

# Logical Topology Design for IP-over-WDM Networks: A Hybrid Approach for Minimum Protection Capacity

Muhammad S. Javed and Krishnaiyan Thulasiraman  
School of Computer Science  
University of Oklahoma  
Norman, OK, USA  
{javed, thulasi}@ou.edu

Guoliang (Larry) Xue  
Department of Computer Science and Engineering  
Arizona State University  
Tempe, AZ, USA  
xue@asu.edu

**Abstract**—The problem of designing high capacity and high bit rate IP-over-WDM networks, which can provide uninterrupted service in the presence of network equipment failures, continues to attract significant interest from the research community. An IP-over-WDM network implements Internet Protocol (IP) directly over physical WDM network by establishing *lightpaths* using IP routers, optical crossconnects (OXC) and optical fibers. Generally an optical fiber carries several lightpaths and all of them get disconnected, if the fiber carrying them fails. Such failures can quickly impact the performance of the entire network. If IP routers can find paths to all the nodes in the network, then the network can continue to provide service without significant performance degradation. This can be achieved by reserving network resources (protection) or provisioning the network with some additional capacity (restoration). Such networks are usually called *survivable networks*. In this paper, we propose four algorithms based on SMART framework proposed by Kurant and Thiran, and a hybrid approach by Shenai and Sivalingam. The algorithms use a combination of protection and restoration mechanisms to make IP-over-WDM networks survivable such that the protection capacity required is not significant.

**Keywords;** *Disjoint paths; IP-over-WDM networks; lightpath routing; survivable networks.*

## I. INTRODUCTION

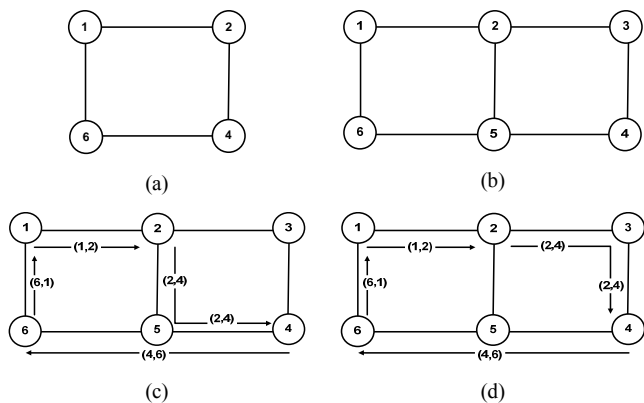
Optical fibers are the transmission medium of choice for long haul telecommunication networks due to their high capacity, low error rate, and cost [1]. By employing state of the art technologies, such as Wavelength Division Multiplexing (WDM) that allows simultaneous transmission of several signals through a single fiber, aggregate bandwidth of several Terabits/sec can be achieved [2]. In the past, the electrical bottleneck limited the throughput to rates that were compatible with the electronic circuitry of the switching/routing and amplifying equipment used in the network. However, the development of all optical amplifiers and crossconnects (OXC) has removed the electrical bottleneck and has made terabits/sec networks a reality. By combining WDM with all optical amplifiers, optical crossconnects and switching from the traditional point-to-point and ring architectures to more complex mesh architectures, high bit rate networks that can

carry a wide variety of data and multimedia traffic have become a possibility.

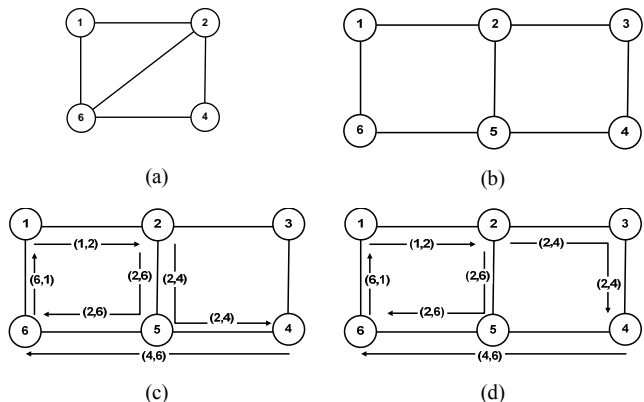
Most of the end user communications today are based on the TCP/IP protocol, therefore, the idea of implementing IP protocol directly over the optical network using optical crossconnects and IP router has gained significant traction among the research community. Such networks are generally referred to as IP-over-WDM networks, and involve *embedding* an IP network into a physical WDM network [2]. In literature, it is common to refer to the IP network as the logical topology, IP links as logical links and IP nodes (switches, routers etc.) as logical nodes. Similarly, the physical WDM network is generally called the physical topology, WDM links (optical fiber cables) are known as physical links and WDM nodes (OXC, optical add-drop multiplexor etc.) as physical nodes. It is also common to assume that a logical node is also a node in the physical topology. A logical topology is embedded into a physical topology by finding a *lightpath* for every IP link (logical link) in the physical topology [3]. A lightpath is an all optical path, which is established by allocating a wavelength between the source and the destination of a logical link in the physical topology. A lightpath, once established, may traverse multiple physical links but does not require any buffering and processing at intermediate nodes. The embedding of a logical link as a lightpath in the physical topology is also called a *mapping* [4].

