Fiber Span Failure Protection in Mesh Optical Networks

Guangzhi Li, Bob Doverspike and Chuck Kalmanek
AT&T Labs — Research, AT&T Shannon Laboratories
Florham Park, NJ 07932-0971, USA

ABSTRACT

A major challenge of optical network design is deciding where and how much spare capacity is needed, so that interrupted traffic may be rerouted in the event of a failure. Given the optical network topology and traffic forecast for optical connections, a network design must choose routes for the connections in the forecast (called *demands* in this context) over network capacity with the objective of minimizing the cost of placing capacity. To provide end-to-end path based restoration for each optical connection, two paths need to be computed: a service path and a restoration path. Optical network design usually considers single failure. When two service paths are not both affected by a single failure, their restoration paths can share the same capacity on the links that they share. In this way, the total spare capacity needed for restoration can be dramatically reduced. However, due to the layered architecture in optical networks, paths that are diverse in one layer are not necessarily diverse in the lower layer. For example, optical networks are typically built on top of a network of fiber cables. Families of coterminous fiber cables are called *fiber spans*. A single fiber-span cut in the fiber layer can cause multiple link failures in the optical layer. In this paper, we investigate scenarios for restoring connection that fail because of fiber span failure in mesh optical networks. Specifically, we provide an algorithm to find two fiber-span disjoint paths for each demand, such that the total spare capacity allocated in the network is minimized. This algorithm can be extended readily to the more general notion of failure of Shared Risk Groups (SRGs) in optical networks, of which fiber-span failure is a sub-case. Another problem that arises in restoration path computation is the existence of a *trap topology* (a subgraph of the network graph). With a trap topology, if a service path is independently routed over a trap topology, then there may not exist a diverse restoration path, even though two diverse paths exist in the network. For simple link-disjoint restoration, the min-cost max-flow algorithm can be used to avoid this problem. For restoration of fiber-span failure, the trap topology problem becomes complicated. It is shown that the problem to find the maximum number of fiber-span disjoint paths between two nodes is NP-hard. The complexity of the problem to find two fiber span disjoint paths is still an open problem. This paper provides two heuristic algorithms to solve this problem. We have implemented fiber span failure protection algorithms in a multi-layer capacity planning toolkit, called *Cplan*, developed at AT&T Labs - Research. An application of fiber span restoration is described at the end of the paper.

Keywords: Spare capacity allocation, Fiber span protection, Mesh restoration, Heuristic algorithms

1. INTRODUCTION

Modern optical networks should be designed to be highly fault tolerant. Customers expect to see uninterrupted service, even in the event of failures such as power outages, equipment failures, natural disasters and cable cuts. To realize this, spare capacity must be added to the network so that traffic that has been interrupted by a failure can be rerouted. Spare capacity for restoration is a sizable fraction of total network capacity, and accounts for a large part of the infrastructure cost of telecommunications networks. It is therefore essential for network designers to have efficient algorithms for designing where and how much spare capacity is needed. This is known by many names, such as restoration capacity planning, survivable network design, resilient capacity reservation, or spare capacity assignment.[8–10,15] The problem is typically modeled by a network topology graph and a forecast of connection requests, usually called *demands* in the context of capacity planning. To provide end-to-end path-based restoration, for each demand the network is required to provide two diverse paths: the service path and the restoration path. When the service path fails, the traffic will be rerouted to the restoration path. Allocation of spare capacity is usually specified to meet a restoration objective in optical networks such that all connections that fail from any single fiber failure can be rerouted to alternate restoration paths with sufficient spare capacity. The network restoration objective usually does not include multiple failures because multiple failures have low probability and restoration from multiple failures can be very expensive (or sometimes impossible). Under the single-failure assumption, when two service paths are not both affected by the same single failure, their restoration paths can share the same capacity on the links that they share. In this way, the total spare capacity needed for restoration can be minimized. The objective is to optimize restoration path selection, such that the total spare capacity allocated is as small as possible, i.e., to maximize the sharing opportunity of spare capacity among multiple restoration paths.

Due to the layered architecture in optical networks,[23,20] this problem is more difficult than the models studied in most studies because a pair of diverse paths in the optical layer are not necessarily diverse in the fiber layer. To understand this, it is helpful to describe optical networks. The optical network consists of *optical cross-connects (OXCs)* and *optical transport systems (OTSs)*. An OTS is a pair of bidirectional wavelength division multiplexer (WDM) terminals that multiplex optical signals at different wavelengths into a single optical fiber for each direction of transmission using wavelength gratings or equivalent technology. Each wavelength into which a signal can be multiplexed is referred to as a *channel*. Longer OTSs require optical amplifiers at intermediate locations. The fibers are organized into cables. A *fiber span* is the collection of all optical fibers that are co-located in the same cable, conduit or substructure between two consecutive points of access (such as a manhole, central office, or amplifier site). If there is a fiber-span failure, all fibers within this fiber span will potentially experience the same failure. An *optical link* in the optical layer refers to all the (coterminous) OTS between a pair of nodes and, for simplicity, we assume that one OTS is long enough to connect any pair of nodes, i.e. we assume back-to-back OTS are not required because of OTS distance limitations.

The most common restoration method for OXCs (the optical layer) is that of choosing a pre-calculated failure-invariant restoration path that is node or edge-disjoint from the service path for each connection. Upon detection of a failure, the OXCs reroute the affected connections to their service path by cross-connecting spare channels of the OTS along the restoration path. However, multiple optical links can route over a common fiber span in the fiber-span layer. Following similar assumptions in the literature,[10,12,16–18,21] this paper focuses on restoration from single fiber-span failure.

