks for the Next Generation Internet (NGI) need to provide multicast services with high throughput and small latency [2]. In order to support multicast in WDM networks, a light-tree concept was proposed in [3], which is a tree in the physical topology and occupies the same wavelength over all the fiber links in the tree. It has been found to provide substantial savings in the network-wide average packet hop distance and the total number of transceivers in the networks [4]. However, due to the physical constraints and characteristics in all-optical WDM networks, multicast routing is a challenging work. First, in the absence of any wavelength conversion device, the same wavelength should be employed over the light-tree, which is referred as the wavelength continuity constraint [4]. Meanwhile, two or more light-trees traversing the same fiber link must be assigned different wavelengths, so that they do not interfere with one another, which is referred as the distinct wavelength constraint [4]. In addition, since all-optical multicast has to distribute packets in the optical domain, branching nodes (or switch nodes) in a light-tree is required to be equipped with light splitters. By employing the light splitting capability, the branching node is able to replicate the incoming packets in the optical domain and forward them to all the required outgoing ports. Usually, a node capable of light splitting is named as a Multicast Capable node (MC) [8]. Otherwise, it is a Multicast Incapable node (MI) [8]. Typically, the network nodes at least possess of the Tap and Continue (TaC [5]) capability to tap into the light signal for local consumption and forward it to only one outgoing port. From the point of optical energy budget, a light splitter reduces the power level of a light signal by a factor equal to the number of optical copies. The reduction of power should be compensated by inter-

Fen Zhou, Miklós Molnár, and Bernard Cousin
IRISA, Campus de Beaulieu, Rennes 35042, France
Tel.: +33-299847194
Fax: +33-299847171
E-mail: {fen.zhou, molnar, bernard.cousin}@irisa.fr

Fen Zhou, and Miklós Molnár
INSA Rennes, 20, Avenue des Buttes de Coësmes, Rennes 35042, France

Bernard Cousin
University of Rennes 1, Campus de Beaulieu, Rennes 35042, France

nal active amplifiers like erbium-doped fiber amplifier (EDFA) [6], which, however, introduce many problems such as Gain Dispersion, Gain Saturation and Noise [7]. Consequently, the complex architectures along with the high-cost of optical amplification make MC nodes much more expensive than the MI nodes. That is why only a subset of nodes in the WDM networks support light splitting, which is characterized as sparse light splitting [8].

Multicast routing in sparse light splitting WDM networks mainly involves two subproblems: light-tree construction and wavelength assignment. The first subproblem is to build a light-tree to include destinations in a multicast session under the degree limitation of MI nodes. Since the MI nodes cannot be a branching node in a light-tree, one light-tree may not be enough to include all the destinations in the multicast session and several ones may be required, which thus forms a light-forest [1]. In a light-tree, generally, one emitter of the source node transmits light signals on only one wavelength to the spanned destinations. In case that the source node can not split, several emitters are deployed to deliver light signals through different outgoing ports while using the same wavelength. Typically, by employing iterative light-tree formation algorithms like Member-Only [1], the set of light-trees computed for the same multicast session are not edge disjoint. Referring to the wavelength distinct constraint [4], the maximum number of wavelengths required per fiber by a multicast session equals to the number of light-trees constructed for this multicast session. Given a multicast group, the more destinations a light-tree is able to span, the fewer wavelengths will be needed. A lot of research has addressed the light-tree formation algorithms [1, 7, 9, 13, 14, 17]. In [9], a QoS routing algorithm is proposed to construct a light-tree with delay constraint and the cost approximation of light-tree is given. However, all the nodes are assumed to be capable of splitting this is why only one light-tree is required. In [1] the Reroute-to-Source algorithm proposes to make use of the shortest path tree from the source to all the destinations. Since the non-splitter MI nodes are not considered during the construction of the light-trees, the nodes connected to a MI node have to use different wavelengths to communicate with the source. Thereby, the number of destinations included in a light-tree is limited, and this is costly in terms of link stress (i.e., the maximum number of wavelengths required per fiber) and the total cost (i.e., the total wavelength channels used for a multicast session). In [7] a Tabu search heuristic is addressed to find the optimal light-tree. As we know Tabu search is just a local search method, the result depends a lot on the initial solution, and hence the number of destinations included in one light-tree cannot be guaranteed. Besides, Tabu search is not time efficient for dynamic multicast traffic. The Member-Only algorithm proposed in [1] takes the splitting capability of nodes into account when constructing a light-tree. It is currently thought to have the best link stress and total cost for multicast routing in all-optical networks with spare light splitting and without wavelength conversion. However, Member-Only is just a simple application of the Minimum Path Heuristic [10], which uses exclusively the shortest path to connect the nearest destination to a light-tree. It may face failures (Figs. 2 and 3), since a shortest path is more likely to traverse Tac-exhausted MI nodes (i.e., a non-leaf MI nodes in a light-tree under construction), which is not permitted by the light splitting constraint. In fact, in case the shortest path does not work, a longer constraint satisfied path can be a good choice to connect a destination to the light-tree. Hence, the number of destinations in a Member-Only light-tree still could be improved (Fig. 6). In [13], a trade off is found between the link stress and the average delay. The DijkstraPro algorithm is proposed to construct the shortest path tree while taking the sparse light splitting into consideration. Although the delay is diminished, the links stress is still a little bigger than that of Member-Only algorithm. In [17], a distance priority is introduced during the spanning of multicast light-trees, which improves the delay of Member-Only light-trees while keeping almost the same link stress.

