Dynamic Routing in Survivable WDM Networks

Arun K. Somani
Dependable Computing & Networking Laboratory
Department of Electrical and Computer Engineering
Iowa State University, Ames, IA 50011
arun@iastate.edu

Murari Sridharan
RSoft Design Group, Eatontown, NJ 07724
murari@rsoftdesign.com

R. Srinivasan
Department of Electrical and Computer Engineering
University of Arizona, Tucson, AZ 85721
srini@ece.arizona.edu

Abstract

Changing trends in backbone transport networks towards dynamic path provisioning and evolving optical technologies have motivated the study of dynamic routing algorithms in the context of Multi Protocol Label Switching (MPLS) based networks. As the network size increases, it is impractical to maintain detailed information regarding every connection that is established in the network. Hence, routing algorithms need to be designed that rely on partial information.

We study the effect of dynamic routing algorithms in the context of routing with partial information. We observe that dynamic routing algorithms perform better than static routing algorithms. However, the network capacity allocated for backup resources is found to be high. In order to alleviate this problem, we study the effectiveness of protecting a connection using a simple segmented path protection scheme. We compare the performance of the partial information routing algorithm with and without segmented path protection through simulation studies, based on metrics such as the call blocking probability and capacity redundancy. Our results indicate that with a simple segmentation scheme, the capacity efficiency of partial information routing can be significantly improved, upto 20 to 30% depending on the topology. The capacity savings obtained leads to a modest improvement in call blocking probability. It is also observed that segmented protection offers better performance as compared to path protection under partial information scenario, which is contrary to the performance obtained with complete information. Hence, it would be a better alternative to employ segmented protection in large networks where obtaining complete network state information is impractical.

I. INTRODUCTION

The Internet is growing faster than ever with traffic across the core of the network quadrupling over the last year. With services ranging from video conferencing to large multimedia downloads, the per user bandwidth consumption has increased considerably. Network survivability is a serious concern for today's businesses as they rely heavily on a reliable and continuously available high-speed communications infrastructure. With millions of wavelength-miles laid out in typical global and nation wide networks, fiber optic cables are among the most

The research reported in this paper is funded in part by the National Science Foundation under grant ANI-9973102 and Defense Advanced Research Projects Agency and National Security Agency under grant N66001-00-1-8949.

prone to failures. For instance, GTS Trans European Network (formerly Hermes Europe Railtel) estimates an average of one cable cut every four days on their network [1]. Therefore, it is imperative to design networks that can quickly and efficiently recover from failures.

Survivable network architectures based on mesh (arbitrary) topologies offer better capacity efficiency and flexible re-routing capability around failed links. This is a result of the route diversity in arbitrary topologies, which is highly sensitive to the average nodal degree. Several survivability paradigms have been explored for surviving single link failures in mesh-based networks [2], [3], [4], [5].

Several methods have been proposed for joint working and spare capacity planning in survivable optical networks [2], [3], [4], [6], [7]. These methods consider a static traffic demand and perform routing and wavelength assignment (RWA) to optimize network cost assuming different cost models and survivability paradigms. The application of these algorithms are restricted for the network provisioning and offline reconfiguration phases due to their high computational complexity. For network operation under dynamic traffic scenarios, direct use of such methods for online reconfiguration remains limited to small networks with few tens of wavelengths.

Several heuristics and decomposition techniques [2], [8], [9] are being explored to significantly reduce the computational complexity of the RWA optimization problems. However, these techniques are centralized and require per-flow information, hence does not scale well with increasing network size. This motivates the need for developing source-based or distributed routing algorithms with partial information that is obtained from traffic engineering extensions to routing protocols.

A. *Related work*

Dynamic routing in wavelength-routed WDM networks has been studied extensively in the literature and several dynamic routing algorithms have been proposed [5], [10], [11], [12], [13], [14]. A methodology for dynamic routing of sub-wavelength traffic in WDM grooming networks is developed in [15]. The effect of grooming complexity and dispersity routing, where higher capacity requests are broken into multiple unit capacity requests, are analyzed. These studies are limited to establishing one connection per request and do not guarantee protection paths.