We note that under this restoration approach, if two service paths do not share any fiber span, their restoration paths can share the spare channel capacity on any links that they have on common. However, what makes this problem so complex is that this is only a sufficient condition for spare capacity sharing. In fact, a single restoration channel on a common link of multiple restoration paths can be shared by non-simultaneous fiber span failures. Since channels can be allocated after failure, the combinatoric sharing of restoration channels applies to groups of connection paths, rather than on a pair-wise basis. For example, consider figure 1. Solid lines reflect service paths and dashed lines restoration paths. All three connections share service paths with one another, but the single failure of any link fails no more than two connections at the same time (assuming for this simple example that one optical links routes over just one fiber span). All three restoration paths intersect on multiple links. For example, link H-J requires two restoration channels and therefore three connections are sharing two restoration channels on link H-J. Thus, any two connections share a fiber span on their service paths, yet may share restoration channels on link H-J.

In general, network restoration includes two parts: spare capacity allocation and service restoration after a failure. Spare capacity allocation ensures enough spare capacity to recover a failure via traffic rerouting. The spare capacity can be dedicated or shared among connections that are affected by non-simultaneous failures. Failure restoration schemes can in general be classified into two categories: dynamic and pre-computed. Dynamic schemes[24] search for alternative paths with sufficient spare capacity by broadcasting restoration messages after a failure occurs. In contrast, in pre-computed[23,22,19] restoration schemes, the restoration paths are pre-computed and spare capacity is reserved. Upon failure, the node responsible for restoring the affected traffic simply activates the restoration paths and reroutes the affected traffic. If the spare capacity (channel inventory) is dedicated and pre-crossconnected to each connection (often called 1+1 protection), then the rerouting only has to occur at the ends and hence is fast. However, such schemes use considerable restoration capacity and are hence viewed as prohibitive in cost in large scale networks.

Due to the high cost of spare capacity, in this paper, we consider schemes that pre-compute restoration paths with shared spare capacity, i.e, where the spare restoration channel capacity is not dedicated to specific connections, but rather chosen after the failure has occurred. The objective is to choose restoration paths to minimize the total cost of spare capacity while meeting network survivability requirements. The literature contains

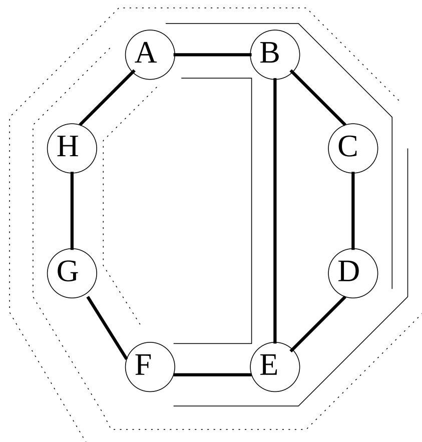

Figure 1. Spare Capacity Sharing Example

a number of research papers dealing with this problem.[10,12,15–17] However, very few papers have considered a multi-layer model, specifically fiber span failure.[26,27] Since in a typical optical network deployment, a fiber span failure often results in the simultaneous failure of multiple optical links, modeling of the fiber-span layer is a critical issue in designing survivable optical networks. Efficient algorithms for solving this new problem are required.[23]

Another problem that arises in the pre-computed restoration scheme is the existence of subnetworks called the *trap topology*.[25] When routing over a trap topology, the pre-selected service path may not have a diverse restoration path, even though diverse paths exist in the network. Our experience shows that trap topology is not only an academic problem, but also it exists in typical carrier backbone networks. Although the network graph may be two-connected, some "poorly chosen" service paths may not have a link-disjoint restoration path. For simple single-layer link failure, the min-cost max-flow algorithm[6] can be used to choose service and restoration paths at the same time and thus avoid this situation. For multi-layered networks (fiber-span failure), this problem is more complex. We have not resolved the algorithm complexity of this more complex problem yet, but we show that it is NP-hard to find the maximum number of fiber-span disjoint (often referred to as *physically disjoint*) paths between two nodes. Two heuristics are provided in this paper to handle this case.

In section 2, we provide the problem formulation, including architecture modeling. An efficient heuristic algorithm to minimize the spare capacity allocation for fiber span failure protection follows in section 3. Section 4 describes the trap topology problem, including two heuristic algorithms. Section 5 discusses our implementation in the *Cplan*[3] toolkit, followed by an application of multi-layer failure restoration using our *Cplan* tools in section 6. Section 7 provides our conclusions.

2. PROBLEM FORMULATION

2.1. Model

$G_s = (N_s, E_s)$ denotes the graph of the fiber-span layer, where N_s is the set of nodes (representing locations, generally, where WDM equipment can be placed) and E_s is the set of edges representing fiber spans. As described before, each edge $e \in E_s$ represents the collection of all channels of all OTS placed between the endpoints of e. Let $N_o \subset N_s$ be the set of nodes where OXCs are placed and E_o the set of edges representing optical links, which are in-turn routed on the fiber spans of G_s. Therefore, $G_o = (N_o, E_o)$ is the graph of the optical layer. For each restorable connection request, a service path P_s and a restoration path P_r in the optical network are computed. Our design objective is to minimize the total spare capacity subject to the following requirements:

- The fiber span network, G_s, and optical network G_o graphs are given and the routing of any optical layer link $e \in E_o$ over path of fiber-spans in E_s is given and fixed.

