A Heuristic Routing Algorithm for Shared Protection in Connection-Oriented Network

Shengli Yuan
Fujitsu Network Communications, Inc.
2891 Telecom Pkwy, Richardson, TX75082
e-mail: shengli.yuan@fnc.fujitsu.com

Jason P. Jue
Department of Computer Science, University of Texas at Dallas
Richardson, TX75083
e-mail: jjue@utdallas.edu

ABSTRACT

In a connection-oriented network, shared protection provides the same level of protection against single path failures as dedicated protection, with potentially higher network utilization. This paper lists the requirements of path protection and proposes a heuristic routing algorithm for shared protection provisioning. Simulations were conducted to verify the algorithm and to compare network utilization of shared protection to that of dedicated protection.

1. INTRODUCTION

A connection-oriented network provides end-to-end paths such as MPLS Label Switched Paths (LSPs) in a MPLS domain, and Optical Light Paths (OLPs) in a DWDM network. To prevent traffic loss, a path may be protected by another path, or so-called protection path. The protection path has the same source and destination as the original, or primary path. When the primary path fails, the protection path is activated to continue carrying traffic. Statistically, the failure probabilities of the paths ought to be independent; thus if the failure probability of one path is $p_f < 1$, the probability of traffic loss is reduced to $p_f^2 < p_f$.

Based on whether the sharing of network resources is allowed, a protection scheme can be categorized as dedicated protection or shared protection. In dedicated protection, different protection paths do not share common resource, which may be a physical transmission line, a SONET channel, or a WDM wavelength. Here we simply refer to these different resources as "links" between two nodes. The failure and activation of one protection path doesn't affect any other protection paths. The provisioning of this type of protection is simple, and its behavior is deterministic. On the other hand, in shared protection, multiple protection paths may go through common links. When a protection path is activated, other protection paths that share common links with it will have to be rerouted. When a common link fails, all protection paths that share the link need to be rerouted, therefore shared protection is more complex to provision and maintain.

However shared protection does offer one advantage over dedicated protection, i.e., it may offer higher network utilization. Assume that every path needs protection. In the dedicated case, the best network utilization would be 50%. On the other hand, for shared protection, since multiple paths share common links, the total number of links required for all the protection paths can potentially be much lower. If the failure probabilities of primary paths are statistically independent, we wouldn't expect multiple paths to fail simultaneously, in which case shared protection provides the same protection as dedicated protection. This concept can be illustrated in the following example.

Assume that there are 10 primary paths in a network, each with a failure probability of 0.01. At any moment the probability of path failure is $1- (1-0.01)^{10} = 0.9562$, of which single path failure probability is $10*0.01*(1-0.01)^9 = 0.9135$. Thus a single path failure counts for 95.534% of total path failures, for which shared protection performs as

well as dedicated protection. For the remaining 4.466% failures, i.e., multiple path failures, the performance of shared protection would depend on how effectively the other protection paths are rerouted when one protection path is activated.

Shared protection provides decent protection with much lower network resources; thus, the network can achieve higher utilization. Shared protection and dedicated protection schemes complement each other to offer more flexible solutions. Only the paths with the most strict protection requirement need to be dedicatedly protected. The remaining paths can be protected under shared protection and free up network resources, to either support more paths, or to protect paths that had no protection before.

The benefits of shared protection have attracted some research interests, especially for the emerging all optical DWDM network. ([1], [2]). This paper tries to establish a generic framework for shared protection routing that is applicable to connection-oriented network including the all optical DWDM network. The next section describes a heuristic routing algorithm for setting up shared protection paths. Section 3 discusses other issues related to shared protection, and section 4 presents simulation results for the shared protection scheme. Section 5 concludes the paper.

2. ROUTING ALGORITHM

We first list the requirements for the shared protection routing algorithm.

Requirement 1. A routing algorithm for path protection is subjected to the risk disjointing constraint, i.e., a primary path and its protection path must not undertake the same risk(s); otherwise the same failure may cause both paths to fail. This requirement applies to both dedicated and shared protections.