Routing dependable connections under dynamic traffic was studied extensively in [16], [17], [18]. Dynamic routing of restorable bandwidth guaranteed tunnels is studied in [19]. They study dynamic routing using Integer Linear Programming (ILP) formulation under three information scenarios: none, complete, and partial information. These approaches do not scale well with the network size either due to the prohibitively large amount of information that needs to be maintained or high computational complexity involved in solving ILP formulations. Recently multi-segment protection approaches are gaining attention in the context of shared risk link groups (SRLG) [20], [21].

In this paper, we study the performance of dynamic routing algorithms under partial information scenario. Despite the improvement in performance achieved through dynamic routing, network capacity allocated for protection paths is observed to be very high. We develop a simple segmented path protection scheme to improve capacity efficiency. We compare the performance of the partial information routing algorithm with and without segmented path protection through simulation studies, based on metrics such as the call blocking probability and capacity redundancy. Our results indicate that with a simple segmentation scheme, the capacity efficiency of partial information routing can be significantly improved, up to 20 to 30% depending on the topology. The capacity savings obtained leads to a modest improvement in call blocking probability. It is also observed that segmented protection offers better performance as compared to path protection under partial information scenario, which is contrary to the performance obtained with complete information. Hence, it would be a better alternative to employ segmented protection in large networks where obtaining complete network state information is impractical.

The rest of the paper is organized as follows. Section II introduces the framework for dynamic routing of primary and backup connections under the partial information model and details the segmented path protection scheme employed. Section III-B discusses the experimental setup, performance metrics, and performance results of different routing algorithms. Section IV presents our conclusions and discusses further possible improvements based on our observations.

II. DYNAMIC ROUTING WITH PARTIAL INFORMATION

In this section, we develop the framework for dynamic routing of primary and backup connections under the partial information model and segmented path protection scheme.

Every link in the network is denoted by a *link-state vector*. The vector consists of a set of properties associated with a link, eg. available bandwidth, primary capacity, backup capacity, hop-length, fiber length etc. Each entity in the vector is referred to as a *metric*. Every path from a source to destination has a *path-vector* that is obtained by combining the link-state vectors of the links in the path. Note that the link vector is a special case of a path vector when the path has only one link.

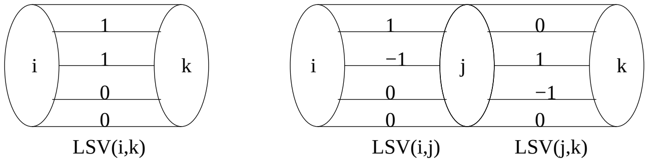
In WDM networks, the metrics can be classified either as path-specific or wavelength-specific. Path-specific metrics are those metrics that depend only on the route from a source to destination and are independent of the wavelength used. One example of path metric is the hop-length. The usage of a wavelength as a primary or backup on a link is an example of a wavelength-specific metric. Various dynamic path selection algorithms can be developed based on the above specified metrics.

Every node in the network is assumed to maintain global network state information through a link-state protocol. The information available for each link (i, j) is the status of each wavelength, set to 1 if used as a primary, 0 if used as backup, and -1 if available. Note that, although we know if a wavelength is used as a primary or backup (partial information), we still do not know which node pair's flow is using the wavelength as a primary, or which flows are sharing the wavelength for backup (full information).

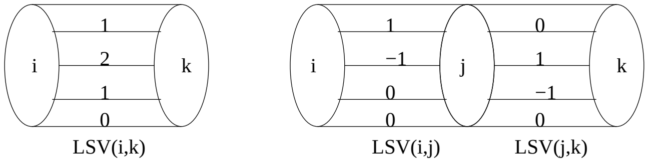
Dijkstra's shortest path algorithm is extended to the above link-state vector, referred to as extended Dijkstra's shortest path (EDSP) algorithm and is employed at every node in the network. The EDSP algorithm uses the link-state vector as defined above instead of a single metric that is traditionally used. The EDSP algorithm has two important operations: (1) combining two path vectors and (2) selecting the best path vector. A path vector is a combination of link state vectors.

We illustrate the path vector combine function using an example shown in Figure 1.

