Lightpath Re-Optimization in Mesh Optical Networks

Eric Bouillet, Jean-Francois Labourdette, Ramu Ramamurthy, and Sid Chaudhuri

***Abstract*—Intelligent mesh optical networks deployed today offer unparalleled capacity, flexibility, availability, and, inevitably, new challenges to master all these qualities in the most efficient and practical manner. More specifically, demands are routed according to the state of the network available at the moment. As the network and the traffic evolve, the lightpaths of the existing demands becomes sub-optimal. In this paper we study two algorithms to re-optimize lightpaths in resilient mesh optical networks. One is a complete re-optimization algorithm that re-routes both primary and backup paths, and the second is a partial re-optimization algorithm that re-routes the backup paths only. We show that on average, these algorithms allow bandwidth savings of 3% to 5% of the total capacity in scenarios where the backup path only is re-routed, and substantially larger bandwidth savings when both the working and backup paths are re-routed. We also prove that trying all possible demand permutations with an online algorithm does not guarantee optimality, and in certain cases does not achieve it, while for the same scenario optimality is achieved through re-optimization. This observation motivates the needs for a re-optimization approach that does not just simply look at different sequences, and we propose and experiment with such an approach. Re-optimization has actually been performed in a nationwide live optical mesh network and the resulting savings are reported in this paper, validating reality and the usefulness of re-optimization in real networks.**

***Index Terms*—PLEASE SUPPLY YOUR OWN KEYWORDS OR SEND A BLANK E-MAIL TO KEYWORDS@IEEE.ORG TO RECEIVE A LIST OF SUGGESTED KEYWORDS.**

I. Introduction

INTELLIGENT mesh optical networks, supported by dense wavelength division multiplexed (DWDM) equipment and optical switches, are firmly established at the core constituent of next-generation optical networks. A key requirement of these optical mesh networks is the ability to route and restore quickly services via fast and capacity-efficient end-to-end restoration schemes [1]–[14].

During operations, requests for services are received and routed using an online routing algorithm that takes all of the information available at the time of the request to make the appropriate routing decision. With connection rates reaching tens of Gigabits per second (Gbps), the ability of the network management system to operate and maintain service continuation during failures has become a challenging requirement. In this work we consider end-to-end (or path) shared mesh restorations as supported by some of the optical switches available in the market.[1] In end-to-end dedicated mesh $(1 + 1)$ protection, the ingress and egress OXCs of the failed connection attempt to restore the signal on a predefined backup path that is disjoint, or diverse, from the primary path. Path diversity guarantees that primary and backup paths will not simultaneously succumb to the same failure. This approach requires large amounts of capacity, that is more than the working capacity since backup paths are longer than working paths. However, the backup path remains "live" in permanence, thus saving crucial path-setup latency when recovery takes place. In shared mesh restoration (Fig. 1), backup paths can share capacity if the corresponding primary paths are mutually diverse. Compared to dedicated mesh $(1 + 1)$ protection, this scheme allows considerable saving in terms of capacity required [1]. For example, since backup paths are usually longer than primary paths, the ratio of backup to working capacity is larger than one. On the other hand, with shared mesh restoration, while the backup can be even longer, but made up of shared channels, the same ratio is typically less than one. In addition, the backup resources can be utilized for lower priority pre-emptible traffic in normal network operating mode. However, recovery is slower than dedicated $(1 + 1)$ mesh protections, because it involves signaling and path-setup procedures to establish the backup path. In particular, we note that the restoration time will be proportional to the length of the backup path and the number of hops, and if recovery latency is an issue this length must be kept under acceptable limits [15]. This latter constraint may increase the cost of the solution, as it is sometimes more cost-effective to use longer paths with available shareable capacity than shorter paths where new shareable capacity must be reserved. Path-based shared mesh protection can itself be classified into several sub-categories.

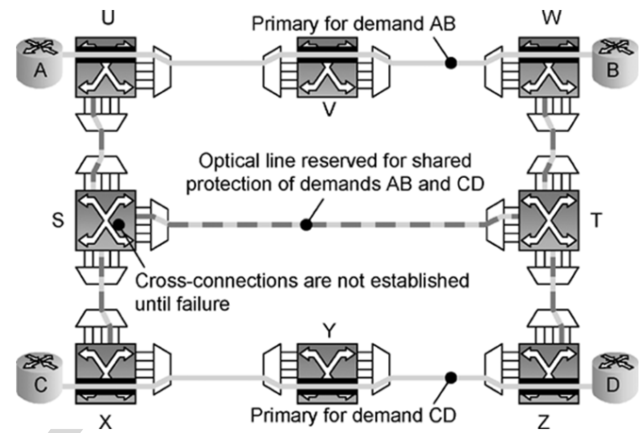

Fig. 1. Shared mesh restoration.