In this article, we try to improve the link stress and the network throughout in all-optical WDM networks with multicast traffic and sparse light splitting constraint. It is assumed that the costly wavelength conversion devices are not available in our optical networks and the backward direction of an optical fiber cannot be used on the same wavelength as its forward. Since the more destinations a light-tree can hold the fewer wavelengths a multicast session will require, we introduce the Hypo-Steiner Heuristic to include as many destinations as possible in a light-tree for a multicast session. By deleting on-leaf MI nodes in the subtree from the original graph and computing the nearest destination plus the shortest paths in the modified graph, the proposed Hypo-Steiner Heuristic is able to easily find more available paths fulfilling the optical constraints to connect destinations to the same light-tree. It always computes the shortest one among the paths satisfying the constraints, which is the shortest path between two nodes in the modified graph after deleting. However, this path is not necessary to be the shortest path between these two nodes in the original graph. That is why we call it Hypo-Steiner Heuristic.

The rest of this article is organized as follows. In the next Section, multicast routing and wavelength assignment problem in sparse light splitting WDM network is formulated and the previous work is reviewed. Then Hypo-Steiner Heuristic is presented and analyzed in Section 3. Numerical results are obtained and evaluated through the comparison among the proposed algorithm and the most efficient and state-of-art algorithms in Section 4. Finally, a conclusion is made in section 5.

2 All-Optical Multicast Routing With Sparse Light Splitting

2.1 System Model and Problem Description

The sparse light splitting WDM network can be modeled by an undirected graph $G(V, E, c, W)$. V represents the vertex-set of G. Each node $v \in V$ is deployed to be either an MI or an MC node beforehand. E represents the edge-set of G, which corresponds to the fiber links between the nodes in the network.

$$V = \{v | v = MI \ or \ v = MC\}$$
$$|V| = N, |E| = M \tag{1}$$

W denotes the number of wavelengths supported in each fiber link. Each edge $e \in E$ is associated with a cost functions $c(e)$. We define the cost $c(e)$ of each edge as

$$c(e) = 1 \ unit \ hop - count \ fiber \ cost \tag{2}$$

Function c is additive over the links of a lightpath $LP(u, v)$ between two end point nodes u and v,

$$c\big(LP(u,v)\big) = \sum_{e \in LP(u,v)} c(e) \tag{3}$$

We consider a multicast session $ms(s, D)$, which requests for setting up a set of light-trees under optical constraints (continuous wavelength constraint, distinct wavelength constraint, sparse light splitting and lack of wavelength conversion) from the source s to a group of destinations D simultaneously. Without loss of generality, assume there are $|D| = K$ destinations in a multicast session and an iterative light-tree construction heuristic builds k light-trees $LT_i(s, D_i)$ in sequence for $ms(s, D)$, where $i \in [1, k]$, and $1 \leq k \leq K \leq N - 1$. Although the i^{th} light-tree $LT_i(s, D_i)$ may span some destinations already spanned in the previous light-trees, D_i is used to denote exclusively the set of newly served destinations in the i^{th} light-tree (these newly served destinations can not be spanned in the previous light-trees, the i^{th} light-tree is built to serve them uniquely). Hence, these k sets D_i are disjoint,

$$\forall i, j \in [1, k] \ and \ i \neq j, \ D_i \cap D_j = \emptyset \tag{4}$$

Since all the destinations in D are spanned by k light-trees, we obtain

$$D = \bigcup_{i=1}^{k} D_i \tag{5}$$

Let a positive integer l_i denote the size of the subset D_i, then we have

$$\forall i \in [1, k], |D_i| = l_i \geq 1$$
$$\sum_{i=1}^{k} l_i = |D| = K \tag{6}$$

First, regarding the optimization of network resources, the total cost (i.e., the total wavelength channels used for a multicast session) and the link stress (i.e., the maximum number of wavelengths required per link by a multicast session) should be minimized. The total cost of a multicast session can be calculated by the sum of cost in all the light-trees build for the multicast session.

$$\begin{aligned} c\big(ms(s,D)\big) &= \sum_{i=1}^{k} c\big[LT_i(s, D_i)\big] \\ &= \sum_{i=1}^{k} \sum_{e \in LT_i(s,D_i)} c(e) \\ &= \sum_{i=1}^{k} \sum_{e \in LT_i(s,D_i)} 1 \end{aligned} \tag{7}$$

Since the light-trees for the same multicast session are not edge-disjoint, they could not be assigned the same wavelength. For a multicast session, link stress equals to the number of light-trees k built for the multicast session.

$$Stress\big(ms(s,D)\big) = k \tag{8}$$

Secondly, considering the network throughput, given W as the number of wavelengths supported in one fiber, the more multicast sessions can be accepted, the better.

$$Throughput = Num(accepted \ sessions)|_W \tag{9}$$

However, not all these parameters could be optimized simultaneously. The main objective in this paper is trying to build a light-tree, which is able to include as many destinations as possible for a multicast session. Accordingly, the number of light-trees built for a multicast session will be minimized.

$$\max\{|D_i| = l_i\} \ and \ \min\{k\} \tag{10}$$

2.2 Member-Only Algorithm

The heuristic considered as the best one is the Member-Only algorithm which derives from the Minimum Path

Heuristic [10]. It involves three nodes sets during the construction of the light-tree. In a subtree LT under construction,

MC_SET: includes source node, MC nodes and the leaf MI nodes. They may be used to span light-tree LT and, thus are also called connector nodes in LT.