As defined earlier, associated with each wavelength in a link-state vector lsvij is a status variable which is set to 1 if used as a primary, 0 if used as backup, and -1 if available. By examining each lsv, we can determine the total primary wavelengths by adding all the wavelengths whose status value is 1, the total backup wavelengths by adding wavelengths whose status value is 0, and total available wavelengths by adding wavelengths whose status value is -1. Although a status variable is indexed and maintained for each wavelength in a link, in the discussions to follow, we use a generic status variable to represent all wavelengths. If the combine function is the maximum of the status variables, i.e., $lsv_{ik}.Status = max(lsv_{ij}.Status, lsv_{jk}.Status)$. By examining the combined $lsv_{ik}.Status$ as shown in Figure 1a, the following information can be obtained. Since we are interested in wavelength continuous paths, any wavelength λ whose status in the combined lsv is 1 cannot be used as there is atleast one link in the corresponding wavelength continuous path where λ is being used as a primary wavelength. Any λ whose



a) lsvik = max(lsvij.Status,lsvjk.Status)



b) lsvik = abs(lsvij.Status)+abs(lsvjk.Status)

Fig. 1. Combine function for two path vectors.

status in the combined lsv is -1 means the wavelength continuous path corresponding to λ is available on all links on the path. Similarly, any λ whose status in the combined lsv is 0 means the wavelength continuous path has λ either being used as a backup or is available. Therefore, to check if a wavelength continuous path can be established or not, it is sufficient to check the status of the wavelengths in the combined path vector.

The second operation of selecting the best path vector from a given set of path vectors is defined by a specific path selection policy. For example, the traditional shortest path algorithm selects a path with minimum hop length. To understand how different combine functions have different effects on the path selection, and hence the routing algorithm, refer to Figure 1. Let us assume that we would like to choose a path which has maximum number of backups (already reserved) available. If we use the combine function in Figure 1a, we will get paths where, in the worst case only one link along the path has the corresponding wavelength being used as backup and on all other links the wavelength is available. In such a scenario, once the path is selected we need to reserve capacity on all links where the wavelength was available. On the other hand, if we assume that the combine function as shown in Figure 1b is used, i.e., $lsv_{ik}.Status = abs(lsv_{ij}.Status) + abs(lsv_{jk}.Status)$. Now in this case, as can be seen in the figure, any λ whose status in the combined lsv is 0, means that λ is being used as backup on all links on the path. In such a scenario, once the path is selected, we do not need to make any excess reservations. It is to be noted here that, we merely use the above illustration to explain the process of combine and selection, actual decision on whether we can multiplex a given connection's backup on to an already reserved backup wavelength will depend on whether such an assignment will violate the 100% protection guarantee. However, it can be easily shown that 100% protection guarantee can be achieved.

A. Routing algorithm

We first discuss the basis upon which the routing algorithms are developed. The key is to use the aggregate information in the partial information scenario to obtain significant gains in network performance. Let us assume that the primary path for an arriving request has been selected. The maximum number of primary connections on a link along the path is referred to as the *conflict* created upon routing the current primary connection, inclusive of the primary. For any potential backup path which is disjoint from the primary, it is required that the path has atleast C wavelengths that are either available or used as backup. Note that, as explained earlier in the section, this depends on the kind of combine function used. Now suppose if we cannot find C backup wavelengths, then assign one wavelength continuous path without any sharing.

It is a fairly simple exercise to see that in both the above cases, 100% protection guarantee is maintained.

We will illustrate this with an example. Figure 2 shows the snapshot of a part of the network.