[1]Other categories include line protection and re-provisioning. These are not considered here.

Manuscript received September 27, 2002; revised June 5, 2003.

E. Bouillet was with Tellium Inc., Oceanport, NJ 07757 USA, and is now with IBM T.J. Watson Research Center, Hawthorne, NY 10532 USA (e-mail: ebouillet@ieee.org).

J.-F. Labourdette was with Tellium Inc., Oceanport, NJ 07757 USA, and is now with Verizon. **<PLEASE PROVIDE A COMPLETE POSTAL ADDRESS.>**

R. Ramamurthy and S. Chaudhari are with Tellium Inc., Oceanport, NJ 07757 USA.

Digital Object Identifier 10.1109/TNET.2005.845540

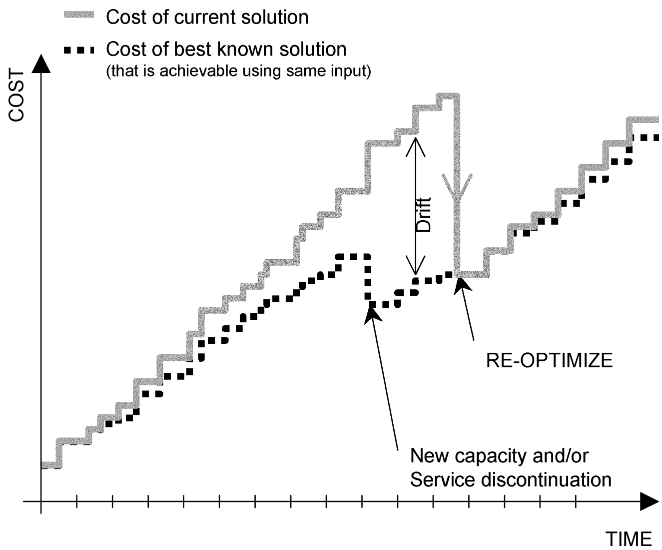

Fig. 2. Current cost versus best possible cost with cost-benefit of re-optimization.

A major distinctive feature is whether the protection channels are pre-assigned to each backup path before failure occurrence, or if the protection mechanism selects the channels from a pool of reserved channels to restore the services after failure occurrence. The advantage of pre-assigning the channels is that it delegates the channel selection function to the routing phase, when speed is less of an issue. Furthermore, since the channel assignment is predefined, the protection mechanism can be distributed without the need for a time-consuming handshake protocol to agree on which channel to select. Other authors have studied the case where protection channels are pooled and not pre-assigned [16]. While further savings can be achieved by doing so over the case where channels are pre-assigned, other methods can be used to continuously improve efficiency of protection capacity in the case of pre-assigned protection channels [11]. In the reminder of this paper the approach based on pre-assigning the channels is assumed.

The primary and protection paths of each new demand are computed according to the current state of the network, which includes the routing of the existing demands. As the network and traffic evolve, the routing of the existing demands becomes sub-optimal. Demand churn and network changes such as the addition/deletion of new links and/or capacity, causes the routing to become sub-optimal, thereby creating opportunities for improvements in network bandwidth efficiency. Increasing customer churn and the continued demand for bandwidth services further exacerbates this problem.

Re-optimization seizes on these opportunities and offers the network operator the ability to better adapt to the dynamics of the network. This is achieved by regularly (or upon a particular event) re-routing the existing demands, temporarily eliminating the drift between the current solution and the best known solution that is achievable under the same conditions, as illustrated in Fig. 2.

Carriers have been using reconfiguration over time to better manage their network assets and increase utilization on those assets, thereby deferring capital spending on new infrastructure. Reconfiguration has also been used to provide better service performance, for example, by rerouting services over shortest paths if such paths become available. Earlier work on reconfiguration was done in the context of Digital Cross-Connect (DCS)-based networks [4], [17]. Later, with the deployment of ATM as a networking technology, reconfiguration of ATM networks was explored, leveraging ATM Virtual Paths [5], [18]–[21]. More recently, reconfiguration has been studied in the general context of optical networks [22]–[29]. Our work explores the reconfiguration, or re-optimization, of optical mesh networks, specifically those where lightpaths are protected by shared mesh restoration as described earlier. In the optical mesh network of interest here, carriers can either reroute only the shared mesh back-up paths of existing lightpaths, so that service is not impacted (it is still carried on the primary or working path during the re-optimization of the back-up paths), or they can reroute both the primary and shared mesh back-up paths (either by impacting customer service, or by first moving service to the back-up path and re-optimizing the primary, and then moving service back to the primary and re-optimizing the back-up). Re-optimizing both primary and back-up paths should improve network bandwidth utilization more than just re-optimizing back-up paths, but at the cost of customer service impact, or additional operational complexities and risks.