In an IP-over-WDM network, a single fiber generally carries several lightpaths simultaneously and usually several fibers are bundled together to form a cable. Therefore, a cable cut can disrupt all the lightpaths passing through the cable and degrade the network performance significantly, if the failure persists. Unfortunately, cable cuts have become a common occurrence due to human or natural events, drawing considerable attention to designing networks that can provide uninterrupted service in the case of fiber or other network equipment failures. Such networks are generally called *survivable networks* [5].

A failure of physical link/links or node/nodes may disconnect one or more lightpaths and brings down the corresponding logical link/links. However, an IP-over-WDM network can continue to operate in the presence of such



**Figure 1. Illustration of mapping and survivability for ring networks [7]. (a) A logical topology (b) A physical topology (c) An unsurvivable mapping (d) A survivable mapping.**



**Figure 2. Illustration of mapping and survivability for general networks [7]. (a) A logical topology (b) A physical topology (c) An unsurvivable mapping (d) A survivable mapping.**

failure/failures, if the logical topology remains connected after the failure. A connected topology allows the IP routers to find alternate routes for the affected lightpaths. A mapping of the logical links in the physical network that remains connected after the failure of physical link/links is called *link survivable mapping* [5]. Similarly, a *node survivable mapping* is a mapping that stays connected after the failure of physical node/nodes [5]. In this paper, we focus only on one link survivable mappings, mappings that keep the logical topology connected after the failure of a single physical link.

*Protection* and *restoration* are the two widely discussed mechanisms for providing survivability in IP-over-WDM networks [6]. Protection in IP-over-WDM networks is generally provided at the physical layer at the design stage. First *primary* or *working lightpaths* are found for the logical links and then *backup* or *protection lightpaths* are calculated that do not use physical links (or nodes) already assigned to the primary lightpaths (i.e. their mappings are disjoint) [1]. In case of a failure, the network traffic carried by a primary lightpath is always switched to its corresponding backup lightpath [1].

Restoration is generally provided at the logical layer by provisioning the network with some additional capacity, which can be utilized by the IP routers to find backup paths after a failure. It is possible to find backup paths only if the logical links are mapped onto physical topology in such a way that the

logical topology remains connected after the failure [4]. This can be achieved by requiring at least two-connected logical and physical topologies and finding link/node disjoint mappings for all (Fig. 1) or some logical links (Fig. 2) [5].

Fig. 1(a) and 1(b) show a ring logical topology and a general physical topology, respectively. Fig. 1 (c) shows an unsurvivable mapping of this ring in the physical topology. It can be observed that the mappings for all the logical links are not disjoint. If the physical link (4, 5) fails then logical links (2, 4) and (4, 6) fail and the logical topology gets disconnected. Fig. 1(d) shows a mapping that is survivable and it can be observed that all the mappings are disjoint. Fig. 2(a) shows a general logical topology and Fig. 2(c) shows an unsurvivable mapping of this logical topology. In this case all the mappings are not disjoint and the logical topology is not survivable e.g. the failure of physical link (4, 5) disconnects the logical topology. Fig. 2(d) shows a survivable mapping. In this case, it can be seen that all the mappings are not disjoint and a physical link failure may disconnect multiple logical links but the logical topology remains connected. For example, if the physical link (5, 6) fails, logical links (2, 6) and (4, 6) get disconnected but it is possible to reach all the logical nodes.

The above example demonstrates some of the difficulties involved in finding survivable mappings. In fact, the problem of finding survivable mapping is shown to be NP-complete by Modiano and Narula-Tam for general logical topologies as well as for ring logical topologies [8]. In this paper, we briefly discuss the existing approaches and propose several new heuristics that can find survivable mappings for all the logical topologies using a combination of protection and restoration strategies. The paper is organized as follows; section II formally defines the problem of finding survivable mappings, section III briefly discusses the existing literature, section IV discusses the new approach and section V presents the simulation model and results.

## II. FORMAL PROBLEM DESCRIPTION

Assume that we are given a two edge-connected (a topology that remains connected after the removal of an edge) physical topology  $G(N, E)$  where  $N$  is the set of physical nodes (vertices), and  $E$  is the set of physical edges (links). A physical edge  $e$  is a pair of terminals,  $(i, j)$ , such that  $i, j \in N$ . Each physical link  $e \in E$  has an associated non-negative cost  $c_e$  and capacity  $u_e$ . A logical topology is defined by a set of source-destination  $(s, t)$  pairs where  $s, t \in N$ , such that together they form a two edge-connected undirected topology  $G^L(N^L, E^L)$ . Let  $K$  be the set of all such  $(s, t)$  pairs i.e.  $K = \{(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)\}$ . Also associated with each  $(s_k, t_k) \in K$  is a non-negative demand  $d_k$ , which represents the units of traffic that must be sent from  $s_k$  to  $t_k$  in  $G$  without violating the capacities of all  $e \in E$ .