MI_SET: includes only the non-leaf MI nodes, whose splitting capability is exhausted. Hence, these nodes are not able to connect a new destination to the subtree LT.

D: includes unserved multicast members which are neither joined to the current light-tree LT nor to the previously constructed multicast light-trees.

In Member-Only algorithm, the shortest paths between all pairs of nodes in the network are computed beforehand and stored. The span of the distribution light-tree LT begins with the source: $LT = \{s\}$, $MI\_SET = \emptyset$, $MC\_SET = \{s\}$ and $D = \{all\ destinations\}$. At each step, try to find the nearest destination from $d \in D$ to $c \in MC\_SET$, whose shortest path $SP(d, c)$ does not involve any node in MI_SET. If it is found, $SP(d, c)$ is added to LT and the sets are updated: d and the MC nodes are added to MC_SET, non-leaf MI nodes are added to MI_SET, and d is removed from D. Otherwise (i.e., no such constraints-satisfying shortest path could be found), the current light-tree LT is finished, and another light-tree assigned a new available wavelength is started using the same procedure until no destination is left in D.

Algorithm 1 Hypo-Steiner Light-tree Algorithm

Input: A graph $G(V, E, c, W)$ and a multicast session $ms(s, D_0)$.
Output: A set of Light-trees $LT_k(s, D_k)$ each on a different wavelength w_k for $ms(s, D_0)$.

$k \leftarrow 1$ {k is the serial number of a light-tree}
$D \leftarrow D_0$
while $(D \neq \emptyset)$ **do**
  $i \leftarrow 1$ {i is the serial number of a renewed graph}
  $G_i \leftarrow G(V, E, c, W)$
  $MC\_SET \leftarrow \{s\}$
  $LT_k \leftarrow \{s\}$
  while $(SP_{G_i}(\mathbf{d}, \mathbf{c}) \neq \infty)$ **do**
    for all ($d \in D$ and $c \in MC\_SET$) **do**
      Compute the shortest path $SP_{G_i}(d, c)$ in G_i.
    end for
    Find the nearest destination $\mathbf{d}$ with $SP_{G_i}(\mathbf{d}, \mathbf{c})$
    $LT_k \leftarrow LT_k \cup SP_{G_i}(\mathbf{d}, \mathbf{c})$
    $MC\_SET \leftarrow MC\_SET \cup \{\text{MC in } SP_{G_i}(\mathbf{d}, \mathbf{c})\} \cup \{\mathbf{d}\}$
    if ($\mathbf{c}$ is an MI node) **then**
      $MC\_SET \leftarrow MC\_SET \setminus \{\mathbf{c}\}$
    end if
    $D \leftarrow D \setminus \{d\}$
    $G_{i+1} \leftarrow G_i \setminus \{\text{edges and MI nodes in } SP_{G_i}(\mathbf{d}, \mathbf{c})\}$
    $i \leftarrow i + 1$
  end while
  Assign wavelength w_k to LT_k
  $k \leftarrow k + 1$ {Star a new light-tree LT_{k+1}}
end while

3 Hypo-Steiner Light-Tree Algorithm

3.1 Hypo-Steiner Heuristic

During the construction of a light-tree LT, non-leaf MI nodes in LT have exhausted their TaC capability and could not be used again to connect another destination to the subtree. In other words, these non-leaf MI nodes are useless for the spanning of the current light-tree LT. Thus, why do not we delete them from the graph? At each step, we compute the shortest paths and the distances from the destinations in set D to the subtree LT in a new graph, say G_i (generated by deleting all non-leaf MI nodes in LT from the original graph G). Then, find the nearest destination and join it to the subtree with the shortest path in G_i. Here, we can see, it is definitely true that the shortest paths between any two nodes in the new graph G_i will not traverse any MI nodes in MI_SET. Hence, by computing the shortest path in the modified graph G_i, when searching the nearest destination to the subtree LT, we need not to check whether its shortest path to LT (precisely speaking, to its connector node in MC_SET) satisfies the splitting constraints or not. The benefits of the Hypo-Steiner Heuristic are based on the following two observations.

First, Hypo-Steiner Heuristic can enumerate all the possible shortest paths which are able to connect a destination to the light-tree while respecting constraints. When using Dijkstra to compute the shortest path from node $d_i \in D$ to a connector node $c_j \in MC\_SET$, only one could be found, although there may be several ones. If the useless non-leaf MI nodes are still in the graph, then the shortest path that traverses a node in MI_SET cannot be avoided in the computation. If, unfortunately, the founded shortest path from a destination d_i to the current subtree LT traverses a node in MI_SET, then d_i could not be joined to the current light-tree LT according to the Member-Only algorithm [1] and consequently a new light-tree on another wavelength will be needed. However, after deleting all the useless non-leaf MI nodes and the used edges from the original graph G, the other shortest path, which does not involve any node in MI_SET, could be definitely found in the new graph if it exists.