In this paper we study two re-optimization algorithms. A complete re-optimization algorithm that re-routes both primary and backup paths, and a partial re-optimization algorithm that re-routes the backup paths only. Re-routing backup paths only is a sub-optimal but attractive alternative that avoids any service interruption since the primary path is not affected (changed). In this paper we show that on average, with periodic re-optimization, these algorithms allow bandwidth savings of 3% to 5% of the total capacity in scenarios where the backup path only is re-routed. Substantially larger bandwidth savings can be achieved when both the working and backup paths are re-routed. In addition, significant bandwidth savings can be achieved by re-optimizing the network after topological changes such as new nodes and/or new link additions. These bandwidth savings are achieved through increased sharing of backup path capacity among several working paths, and substantial reductions in average path length, which also translates into shorter restoration times.

The paper is organized as follows. In Section II we discuss the online algorithm cost model and the main function used to compute the shared mesh restored paths that achieve the desired compromise between cost and restoration latency. In Section III, we describe the re-optimization algorithm. This algorithm uses the routing function discussed in Section II. Section IV is a collection of proofs where we demonstrate the existence of cases for which neither an algorithm that tries different sequences to route the demands nor the proposed re-optimization algorithm can achieve the optimum solution. We also reveal the existence of cases for which re-optimization achieves the optimum, whereas trying all possible sequence to routes the demands does not. The effectiveness of the re-optimization algorithm is measured for real customer networks and the results presented in Section V. We conclude this paper in Section VI.

II. Routing Algorithm

A. Cost Model

In the following, we use the term Shared Risk Optical Group (SROG) to indicate a group of optical resources that share a common risk of failure. We define our cost model as follows.

- **The network link cost** is for each link the cost of using a channel in the link. It take into account the costs of the optical interfaces and the transponders, as well as the prorated cost of the common equipments such as optical amplifiers (OA), fibers, and WDM systems, which is equally distributed over the multiple channels they support. Such cost has a component that is usually proportional to the length of the link, since longer links use OA and possibly regenerators. The network link costs are often administratively defined by the network operator to reflect internal cost structures.
- **The network cost** is the sum of the respective network link costs for all the channels that are required to accommodate a prescribed demand set. The objective of the algorithm is to find a solution that minimizes this total cost. If the network link costs are uniform, then from an algorithm standpoint the network cost is proportional to the number of channels required to accommodate the demand.
- **The routing link cost** is a cost metric determined and used internally by the routing algorithm. It coalesces the network link cost with the concepts of path diversity and sharing of protection capacity. The metric or policy used for assigning routing link costs to the links is different for primary paths and backup paths. For primary paths it is the network link costs C_e of using the links. For backup path it is a function of the primary path. A backup link e is assigned: 1) infinite routing link cost if it intersects with an SROG of the primary path; 2) routing link cost w_e if new capacity is required to route the path; and 3) routing link cost $s_e \leq w_e$ if the path can share existing capacity reserved for pre-established backup paths on the link.

Quite evidently, the underlying idea expressed in the definition of the routing link cost is to encourage "sharing", whereby existing capacity can be reused for routing multiple backup paths. The condition for sharing is that the backup paths must not be activated simultaneously, or in other words that their respective primaries must be pair-wise SROG-disjoint so that they do not fail simultaneously. The ratio s_e to we can be adjusted for the desired level of sharing. For smaller values of s_e, backup paths will be selected with the minimization of the number of nonshareable links (routing link cost w_e) in view, eventually leading to arbitrary long paths (as expressed in number of hops) that consist uniquely of shareable links (routing link cost s_e.) For larger values of s_e routing is performed regardless of sharing opportunities and backup paths will end-up requiring substantially more capacity.

B. Illustrative Online Routing Algorithm

Assume that routing of a lightpath is performed in two steps: 1) computation of a primary and backup pair of routes, and 2) assignment of channels along the routes. Ideally, the two steps are solved simultaneously and step 1 is optimized so that channel-assignment in step 2 reuses existing capacity for backup paths. For the only purpose of illustrating the cost model described above, we present a K-shortest-path-based algorithm (Fig. 3) **<CITATION OF FIG. 3 ADDED; PLEASE CONFIRM.>**, keeping in mind that any other algorithm whose objective minimizes this cost model can also be used. The algorithm takes as input: 1) A network object N that encapsulates the state information of the switches, optical channels (busy and available), the network link cost c_e for every link e, and the existing demands with their routes; 2) the end nodes A and Z of the demand; and 3) a candidate primary path p_0 if partial re-optimization is desired. It operates as follows.