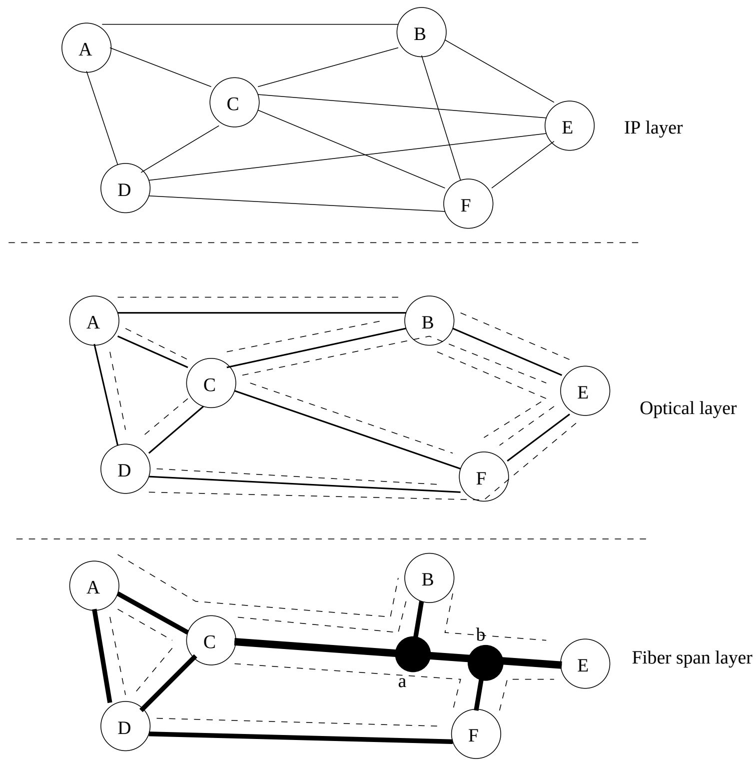

Figure 2. Three-layer architecture example

- P_s and P_r do not share any common fiber span (if topologically possible).
- The spare capacity on a link can be shared by multiple restoration paths if their corresponding service paths are not subject to simultaneous fiber span failures. For example, we can use a spare channel on a link of restoration path P_{r1} to reroute service path P_{s1} due to any failure of fiber spans along P_{s1}. That same spare channel can be used to reroute service path P_{s2} to restoration path P_{r2} due to any failure of fiber spans along P_{s2} as long as P_{s1} and P_{s2} do not fail at the same time.
- Enough spare capacity should be allocated on the links of a restoration path such that under any single fiber span failure there is enough spare capacity to restore all failed service paths.

2.2. Structure of an optical network

An optical network typically has a layered architecture: the optical layer provides connections for service layer links and the optical layer itself rides on the fiber span layer. Figure 2 is a simple example of three-layer network. From the point of view of a network designer, links in a higher layer are provided by connections in a lower layer.

In Figure 2, the fiber span layer has eight nodes and nine fiber spans. The open circles denote nodes where equipment can be placed and the darkened circles represent nodes where only fiber splice points may occur. In the optical layer, only six nodes are deployed with OXCs where optical connections may be added/dropped and there are nine optical links. In the IP layer, there are six nodes with routers and 11 links. Note that as one examines the network layers from bottom to top, the probability of multiple link failure increases. For example, if the single fiber span b-E is cut, then two links fail in the optical layer and four links fail in the IP layer.

This paper focuses on the optical layer and fiber-span layers with restoration (rerouting of connections) at the optical layer against single fiber-span failure. More complex work is underway to study integrated restoration at both IP and optical layers, but such models are not considered here. In order to simplify our presentation, we illustrate the fiber spans and optical links in Table 1 and Table 2 respectively.

Span ID	Source	Destination
0	A	C
1	C	a
2	a	B
3	a	b
4	b	E
5	b	F
6	D	F
7	C	D
8	A	D

Table 1. Fiber Span Table

Link ID	Source	Destination	Span IDs
0	A	B	0:1:2
1	A	C	0
2	C	B	1:2
3	C	F	1:3:5
4	C	D	7
5	A	D	8
6	D	F	6
7	F	E	5:4
8	E	B	4:3:2

Table 2. Optical Link Table

The span IDs in the optical link table represent the link-span relationship. For example, the link with ID=0 traverses fiber spans with IDs 0, 1, and 2 sequentially.

To select an end-to-end restoration path, it is important to know the routing of optical links over fiber spans. In Figure 2, if the service path for a connection from A to C traverses directly over the optical link A-C (optical link with ID=1), it appears from the optical layer that the restoration path A-B-C (optical links with IDs = 0 and 2) would be adequate. However, these two paths are not fiber-span diverse in the fiber-span layer. The goal is to assign a service path and a span-disjoint restoration path for each demand with an objective function to minimize the total OTS channel capacity required or the total spare OTS channel capacity required.

3. ALGORITHM FOR RESTORATION PATH SELECTION

Spare capacity allocation with restoration of single link-span failure can be formulated as an integer programming problem that is NP-hard.[17] If each optical link only traverses a single fiber span, the problem of multi-layer fiber span failure reduces to that of classical single-layer link failure. Therefore, accepting the results of,[17] we conclude that spare capacity restoration allocation under multi-layer fiber-span failure is NP-hard. In this section, we provide a simple and efficient algorithm to compute two fiber span disjoint restoration paths for each demand. This algorithm automatically minimizes the total spare capacity.

3.1. An algorithm to select two span-disjoint paths

In single-layer networks, the conventional Dijkstra algorithm computes a shortest path tree from a node S to every other nodes in a graph. Of course, given the network topology and two nodes, say S and D, Dijkstra's algorithm is able to compute a shortest path from S to D. After that, if we remove all links of this shortest path from the network topology and run Dijkstra's algorithm again, we will find another shortest path from S to D.

Link-Disjoint (input: a network topology with nodes and links, two different nodes S and D; output: two link-disjoint shortest paths from S to D)
1. Run Dijkstra's algorithm to find first shortest path from S to D
2. Delete all links on the first path from the network.
3. Run Dijkstra's algorithm on the new network to find the second shortest path from S to D
4. return these two shortest paths

Figure 3. Find two link-disjoint paths

SPAN-Disjoint (input: optical-layer and fiber-span layer networks and relationships as defined by G_o and G_s; two different nodes S and D; output: two span-disjoint shortest paths from S to D)
1. Run Dijkstra's algorithm in G_o to find first shortest path from S to D
2. For each link e on the first path
3. Delete all links that share at least one fiber-span with e
4. End for
5. Run Dijkstra's algorithm on the new network to find the second shortest path from S to D
6. return these two shortest paths

Figure 4. Find two span-disjoint paths

Obviously, these two shortest paths from S to D are link-disjoint. We can use the first path as the service path and the second path as the restoration path. See Figure 3.