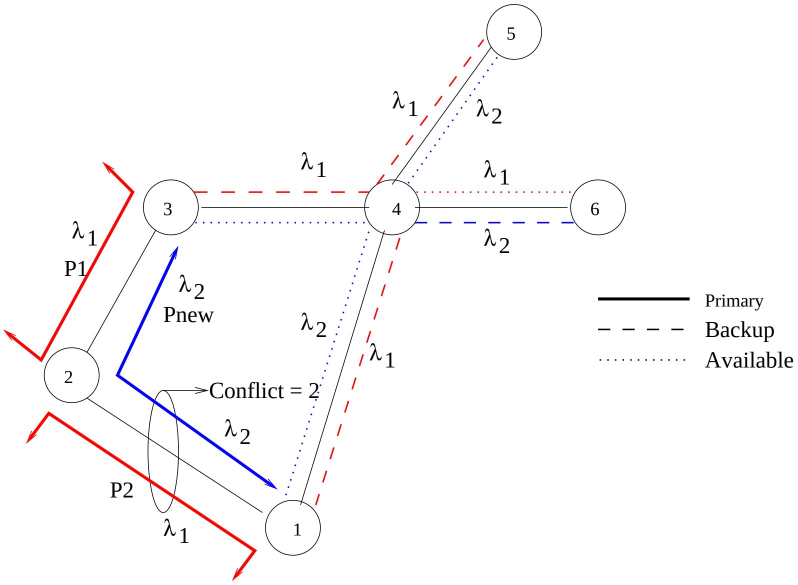


Fig. 2. Snapshot of a part of the network to illustrate protection guarantee

Assume that there are two wavelengths λ_1, λ_2 on each link. There are two primary paths P1 and P2 which use, as part of its route, links 2-3 and 2-1 respectively. Assume that a primary path for the new request between (3,1) has been selected to be 3-2-1 (Pnew) on λ_2. Note that the primary wavelengths cannot be shared. The maximum conflict created by Pnew is $C = 2$. Now the claim is that as long as there are C wavelength-continuous backups assigned on the selected backup path for this new request, 100% protection guarantee can be ensured. The other case is to assign one wavelength continuous path on 3-4-1 without any sharing. It is trivial to see that in the latter case, since one dedicated wavelength continuous path has been assigned for the new request, 100% guarantee is ensured.

Let us examine the first case of the claim that there are C wavelength-continuous backup paths assigned. Suppose, if it is not the case then it would result in the violation of 100% guarantee. To understand this better consider the example shown in Figure 2. Suppose that we were to assign one wavelength continuous path on 3-4-1 using λ_1. Here on both 3-4 and 4-1 λ_1 has been already reserved as backup for some earlier request(s). Assume that the backup path for P2 had been assigned λ_1 on link 3-4. Suppose of link 1-2 fails, both Pnew and P2 contend for the same wavelength λ_1, thereby violating 100% protection guarantee. Note here that instead of λ_1, if we were to assign the backup path for the new request on 3-4-1 using λ_2 which is available on both links, then this becomes a dedicated backup for Pnew and as shown earlier is a trivial case.

It is clear that in some cases, due to lack of information, more backup wavelengths may be reserved than required. In the above example, if P2 were not using any of links 3-4 or 4-1 for backup, we could have assigned the backup path for the new request on 3-4-1, sharing the already reserved wavelength λ_1. The inherent assumption here is that the algorithm had ensured protection guarantee for all earlier requests before the arrival of the new request. This is the best decision that any routing algorithm can make to provide 100% protection guarantee given the partial information scenario. So we need to exploit the excess reservations that may be made in some cases to our advantage by reusing such links where backup wavelength chunks are already available.

It is to be noted that, we need to ensure C wavelength continuous paths, instead of merely checking for C wavelength capacity to be either available or reserved as backup on all links on the backup path. This is because when the backup has to be actually assigned for this request, we may not find a common wavelength that is available on all paths, although the required capacity is present. Although we provide C possible backup wavelengths for some arriving requests, we need to switch the connection onto one specific wavelength when a failure occurs. There are some issues as to the order in which the affected requests pick the backup wavelength to switch to. A wrong assignment here may result in some requests being blocked. This can be avoided by some extra processing to ensure that the switch to the backup path is done without blocking any requests.

The optimality metric for our dynamic algorithm is stated as follows: route the primary on the shortest path, and try to route backup on links where backup wavelengths are already reserved by earlier requests, at the same time ensuring 100% protection guarantee. In [22], we developed several routing algorithms based on our optimality definition and all the observations discussed earlier. We choose the algorithm that performed the best and try to improve its performance using segmented path protection. The algorithm is as follows.

Least Conflict Path Unconstrained Primary ($LCPUP$): The primary is routed using the least conflict path, i.e., minimize the maximum conflict (*minmaxconflict*) created on the primary path. This is achieved in the EDSP framework, by using a maximum metric in the link-state vector combine function for primary capacity, and choosing the minimum value when comparing path vectors to update the Dijkstra cost. The metric used for routing the backup path is to maximize the minimum backup capacity available (*maxminbackup*). This metric attempts to find a path for which minimum backup capacity available on the path is maximum. This can be achieved using the EDSP framework, by using a minimum metric in the link-state vector combine function for backup capacity, and choosing the maximum value when comparing path vectors to update the Dijkstra cost.