Compute_Pair_of_Paths(Network N, node A, node Z, candidate primary path p_0):

1) If p_0 is non-null, set $P = \{p_0\}$ and go to 4, otherwise compute a list of candidate primary paths:
2) For every link e in N set routing link cost to network link cost c_e of link.
3) Using metric defined in 2, compute set P of k minimum routing cost paths connecting node-pair $A - Z$, or all feasible paths if there are less than k of them.
4) Set min_routing_cost = infinity, and $\{p^*, q^*\} = INFEASIBLE$.
5) For each path p in P, do
   a) Assign routing link cost to every link e
      i) If e intersects SROG of primary path p, set routing link cost to infinity.
      ii) If e has at least one channel that is shareable with p, set routing link cost to $s_e = \epsilon c_e$.
      iii) Otherwise, set routing link cost to $w_e = c_e$.
   b) Using metric defined in 5a, compute minimum routing cost path q connecting node pair $A - Z$.
   c) If q does not exist, continue step 5. with next path p in P.
   d) If min_routing_cost < combined routing costs of p and q, then $\{p^*, q^*\} = \{p, q\}$ and min_routing_cost = combined routing costs of p and q.
6) Return $\{p^*, q^*\}$

Fig. 3. Online routing algorithm.

If the minimum network cost is sought (maximum sharing), the value of ϵ in step 5(a)ii, determining the routing link cost of "shareable" protection channels, is set to 0. Otherwise if shorter backup lengths and faster restoration are desired, ϵ is set to a small positive value. Extensive study has already been performed for $\epsilon = 0$ in [1]. In [30] we studied the effect of varying ϵ between 0 and 1. When ϵ tends toward 1, we expect the lengths of primary and backup paths, as expressed in number of hops, to resemble that of dedicated $(1 + 1)$ mesh protection, though sharing is still implemented when available on the backup path and the capacity required remains lower than for dedicated $(1 + 1)$ mesh protection. Earlier experiments [30]–[32] indicate that a value of ϵ in the range [0.2–0.4] returns the best tradeoffs between network cost and restoration latency (i.e., average and maximum restoration path length.) In the remainder of this paper we use $\epsilon = 0.3$.

III. Re-Optimization Algorithm

The re-optimization algorithm takes as input: 1) A network object N that encapsulates the state information of the switches, optical channels (busy and available), and existing demands

Reopimize_Demands(Network N, list of demands D with respective re-optimization types)

1) Set REPEAT = 0
2) For each demand d in D
   a) Let A, and Z denote the end-points of the demand
   b) Set p_0 = current primary path, and q_0 = current backup path of demand d.
   c) In network N, free paths p_0, and q_0.
   d) If partial re-optimization is desired, then $\{p^*, q^*\}$ = *Compute_Pair_of_Paths*(N, A, Z, p_0) (note that $p^* = p_0$), else $\{p^*, q^*\}$ = *Compute_Pair_of_Paths*(N, A, Z, null)
   e) If combined routing costs of p^* and q^* is less than combined routing costs of p_0 and q_0, then in network N, route demand d on paths p^* and q^*, and set $REPEAT = 1$. Otherwise, route demand d back to paths p_0 and q_0.
3) If $REPEAT > 0$, repeat from step 1.

Fig. 4. Re-optimization algorithm.

with their routes; and 2) a list D of demands to be re-optimized with their respective re-optimization types (complete or partial). It operates as follows.

The key idea behind the re-optimization algorithm using successive routing is not new [33]. An approach similar to ours has recently been proposed in the case where protection channels are pooled [34]. Nevertheless, it is the first time to our knowledge that it has been applied to re-optimize shared mesh restored lightpaths in a real life network. We prove in the next section that there are instances for which this algorithm achieves the optimum routing configuration, while a sequential algorithms that tries to find a best ordering for routing the demands (such as in [35]) would fail to find the optimum. The re-optimization algorithm is generic enough so that it is also applicable to re-optimize mixed protection types, i.e., combination of unprotected, dedicated mesh and shared mesh protected demands of various rates. It is also fast and easy to enhance with additional rules that improve the quality of the re-optimization. It can for instance be improved to selectively re-optimize a specified set of demands. Finally, this algorithm provides the means to carry out the re-optimized solution by executing step 2 in the real network. The risks involved in step 2e are limited, since only one demand is re-routed at a time, and the operation does not impact the service if partial re-optimization is used.

IV. Complexity of Optimizing the Routing of Shared Lightpaths

Note that the optimum routing of shared mesh restored demands is a very difficult problem (NP-hard) [1]. In this section we discuss cases where the online routing algorithm or re-optimization algorithms as defined in Section III fail to find the optimum solution. We provide here the theorems and proofs of such cases, as well as problem instances that can be used as comparison points to compare different optimization algorithms.

In the presentation of the proofs we use the following notation.