Example 1: In the NSF network of Fig. 1, a multicast session $ms_1\big(s : 8, D : (4, 6)\big)$ request arrives, and only source s is an MC node. Using Member-Only algorithm, node 4 is firstly connected to node 8 using the shortest path $SP(8{-}7{-}5{-}4)$. Now, $MC\_SET = \{4, 8\}$, $MI\_SET = \{5, 7\}$. Next, compute the shortest paths from node 6 to the nodes in MC_SET. $SP(4 - 5 - 6)$

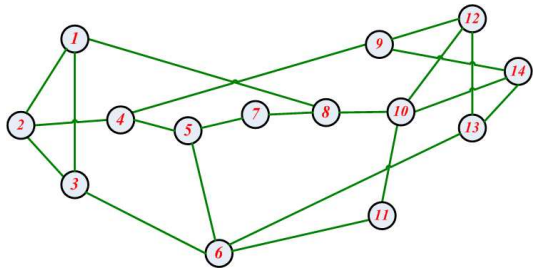

Fig. 1 NSF Network Topology

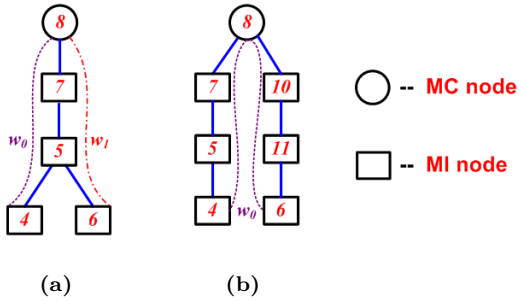

Fig. 2 For multicast session ms_1, (a) Light-tree built by Member-Only; (b) Light-tree built using Hypo-Steiner Heuristic.

involves non-leaf MI node 5, so it does not work. If $SP(8-7-5-6)$ is unfortunately computed out, then a new tree should be employed to accommodate node 6 as in Fig. 2(a). But, with the help of Hypo-Steiner Heuristic, node 5 and 7 are deleted from the original graph G_1. Hence, $SP(8-7-5-6)$ (Fig. 2(a)) could be definitely avoided, and another shortest path $SP(8-10-11-6)$ (Fig. 2(b)) could be found in the new graph G_2, which does not traverse any node in MI_SET.

Secondly, in case that the shortest path does not work, the Hypo-Steiner Heuristic tries to find a second shortest constraint-satisfied path to connect a destination to the light-tree while Member-Only will stop the spanning of the current light-tree. Only making use of the shortest path in the original graph, not all the possible paths satisfying the splitting constraints could be enumerated by Member-Only algorithm. This is because that the shortest path from node $d_i \in D$ to a connector node $c_j \in MC\_SET$ in the original graph is always used to span the subtree. However, most of these shortest paths involve some nodes in MI_SET, which are not permitted by the splitting constraint. In fact, if all the MI nodes and the used edges in LT are deleted from the graph to get G_2, then in the following step, a secondly shortest path between two nodes (it is the shortest path in the new graph G_2) may be found by just using Dijkstra's algorithm. The most important thing is that it satisfies the light splitting constraint.

Example 2: Assume a multicast session $ms_2\big(s : 8, D : (9 \sim 11)\big)$ is required in the NSF network, where

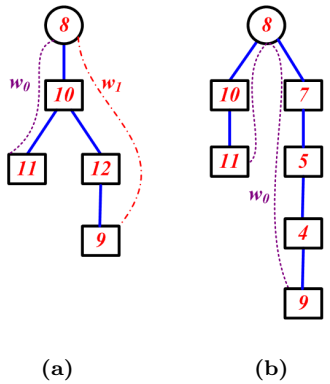

Fig. 3 For multicast session ms_2, (a) Light-tree built by Member-Only; (b) Light-tree built using Hypo-Steiner Heuristic.

only node 8 is an MC node. According to the Member-Only algorithm, node 10 is first added to the subtree, then node 11. At this moment, $MC\_SET = \{8, 11\}$, $MI\_SET = \{10\}$. Noting that shortest paths $SP(8-10-12-9)$, $SP(8-10-14-9)$ and $SP(11-10-9)$ involve non-leaf MI node 10, which belongs to MI_SET. Consequently, node 9 cannot be joined to the current subtree (using wavelength w_0), and a new light-tree on wavelength w_1 should be used as shown in Fig. 3(a). However, by implementing the Hypo-Steiner Heuristic, MI node 10, links $(8, 10)$ and $(10, 11)$ are deleted from graph G_1 to get G_2. It is interesting to find that the shortest path $SP_{G_2}(8-7-5-4-9)$ in the new graph G_2 is able to connect node 9 to the current subtree, which is the shortest constraint-satisfied path between them. So, only one wavelength w_0 is required as shown in Fig. 3(b).