For each risk, we can assign a unique number, a **Risk ID**. If a link in the network is subjected to multiple risks, the collection of the risk IDs describes all the risks for the link, and the collection of the risk IDs of all of a path's links describes the path's total risks. This collection of risk IDs is called the **Risk Vector**. For instance, in a DWDM network, a lightpath consists of two links, l_1 and l_2. l_1 runs across two bridges, A and B. l_2 crosses one bridge, C. The failure of any bridge can cause the lightpath to fail. If we assign risk ID 2 to A, risk ID 5 to B and risk ID 3 to C, then {2, 5} is l_1's risk vector and {3} is l_2's risk vector. The lightpath's risk vector is then {2, 5, 3}. With the concept of risk vector, the risk disjointing constraint requires that there must not be any common risk IDs in the risk vectors of a primary path and its protection path.

Figure 1 gives another example. There are two optical lightpaths in a DWDM network, l_1 = abc and l_2 = abdc. Link ab of l_1 and link ab of l_2 are on the same fiber between node a and b, so both links are subjected to the same fiber failure. We can assign a common risk ID 2 to the fiber ab.

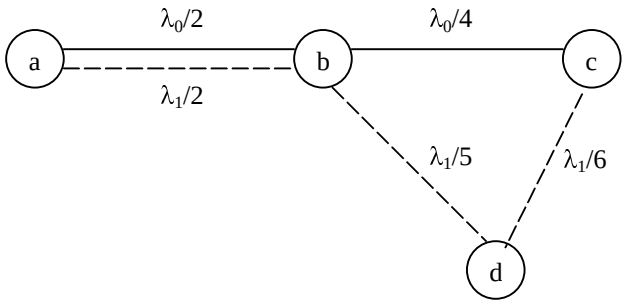

Figure 1: Risk IDs in a DWDM network

Link bc of l_1 and link bd, dc of l_2 are on different fibers in disjoint terrain. The risks of fiber failure are different. We assign a risk ID of 4 to bc, 5 to bd and 6 to dc. Combining all links, we have l_1's risk vector {2, 4} and link l_2's risk vector {2, 5, 6}. It is clear that l_1 and l_2 do not satisfy the risk disjointing constraint because of their common risk ID 2.

Requirement 2. Both the primary path and protection path must be routed in order to claim success. This requirement applies to dedicated protection as well. It is up to the network operator to handle the failure. For instance, the operator may decide that the request is blocked, or may make the primary path unprotected. Due to the risk disjointing constraint, if the risk IDs of the primary path's risk vector appear on many unused links, those links will have to be excluded from the protection path routing. Clearly, our routing algorithm should have a preference for the links with uncommon risk IDs. This applies to the routing of both the primary path and protection path.

Requirement 3. If a link is already taken by a protection path, that link should be shared as much as possible by subsequent protection paths, up to the maximum number allowed on that link. The purpose is to reduce the number of total links taken by protection paths in the network. Therefore shared links should be given higher preference for routing the protection path.

Requirement 4. If multiple protection paths share common links, those protection paths should not activate simultaneously. In order to achieve this, the routing algorithm must disallow protection paths from sharing common links if their primary paths have common elements in their risk vectors.

Requirement 5. For multiple routing requests, we can process the requests either one at a time, or all at once. The latter has a higher chance of obtaining more optimal routes, but in a distributed network, routing requests often arrive at different nodes of the network. It is more practical and simpler to route requests one at a time. Once we develop the algorithm for a single request, we can handle the multi-request case by running iterations of the algorithm and choosing the most optimal routes.

Requirement 6. Specific networks may impose additional requirements. For instance, a DWDM network without wavelength conversion has the wavelength continuity constraint, in which case we may need to run iterations of the algorithm, one for each wavelength.

The heuristic routing algorithm we are proposing is a modified OSPF routing algorithm. Every node has global network topology and complete information of every link in the network. In addition to OSPF generic link state information, all nodes have information on every link about,

1. The link's risk IDs.
2. Whether the link is already taken by a primary path.
3. Whether the link is running a protection path. If so, the risk IDs of the primary paths are also known. If many protection paths share this link, the amount data on this item is potentially large.
4. The maximum number of shared protection paths the link supports. By lowering this number, we can decrease the amount of data for item 3.