- P is an instance of shared mesh restored routing problem, it consists of a prescribed capacited network, point-to-point demands, and protection type for each demand.
- $Sol(P)$ represents the set of all possible solutions of P, which includes the routing of each demand, and the channels used (and shared) by each demand.
- $Opt(P) \subseteq Sol(P)$ is the subset of optimum solutions of P, solutions that utilize the minimum number of channels over all possible solutions, i.e., $cost\ Opt(P) = min_P\ cost\ Sol(P)$.

Assume that we use an online routing algorithm to solve this problem. An online routing algorithm is any algorithm that satisfies all three conditions: 1) demands are routed in sequence; 2) routed demands are immutable; and 3) the primary-backup pair of every new demand is selected so that the resulting total number of channels is minimized.

- S is an ordered sequence of all demands in P, and $Sol(P,S) \subseteq Sol(P)$ is the corresponding solution if an online routing algorithm is used in conjunction with this sequence.
- In addition, for an ordered sequence S, let $Reopt(P,S) \subseteq Sol(P)$ designates the re-optimized solution $Sol(P,S)$.

In this section, a re-optimized solution is the result of applying the re-optimization algorithm described in Fig. 4 on an existing solution, which is itself obtained using an online algorithm, possibly after trying all possible sequences.

Lemma 1: By definition of optimality the cost of an optimum solution is the minimum over the costs of all possible solutions, including solutions found using an online routing algorithm with demands routed in any sequence

$$\text{cost}\ Sol(P,S) \geq \text{cost}\ Opt(P) \quad \forall\, S.$$

A. No Prior Placement of Protection Channels or Primary Paths

In this subsection we prove three basic theorems on optimality of an-line routing and re-optimization in the case where no prior placement of protection channels or primary paths exists.

Theorem 1: There are network instances for which no sequence exists for online routing that can achieve the optimum routing configuration.

$$\exists P \text{ so that cost } Sol(P,S) > \text{cost } Opt(P) \quad \forall\, S.$$

Proof: With the help of Fig. 5 we demonstrate the existence of at least one instance P for which the theorem is true. Part (i) of the figure illustrates P_0, a 12 nodes network, with two demands (a,b) and (c,d). We solve P_0 using an online routing algorithm and all possible sequences $S_1 = \{(a,b);(c,d)\}$ and $S_2 = \{(c,d);(a,b)\}$. Parts (ii) and (iii) of the figure depict two possible solutions. The dotted lines in the solution represent sharable protection channels. The solution in part (ii) could result from either sequence S_1 or S_2. The other example shown in part (iii) results from sequence S_1 only. There are other solutions not shown in this figure; we show, however, from the definition of the online routing algorithm that in this particular example, the cost is the same for all solutions of each given sequence S_i. The symmetry of the network guarantees the independence on the order of the sequence, and the first demand is always routed along the single hop primary and its

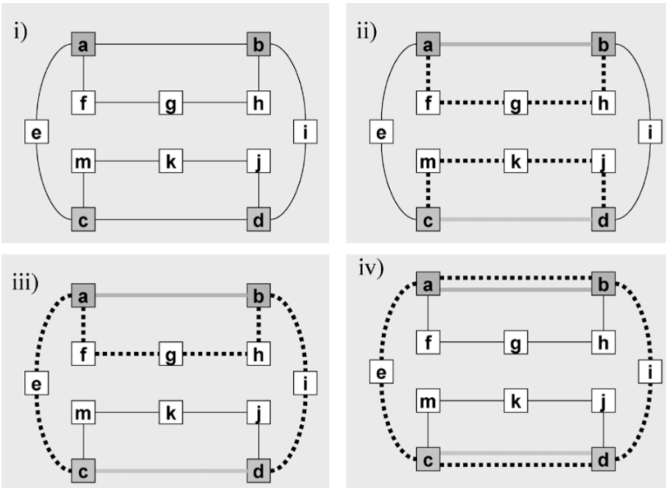

Fig. 5. (i) Network with demands (a,b) and (c,d). (ii) and (iii) Two sub-optimum solutions using best sequence of online routing. (iv) An optimum solution.

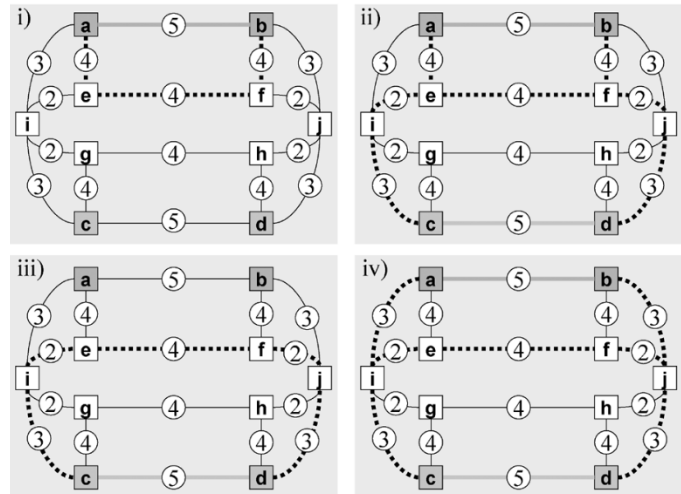

Fig. 6. (i) Network with demands (a,b) routed and (c,d) to be routed. (ii) Min-cost routing of demand (c,d) assuming configuration (i). (iii) Configuration with demand (a,b) removed, and (iv) with demand (a,b) re-routed.