A *mapping*  $M_k$  of a pair  $(s_k, t_k) \in K$  is a path  $P_k$  that connects  $s_k$  to  $t_k$  in  $G$ . Let  $A$  be a set of source-destination pairs such that  $A \subseteq K$ , then  $M_A$  is a set of mappings for source-destination pairs in  $A$ . Two mappings  $M_1$  and  $M_2$  are disjoint, if and only if  $\neg \exists e (e \in M_1 \wedge e \in M_2)$ . This definition can be easily extended to a set of mappings ( $M_A$ ). These mappings/paths will

be referred to as mutually disjoint mappings/paths. In contrast, the term two-disjoint mappings/paths will be used to refer to two disjoint mappings/paths between a single source-destination pair  $(s_k, t_k) \in K$ , and will be denoted by  $M_k'$  and  $M_k''$ .

Given  $G$  and  $K$ , the problem is to find mappings for some or all  $(s_k, t_k) \in K$  in  $G$  such that the removal of an edge  $e \in E$  does not disconnect the logical topology  $G^L$ . Such a mapping is called *link-survivable mapping*. If  $G^L$  is a ring, then the problem of finding link-survivable mappings is equivalent to finding disjoint paths for all the  $(s_k, t_k) \in K$ . However, in the case of a general topology, disjoint paths must be found for only a subset of  $(s_k, t_k) \in K$ .

### III. LITERATURE REVIEW

A simple approach to make a two-connected IP-over-WDM network survivable is to establish two link disjoint lightpaths in the physical topology for each logical link. However, this requires significant reservation of resources. Therefore, most of the approaches in the literature concentrate on the model described in section II i.e. (i) find mutually disjoint paths for some or all of the logical links such the logical topology remains connected after a physical link failure (ii) provision the physical network with some spare capacity that can be used to reroute the traffic carried by the failed physical link. Given the NP-complete nature of the problem, most of the approaches use ILP formulations. ILP formulations do not scale well when problem size grows, therefore these approaches also provide heuristics that relax some of the ILP constraints or exploit certain characteristics of the topology under consideration.

Ref. [4] provides a comprehensive framework under the protection interoperable design paradigm. [4] defines a protected group with protection level  $k$ , as a group of logical links that can stay connected in the presence of  $k$  logical link failures and provides three constraints that must be respected by a solution (i) *the capacity constraint* requires that a solution must respect the capacity constraints of the physical links (ii) *the bottleneck constraint* states that a physical link/node failure should not cause more than  $k$  logical link failures (iii) *the connectivity constraint* requires that the logical topology must remain connected for any single link/node failure in the physical topology. [4] acknowledges that the problem is NP-complete and provides a Tabu search based heuristic.

Ref. [8] provides an ILP formulation based on the observation that a single physical link failure can disconnect the logical topology if and only if it carries the entire cutset of the logical topology. A feasible ILP solution implies that a survivable mapping exists, but the formulation requires examination of all the cutsets. [8] notes that when the logical topology is a cycle (ring), the problem is reduced to finding disjoint mapping for the IP links. [8] also proves that the ILP formulations are NP-complete and provides fast heuristics that consider only a subset of all the cutsets.

Ref. [9] and [10] are based on observations in [8]. [9] does not evaluate all the cutsets but uses a probability function as an estimate of cutsets. [10] applies the work in [8] to Shared Risk Link Groups (SRLG, a set of links that share the risk of failing

simultaneously) and incorporates wavelength assignment in the ILP. [10] also proves that only primary cutsets should be considered to design survivable networks, which can significantly improve the performance specially when applied to hierarchical cyclic graphs.

Ref. [11] suggests a hybrid approach that uses a combination of restoration and protection. [11] suggests providing protection for *critical* links and restoration for the remaining links. The criticality of a link depends upon the ratio of number of lightpaths using a particular link to the total number of wavelengths supported by the link

In this paper, we propose a hybrid (protection and restoration) approach to provide survivability in IP-over-WDM networks based on the SMART framework proposed by Kurant and Thiran [5, 13]. The proposed approach can find survivable mappings for all the logical topologies using a combination of restoration and protection.

### IV. SMART FRAMEWORK

In [5] Kurant and Thiran propose a framework called SMART (Survivable Mapping Algorithm by Ring Trimming), which finds a survivable mapping for a logical topology by repeatedly considering only *pieces* (subgraphs) of the logical topology. The SMART does not impose capacity constraints and assumes full wavelength conversion.

SMART takes a logical topology and a physical topology as input, and proceeds by picking subgraphs in the logical topology and finding disjoint mapping for them. If disjoint mapping are found for a subgraph, the algorithm *contracts* such a subgraph by collapsing the links and merging the nodes in the piece (Fig. 3). The logical topology after contraction of a subgraph is called the *contracted logical topology* or simply *contracted topology*. SMART then proceeds by picking another subgraph in the contracted topology until the contracted logical topology is reduced to a single node or a survivable mapping could not be found for all the subgraphs possible in the contracted topology.