Based on the above information, we will modify the link costs such that the OSPF algorithm generates the routes that meet all of the requirements. The algorithm is run at the source node and returns explicit routes.

For routing a primary path, we modify the cost of every link as follows:

1. Set the cost to infinity if a link is already taken by a primary or protect path.
2. Increase the cost if the risk IDs of the link have high occurrence in the network. For instance, for each occurrence of the risk ID, we increase the link cost by a certain percentage, or by a fixed amount.
   Generically, the resulting cost, c_i', of the i-th link, is a function of the original cost c_i and the number of occurrences of its risk ID, n_{ri}, i.e., $c_i' = f(c_i, n_{ri})$, and $c_i' > c_i$. This function may be either linear or non-linear.
3. Increase the cost if the link has the same risk IDs of existing primary paths.

Both 2 and 3 potentially increase the degree of sharing when routing the protection path, but item 3 requires the source node to be aware of the risk IDs of all existing primary paths, which may be a difficult task.

After routing the primary path, we modify the link cost to route its protection path:

1. Set the cost to infinity if a link is already taken by a primary path. Effectively this link is removed.
2. Set the cost to infinity if a link is running the maximum number of protection paths. Effectively this link is removed.
3. Set the cost to infinity if a link has a common risk ID with the risk vector of the primary path. If *Requirement 1*, the risk disjointing constraint, is tolerant, then the link cost may be set to a large positive number instead of infinity.
4. Set the cost to infinity if a link is running a protection path whose primary path has common risk IDs with the current primary path. If *Requirement 4* is tolerant, the link cost may be set to a large positive number instead of infinity.
5. Increase the cost if the risk IDs of the link have high occurrence in the network as in item 2 of the above primary path routing algorithm.
6. For the remaining links, decrease the cost if a link is running at least one but less than the maximum number of protection paths. The lower link cost makes the link more preferable and increases the degree of sharing. For instance, for each shared protection path still allowed on this link, we decrease the link cost by a certain percentage, or by a fixed amount, until it reaches minimum. The minimum cost should be a positive number.
   Generically, the resulting cost, c_i', of the i-th link, is a function of the original cost c_i and the number of protection path still allowed on this link, n_{pi}, i.e., $c_i' = g(c_i, n_{pi})$, and $c_i > c_i' > 0$. This function may be either linear or non-linear.

Now we illustrate the algorithm in an example with the network shown in Figure 2.

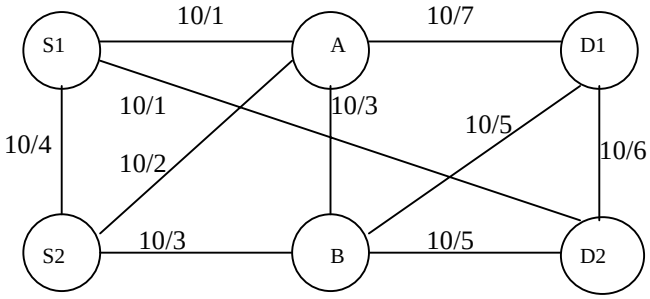

Figure 2: Example network with link costs and risk IDs

There are two routing requests. The first request asks for a path from node S1 to node D1. The second request asks for a path from S2 to D2. Each link has a cost of 10 and risk IDs as marked in the figure. All links and paths are bi-directional. With dedicated protection, one of the paths would have to be unprotected.

With shared protection, S1D1 is routed first. For the primary path, since it is the first path in the network, we only need to modify the link costs based on risk ID occurrence before running OSPF. For each extra occurrence of a risk ID, we increase the link cost by 10%. The resulting network is shown in Figure 3. Running OSPF yields a primary path S1-A-D1. Its risk vector is {1,7}. The total cost is 20.