B. Segmented Path Protection

We extend the $LCPUP$ algorithm for segmented path protection as follows. We use multiple protection segments for each working path. The algorithm is as follows. The primary is routed using the *minmaxconflict* metric. For the backup path, we calculate the excess spare capacity reservation required to route the backup using the *maxminbackup* metric (same as the $LCPUP$ case). Then we compute the excess spare reservation obtained with segmentation. As mentioned earlier, the *maxminbackup* metric tries to route backup path using links where backup wavelengths are already reserved by earlier requests. Excess spare reservation is computed as the number of available wavelengths (not reserved by any earlier request) that have to be reserved to route the current request.

We employ two simple segmentation schemes. In the first scheme $Seg2$, we find the intermediate node ($\lfloor L/2 \rfloor$, Where L is the length of the primary path). Then we find two backup paths using the *maxminbackup* metric, one from source to the intermediate node and the second from intermediate node to the destination. Since we split the working path into two segments that are link disjoint, their backup segments can share wavelengths. This is expected to reduce the spare capacity usage. In the second scheme $SegLink$, we split the working path into L link segments (where L is the hop length of the working path). For each link, we compute a backup using the *maxminbackup* metric. Finally we choose route the request using the scheme that minimizes the excess reservation.

III. PERFORMANCE EVALUATION

The performance of the algorithms described in the previous section are evaluated on the 14 node 22 link NSFNet network topology. When a request arrives at a node, primary and backup paths are chosen using the above mentioned routing algorithms. FirstFit wavelength allocation policy is used for both primary and backup paths, with primary choosing from wavelengths in descending order, and backup in ascending order.

A. Experimental setup

The experimental setup for the simulation is based on the following assumptions.

1) The arrival of requests at a node follows a Poisson process with rate λ and are equally likely to be destined to any other node;
2) The holding time of the requests follow an exponential distribution with unit mean;
3) The capacity requirement of a request is restricted to a full wavelength; and
4) Every link has 32 wavelengths.

The requests are generated independently at a rate of $N\lambda$, where N denotes the number of nodes in the network. The requests are equally likely to have any of the N nodes as its source. The generated requests are fed to the different networks running in parallel and their performances are measured. A total of 5×10^5 requests were generated with performance metrics being measured in batches of 10^5 requests. The average of the performance metrics over observed five set of values are reported in the results.

The performance metrics measured are the request blocking probability, redundancy. Since every request is assigned a primary and a backup path, and since the network revenue is gained as a result of accepting more primaries, and the backup capacity is idle until a failure occurs, all the measurements and performance comparisons are made specific to the primary paths. The blocking probability is computed as the ratio of the number of blocked requests to the total number of requests generated. Redundancy is defined as the ratio of total spare to working capacity in the network.

B. Results

The capacity redundancy of $LCPUP$ is compared with two segmented protection schemes $Seg2$ and $SegLink$. We choose the two segment protection schemes by examining the average primary path lengths of the accepted connections which is around 2.11 for NSFNet. So it makes sense to split the working path into either two segments or L segments for every link in the path. We chose to compare with $LCPUP$ based on the experiments performed in [22]. $LCPUP$ performed the best for partial information routing.

The capacity redundancy for $LCPUP$ versus $Seg2$ is shown in Table I. The redundancy in the network is reduced by using $Seg2$ scheme. When the load in the network is low, 17 - 34% reductions in capacity is obtained. The corresponding blocking probability values for $LCPUP$ versus $Seg2$ are shown in Table II. The blocking improves modestly for the corresponding loads. Around 10-30% improvements are obtained for low loads, and for arrival rate 50, there is 72% improvement in blocking performance. As the load increases, performance of these schemes converge as expected, in fact $Seg2$ performing worse at high loads.

The capacity redundancy of the $SegLink$ scheme is compared with $LCPUP$ and the results are shown in Table III. Similar trends are seen in this case with some slight improvements at higher loads as compared to $Seg2$ scheme. For low loads, 19 - 36% reductions in capacity is obtained. The blocking probability values for $LCPUP$ versus $SegLink$ are shown in Table IV. Again modest improvements in call blocking can be seen. For higher loads, performance converges as expected.

The important thing to note here is that a simple $Seg2$ scheme works well to improve network performance, rather than a complicated segmentation scheme which has a high overhead on call setup time. The $Seg2$ scheme can be easily extended with minimal increase in state and fairly simple signaling. The reduced segement size can also improve restoration time guarantees.

TABLE I

Capacity Redundancy ($LCPUP$ Vs $Seg2$).