3.2 Overview of Hypo-Steiner Light-tree Algorithm

Theorem 1 *Given MI_SET and MC_SET for a subtree LT under construction, if $\exists$ one or several paths $P(d, c)$ satisfying (a) $d \in D$ and $c \in MC\_SET$, (b) $\forall v \in P(d, c)$, $v \notin MI\_SET$, HSLT algorithm is able to span at least one more destination to LT, and the path used to connect a destination to LT is the shortest one among all the existing paths $P(d, c)$.*

Proof In order to get a new modified graph G_i, algorithm HSLT deletes all the nodes in MI_SET from the previous graph G_{i-1}, thus no path in G_i contains any non-leaf MI node in LT. If condition (b) $\forall v \in P(d, c), v \notin MI\_SET$ is satisfied, then no node in $P(d, c)$ will be deleted from the previous graph G_{i-1}. Thus the paths $P(d, c)$ will appear in the modified graph G_i. HSLT computes all the shortest paths from $d \in D$ and $c \in MC\_SET$ in the modified graph G_i. It is evident that the shortest one among the paths $P(d, c)$ will be

found. Then the nearest destination is connected to LT with the shortest path found. After that, HSLT will continue the span procedure of LT until blocked.

Theorem 2 *Member-Only will fail to span another destination in LT, if none of the paths $P(d,c)$ is the shortest path in the network.*

Proof If all the paths $P(d,c)$ satisfying condition (a) and (b) are not the shortest path in the original topology, it means all the shortest paths $SP(d,c)$ with $d \in D$ and $c \in MC\_SET$ in the original graph have the following problem

$$SP(d,c) \cap MI\_SET \neq \emptyset \tag{11}$$

As a result, Member-Only will block the construction of the current light-tree LT and continue another one to serve the destinations left in D (for instance Examples 1 and 2).

Through the above two theorems, we can see, under the same situation, HSLT has a bigger capability to span more destinations in one light-tree than Member-Only. This is why a HSLT light-tree is able to serve more destinations in most cases, especially for the first built light-tree. Once the number of destinations served in one light-tree increases, the number of light-trees required for a given multicast session will decline, which leads to the decrease of link stress. Since one multicast session requires fewer wavelengths, for a given number of wavelengths W supported per fiber link, more multicast sessions are reasonable likely to be established simultaneously. This may help to improve the network throughput in the scenario that the network traffic load is not too high.

Although many heuristics [1,11,13,16,17] have been proposed for the construction of light-trees under the sparse light splitting constraint, none of them discuss about the cost bound of the light-trees. Since it is NP-Hard to compute the light-trees with the minimum cost, the cost bound is an important quality of the light-tress. Next, we give the cost bound of the HSLT light-tress.

Theorem 3 *Given a multicast session $ms(s,D)$, the total cost of the light-trees built by HSLT algorithm is bounded to*

$$K = |D| \leq c\big(ms(s,D)\big) \leq \frac{N(N-1)}{2}. \tag{12}$$

Proof When finishing the first light-tree LT_1, the destinations left in $D \setminus D_1$ could not be included in LT_1, otherwise we don't have to span them in a second light-tree. For the same reason, after the j^{th} light-tree is built, the destinations left in $D \setminus \big(\bigcup_{i=1}^{j} D_i\big)$ could not be included in the j^{th} light-tree. Since equation (4), in the j^{th} light-tree, there are at most $N - \sum_{i=j+1}^{k} l_i$ nodes. Thus the cost of the j^{th} light-tree complies

$$c\big(LT_j(s,D_j)\big) \leq N - 1 - \sum_{i=j+1}^{k} l_i \tag{13}$$

Then, we obtain the total cost of the set of light-trees built for multicast session $ms(s,D)$

$$\begin{aligned} c\big(ms(s,D)\big) &= \sum_{i=1}^{k} c\big[LT_i(s,D_i)\big] \\ &\leq \sum_{i=1}^{k} \big(N - 1 - \sum_{i=j+1}^{k} l_i\big) \\ &= k(N-1) - \sum_{j=1}^{k}\sum_{i=j+1}^{k} l_i \end{aligned} \tag{14}$$

According to equation (6), it holds that

$$\sum_{i=1}^{k}\sum_{i=j+1}^{k} l_i \geq \frac{1}{2}k(k-1) \tag{15}$$

Thus,

$$\begin{aligned} c\big(ms(s,D)\big) &\leq k(N-1) - \frac{1}{2}k(k-1) \\ &= -\frac{1}{2}\Big[k - \big(N - \frac{1}{2}\big)\Big]^2 + \frac{1}{2}\big(N - \frac{1}{2}\big)^2 \end{aligned} \tag{16}$$

When $k = N-1$, we get the following inequality

$$c\big(ms(s,D)\big) \leq \frac{N(N-1)}{2}. \tag{17}$$

Moreover, in the j^{th} light-tree $LT_j(s,D_j)$, there are at least $l_j + 1$ nodes when the j^{th} light-tree includes no other nodes than the source node and the destinations in set D_j, i.e., $LT_j(s,D_j) = \{s\} \cup D_i$. Then, the cost of $LT_j(s,D_j)$ holds

$$c\big(LT_j(s,D_j)\big) \geq l_j \tag{18}$$

Thereby, we obtain the total cost of multicast session $ms(s,D)$,

$$c\big(ms(s,D)\big) \geq \sum_{j=1}^{k} l_i = K = |D| \tag{19}$$

3.3 Time Complexity of Hypo-Steiner Light-tree Algorithm

At every step, a new graph G_i is generated. In order to find the nearest destination to the subtree built, Dijkstra's algorithm should be used compute all the shortest paths in the new graph G_i from each destination