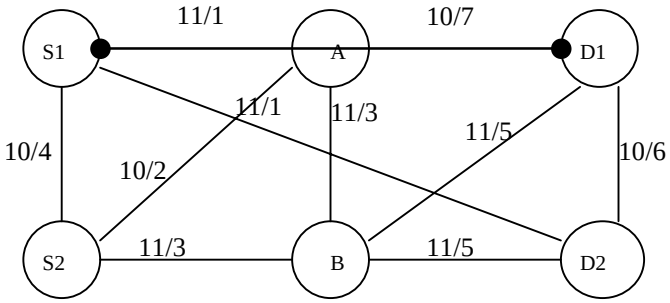

Figure 3: Network with modified link costs

Next we route S1D1's protection path. Since it is the first protection path in the network, we only need to remove the primary path and the links with common risk IDs with the primary path from Figure 3. We then obtain the protection path S1-S2-B-D1, as shown in Figure 4. Its risk vector is {3, 4, 5}. This path satisfies the risk disjointing constraint. The total cost is 30.

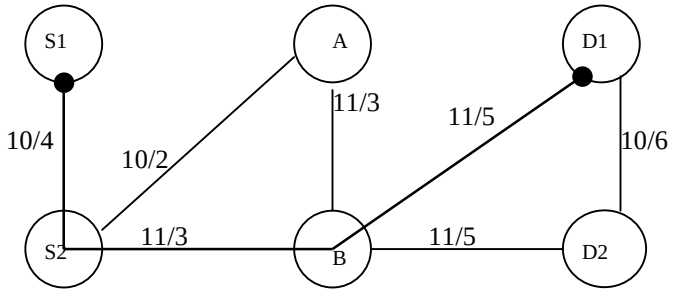

Figure 4: Network after routing the first request

We now process the second request. To route the primary path from S2 to D2, we need to remove the links on primary path S1-A-D1 and its protection path S1-S2-B-D1 from Figure 3. We also need to increase the cost on links that have common risk ID with the risk vector of primary path S1-A-D1, {1,7}. The resulting network topology is shown in Figure 5. OSPF yield the second primary path S2-A-B-D2. Its risk vector is {2, 3, 5}, and its total cost is 30.

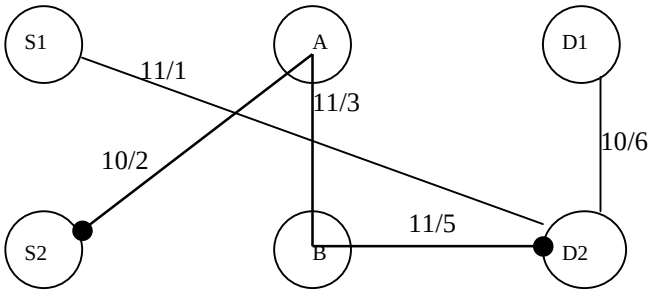

Figure 5: Network with modified link costs

To route the protection path, we need to remove all links of the two primary paths from Figure 3 and links with risk ID 2 or 3 or 5. Then we decrease by 10% the cost on links that are running the first protection path. The resulting topology is shown in Figure 6. The protection path becomes S2-S1-D2. Its risk vector is {1, 4}, and its total cost is 20.

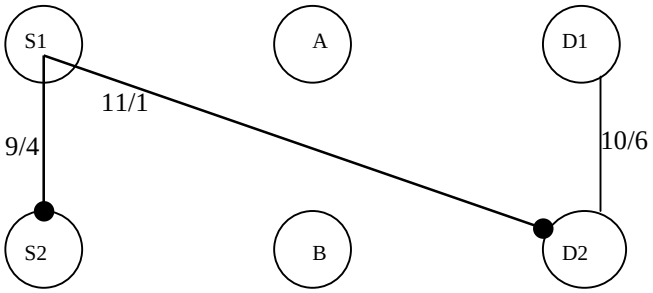

Figure 6: Network after routing the second request

Finally we obtain the network with all paths routed as shown in Figure 7. Solid lines indicate the primary path; dash lines indicate the protection path. Link S1S2 is shared by two protection paths. With shared protection, both primary paths are protected, which is infeasible with dedicated protection.