If we need two node-disjoint paths, we delete all nodes along the first path except nodes S and D including all links associated with these nodes. In multi-layer networks, if the link-fiber-span relationships are given and we are interested in restoration against fiber-span failure, we can compute two span-disjoint paths in a similar way. See Figure 4.

The algorithm in Figure 4 only provides two span-disjoint paths. How can we minimize the spare capacity that is allocated? The idea is to find a span-disjoint path that maximizes the amount of spare capacity sharing with already chosen restoration paths. For example, in Figure 5, suppose we need to route 2 demands (connections) The source and destination (s/d pair) of demand 1 is A and B. The service path is A-B and restoration path is A-C-D-E-B. To protect demand 1, one unit (WDM channel) of spare capacity is allocated in links along the restoration path (assuming that each demand requires one unit of resource). Suppose demand 2 has source F and destination I. Assume that the service path is F-G-H-I. What is a good restoration path? If we use hop count as the routing metric, F-A-B-I would be the shortest diverse path. This path requires one extra spare unit of capacity on each link and thus the total extra spare capacity will be 3 units. However, if path F-C-D-E-I is selected as the restoration path, we only need to allocate one unit of spare capacity on optical link F-C and one unit of spare capacity on optical link E-I. The total extra spare capacity will be 2 units, because the spare capacity on optical links C-D and D-E can be shared between these two restoration paths. Next, we will describe a simple and efficient algorithm to realize this idea.

3.2. An algorithm to minimize spare capacity

The process of computing a service path and restoration path for a connection relies on information about available optical link resources. A network algorithm implemented by a centralized system is able to maintain global link state information. A popular heuristic is to select the "min weight" path among all suitable paths where the weights are approximated by hop count, link usage, or "fill". However, such heuristics may yield poor results because since a restoration path which can share spare capacity resources may not be captured by a simple weight approximation. We prove this assertion in.[5] The algorithm presented here takes into account several metrics in the restoration path selection process to minimize spare capacity needed for restoration. We focus here on the path selection algorithm and omit details associated with managing the network state information such as allocated service and restoration capacity. Distributed signaling protocols to implement this algorithm are the focus of our current work.

Essentially, we present a greedy algorithm that routes each demand on the path with minimum marginal

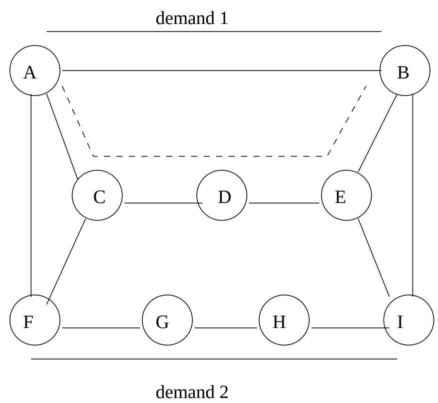

Figure 5. Example of Shared Restoration Capacity

restoration capacity. To do this, first we define the matrix $failneed_{sk}$ which is the amount of spare capacity needed on optical link $k \in E_o$ to reroute all failed connections when fiber span $s \in E_s$ fails. This spare capacity would be used by the OXCs to reroute the connections whose service paths route over fiber span s to their restoration paths. We also define $maxfailneed_k$ as the amount of spare capacity needed on optical link $k \in E_o$ to meet a restoration objective to restore 100% of the failed connections resulting from any single fiber span failure. Note that $maxfailneed_k = max\{failneed_{sk} : s \in E_s\}$.

We compute the service and restoration paths for each connection in sequence. The $failneed_{sk}$ and $maxfailneed_k$ variables keep the current network state information. We will show how to compute the restoration path for a new connection d. The restoration path should be fiber-span (physically) disjoint from the service path. In addition to the previously defined variables, we also define the following variables to complete the restoration path selection process for connection d on optical link k.

- M_k, the amount of spare capacity needed on optical link k over all possible failures of the selected service path for d, i.e., $M_k = max\{failneed_{sk} : s \text{ is on the service path for } d\}$.
- Routing weights, w_k, and initial weights, w_{ok} of optical link k.

w_{ok} is initialized to a value reflecting the hop count (e.g.,$w_{ok} = 1$) or approximate cost of unit capacity.

The method to compute the restoration path then becomes a shortest path algorithm, for example, Dijkstra's algorithm. First, we delete all links which share at least one fiber span with the links in the service path. Then we set the weight on the remaining optical links to compute a shortest path based on the new weight on each optical link. If dedicated restoration (1+1 protection) is specified, we simply set the weight as the initialized weight. If shared restoration is specified, we compute the weights as follows:

$$w_k = \begin{cases} \epsilon & \text{if } C_k \geq size(d), \\ w_{ok} & \text{if } C_k \leq 0, \\ w_{ok} * (size(d) - C_k)/size(d) & \text{if } 0 < C_k < size(d). \end{cases}$$

where $C_k = maxfailneed_k - M_k$ represents the "spare" restoration capacity allocated on optical link k, and $size(d)$ is the number of units of capacity required by demand d. For example if d is an SONET OC-192 connection (10Gb/s) and the units are OC-48 (2.5Gb/s), then $size(d) = 4$. ϵ is a small positive value which is much less than the weight of a link. If the restoration path for demand d is routed over optical link k, and if $size(d)$ does not exceed C_k, then the total spare capacity needed on link k does not increase if demand d is routed over it. Note that we set the weight to ϵ rather than zero when the spare capacity is sufficient to meet the demand so that the algorithm will reuse capacity on shorter rather than longer paths.