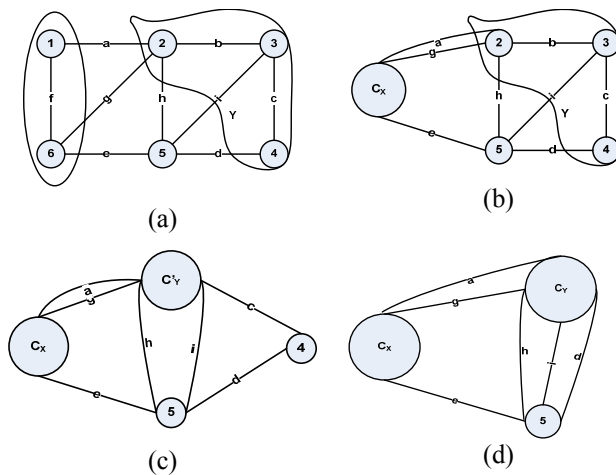


Figure 3. The concept of contraction of an edge and a subgraph (a) Subgraphs to be contracted (b) Logical topology after contracting link  $f$  (1-6) (c) Logical topology after contracting links  $b$ ,  $c$  and  $f$  ((1-6), (2-3), and (3-4)) (d) Logical topology after contracting links  $f$ ,  $b$ , and  $c$  ((1-6), (2-3), and (3-4)).

Two-connectivity is required for logical and physical topologies to provide survivability, therefore, the pieces considered in each step should be at least two-connected also. This can be easily achieved by selecting rings (cycles) in each iteration of SMART. However, even if the subgraph selection is restricted to cycles only, an undirected complete graph has  $\frac{1}{2} \sum_{i=3}^{|N|} \binom{|N|}{i} (i-1)!$  cycles, a number that grows faster with  $|N|$  than the exponential  $2^{|N|}$  [12]. Therefore, for all practical purposes SMART must be restricted to considering a small subset of such subgraphs. Also, the algorithm does not keep track of the subgraphs already considered and may revisit them.

Ref [5] also provides a heuristic version of SMART (called SMART-H), which only picks rings as subgraphs and uses a simple heuristic to find disjoint mappings [5, 13]. The heuristic assigns a weight of 1 to all physical links and applies a shortest path algorithm to find mappings for all the links in the subgraph using this weight. The mappings, if disjoint, are accepted as the mapping for the subgraph and the subgraph is contracted. Otherwise, the weights of links common to the mappings are incremented by 1 and the disjoint path algorithm is applied again. The procedure is repeated until disjoint mappings are found or a termination condition is met. If disjoint mappings cannot be found for a particular subgraph, the algorithm proceeds by picking another subgraph. After unsuccessfully examining a certain number of subgraphs, SMART-H terminates and declares logical-physical topology pair unsurvivable.

If SMART-H terminates successfully i.e. the logical topology is reduced to a single node, then a survivable mapping for the logical topology has been found. Otherwise, either a survivable mapping does not exist for the entire logical topology or the SMART-H terminated prematurely. In either case, SMART-H returns some survivable pieces and the remaining contracted topology that can be processed further by an appropriate method. In [7], we have explored a more sophisticated heuristic to find subgraphs and mappings, which can find survivable mappings for a large number of topologies.

SMART also proposes that adding a logical or a physical link after the unsuccessful termination of SMART and reapplying SMART may make a logical topology survivable. The additional link is added by picking two nodes in the contracted logical topology and connecting them in the logical or the physical topology. However, depending upon the number of remaining nodes there could be a lot of choices available. In this paper, we provide a new approach to make an unsurvivable contracted logical topology survivable using a combination of restoration and protection. The new approach is discussed in the next section.

## V. THE NEW APPROACH

As mentioned earlier, it is possible to formulate the problem of finding survivable mapping for a logical topology in a physical topology as an ILP and solve optimally [8]. In some cases, survivable mappings for some logical-physical topology pairs may not exist and must be rejected by the network operator, which would result in loss of revenues.

Another option is to add logical or physical links, which can make the logical-physical topology pair survivable [5]. Adding physical links by laying new fibers is usually very expensive and in some cases impossible. On the other hand, logical links can be easily added by establishing extra lightpaths in the physical topology. Given the number of choices available, the nodes between which additional lightpaths must be established may be very difficult to determine. Also deciding the number of links to be added to make the logical-physical topology pair survivable may not be known. Therefore, we propose a new approach in this section that utilizes concepts from [5] and [11]. The new approach proposes a combination of restoration and protection to find survivable mapping for all the logical topologies.

Given a cycle  $C$  and a set of mappings for the links in  $C$ , let *critical links* ( $\pi$ ) be a set of logical links which when mapped together make the logical topology unsurvivable. For example, in Fig. 4 logical links (3, 2) and (4, 1) are critical links because their mappings are not link disjoint. Failure of physical link (3, 2) can disconnect logical links (3, 2) and (4, 1) and the logical topology is no longer connected i.e.  $\pi = \{(3, 2), (4, 1)\}$ . Therefore we propose that for such links protection should be provided. Protection can be easily provided by finding two link disjoint paths for all or some of the critical links. Two disjoint paths are easier to find using a max flow algorithm, Suurballe algorithm or any other appropriate algorithm [14]. In contrast to finding mutually disjoint mappings (NP complete), finding two disjoint paths in a two connected topology is fairly simple.