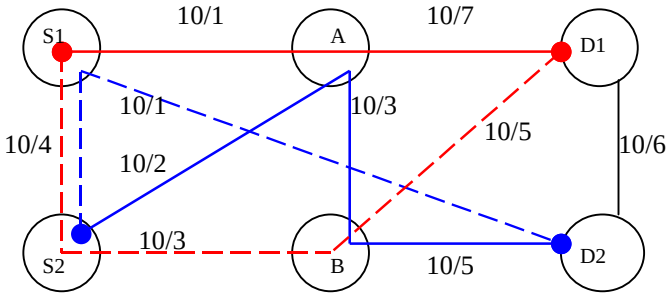

Figure 7: Network with both requests routed

3. ISSUES AND IMPROVEMENTS

Additional Data and Computation Requirements. Compared with the algorithm for dedicated protection, the algorithm for shared protection requires links of protection paths to have the extra knowledge of the risk IDs of the related primary paths. If the maximum number of shared protection path allowed is M, and the number of nodes in the network is N, then on each link, the number of risk IDs is on the order of O(M*N). In order to find out whether the paths have any common risk IDs with the target primary path, the algorithm needs an extra O(NlogN + (M*N)log(M*N)) computations on each link.

Path Removal. When a primary path and its protection path are torn down, the resources that were once occupied are freed up. If we reroute the remaining paths, we may get more optimal routes ([3]). This applies to both dedicated protection and shared protection.

With either type of protection, rerouting primary routes may cause traffic hits. It may be more practical to reroute only the protection paths from time to time.

Protection Activation. When multiple protection paths share common link(s), only one can be activated at a time. In order to allow multiple activation, we can do one of the following after a protection path is activated:

- Reroute the failed primary path. Once the new primary path is established, move traffic onto it from the protection path, then deactivate the protection path. This approach requires one reroute, plus signaling for path deactivation. The probability of traffic hit is high when traffic is moved to the new primary path.
- Leave the traffic on the activated protection path and make it the new primary path. Establish a new protection path for it, as well as reroute all other protection paths that shared common link(s) with it. This approach doesn't introduce a traffic hit, but it requires rerouting multiple protection paths as well as signaling associated with the rerouting.

The network operator should decide which option to take. If the end user has a high tolerance for traffic hits, the first approach is clearly more suitable, since, when the network utilization is relatively high, rerouting multiple protection paths has higher failure probability than rerouting only the primary path. On the other hand, if rerouting multiple protection paths is not an issue, then the second approach may be considered.

Signaling. Shared protection requires more signaling than dedicated protection. In addition to path establishment and removal, protection activation needs extra signaling as described above. Signaling can be done either in-band or out-of-band, depending on the network type.

Maximizing Link Sharing. *Requirement 4* prohibits link sharing among protection paths whose primary paths have common risk IDs. However, if we break each primary path into multiple segments, we may find risk disjointing segments of the primary paths. We can then establish segment-based protection instead of path based, and achieve higher sharing this way ([4]).

4. SIMULATIONS

We ran our simulations using the 16-node, 25-link NSFNET backbone topology as shown in Figure 8. The cost of every link is assumed to be 100, and the risk IDs are as marked in Figure 8.