corresponding 4-hops backup. Because the objective of the algorithm is to minimize the number of channels required for each new demand, and there is only one possible configuration for the first demand, the cost of routing the second demand is the same for all possible solutions resulting from this algorithm. Now, we find that the minimum of cost $Sol(P_0, S_1)$ and cost $Sol(P_0, S_2)$ requires two primary channels, and eight channels are reserved for protection, that is a total of 10 channels. In comparison, the optimum solution $Opt(P_0)$ shown in part (iv) of the figure requires two primary channels and six channels are reserved for protection—therefore, for this example, cost $Sol(P_0, S) >$ cost $Opt(P)\ \forall S$. ■

Using the same example as given in Fig. 5 we can demonstrate a similar result for the re-optimization algorithm, in which the following theorem applies.

Theorem 2: There are network instances for which no re-optimization exists that can achieve the optimum routing configuration

$$\exists P \text{ so that cost } Reopt(P,S) > \text{ cost } Opt(P) \quad \forall S.$$

Proof: Removing any demand (a,b) or (d,c) from case (ii) or (iii) in Fig. 5 and attempting to re-route it would only achieve the same result. ■

In this particular case the re-optimization did not bring further improvement with respect to the online algorithm. This result is not to be generalized though. And in fact we prove in the next theorem that re-optimization can achieve the optimum result, while there exists no sequence for which an online routing algorithm can.

Theorem 3: There are network instances for which a re-optimization exists that can achieve the optimum routing configuration, while no sequence exists for which online routing can achieve the optimum routing configuration

$$\exists P \text{ so that cost } Sol(P,S) > \text{cost } Reopt(P,S) = \text{cost } Opt(P) \quad \forall S.$$

Proof: With the help of Fig. 6 we demonstrate the existence of at least one instance P for which the theorem is true. Part (i) of the figure illustrates P_1, a network with two demands (a,b) and (c,d). Unlike the previous example where the cost was identical for every link, the links of this example may traverse a varying number of adjacent channels and intermediate degree-2 nodes. The circled values on the links represent the number of such channels (hops) and are used to indicate the effective cost of using the corresponding links. Part (i) of the figure shows the min-cost routing of demand (a,b). Part (ii) shows the subsequent min-cost routing of demands (c,d) following the routing given in part (i). Had the two demands been routed in the reverse order, a homologous solution would have been obtained with the routes of demands (a,b) and (c,d) reversed. Note that these are the solutions returned by the online routing algorithm, using any possible sequence (a,b), (c,d) or (c,d), (a,b). These solutions require 10 working channels, and 22 protection channels. Applying the re-optimization algorithm to the solution of part (ii), we remove demand (a,b), as shown in part (iii), and re-route it using the min-cost algorithm, resulting in the solution shown in part (iv). The latest solution requires 10 working channels and 20 protection channels, an improvement of two channels compared to the best possible solution obtained by way of the online routing algorithm. By inspection we can show that this is the minimum cost achievable for this network. ■

B. Prior Placement of Protection Channels or Primary Paths

In this subsection, we prove three theorems on optimality of re-optimization and online routing with prior placement of protection channels or primary paths:

As an extension to Theorem 1, we can show that the knowledge of the optimal placement of the protection channels is insufficient to determine the optimum solution using either the online routing or the re-optimization algorithm. This is the topic

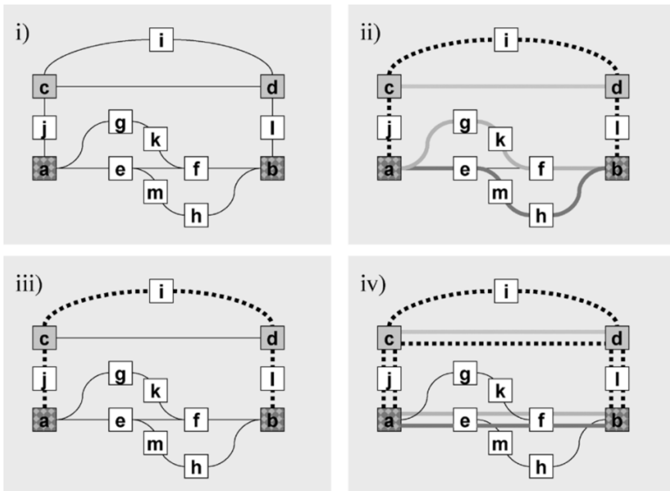

Fig. 7. Shared mesh restoration architecture. (i) Network with two demands (a, b) and one demand (c, d). (ii) An optimum solution, (iii) optimum set of channels reserved for protection, and (iv) sub-optimum solution obtained from (ii) using a greedy online routing algorithm.

of the next two theorems. We begin these theorems with an extension of our terminology. For any instance P, let P^+ denotes the instance with the protection channels optimally placed.