Fig. 4(c) illustrates the new approach. Since  $\pi = \{(3, 2), (4, 1)\}$ , protection can be provided for (3, 2) or (4, 1) or both. Fig. 4(c) shows the mappings when protection is provided for (3, 2) by setting up two lightpaths rather than one. Now it can be seen that the failure of any single physical link will not disconnect the logical topology. For example, if physical link (3, 2) fails lightpaths (4, 1) and (3, 2) get disconnected but logical link (3, 2) remains connected through the alternate lightpath for (3, 2). Hence the entire logical topology remains connected. Fig. 4 also illustrates that there is no need to provide protection for (4, 1), since routing (4, 1) and (3, 2) together no longer makes the topology unsurvivable. In fact, if there are  $q$  critical links then we have to provide protection for only  $q - 1$  critical links in the worst case.

Finding an optimal set of critical links in a topology may not be possible. Therefore, we propose utilizing SMART-H as a heuristic to find such links. The set of critical links obtained using SMART-H may not be optimal i.e. some of these links may not be critical. Some of these links may have been declared critical because the heuristic used by SMART-H to find disjoint mappings terminated unsuccessfully, although disjoint mappings were possible using an ILP or some other algorithm. Therefore, we call the set of links deemed critical by SMART-H as *pseudo-critical links* ( $\eta$ ).

To find pseudo-critical links using SMART-H, a simple modification to SMART-H is made by introducing a new variable, *best mapping* ( $M_{Best}$ ).  $M_{Best}$  is a set of mappings that has the fewest number of non-disjoint mappings. The logical links that have non-disjoint mappings are the pseudo-critical links. Also, let  $|M_{Best}|$  be the number of links in the chosen cycle

TABLE I. NO. OF SURVIVABLE LOGICAL-PHYSICAL TOPOLOGY PAIRS

Method	Physical degree = 3 Logical degree = 2.5			Physical degree = 3 Logical degree = 3		
	No. of Physical Nodes					
	100	200	300	100	200	300
SMART-H	392	390	430	835	855	785
SMART-H+1	666	695	725	925	965	950
HA-1	1000	1000	1000	1000	1000	1000
HA-2	1000	1000	1000	1000	1000	1000
HA-3	1000	1000	1000	1000	1000	1000
HA-4	1000	1000	1000	1000	1000	1000

TABLE II. AVERAGE NUMBER OF PROTECTED LOGICAL LINKS

Method	Physical degree = 3 Logical degree = 2.5			Physical degree = 3 Logical degree = 3		
	No. of Physical Nodes					
	100	200	300	100	200	300
SMART-H	--	--	--	--	--	--
SMART-H+1	1	1	1	1	1	1
HA-1	7	7.1	6.2	6.3	6.5	5.5
HA-2	6.3	6.1	4	3	3.2	2.1
HA-3	1.9	1.8	1.7	1.54	1.3	1.3
HA-4	2.3	2	1.9	1.54	1.4	1.3

TABLE III. AVERAGE EXECUTION TIME (SECONDS)

Method	Physical degree = 3 Logical degree = 2.5			Physical degree = 3 Logical degree = 3		
	No. of Physical Nodes					
	100	200	300	100	200	300
SMART-H	18	41	54	3.16	5.6	12.38
SMART-H+1	25	53	71	4.2	6.5	14.3
HA-1	2.3	5.8	8	1.65	2.7	5.3
HA-2	1.4	4.1	6.8	1.58	3.5	5
HA-3	50	107	137	7.6	12.2	27.7
HA-4	41	95	117	6.9	11	25.6

that have non-disjoint mappings in  $M_{Best}$ . The modified algorithm is shown in Fig. 5.

SMART-H proceeds by recursively picking cycles. When survivable mappings for a particular cycle  $C$  are not returned by the mapping algorithm, it proceeds by picking a new cycle. However, when the modified mapping algorithm (shown in Fig. 5) fails to find survivable mappings for a cycle, it returns a set of pseudo-critical links ( $\eta$ ). By providing protection for links in  $\eta$  and accepting the mappings returned by the mapping algorithm for the links in  $C$  but not in  $\eta$ , the entire cycle becomes survivable. The modified algorithm called Hybrid Algorithm-1 (HA-1) is shown in Fig. 6.

The algorithm in Fig. 6 (HA-1) determines pseudo-critical links on a per cycle basis that would result in a large number of logical links that are protected. To reduce the number of such links, we propose three variants of the HA-1 in Fig. 7, 8, and 9.

The algorithm shown in Fig. 7 (HA-2) also proceeds by picking cycles. If a cycle cannot be mapped in disjoint manner, the mapping algorithm returns  $\eta$ , a set of links for which

disjoint mapping could not be found. Now pick a link  $l \in \eta$  and provide protection for such an  $l$ . After providing protection,  $l$  is contracted and the algorithm proceeds normally by picking another subgraph from the contracted topology.

Fig. 8 shows a variant (HA-3) that waits for the SMART-H to terminate unsuccessfully. An unsuccessful termination returns  $\eta$  and  $M_{Best}$  for the last subgraph considered. The algorithm then provides protection for  $(|M_{Best}| - 1)$  links in  $\eta$ .