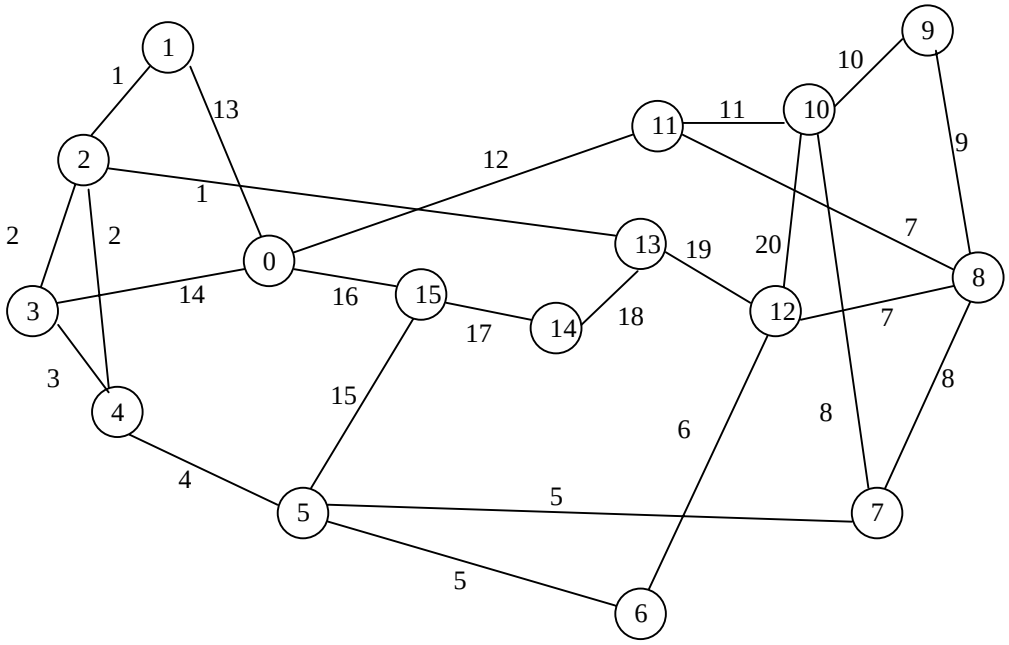

Figure 8: NSFNET Backbone

Let the bandwidth of each link be BW. A primary path takes one unit of bandwidth, as does a dedicated protection path. When multiple protection paths share a common link, they take one unit of bandwidth. We ran simulations with various values of BW.

We used standard Dijkstra's Shortest Path algorithm for dedicated protection and the heuristic algorithm described earlier for shared protection.

For shared protection, we increased a link's cost by 100% for each occurrence of its risk ID when routed primary and protection path. We also decrease a link's cost by 50% if it ran a protection path when routed protection path.

We randomly generated 500 source-destination pairs as the routing requests. Then we compared the number of successes for dedicated protection and shared protection. For shared protection, we also changed the maximum number of shared protection paths, M, allowed on each link.

We list the simulation results in Table 1. The first row contains the numbers of path pairs being successfully routed with dedicated protection, with various link bandwidths, BW. The remaining rows contain the results for shared protection, with different link bandwidths, and sharing degrees, M.

Protection Type		Number of Path Pairs Routed			
		BW = 2	BW = 5	BW=10	BW=20
Dedicated		8	18	36	67
Shared	M = 2	10	26	53	98
	M = 4	12	34	69	128
	M = 8	13	37	74	134
	M = 16	13	37	74	134
	M = 32	13	37	74	134

Table 1. Simulation Results

Two observations can be made from the results. First, shared protection routes more requests than dedicated protection. It confirms our earlier analysis that shared protection offers higher network utilization. It is also worth noting that network utilization can increase fairly significantly even with the minimum amount of sharing. For example, when the link bandwidth is 10, 36 primary-protection path pairs are routed successfully under dedicated protection. But with only a sharing of two, 53 pairs are routed under shared protection, an increase of nearly 50% in network utilization.

Secondly, in shared protection, higher degreed of sharing beyond 8 do not provide significant additional gains of network utilization. This is an area deserves further investigation.

5. CONCLUSION

Shared protection provides a decent level of protection with less network resource than dedicated protection. It complements the current two level protection of no protection or dedicated protection, and offer three-level protection. It does so at the expense of extra signaling, data, computation, and path rerouting. Practical implementations can use a small degree of sharing to reduce the extra expense while still achieving higher network utilization.

6. REFERENCES

[1] Ching-Fong Su, Xun Su, “Protection path routing on WDM networks”, OFC2001, TuO2-1.
[2] S. Ramamurthy, B. Mukherjee, “Survivable WDM Mesh Networks: Part I – Protection”, Proceedings, IEEE INFOCOM ’99, Vol. 2, 1999, pp.744-751
[3] Vishal Anand, Chunming Qiao, “Dynamic Establishment of Protection Paths in WDM Networks: Part I”, Proceedings, 9th International Conference on Computer Communication and Networks (IC3N 300), pp.198-204
[4] Pin-Han Ho, H.T. Mouftah, “SLSP: A new path protection scheme for the optical Internet”, OFC2001, TuO1-1