It is possible to extend this algorithm where the lowest layer is represented more generally by Shared Risk Groups (SRGs). SRG is generic transport network protection/restoration terminology which has been discussed

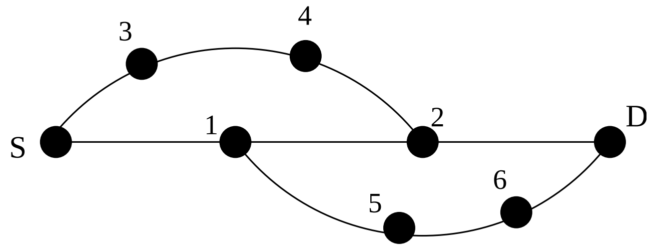

Figure 6. Trap Topology Example

in the Internet Engineering Task Force (IETF) and Optical Internetworking Forum (OIF) standards bodies.[28,29] An SRG represents a common physical network resource over which higher (optical) layer links route. If the common network resource fails, the whole set of links will experience the same failure. Depending on the network reliability objective, the common resource type could be a fiber, fiber sub-segment, fiber segment, fiber span, node, manhole, conduit, etc. Sometimes, if the resource is like a network edge, it is called a Shared Risk Link Group (SRLG). In this algorithm, if we replace the fiber span index with the SRG index, the algorithm will automatically provide two SRG-disjoint paths for each demand and minimize the total reserved restoration resources.

4. TRAP TOPOLOGY HEURISTICS

The above algorithm may work poorly when routing over a trap topology. In Figure 6, when the traffic flow between node S and D has service path via nodes 1 and 2, this path does not have any diverse restoration path available although there exist two diverse paths.

Min-cost max-flow algorithms[6] can be used to avoid this problem under link failure. If we assign each optical link a capacity of 1 unit and unit cost of link weight, we can apply the flow-augmentation method of the min-cost max-flow algorithm to compute two link-disjoint paths and use the shorter one for the service path and the other for the restoration path. How can we solve the trap topology problem for multi-layer networks (fiber-span failure)? One possibility is to use the min-cost max-flow algorithm. However, although the problem complexity is still unknown, the problem to find the maximum number of fiber-span disjoint paths between two nodes is proven to be NP-hard.[4] This is different from single-layer simple link failure, where the complexity of finding the maximum number of link-disjoint paths between any two nodes is polynomial. Another possibility is to compute two fiber span disjoint paths in the fiber-span layer. Unfortunately, the optical-layer links do not necessarily terminate at each fiber span end-node and, as such, a feasible fiber-span path may not be a feasible path in the optical layer. Practical and efficient algorithms are required to handle this situation.

4.1. Computational complexity

This section provides a simple outline of the proof of NP-hard complexity.

THEOREM 4.1. SPAN-DISJOINT *(input: optical-layer and fiber-span layer networks and relationships as defined by G_o and G_s; two different nodes S and D; output: two span-disjoint shortest paths from S to D)*

Given an multi-layered network (optical layer over a fiber-span layer) as defined by G_o and G_s and the routing of edges in G_o over those in G_s, the problem of finding the maximum number of fiber-span-disjoint paths between any two nodes is NP-hard.

Proof. We only provide the outline of the proof here. See[4] for more detail. The technique used is a simple reduction from the INDEPENDENT SET problem. Let $G = (V, E)$ be an undirected graph, and let $I \subset V$. We say that the set I is independent if whenever $i, j \in I$, there is no edge between i and j in G. The INDEPENDENT SET problem is that: Given an undirected graph $G = (V, E)$, and a goal K, is there an independent set I with $|I| = K$? We know that INDEPENDENT SET problem is NP-hard.[7] Now, given an instance of the INDEPENDENT SET problem, we construct a fiber-span network and an optical network with link-to-fiber-span relationships as well as two nodes in the optical network such that the independent set problem has a solution of K if and only if there are K number of fiber-span disjoint paths between the two nodes in the optical network.

This says that the multi-layered maximal number of fiber-span-disjoint paths problem is at least as hard as the INDEPENDENT SET problem, which proves that it is NP-hard. This proves our claim. ∎

4.2. Two algorithms for trap topology networks

The algorithm in section 3 can be used to find two fiber-span-disjoint paths, but it is not sufficient to handle trap topology networks. As we pointed out before, due to the trap topology, the service path computed by this algorithm may prevent a fiber-span-disjoint restoration path from being found. So, if we find the first service path, but cannot find a second fiber-span-disjoint restoration path, we need to provide some algorithms to avoid this situation. In this section, we propose two heuristic algorithms to find two span-disjoint paths if the algorithm in section 3 fails.

4.2.1. BasicLink method

We separate the links in the optical network into two groups: basic links and express links. If a link is not a basic link, then it is an express link. A link is a basic link if it satisfies one of the following two conditions:

1. the link only traverses one fiber span,
2. the link traverses multiple fiber spans, but it is the only link traversing each of the fiber spans.

If we can find a first shortest path, but can't find a second fiber-span-disjoint path using the algorithm in section 3, then we construct a new network with the original nodes and only basic links, which we call the *basic link network*. In this basic link network we use the min-cost max-flow method to find two link-disjoint paths (which are also fiber-span disjoint) and choose the shorter one as the service path. We may use one of the other paths from the max-flow algorithm as the restoration path, but the algorithm in section 3 may be able to find a better span-disjoint path.

This is a heuristic algorithm because it uses only basic links for the service path selection. However, if all links are basic, for example, no link-fiber-span relationship is considered, this algorithm reduces to solving the problem of restoration from single-layer link failure. This algorithm is simple but it has the drawback of limiting the path search to the basic links only. Then, this algorithm may fail too, although two fiber-span-disjoint paths exist. A more complicated and exhaustive searching method is provided at follows.

4.2.2. Bypass method

The idea of bypass method is to construct a single layer sub-network over the original optical network and try to find two link disjointed paths on the constructed sub-network. Then we choose one of the paths as the service path. Hopefully, this new service path does not prevent the fiber-span disjoint restoration path selection. This method may fail too, but we show that the constructed sub-network has some good characteristics.

