Protecting Multicast Sessions in WDM Networks with Sparse-Splitting Constraints

Xiong Wang, Sheng Wang, and Lemin Li

ABSTRACT—In this letter, we study the multicast protection problem in sparse-splitting wavelength-division multiplexing (WDM) optical network, and propose a novel multicast protection algorithm called the shared source-leaf path-based protection (SLPP) algorithm. Unlike the proposals in previous studies, the backup paths derived by SLPP can share wavelength with the primary tree in sparse-splitting WDM networks. Simulations are used to evaluate the effectiveness of the SLPP algorithm.

Keywords—WDM networks, sparse splitting, wavelength continuity, multicast, light-tree, light-forest.

I. Introduction

In order to realize all-optical multicasting efficiently, nodes in a wavelength-division multiplexing (WDM) network need to have light-splitting capability. A node capable of splitting light can forward an incoming message to multiple output channels, and, therefore, is multicast capable (MC). An MC node, however, is expensive to implement. The concept of sparse splitting was first introduced in [1]. With sparse splitting, only a small percentage of nodes in the network are MC; the rest are multicast incapable (MI). Due to the immaturity of all-optical wavelength conversion technology, we also consider the wavelength continuity constraint in this paper.

Survivability is an important issue in WDM networks [2]. Protecting a multicast session from a single link failure in WDM networks with or without a sparse-splitting constraint has been studied in recent literature [2]-[6]. Existing multicast protection algorithms can be classified into the following four types: link-disjoint tree [2], [3]; link protection [4]; segment protection [2], [5]; and path-based [4], [6] algorithms. However, with the sparse-splitting constraint, the backup paths derived by existing algorithms cannot share wavelength with the working tree. For example, in Fig. 1(a), the backup path (s–u–d2) for destination d2 cannot share the wavelength with the primary tree on link (s–u), because node u is an MI node and cannot split a signal to destination d2 when link (s–u) fails. Furthermore, without wavelength converters, some of the current algorithms cannot even find a link-disjoint backup path. For example, as shown in Fig. 1(a), the segment protection algorithm cannot find a backup path for segment (s–d1) if there is no wavelength converter available because the links used by the primary tree must be removed when backup paths are computed. Actually, we can choose path (s–u–v–d1) as the backup path for segment (s–d1), and the backup path can share wavelengths with the primary tree on links (s–u) and (u–v).

In this work, we propose a novel multicast protection algorithm, called the source-leaf path-based protection (SLPP) algorithm, to enhance wavelength sharing between the primary tree and backup paths in a sparse-splitting network. Simulations prove that SLPP achieves a higher degree of resource sharing than existing schemes, and this is the main contribution of this work.

II. The Proposed Algorithm

1. Problem Description

We define a physical topology $G(N, E, W)$ for a given WDM mesh network, where N is the set of nodes, E is the set of fiber links, and W is the set of available wavelengths per fiber link. Let N_{MC} and N_{MI} denote the MC and MI node sets, respectively, and

Manuscript received Jan. 18, 2007; revised Mar. 12, 2007.

This work was supported by the National Natural Science Foundation of China (NSFC) under the grant number 60472008.

Xiong Wang (phone: + 86 28 83201113, email: wangxiong@uestc.edu.cn), Sheng Wang (email: wsh_keylab@uestc.edu.cn), and Lemin Li (email: lml@uestc.edu.cn) are with the Key Lab of Broadband Optical Fiber Transmission and Communication Networks, the University of Electronic Science and Technology of China, Chengdu, China.

$N = N_{MC} \cup N_{MI}$. We assume that all MI nodes are equipped with tap-and-continue (TaC) devices, which can tap a small fraction of input optical power for the local station while switching the remaining power to any one of the outputs. None of the network nodes have wavelength converters. The weight function $c: E \rightarrow R^{+}$ represents the network cost of using a particular edge.

A multicast session is represented by (s, D), where s is the source node from which data is sent to a set of destination nodes D. With the sparse-splitting constraint, several light-trees (light-forest) may be needed to multicast data to all the destinations [1]. We assume that the primary tree set F (light-forest) has been derived for the multicast session using existing approaches, such as the member-only algorithm proposed in [1]. Our objective is to find the minimum cost backup topology for each light-tree in F by enhancing wavelength sharing between the primary tree and backup paths.

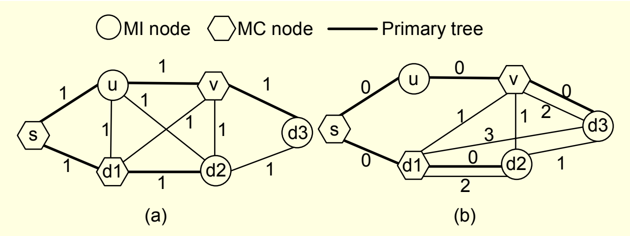

Fig. 1. (a) Sample network and a light-tree and (b) auxiliary graph for the light-tree.