Arrival Rate	LCPUP	Seg2	% Improvement
50	2.7203162	1.7702888	34.92341809
60	2.4216648	1.7567562	27.45667361
70	2.1073996	1.7358794	17.62931909
80	1.8761902	1.6967922	9.561823743
90	1.7212094	1.6363718	4.928952863
100	1.6167616	1.5739436	2.648380565
130	1.4596456	1.4471596	0.855413122
150	1.4025966	1.397595	0.35659576
200	1.3192184	1.3191904	0.002122469

TABLE II

Blocking Probability ($LCPUP$ Vs $Seg2$).

Arrival Rate	LCPUP	Seg2	% Improvement
50	2.20000E-05	6.00000E-06	72.72727273
60	9.80000E-05	6.60000E-05	32.65306122
70	1.71600E-03	1.53800E-03	10.37296037
80	1.20780E-02	1.17780E-02	2.483854943
90	3.57640E-02	3.55960E-02	0.469746113
100	7.17920E-02	7.18300E-02	-0.052930689
130	1.81166E-01	1.81500E-01	-0.184361304
150	2.42646E-01	2.42518E-01	0.052751745
200	3.55040E-01	3.55960E-01	-0.259125732

TABLE III

Capacity Redundancy ($LCPUP$ Vs $SegLink$).

Arrival Rate	LCPUP	SegLink	% Improvement
50	2.7203162	1.731061	36.36544899
60	2.4216648	1.7153626	29.16597706
70	2.1073996	1.6979632	19.42851275
80	1.8761902	1.6638772	11.31617679
90	1.7212094	1.6140468	6.22600597
100	1.6167616	1.5556842	3.777761669
130	1.4596456	1.4427846	1.155143413
150	1.4025966	1.3947604	0.558692357
200	1.3192184	1.3181978	0.077363991

TABLE IV

BLOCKING PROBABILITY ($LCPUP$ VS $SegLink$).

Arrival Rate	**LCPUP**	**SegLink**	**% Improvement**
50	2.20000E-05	1.20000E-05	45.45454545
60	9.80000E-05	5.20000E-05	46.93877551
70	1.71600E-03	1.58600E-03	7.575757576
80	1.20780E-02	1.07100E-02	11.32637854
90	3.57640E-02	3.43600E-02	3.925735376
100	7.17920E-02	7.15200E-02	0.378872298
130	1.81166E-01	1.81634E-01	-0.258326618
150	2.42646E-01	2.41798E-01	0.349480313
200	3.55040E-01	3.58306E-01	-0.91989635

IV. CONCLUDING DISCUSSIONS

We study the performance of source based routing algorithms with partial information routing. Since we rely on partial information for making dynamic routing decisions, the capacity redundancy in the network is high. To improve capacity efficiency, we develop a dynamic routing algorithm that uses a simple segmented path protection scheme. We compare the performance of the partial information routing algorithm with and without segmented path protection through simulation studies. The important point to observe is that a simple $Seg2$ scheme works well to improve network performance, rather than a complicated segmentation scheme which has a high overhead on call setup time. The $Seg2$ scheme can be easily extended with minimal increase in state and fairly simple signaling.

Since segmented protection offers better performance as compared to path protection under partial information scenario, it would be a better alternative to employ segmented protection in large networks where obtaining complete network state information is impractical.

We also observe that the information collection remains the bottleneck for improving the overall performance of the routing algorithm. Per-flow information does not scale well for large networks. So converting per-flow information into an aggregated matrix which the partial information protocols can then work with is the key to improving performance. We also plan to conduct studies on different topologies to identify good segment sizes. Adjusting the segment sizes can also improve restoration time guarantees. For any m-segment protection scheme, identifying which m nodes to use for segmentation could also be a key factor.

REFERENCES

[1] P.F. Fonseca, "Pan-european multi-wavelength transport networks: Network design, architecture, survivability and sdh networking," *Proceedings of the 1st International Workshop on Reliable Communication Networks*, May 17-20, Paper P3 1998.

[2] B.T. Doshi, S. Dravida, P. Harshavardhana, O. Hauser, and Y. Wang, "Optical network design and restoration," *Bell Labs Technical Journal*, pp. 58–83, January-March 1999.