Assume we have chosen the first shortest path p from S to D in the optical network with nodes $a_1, a_2, ..., a_k$ where a_1 is S and a_k is D, and links $e_1, ..., e_{k-1}$, but failed to find the second fiber-span-disjoint restoration path using algorithms in section 3. We construct a directed graph H with k nodes labeled $1, ..., k$, and we define edges on H as follows. We delete all links that share at least one fiber span with links in path p from the optical network. We then run Dijkstra's algorithm for each node pair $\forall a_i, a_j$ with $1 \leq i < j \leq k$. If we find a shortest path between a_i and a_j, we draw a directed edge from i to j. H is also given a back edge from i to $i-1$ for each $i = 2, ..., k$. Then we run Dijkstra's algorithm on H with a hop count metric from 1 to k. If we find a path on H, then we are able to find two link disjoint paths on H without considering the arc direction.

Figure 7 shows a simple example of directed graph H. Subgraph (a) shows H with 6 nodes and all possible edges, while Subgraph (b) shows a shortest path between 1 and 6, which is 1-4-3-6. Then we are able to find two link disjoint paths from node 1 to node 6 on H ignoring the arc direction: 1-4-5-6 and 1-2-3-6. Of course these two paths may not be fiber-span disjoint, because links 1-4 and 3-6 may share the same fiber span. As a result, we choose one of them as the service path and run the algorithm in section 3 on the optical network to find the second fiber-span-disjoint restoration path if one exists.

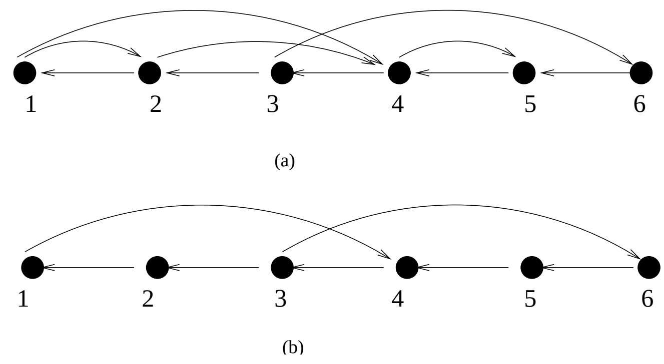

Figure 7. Directed Graph H Example

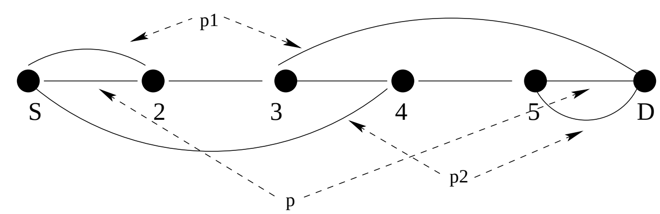

Figure 8. Example of $p, p1, p2$

In general, if there is a shortest path from S to D on H, there exists two link disjoint paths on H ignoring the arc direction. Assume there are n bypass links on the shortest path from S to D on H, which are indexed sequentially from S to D with $1, 2, ..., n$. We choose one path with the odd indexed bypass links combining links from original service path and another path with the even indexed bypass links connected by links of original service path. As we have pointed out before, these two paths may not be fiber span-disjoint. So, we choose the shortest path as the service path, then run the algorithm in section 3 to find a possible fiber span-disjoint path as the restoration path.

In the fiber-span network, if a sequence of fiber-spans forms a line, that is, the interior nodes have degree of two and there is no demand originating or terminating from these interior nodes, we call the list of spans a {it superspan }. Notice that a superspan may consist of only one fiber-span from this definition.

LEMMA 4.2. *Suppose there is an optical link for each superspan. If two paths (s_1, d_1) and (s_2, d_2) share one superspan, we can find a path from s_1 to d_2.*

Proof. If there is an optical link for each superspan, then there is an OXC located at each end of each superspan. Suppose (s_1, d_1) and (s_2, d_2) share a superspan (u, v), then there are OXCs at nodes s_1, u, v, d_2. We can find a path from s_1 to d_2 via $s_1 - u - v - d_2$. ∎

THEOREM 4.3. *Suppose there is one optical link over each superspan. If we can not find a path from S to D on H, then there exist no two fiber-span-disjoint paths on the optical network. If we can find a shortest path in H between S and D, then we can find two fiber span disjoint paths.*

Proof. Assume that we find a service path p between S and D and the algorithm in section 3 can not find the second fiber-span-disjoint path. We create the graph H but can not find a path from S to D on H. If there exist two fiber-span-disjoint paths in the optical network p_1, p_2 between S and D, then p_1 and p_2 must share at least one fiber span with p. The shared fiber span between p_1 and p must be different from the shared fiber span between p_2 and p, otherwise p_1, p_2 could not be fiber-span disjoint. The relationships between p, p_1, and p_2 show that there is at least one path from S to D on H. See figure 8. This contradicts our assumption and proves the first part of the theorem.

For the second part of the theorem, we claim that if we find a shortest path on H between S and D, the edges on this shortest path do not share any fiber span. According to Lemma 1, if two edges b_1 and b_2 share a fiber span, there must be a longer edge b_3 to cover the segments of p covered by b_1 and b_2. The shortest path on H encourages the choice of few edges. So, we find two link-disjoint paths on H, which will be fiber-span-disjoint paths on the original optical network. ∎

Notice that without considering the link-fiber-span relationship information, we may assume that each optical link routes over one fiber span and that all these fiber spans are different. Then our assumption that an optical link corresponds to each fiber span is satisfied.

5. IMPLEMENTATION

We implemented all these algorithms in the *Cplan* toolkit, a collection of multi-layer modeling and optimization routines designed to perform various capacity planning tasks.[3] The restoration path in *Cplan* could be specified to be either abstractly link-disjoint (with respect to the upper layer, e.g., the optical layer) or span-disjoint (with respect to the lower layer, e.g., the fiber-span layer). In our implementation, we first indexed all links from 1 to k and all spans from 1 to s. With the link-span relationship information for each link, we constructed several array data structures. $LinkLink[i]$ points to a list of links that each share at least one span with optical link i. $LinkSpan[i]$ points to a list of spans that link i passes through. $SpanLink[j]$ points to a list of links that pass through span j. $IsBasicLink[i]$ is a boolean variable indicating whether optical link i is a BasicLink. If there is no link-span relationship information, we assume that each link corresponds to one fiber span, i.e., equivalently a single layer network. Then we have $LinkLink[i] = i, LinkSpan[i] = i, SpanLink[i] = i, IsBasicLink[i] = true, i = 1, ..., s, s = k$.