2. Overview of SLPP Algorithm

The SLPP algorithm finds a link-disjoint backup path for each leaf node on the primary tree. The destinations on the primary path from the source to a certain leaf node can be protected by the circle formed by the primary and backup paths of the leaf node. For example, in Fig. 1(a), the backup path for leaf node d2 is (s–u–v–d2), and destination nodes d1 and d2 can be protected by circle (s–u–v–d2–d1–s). In order to share wavelength with the primary tree, SLPP chooses path (s–u–v–d2) instead of the shortest path, (s–u–d2), as the backup path for leaf node d2. This is because the backup path (s–u–d2) cannot share the primary wavelength on link (s–u), while the backup path (s–u–v–d2) can share the primary wavelengths on links (s–u) and (s–v).

To achieve wavelength sharing, the SLPP algorithm first constructs an auxiliary graph AG for each light-tree T in F as follows: 1) Let AG initially equal T. 2) Put the source node, the leaf nodes, and the MC nodes involved in T to B. Delete the links involved in T from G. 3) For nodes $u \in B$ and $v \in B$, add an edge between them in AG if there is a shortest path between u and v in G. An auxiliary graph for the light-tree in Fig. 1(a) is shown in Fig. 2(b). After constructing the auxiliary graph AG, find a link-disjoint backup path for each leaf node in AG. As shown in Fig. 2(b), the derived backup paths for the leaf nodes d2 and d3 are (s–u–v–d2) and (s–d1–d2–d3), respectively.

2. Procedure of SLPP Algorithm

As stated above, the SLPP algorithm can be divided into two phases. In the first phase, an auxiliary graph AG is generated. In the second phase, the backup paths for leaves are derived in AG The details of the SLPP algorithm are as follows:

Input. Network $G(V, E, W)$, primary light-forest F, and multicast session (s, D)

Output. Backup topology BT used to protect the primary light-forest F.

Phase 1. Construct auxiliary graph AG.

Step 1. Get light-tree T from F. Let AG=T, and set the cost of links in AG to zero. Let LD be the set of leaf nodes on tree T, and let M denote the set of MC nodes involved in T. $B = s \cup LD \cup M$.

Step 2. Set the link $e \in T$ to infinite in G.

Step 3. For each $u \in B$, compute the shortest path $p(u, v)$ from u to all other nodes in G. If there is a shortest path $p(v, u)$ between $u \in B$ and $v \in B$, add an edge between u and v in AG, and set the cost of the edge to the cost of the shortest path.

Phase 2. Find the backup path for each leaf node in AG.

Step 4. Get leaf node l from set LD, and let $pp(s, l)$ denote the primary path from s to l in T.

Step 5. Set the cost of every unmarked link $e \in pp(s, l)$ to infinite in AG.

Step 6. Find a shortest backup path $bp(s, l)$ in AG. Add every link represented by $e \in bp(s, l)$ to BL, and mark every link $e \in pp(s, l)$ in AG.

Step 7. Set the cost of every link $e \in pp(s,l) \cup bp(s,l)$ to zero in AG, and remove l from LD.

Step 8. If $LD = \phi$, go to step 9; otherwise, go to step 4;

Step 9. Remove T from F. If $F = \phi$, return BL; otherwise, go to step 1.

The time complexity of SLPP mostly depends on the running times of Dijkstra's algorithm, whose time complexity is $O(|N|^2)$. In the worst case, the complexity of constructing the auxiliary graph in phase 1 is $O(|S||D||N|^2)$, where $|S|$ and $|D|$ are the number of MC nodes and destination nodes, respectively, and the total complexity of phase 2 is $O(|D||N|^2)$. Thus, the overall complexity of the algorithm is $O(|S||D||N|^2)$.

III. Numerical Results

We conducted simulations to study the performance of the SLPP algorithm. We compared our proposed algorithm with the disjoint-segments (DS) and source-destination path pairs (SDPP) [2] algorithms. We did not compare our algorithm with the algorithm proposed in [6], because it cannot be used in WDM networks without wavelength converters. In our simulations, we assume the following: 1) The 24-node USA national network shown in [4] is used. 2) Each link in the network carries 64 wavelengths, and the link cost of each link is 1. 3) The MC nodes are placed on the network in decreasing order of degree of nodes, and the number of MC nodes is 8. None of the nodes in the network have wavelength converters. 4) Multicast sessions arrive with Poisson distribution, and their holding time is negative exponentially distributed. 5) Multicast sessions are uniformly distributed among all nodes. 6) The MO algorithm [1] is used to derive the initial light-trees. With the above assumptions, we injected 10^5 randomly generated multicast requests with destination size k into the sample network, and compared the average network cost and the blocking probability of the above algorithms. The average network cost is defined as the average sum of the wavelength link cost used by the multicast sessions.