$d \in D$ to the connector nodes in MC_SET. So, the time complexity of each step is $|D|_i \times O(Dijkstra)$, where $|D|_i$ denotes the number of destinations left in set D at the i^{th} step. There are $K = |D|$ steps in total, and $|D|_i = K - i$. The most recent worst-case time complexity of Dijkstra is $O(NlogN + M)$ [15]. Therefore, the overall time complexity is

$$\frac{1}{2}K(K+1) \times O(Dijkstra) = O\big(K^2(NlogN + M)\big) \quad (20)$$

In fact, if we modify the Dijkstra's algorithm, the complexity could be reduced a lot. At each iteration, in order to find the nearest destination in set D, the connector nodes in MC_SET could be viewed ensemble as the source (set their distances to the source as 0), and thus only one shortest path tree is required in order to compute the distances and search the nearest destination. There are $K = |D|$ destinations in one multicast session, so Dijkstra's algorithm is only used for K times. Thus, the time complexity of the proposed algorithm could be diminished to

$$K \times O(Dijkstra) = O\big(K(NlogN + M)\big) \quad (21)$$

4 Simulation And Numerical Results

In this section, we use the simulation to evaluate the average case performance of the proposed Hypo-Steiner light-tree (HSLT) algorithm. It is compared with the existing algorithms like Reroute-to-Source (R2S) and Member-Only (MO) algorithm [1]. The latter one is currently shown to provide the best link stress and total cost.

4.1 Simulation Model and Evaluation Metrics

The USA Longhaul network (28 nodes and 43 links) in Fig. 4 is employed as platform for the simulation. Both the members of a multicast group and the MC nodes are selected uniformly and independently in the topology. For each network configuration (given a number of MC nodes and a group size), 10000 random multicast sessions are generated and the result is the average of 10000 computations. In the simulation, five performance metrics are considered:

- **Link Stress**. It is the number of wavelengths required per fiber, which equals to the number of light-trees k built for a multicast session.
- **The number of destinations included in the first light-tree**, with $|D_1| = l_1$. In most cases, with HSLT and MO two light-trees are sufficient to cover all the group members (Fig. 5). The destinations served by the second light-tree are always fewer. Thus only the first light-tree can signify the capacity of spanning destinations.
- **Total Cost**. It is the wavelength channels used in all the light-trees built for a multicast session $ms(s, D)$.
- **Throughput**. It is the number of multicast sessions accepted simultaneously for a given W.
- **Efficiency of Wavelength Usage**. It can be calculated as

$$Eff(\lambda) = \frac{1}{MW} \sum_{e \in E} \sum_{\lambda=1}^{W} e(\lambda) \quad (22)$$

, where $e(\lambda)$ is defined as the utilization state of wavelength λ on fiber link e

$$\forall e \in E, \ and \ \lambda \in [1, W],$$
$$e(\lambda) = \begin{cases} 1, \text{ if } \lambda \text{ is used on link } e \\ 0, \text{ if } \lambda \text{ is available on link } e \end{cases} \quad (23)$$

And we set $W = 20$ in our simulation.

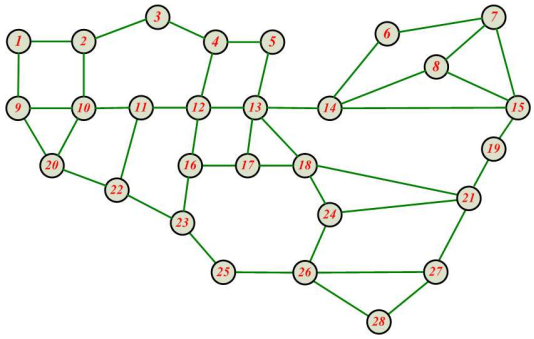

Fig. 4 USA Longhaul Network

4.2 Performance Analysis

The multicast session group size (including the source), $K+1$, is set to 7 and 14 respectively, which corresponds to the group member ratio of 25% and 50%, where

$$Ratio = \frac{(K+1)}{N}. \quad (24)$$

First, we compare the quality of different light-trees built by Reroute-to-Source (R2S), Member-Only (MO) and HSLT, respectively.

(1) The link stress of the three types of light-trees is plotted in Fig. 5. HSLT can always get the best link stress among the three algorithms. If there is no splitter, the HSLT algorithm is able to save 0.6 and 2.9 wavelengths for a multicast session instead of using the other two algorithms when Ratio=50%. The advantage of a HSLT light-tree becomes more evident as the group size grows, while it becomes less significant when the light splitting ratio grows.

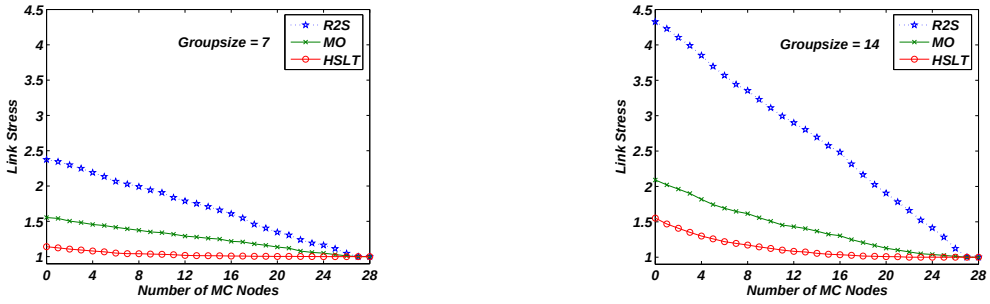

Fig. 5 Comparison of the Link Stress when (a) groupsize = 7; (b) groupsize = 14 in the USA Longhaul Network