To implement the algorithm presented in section 3, we maintain one matrix $failneed$ and one array $maxfailneed$. For each pair of link i and fiber span j, $failneed[i][j]$ is the amount of traffic, from the demands considered so far, that is rerouted onto link i when fiber span j fails. For a link i, $maxfailneed[i]$ is the restoration capacity required on i so far, namely the maximum of $failneed[i][j]$ over all fiber spans. After the restoration path is selected for the demand being processed, the matrix $failneed$ and array $maxfailneed$ are updated immediately. If we can not find the second restoration span-disjoint path, we use a *Route-Repair* procedure to choose a better service path. The implementation is based on two steps. First we use the *BasicLink* method to find two span-disjoint paths. If the first step fails, we use the *Bypass* method to try again. If both fail, we do not route the particular connection.

In our implementation, we process demands one at a time, in sequence without any assumption about future demands. This is how connection requests are normally provisioned in operational networks. However, if the whole connection forecast is assumed to be known and deterministic (which is often assumed in deterministic network design literature) then the order of examining demands is an aspect of the solution heuristics and the quality of the solution (optimality gap) could be dependent on the order. We did not examine this aspect in this paper. The program flow diagram is given in Figure 9.

6. AN APPLICATION

This section presents an application of *Cplan* for planning of restoration with multi-layer architecture. We used a network of 139 nodes and 208 links representative of a typical inter-city fiber span network, which has been used in architecture studies for transport networks.[1] Then we create the optical network with the fiber span network nodes and links plus some extra longer express links that route over the fiber span links. Comparing the restoration cost saving by rerouting the restoration paths over the express links, we show the benefits of creating express links. This application requires to consider the fiber span failure instead of the single layer optical network link failure. The results come from our *Cplan* implementation against single fiber span failure.

The demand set we consider here is one that highlights the need for longer restoration paths. We model our traffic as high rate (OC-48) private line demands that a large inter-city transport carrier would experience. The source of this demand may consist of links between IP routers or data switches, as depicted in Figure 2. We create 1159 demands, with average length 660 miles and a total length of 765791 miles. If restoration

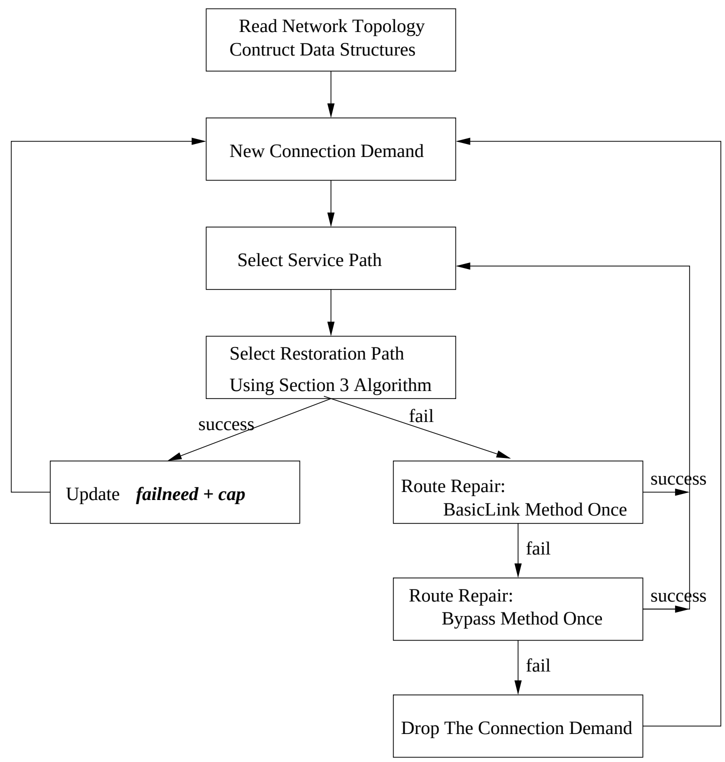

Figure 9. Implementation Flow Diagram

capacity is provisioned in units of single OC48s (2.5Gb/s OTS channels), the resulting profusion of short OC48s unnecessarily increases the cost of the transport network. This was a significant problem[1] assuming OC48s were to be restored in a layer of optical cross-connects. Each OC48 required termination equipment, such as ports on optical cross-connects at either end, so having many short OC48s negatively impacted the estimated network cost.

To generate longer restoration links, we first selected the 15 nodes that terminate the most demand in the demand set and used the tool *trim*[2] to create a set of express links in our fiber-span backbone. Each multi-hop express link rides on the original fiber span network. This process was repeated with the top 10 demand-termination nodes to generate a set of "super-express" links. Although the original network links were fiber span disjoint, the new express and super-express links are not, since they share fiber spans with the links over which they ride.

Three runs are presented here. The first is a run of *Cplan* using the original fiber span network, the second uses the express links, while the third uses both express and super-express links. The results of the runs are presented in Table 3. For each run, the table gives the number of restoration OC48s, the number of miles of restoration OC48s, and a simple estimate of the cost of the restoration OC48s. The estimated cost is just the total mileage plus 300 times the number of OC48s. This estimate reflects the tradeoff between the price of equipment needed at the end of an OC48, such as router ports and optical transponders, and equipment costs that vary with OC48 length, such as optical amplifiers.

The results show that there can be a significant benefit from creating longer restoration links. The estimated cost using both express and super-express links is 13% lower than the estimated cost without any expressing. Without the super-express links, the savings is up to 9%.