Fig. 9 shows another variant (HA-4) that also waits for the SMART-H to terminate unsuccessfully. An unsuccessful termination returns  $\eta$  in the last subgraph considered. The algorithm picks a link  $l \in \eta$ , provides protection for  $l$  and contracts it. The algorithm then proceeds by choosing another subgraph from the contracted logical topology.

## VI. SIMULATION STUDY AND RESULTS

Simulation studies were conducted using VC++ 8.0 to evaluate SMART-H and the proposed algorithms. For simulation studies, random logical and physical topologies with varying number of nodes and degree were generated. The topologies were selected for processing, if the topologies were at least two connected.

Physical topologies were random topologies with 100, 200 and 300 nodes ( $|N|$ ) with degree 3. The logical topologies were also random topologies containing a random subset of physical nodes with degree 2.5 and 3.0. The number of logical nodes in the logical topology was set to  $0.75 \times |N|$ . The total number of such logical-physical topology pairs was 1000 for each case (40 physical and 25 logical topologies).

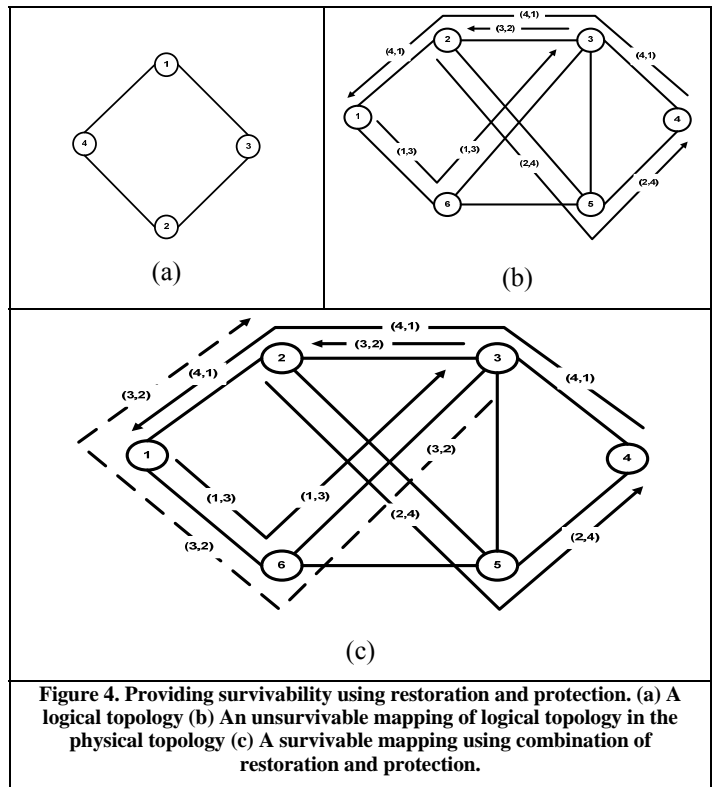


Figure 4. Providing survivability using restoration and protection. (a) A logical topology (b) An unsurvivable mapping of logical topology in the physical topology (c) A survivable mapping using combination of restoration and protection.

To make the disjoint mapping algorithm practical (Fig. 4), the number of iterations was limited to 100. To find a cycle, two nodes ( $s$  and  $t$ ) were randomly picked and two link disjoint paths ( $M_1$  and  $M_2$ ) were found using a max flow algorithm. The cycle was then constructed by  $P_1 \cup P_2$ . The number of subgraphs to be examined before unsuccessful termination was also 100 (MAX\_ITER). SMART-H and the proposed algorithms do not keep track of subgraphs for which survivable mappings could not be found. Therefore, MAX\_ITER was reduced to MAX\_ITER - (MAX\_ITER - 10) when the subgraph size was equal to the contracted topology size for 10 consecutive iterations.

The statistics of interest were the number of logical topologies for which survivable mappings could be found in a physical topology, number of links added to a logical topology in each algorithm and the execution time of the algorithms. Table I, II and III summarize the results.

In Table I, it can be seen that the proposed algorithms can map all the logical topologies in survivable manner by providing protection for some logical links. SMART-H can also map more topologies after the addition of a logical link (SMART-H+1) but some topologies still remain unsurvivable. It can also be observed that the number of mapped topologies, for a particular physical-logical degree, is relatively constant. Increasing the logical degree significantly increases this number, which is expected because the subgraphs becomes smaller in size and the probability of finding a survivable mapping increases.

Table II shows the average number of protected links in the proposed approaches. In case of SMART-H+1, only one additional logical link was added that significantly increased the number of mapped topologies. HA-1 and HA-2 provide protection for a large number of logical links but execution time is comparatively very small (Table III). HA-3 and HA-4 require protection for fewer links but the execution time is significantly higher since they must consider a larger number of subgraphs. It can also be noted that when the logical degree is increased, protection is required for fewer links and the execution time is also less.