[3] S. Ramamurthy and B. Mukherjee, "Survivable wdm mesh networks, part i:protection," *IEEE INFOCOM*, vol. 2, pp. 744–751, March 1999.

[4] M. Sridharan, A.K. Somani, and M.V. Salapaka, "Approaches for capacity and revenue optimization in survivable wdm networks," *Journal of High Speed Networks*, vol. 10, no. 2, pp. 109 – 125, August 2001.

[5] H. Zang and B. Mukherjee, "Connection management for survivable wavelength-routed wdm mesh networks," *Optical Networks Magazine*, vol. 2, no. 4, pp. 17–28, July/August 2001.

[6] R. R. Iraschko and W. D. Grover, "A highly efficient path-restoration protocol for management of optical network transport integrity," *IEEE Journal of Selected Areas in Communications*, vol. 18, no. 5, pp. 779–794, May 2000.

[7] Y. Miyao and H. Saito, "Optimal design and evaluation of survivable wdm transport networks," *IEEE Journal of Selected Areas in Communications*, vol. 16, no. 7, pp. 1190–1198, September 1998.

[8] D. Banerjee and B. Mukherjee, "A practical approach for routing and wavelength assignment in large wavelength-routed optical networks," *IEEE Journal of Selected Areas in Communications*, vol. 14, no. 5, pp. 903–908, June 1996.

[9] M. Sridharan, M.V. Salapaka, and A.K. Somani, "A practical approach to operating survivable wdm networks," *IEEE Journal of Selected Areas in Communication*, vol. 20, no. 1, pp. 34–46, January 2002.

[10] M. Alanyali and E. Ayanoglu, "Provisioning algorithms for wdm optical networks," *IEEE INFOCOM*, vol. 2, pp. 910–918, March 1998.

[11] J. Jue and G. Xiao, "An adaptive routing algorithm for wavelength-routed optical networks with a distributed control scheme," in *Proceedings of the Nineth International Conference on Computer Communications and Networks*, October 2000, pp. 192–197.

[12] L. Li and A.K. Somani, "Dynamic wavelength routing using congestion and neighbourhood information," *IEEE/ACM Transactions on Networking*, vol. 7, no. 5, pp. 779–786, October 1999.

[13] E. D. Lowe and D. K. Hunter, "Performance of dynamic path optical networks," in *IEE-Proceedings of Optoelectronics*, August 1997, pp. 235–239.

[14] S. Ramamurthy and B. Mukherjee, "Fixed alternate routing and wavelength conversion in wavelength-routed optical networks," in *Proceedings of the Global Telecommunications Conference, GLOBECOM'98*, November 1998, pp. 2295–2303.

[15] R. Srinivasan, "Dynamic routing in wdm grooming networks," *Technical Report (DCNL-ON-2001-001), Dependable Computing & Networking Laboratory, Department of Electrical and Computer Engineering, Iowa State University*, August 2001.

[16] G. Mohan, C.S.R Murthy, and A.K. Somani, "Efficient algorithms for routing dependable connections in wdm optical networks," *IEEE/ACM Transactions on Networking*, vol. 9, no. 5, pp. 553–566, October 2001.

[17] R. Ramamurthy, Z. Bogdanowicz, S. Samieian, D. Saha, B. Rajagopalan, S. Sengupta, and S. Chaudhuri, "Capacity performance of dynamic provisioning in optical networks," *Journal of Lightwave Technology*, vol. 19, no. 1, pp. 40–48, January 2001.

[18] S. Thiagarajan and A.K. Somani, "Traffic grooming for survivable wdm mesh networks," *Opticomm: Optical Networking and Communications*, August 2001.

[19] M. Kodialam and T.V. Lakshman, "Dynamic routing of bandwidth guaranteed tunnels with restoration," in *Proceedings of IEEE INFOCOM'00*, April 2000, pp. 902–911.

[20] D. Papadimitriou et al, "Inference of shared risk link groups," Internet Draft draft-many-inference-srlg00.txt, work in progress, February 2001.

[21] P. Ho and H. T. Mouftah, "A framework for service-guaranteed shared protection in wdm mesh networks," *IEEE Communications Magazine*, vol. 40, no. 2, pp. 97–103, February 2002.

[22] M. Sridharan, R. Srinivasan, and A. K. Somani, "Dynamic routing with partial information in mesh-restorable optical networks," *Optical Networks Design and Modeling*, February 2002.