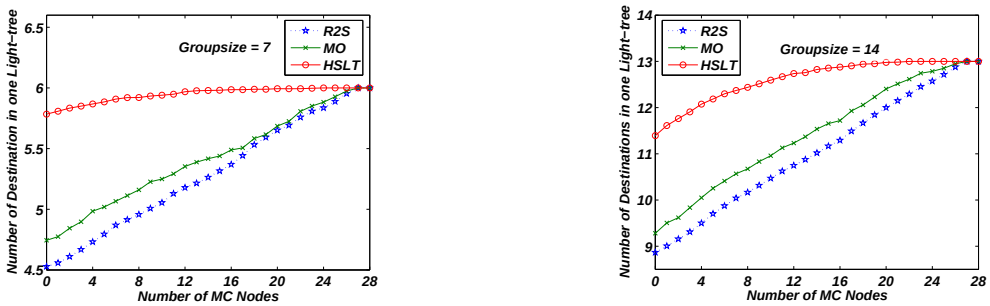

Fig. 6 Comparison of the Number of Destinations included in a light-tree when (a) groupsize = 7; (b) groupsize = 14 in the USA Longhaul Network

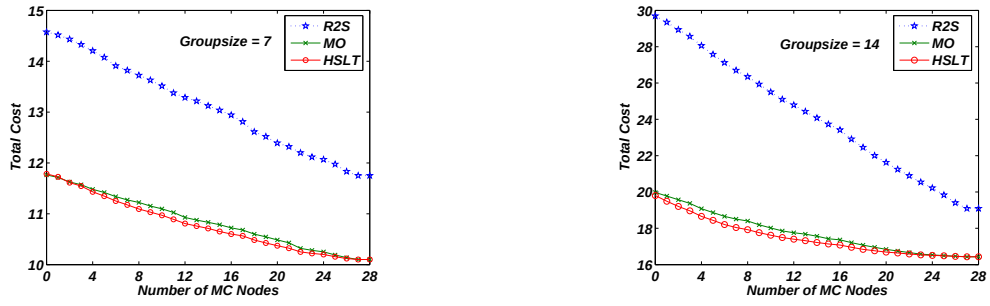

Fig. 7 Comparison of the Total Cost when (a) groupsize = 7; (b) groupsize = 14 in the USA Longhaul Network

(2) As plotted in Fig. 6(a) when group member Ratio = 25%, HSLT can span up to about 1.5 destinations (1.5/6 = 25%) more than R2S and one destination (1/6 = 17%) more than MO if there is no splitter. And as shown in Fig. 6(b) where Ratio = 50%, up to 2.5 (2.5/13 = 19%) and 2 (2/13 = 15%) more destinations could be included by a HSLT light-tree compared to R2S and MO, respectively. These results indicate that as the group size grows, more destinations could be spanned in a HSLT light-tree than that in the light-trees computed by the other two algorithms. The two figures also show that the advantage of a HSLT light-tree becomes significant as the number of MC nodes decreases.

(3) From the curves plotted in Fig. 7, HSLT has a slightly lower total cost than MO algorithm, which was supposed to achieve the best cost of all the current algorithms.

The above observations can be explained as follows. When the group size is big, in the modified graph more destinations could be joined to a HSLT light-tree via a hypo-shortest path while the other two algorithms can-

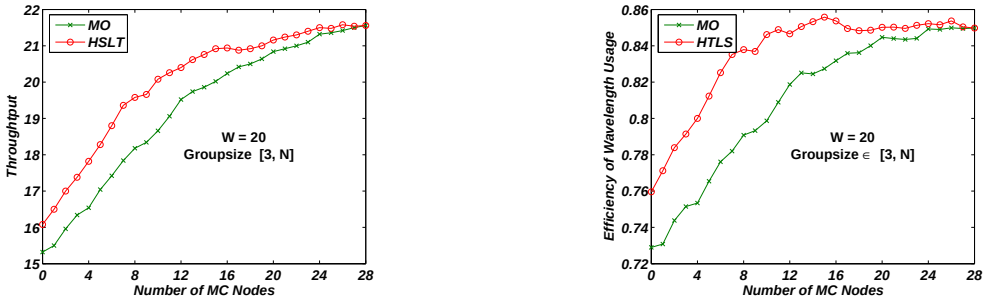

Fig. 8 Comparison of the (a) Throughput; (b) Efficiency of Wavelength Usage in the USA Longhaul Network

not because they use and only use the shortest path in the original topology. This is why the HSLT algorithm has a bigger capacity to include more destinations in a light-tree than the others. However, as the ratio of MC nodes increases, when building a light-tree, the size of set MC_SET grows while the size of set MI_SET reduces, thereby leading to the release of light splitting constraint. Hence, the numbers of destinations included in these three kinds of light-trees grow until they reach the same value. Hence, the HSLT light-tree is more advantageous in sparse light splitting WDM networks. As far as the link stress, the HSLT algorithm is able to achieve the best performance; since the more destinations could be spanned in a light-tree, the fewer light-trees will be required by a multicast session.