## VII. CONCLUSION

In this paper, we have proposed four algorithms to find survivable mappings for a logical topology in a physical topology using a combination of protection and restoration. The algorithms are able to find survivable mapping for any two-connected logical-physical topology pair but require reservation of additional resources such as lightpaths. Simulation results show that the additional spare capacity needed in the case of HA-3 and HA-4 is very small and they are able to find survivable mappings for all the logical-physical topology pairs but they exhibit higher execution time. HA-1 and HA-2 are extremely fast but additional capacity required is much higher.

## ACKNOWLEDGMENT

The work of K. Thulasiraman has been supported in part by NSF ECS grants ANI-0312435 and ECS 04-26831 and the

work of G. Xue has been supported by NSF ITR grant ANI-0312635.

## REFERENCES

- [1] T. Stern and K. Bala, *Multiwavelength Optical Networks: A Layered Approach*, Addison-Wesley 1999.
- [2] G. Iannaccone, C.-N. Chuah, S. Bhattacharyya, and C. Diot, "Feasibility of IP restoration in a tier-1 backbone," Sprint ATL Research Report Nr. RR03-ATL-030666.
- [3] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications an approach to high bandwidth optical WDM," *IEEE Trans. Comm.*, vol. 40, no. 7, July 1992.
- [4] O. Crochat, J. L. Boudec, and O. Gerstel, "Protection interoperability for WDM optical networks," *IEEE/ACM Trans. Networking*, vol. 8, no. 3, June 2000.
- [5] M. Kurant and P. Thiran, "On survivable routing of mesh topologies in IPover-WDM networks," in *Proc. IEEE INFOCOM 2005*, vol. 2, Mar. 2005.
- [6] L. Sahasrabudde, S. Ramamurthy, and B. Mukherjee, "Fault management in IP-over-WDM networks: WDM protection versus IP restoration," *IEEE J. Sel. Areas Comm.*, vol. 20, no. 1, Jan. 2002.
- [7] M. Javed, K. Thulasiraman and G. Xue, "Lightpaths Routing for Single Link Failure Survivability in IP-over-WDM Networks", *Journal of Communication and Networks*, Dec. 2007.
- [8] E. Modiano and A. Narula-Tam, "Survivable lightpath routing: A new approach to the design of WDM-based networks," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 4, May 2002.
- [9] F. Ducatelle and L. Gambardella, "FastSurv: A New Efficient Local Search Algorithm for Survivable Routing in WDM Networks", *GLOBECOM 2004*.

INPUT: An at least 2-connected physical topology  $G(V, E)$ , a cycle  $C$ .

OUTPUT: Best mapping  $M_{Best}$ ,  $|M_{Best}|$  and  $\eta$

### Procedure find mappings

Best mapping  $M_{Best} \leftarrow \phi$

Number of non-disjoint mappings in  $M_{Best}$ ,  $|M_{Best}| \leftarrow \alpha$

Pseudo-critical links  $\eta \leftarrow \phi$

Let  $\forall e \in E, u(e) = 1$

**While** the termination condition is not met **do**

Mapping  $M \leftarrow \phi$

**For all** links  $l_i \in C$  **do**

$s \leftarrow$  source of link  $l_i$

$t \leftarrow$  destination of link  $l_i$

$M[i] \leftarrow$  a shortest path between  $s$  and  $t$  in  $G$  using  $u$

**End For**

**If**  $(M[0] \cap M[1] \dots \cap M[i]) = \phi$  **then**

Disjoint mappings have been found.

$M_{Best} \leftarrow M$ ,  $\eta \leftarrow \phi$

**Break**

**Else**

$\beta \leftarrow$  Physical links belonging to more than one mapping in  $M$

$\forall e \in \beta, u(e) = u(e) + 1$

$\delta \leftarrow$  number of non-disjoint mappings in  $M$

**If**  $\delta < |M_{Best}|$  **then**

$M_{Best} \leftarrow M$ ,  $|M_{Best}| \leftarrow \delta$

$\eta \leftarrow l_i \in C$  for which disjoint mappings could not be found

**End If**

**End While**

**Return**  $M_{Best}$ ,  $|M_{Best}|$  and  $\eta$

Figure 5. Mapping algorithm

- [10] A. Todimala and B. Ramamurthy, "A Scalable Approach for Survivable Virtual Topology Routing in Optical WDM Networks", IEEE J. on Selected Areas in Communications, Vol. 25, No. 6, Aug 2007.
- [11] R. Shenai and K. Sivalingam, "Hybrid Survivability Approaches for Optical WDM Mesh Networks", J. of Lightwave Technology, Vol. 23, No. 10, Oct 2005.
- [12] P. Mateti and N. Deo, "On algorithms for enumerating all circuits of a graph," SIAM Journal on Computing, vol 5, no. 1, Mar. 1976.
- [13] M. Kurant and P. Thiran, "Survivable routing of mesh topologies in IP-over-WDM networks by recursive graph contraction," IEEE J. Sel. Areas Commun., vol. 25, no. 5, June 2007.
- [14] J.W. Suurballe and R.E. Tarjan, "A quick method for finding shortest pairs of disjoint paths", *Networks*, Vol. 14, 1984.