Theorem 4: Even if the shared channels, part of the optimal routing configuration are given, there are instances for which no online routing sequences exists that achieve the optimal solution

$$\exists\ P \text{ so that cost } Sol(P^+, S) > \text{cost } Opt(P) \quad \forall S.$$

Proof: Again we demonstrate this theorem by way of the example. Using the network given in Fig. 7, part (i) of the figure illustrates a 13-node network used to transport 3 demands: one demand between (c, d) and two demands between (a, b). We introduce demand (c, d) in order to limit the number of optimum solutions to one possible configuration, shown in part (ii) of the figure. Assuming next that the protection channels are allotted according to this optimum configuration as shown in (iii), we determine by inspection that regardless of the sequence order, there are only three possible solutions achievable by the online routing algorithm. All three solutions require the same total number of channels. The solution that requires the least working capacity is shown in part (iv) of the figure. Comparing the optimum solution depicted in part (ii) with the solution (iv), we observe that the latter requires three more channels, which proves the theorem. ∎

Using the same example as given in Fig. 7 we can demonstrate a similar result for the re-optimization algorithm, and the following theorem applies:

Theorem 5: Even if the shared channels, part of the optimal routing configuration are given, there are instances for which no re-optimization exists that achieves the optimal solution

$$\exists P \text{ so that cost } Reopt(P^+, S) > \text{cost } Opt(P) \quad \forall S.$$

Proof: Removing any demand (a, b) or (c, d) from case (iv) in Fig. 7 and attempting to re-route it would only achieve the same result. ∎

To conclude this section, we show that even if the primary paths are given, there are problem instances for which there are no sequence or re-optimization that achieves the optimum solution. For the purpose of the theorem, for any instance P, let P^* denotes the instance with the optimal primary paths.

Theorem 6: Even if the primary paths, part of the optimal routing configuration are given, there are instances for which no online routing sequence exists that achieves the optimal solution

$$\exists P \text{ so that cost } Sol(P^*, S) > \text{cost } Opt(P) \quad \forall S.$$

Proof: The proof is derived directly from the example of Fig. 5. We observe that in this example the sequential algorithm, or re-optimization algorithm, always used the optimum primary paths, but failed to find the optimum backup paths. Thus, prescribing the primary paths in this example would results in the same sub-optimal solution. ∎

The proof is also valid for the re-optimization algorithm, and hence our final theorem:

Theorem 7: Even if the primary paths, part of the optimal routing configuration are given, there are instances for which no re-optimization exists that achieves the optimal solution

$$\exists P \text{ so that cost } Reopt(P^*, S) > \text{cost } Opt(P) \quad \forall S.$$

Proof: Using case (ii) or (iii) of Fig. 5, we observe that removing either demand (a, b) or (c, d) and trying to reroute it would only achieve the same result. ∎

In conclusion, while it is sometimes impossible to achieve optimum routing with either online routing or re-optimization, there are cases where it can be achieved through re-optimization, while sequential routing only cannot achieve it (Theorem 3). The reverse is not true, since the first iteration of re-optimization is by definition a sequential algorithm, and any subsequent iteration is an improvement of the first iteration. We will show in the next section that the re-optimization algorithm achieves bandwidth savings in many circumstances.

V. Experiments

A. Calibration

In the following experiments, all the randomly generated graphs consist of rings traversed by cords connecting randomly selected pairs of nodes. Very often, but not always, it is possible to embed such a ring on a real network (the embedding requires finding a Hamiltonian circuit in the network), as demonstrated in Fig. 8 with the ARPANET network. Each link of the random networks is assigned an arbitrary network link cost of using a channel in it. The costs are integers uniformly distributed between 5 and 10. We then compare the algorithms according to their ability to minimize the total network cost, that is the sum of the respective link costs for all the channels used in the network.