Run Description	Restoration Pipes	Restoration Miles	Estimated Cost
Express and super-express	2081	614018	1238318
Express	2290	615900	1302900
No expressing	3124	495957	1433157

Table 3. Comparison of Using Express Links

7. CONCLUSION

We investigated the problem of restoration of multi-layer mesh optical networks from fiber span failure. We provided a simple and efficient algorithm to minimize the total spare capacity allocation. Although the min-cost max-flow algorithm can be used to deal with trap topology networks in the case of single-layer link failure protection, the complexity of the multi-layer problem is still unclear. The complexity of the problem to find the maximum number of link-disjoint paths between two nodes in single-layer networks is provably polynomial, but the complexity of the problem to find the maximum number of fiber-span-disjoint paths in multi-layer networks is NP-hard. Two heuristics were presented to remedy this situation by choosing a "better" service path. We implemented these algorithms in the *Cplan* toolkit, developed at AT&T Labs - Research. A restoration application for a realistic inter-city network illustrated the importance of incorporating fiber span failures when applying some capacity planing software tools.

REFERENCES

1. R. Doverspike, S. Phillips, J. Westbrook, "Transport network architectures in an IP World", *IEEE INFOCOM'00*, March, 2000.
2. S. Phillips, "Trim: a tool for design backbone networks", *Technical Memorandum, AT&T labs*(internal publication), 2000.
3. S. Phillips, J. Westbrook, "Cplan: a network capacity and restoration planning toolkit", *Technical Memorandum, AT&T labs* (internal publication), 1998.
4. G. Li, R. Doverspike, and C. Kalmanek, "Fiber Span Failure Protection in Mesh Optical Networks", *OptiComm'2001*, Denver, CO 2001.
5. G. Li, D. Wang, C. Kalmanek, R. Doverspike, "Efficient Distributed Path Selection for Shared Restoration Connections", submitted to Infocom 2002.
6. B. Carre, "Graphs and networks", *Clarendon Press*, Oxford, 1979.
7. M. Garey and D. Johnson, "Computers and intractability: a guide to the theory of NP-completeness", *W.H. Freemand and Company*, New York-San Francisco, 1979.
8. B. Ryu, M. Murata, and H. Miyahara, "Design method for highly reliable virtual path based ATM networks", *IEICE Transactions on Communications*, vol E79-B (10), pp. 1500-1514, 1996.
9. W. Grover, R. Iraschko, and Y. Zheng, "Comparative methods and issues in design of mesh restorable STM and ATM networks", *Telecommunication Network Planning*, pp. 169-200, 1999.
10. Y. Xiong and L. Mason, "Restoration strategies and spare capacity requirements in self-healing ATM networks", *IEEE/ACM Transaction on Networking*, vol 7(1), pp. 98-110, 1999.
11. B. Caenegem, W. Parys, F. Turck, and P. Demeester, "Dimensioning of survivable WDM networks", *IEEE Journal on Selected Areas of Communications*, vol 16(7), pp. 1146-1157, 1998.
12. S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks part I-protection", *INFOCOM'99*, 1999.
13. M. Kodialam and T. Lakshman, "Dynamic routing of bandwidth guaranteed tunnels with restoration". *INFOCOM'00*, 2000.
14. B. Jager and D. Tipper, "On fault recovery priority in ATM networks", *IEEE international communication conference'98*, 1998.
15. T. Oh, T. Chen, and J. Kennington, "Fault restoration and spare capacity allocation with QoS constraints for MPLS networks", *IEEE Global Communication Conferences'00*, 2000.

16. H. Sakauchi, Y. Nishimura, and S. Hasegawa, "A self-healing network with an economical spare-channel assignment", *IEEE Global Communications Conference'90*, pp. 438-442, 1990.
17. Y. Liu, D. Tipper, and P. Siripongwutikorn, "Approximating Optimal Spare Capacity Allocation by Successive Survivable Routing", *IEEE INFOCOM'01*, 2001.
18. M. Herzberg, S. Bye, and A. Utano, "The hop-limited approach for spare-capacity assignment in survivable networks", *IEEE/ACM Transactions on Networking*, vol. 3, pp. 775-784, 1995.
19. R. Iraschko, M. MacGregor, and W. Grover, "Optimal capacity placement for path restoration in mesh survivable networks", *IEEE ICC'96*, pp. 1568-1574, 1996.
20. Y. Ye and S. Dixit, "On Joint Protection/Restoration in IP-centric DWDM-based optical transport networks", *IEEE Communications Magazine*, pp. 174-183, June 2000.
21. M. Sridharan, M. Salapaka, A. Somani, "Operating mesh-survivable WDM transport networks", *Terabit optical networking: architecture, control, and management issues*, Proceedings of SPIE, vol 4213, pp. 113-123, 2000.
22. A. Fumagalli and L. Valcarenghi, "Fast optimization of survivable WDM mesh networks based on multiple self-healing rings", *All Optical Networking 1999: architecture, control, and management issues*, Proceedings of SPIE, vol 3843, pp. 44-55, 1999.
23. R. Doverspke and J. Yates, "Challenges for MPLS in optical network restoration", *IEEE Communications Magazine*, February, Vol. 39 (2), pp. 89-97, 2000.
24. W. Grover, "The self healing network: a fast distributed restoration technique for networks using digital cross connect machines", *IEEE GLOBECOM'87*, pp. 1090-1095, 1987.
25. D. Dunn, W. Grover, and M. MacGregor, "Comparison of k-shortest paths and maximum flow routing for network facility restoration", *IEEE Journal on Selected Areas of Communications*, vol 2 (1), pp. 88-99, 1994.
26. Ramesh Bhandari, "Survivable networks: algorithm for diverse routing", *Kluwer Academic Publishers*, 1999.
27. J. Strand, A. Chiu, and R. Tkach, "Issues for routing in the optical Layer", *IEEE Communication Magazine*, Feb, 2001.
28. K. Kompella et al., "Link bundling in MPLS traffic engineering", IETF Internet draft, draft-kompella-mpls-bundle-05.txt, working in progress, March 2001.
29. D. Papadimitriou et al., "Inference of Shared Risk Link Groups", OIF contribution oif2001-066, July 2001.