INPUT: An at least 2-connected physical topology  $G(V, E)$  and a logical topology  $G^L(N^L, V^L)$ .  
 OUTPUT: One link survivable mapping  $M$  of  $G^L$  in  $G$ .

**While** the logical topology is not reduced to a single node **do**  
 $C \leftarrow$  a short cycle in  $G^L$   
**Call** the mapping procedure with cycle  $C$   
**If**  $\eta$  returned by the mapping algorithm is empty **then**  
 Contract  $G^L$  by collapsing the edges and merging the nodes in  $C$ .  
**Else**  
**For**  $(|M_{Best}| - 1)$  links in  $\eta$  **do**  
 $l_i \leftarrow$  a link in  $\eta$   
 $s \leftarrow$  source of link  $l_i$   
 $t \leftarrow$  destination of link  $l_i$   
 Find two disjoint paths  $P_1$  and  $P_2$  between  $s$  and  $t$  in  $G$ .  
 Add  $P_1$  and  $P_2$  to  $M$  as the mappings for  $l_i$ .  
**End For**  
 Copy mappings of link  $(l_i \in C \wedge l_i \notin \eta)$  from  $M_{Best}$  to  $M$ .  
 Contract  $G^L$  by collapsing the edges and merging the nodes in  $C$ .  
**End if**  
**End While**

**Figure 6. Hybrid Algorithm 1 (HA-1)**

INPUT: An at least 2-connected physical topology  $G(V, E)$  and a logical topology  $G^L(N^L, V^L)$ .  
 OUTPUT: One link survivable mapping  $M$  of  $G^L$  in  $G$ .

**While** the logical topology is not reduced to a single node **do**  
**While** the termination condition is not met **do**  
 $C \leftarrow$  a short cycle in  $G^L$   
**Call** the mapping procedure with cycle  $C$   
**If**  $\eta$  returned by the mapping algorithm is empty **then**  
 Contract  $G^L$  by collapsing the edges and merging the nodes in  $C$ .  
**End If**  
**End While**  
**For**  $(|M_{Best}| - 1)$  links in  $\eta$  **do**  
 $l_i \leftarrow$  a link in  $\eta$   
 $s \leftarrow$  source of link  $l_i$   
 $t \leftarrow$  destination of link  $l_i$   
 Find two disjoint paths  $P_1$  and  $P_2$  between  $s$  and  $t$  in  $G$ .  
 Add  $P_1$  and  $P_2$  to  $M$  as the mappings for  $l_i$ .  
**End For**  
 Copy mappings of link  $(l_i \in C \wedge l_i \notin \eta)$  from  $M_{Best}$  to  $M$ .  
 Contract  $G^L$  by collapsing the edges and merging the nodes in  $C$ .  
**End While**

**Figure 8. Hybrid Algorithm 3 (HA-3)**

INPUT: An at least 2-connected physical topology  $G(V, E)$  and a logical topology  $G^L(N^L, V^L)$ .  
 OUTPUT: One link survivable mapping  $M$  of  $G^L$  in  $G$ .

**While** the logical topology is not reduced to a single node **do**  
 $C \leftarrow$  a short cycle in  $G^L$   
**Call** the mapping procedure with cycle  $C$   
**If**  $\eta$  returned by the mapping algorithm is empty **then**  
 Contract  $G^L$  by collapsing the edges and merging the nodes in  $C$ .  
**Else**  
 Pick an edge  $l_i \in \eta$   
 $s \leftarrow$  source of link  $l_i$   
 $t \leftarrow$  destination of link  $l_i$   
 Find two disjoint paths  $M_1$  and  $M_2$  between  $s$  and  $t$  in  $G$ .  
 Add  $M_1$  and  $M_2$  to  $M$  as the mappings for  $l_i$ .  
 Contract  $l_i$  by collapsing the edge and merging the  $s$  and  $t$ .  
**End if**  
**End While**

**Figure 7. Hybrid Algorithm 2 (HA-2)**

INPUT: An at least 2-connected physical topology  $G(V, E)$  and a logical topology  $G^L(N^L, V^L)$ .  
 OUTPUT: One link survivable mapping  $M$  of  $G^L$  in  $G$ .

**While** the logical topology is not reduced to a single node **do**  
**While** the termination condition is not met **do**  
 $C \leftarrow$  a short cycle in  $G^L$   
**Call** the mapping procedure with cycle  $C$   
**If**  $\eta$  returned by the mapping algorithm is empty **then**  
 Contract  $G^L$  by collapsing the edges and merging the nodes in  $C$ .  
**End If**  
**End While**  
 Pick an edge  $l_i \in \eta$   
 $s \leftarrow$  source of link  $l_i$   
 $t \leftarrow$  destination of link  $l_i$   
 Find two disjoint paths  $P_1$  and  $P_2$  between  $s$  and  $t$  in  $G$ .  
 Add  $P_1$  and  $P_2$  to  $M$  as the mappings for  $l_i$ .  
 Contract  $l_i$  by collapsing the edge and merging the  $s$  and  $t$ .  
**End While**

**Figure 9. Hybrid Algorithm 4 (HA-4)**