Secondly, we investigate the network throughput and efficiency of wavelength usage when the number of wavelengths supported in a fiber link is set to $W = 20$, group size $K + 1$ is a randomly variable uniformly generated from $[3, N]$, and the First-Fit [12] wavelengths assignment algorithm is employed. In Fig. 8(a), with the help of HSLT light-trees, up to 1.6 more multicast sessions could be accepted in the network, which indicates a 9% improvement of network capacity. And just because more wavelengths in fiber links could be used to establish some extra multicast sessions, the HSLT reaches high to 85% wavelengths usage as shown in Fig. 8(b).

5 Conclusion

Sparse light splitting constraint and the absence of wavelength converters complicate multicast routing in WDM mesh networks. To solve the light-trees construction problem for multicast routing, the Hypo-Steiner Heuristic is introduced to overcome the drawback of Member-Only algorithm [1], which only makes use of the shortest path in the original topology to connect a destination to a light-tree under construction. The Hypo-Steiner Heuristic always deletes the non-leaf MI nodes in the light-tree under construction from the topology, and searches a hypo-shortest path in the modified topology. At each step, after the pruning operation, there is no non-leaf MI node any more in the modified topology, thus the shortest path found (a hypo shortest path in the original topology) is always constraints-satisfied. With the Hypo-Steiner Heuristic, more possible paths could be found to connect a destination to the light-tree under construction. That is why more destinations could be spanned in a HSLT light-tree than Member-Only. This consequently results in the decrease of link stress, which equals to the number of light-trees required for a multicast session. Regarding the wavelength channel cost consumed by per multicast session, the total cost of light-trees built by Hypo-Steiner Heuristic is proved to be upper bounded to $N(N-1)/2$, where N is the number of nodes in the network. Extended simulations are also implemented. The numeric results not only validate the analysis proof but also show that the Hypo-Steiner Heuristic is more advantageous than the existing light-tree construction algorithms in sparse light splitting all-optical WDM mesh Networks.

References

1. Xijun Zhang, John Wei, Chunming Qiao. Constrained multicast routing in WDM networks with sparse light splitting. IEEE/OSA Journal of Lightware Technology, 18(12): 1917-1927, 2000.
2. J.S. Turner. New Directions in Communications (or Which Way to the Information Age?). IEEE Communications Magazine, 40(5): 50-57, 2002.
3. Laxman H. Sahasrabuddhe, Biswanath Mukherjee. Light-trees: optical multicasting for improved performance in wavelength-routed networks. IEEE Communications Magazine, 37(2): 67-73, 1999.
4. Biswanath Mukherjee. WDM optical communication networks: progress and challenges. IEEE Journal on Selected Areas in Communications, 18(10): 1810-1824, 2000.

5. Maher Ali, Jitender S. Deogun. Cost-effective implementation of multicasting in wavelength-routed networks. IEEE/OSA Journal of Lightwave Technology, 18(12): 1628-1638, 2000.
6. E. Desurvire (1991). Erbium-doped fiber amplifiers: principles and applications. New York, NY, USA: Wiley.
7. Shuguang Yan, Jitender S. Deogun, Maher Ali. Network routing in sparse splitting optical networks with multicast traffic. Computer Networks, 41: 89-113, 2003.
8. R. Malli, Xijun Zhang, Chunming Qiao. Benefit of multicasting in all-optical networks. SPIE Proceeding on All-Optical Networking, 2531: 209-220, 1998.
9. Xiaohua Jia, Dingzhu Du, Xiaodong Hu, Mankei Lee, Jun Gu. Optimization of wavelength assignment for QoS multicast in WDM networks. IEEE Transaction on Communications, 49(2): 341-350 2001.
10. H. Takahashi, A. Matsuyama. An approximate solution for the Steiner problem in graphs. Math. Japonica, 24(6): 573-577, 1980.
11. Ashraf Hamad, Tao Wu, Ahmed E. Kamal, Arun K. Somani. On multicasting in wavelength-routing mesh networks. Computer Networks, 50: 3105-3164, 2006.
12. Yingzhou Zhu, G.N. Rouskas, and H.G. Perros. A comparison of allocation policies in wavelength routing networks. Photonic Network Communications, 2(3): 265-293, 2000.
13. Fen Zhou, Miklós Molnár, Bernard Cousin. Avoidance of multicast incapable branching nodes for multicast routing in wdm networks. Photonic Network Communications, to appear, 2009.
14. Fen Zhou, Miklós Molnár, Bernard Cousin. Is light-tree structure optimal for multicast routing in sparse light splitting WDM networks? The 18^{th} International Conference on Computer Communications and Networks, San Francisco, USA, August 2, 2009.
15. Michael Barbehenn. A note on the complexity of Dijkstra's algorithm for graphs with weighted vertices. IEEE Transaction on Computers, 47(2): 263, 1998.
16. N. Sreenath, K. Satheesh, G. Mohan, C. Siva Ram Murthy. Virtual source based multicast routing in wdm optical networks. Photonic Network Communications, 3(3): 213-226, 2001.
17. Fen Zhou, Miklós Molnár, Bernard Cousin. Distance priority based multicast routing in WDM networks considering sparse light splitting. The 11^{th} IEEE International Conference on Communication System, pp709-714, Guangzhou, China, November, 2008.