Figure 2(a) shows the average network cost consumed by a single multicast session when the session size varies from 1 to 12. From this figure, we immediately see that the SLPP algorithm consumes the least network cost, and the SDPP algorithm consumes the most network cost. This is because the protection paths derived by SLPP can share wavelengths with the working tree and existing protection paths, whereas a backup path derived by SDPP cannot share wavelengths with either the primary path or backup paths for other destinations.

In Fig. 2(b), we further compare the blocking probability of the three algorithms when the session size varies from 1 to 23 and the traffic load is fixed to 80. As expected, the blocking probability of the SLPP algorithm is far lower than that of DS and SDPP.

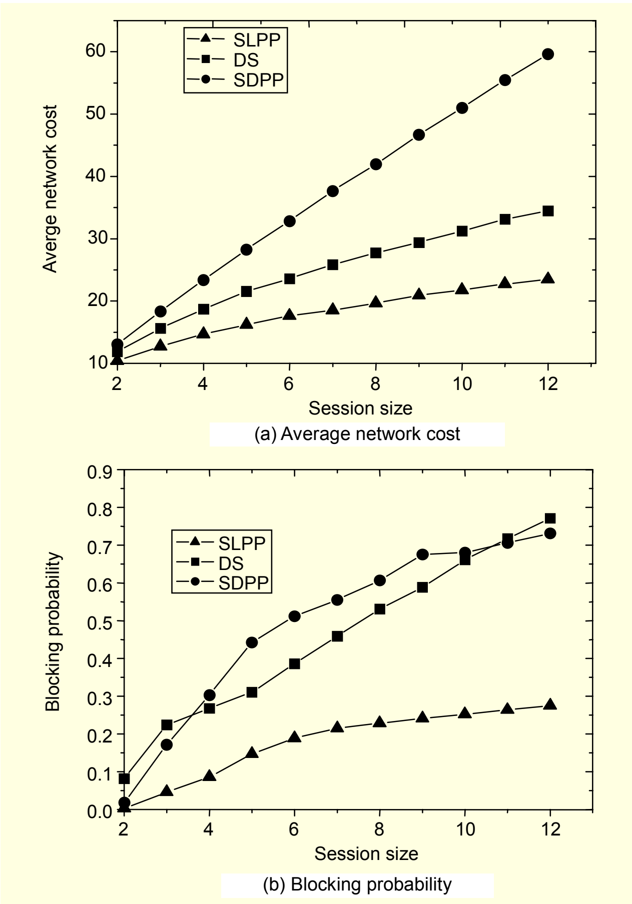

Fig. 2. Network performance vs. session size with traffic load 80 Erlang.

IV. Conclusion

In this letter, we proposed a multicast protection algorithm called SLPP, which considers sparse-splitting and wavelength continuity constraints. The backup paths derived by SLPP can share wavelength with the primary tree. Simulation results show that SLPP yields better results than previous algorithms.

References

[1] X.J. Zhang, J.Y. Wei, and Chunming Qiao, "Constrained Multicast Routing in WDM Networks with Sparse Light Splitting," *Journal of Lightwave Technology*, vol. 18, no. 12, 2000, pp. 1917-1927.

[2] N. Singhal, L. Sahasrabuddhe, and B. Mukherjee, "Provisioning of Survivable Multicast Sessions Against Single Link Failures in Optical WDM Mesh Networks," *IEEE/OSA Journal of Lightwave Technology*, vol. 21, no. 11, Nov. 2003, pp. 2587-2594.

[3] T. Rahman and G. Ellinas, "Protection of Multicast Sessions in WDM Mesh Optical Networks," *Proc. OFC*, 2005, pp. 870-872.

[4] Y. Qing and G. Ning, "Protecting Dynamic Multicast Sessions in Optical WDM Mesh Network," *Proc. ICICS*, 2005, pp. 1187-1190.

[5] C. Lu, S. Wang, and L. Li, "Novel Shared Segment Protection Algorithm for Multicast Session in Mesh WDM Networks," *ETRI Journal*, vol. 28, no. 3, 2006, pp. 329-336.

[6] N. Sreenath and T.S. Prasad, "Protecting Multicast Sessions from Link and Node Failures in Sparse-Splitting WDM Networks," *Proc. IWDC*, 2005, pp. 195-200.