We first apply the complete re-optimization algorithms to small random generated networks, varying in size (i.e., number of nodes), with demands preliminary routed using the online routing algorithm, and compare the solutions with results obtained by way of an ILP solver (see the formulation in the Appendix.) The ILP solver is CPLEX 7.1 from ILOG. CPLEX exploits a branch-and-cut algorithm, in which it solves linear sub-problems after setting a subset of formulation variables to

TABLE I
COMPARISON OF COMPLETE RE-OPTIMIZATION WITH ILP-BASED SOLUTION

			Online routing algorithm			*Re-optimized (complete)*			*ILP Solver (CPLEX)*		
number of			*Primary channels*	*Backup channels*	*Total cost*	*Primary channels*	*Backup channels*	*Total cost*	*Primary channels*	*Backup channels*	*Total cost*
Node	*Link*	*Dem.*									
10	13	90	404	392	5624	396	376	5428	396	376	5428
12	15	99	636	568	9568	552	570	8918	552	566	8894
15	19	95	500	358	6786	514	326	6626	(1)	(1)	6476

(1) ILP failed to find the solution after several hours. Total cost is a lower bound.

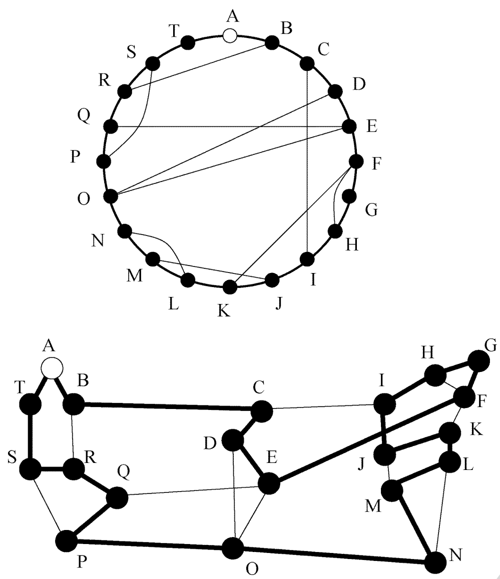

Fig. 8. Chordal ring (top) embedded on Arpanet (bottom).

integer values. In this process the ILP produces a progress reports according to the solutions it finds: lower bounds of the network cost if some of the solution's variables have fractional values, or upper bounds if the solution is feasible and all its variables have integer values. When the upper bounds and lower bounds finally converge, the solution is known to be feasible and optimal. However, the convergence time of this process can be exponential to the size of the problem. We therefore put a limit of 10 hours maximum on each problem, and interrupted the process if no feasible solution could be found within that time frame, in which case not the optimum value, but a lower bound of it was used in the results of our experiments.

Our observations, summarized in Table I, indicate that for these networks, re-optimization allows capacity savings of 2% to 6%, and is within 1% of the network cost of the optimal solution, or 2% of a known lower bound if the optimal solution cannot be determined.

B. Real Networks

Next, we apply the algorithm to re-optimize the routes of four different networks, Net-A, Net-B, Net-C, and Net-D. Net-A is a network built by Dynegy [36]. It consists of 48 nodes, 75 links, and a number $Q \approx 100$ (order of hundred) of shared mesh restored demands with their routes provided by the operator. Net-A consists of three periods (Net-A.1, Net-A.2, and Net-A.3), measured over a 14–month interval and capturing the actual growth of the network. This scenario has a limited number of spare channels, and offers very little room for rearranging the paths. Net-B consists of 25 nodes, 30 links, and 290 demands.

TABLE II
PARTIAL RE-OPTIMIZATION—STATIC INFRASTRUCTURE. ONLY BACKUP PORTS COUNT ARE PROVIDED, SINCE PRIMARY PORTS REMAIN THE SAME WITH PARTIAL RE-OPTIMIZATION

Scenario name	*Backup port count*				*Average backup hops*		*Max backup hops*	
	before	*after*	*%save*	*%save of total ports*	*before*	*after*	*before*	*after*
Net-B	2520	2452	3%	1%	5.83	5.76	11	10
Net-C	2340	2242	4%	2.1%	4.61	4.53	15	13
Net-D	504	470	7%	3%	4.37	4.10	11	9

Net-C is a 45-node network with 75 links and 570 demands. Net-D is a 60-node network with 90 links and 195 demands. The demands of all four networks have OC48 rates and above. The dimensions and the technology used in the networks are such that we may assume uniform network link costs. Under this assumption, the network costs are proportional to the number of channels or number of ports necessary to accommodate their respective demands.

The demands of Net-B, Net-C, and Net-D are provided unrouted with sources and destinations information only. Henceforth, we created an initial routing configuration for these three scenarios by routing their demands sequentially following an arbitrary order, using the *Compute_Pair_of_Paths* online routing procedure described in Section II. We added new channels as needed during that process assuming that the network had infinite capacities. The demands of each scenario are then re-optimized, once partially and once completely, using the *Reoptimize_Demands* procedure of Section III.

Using networks Net-B, Net-C, and Net-D, we perform a series of experimentation assuming two different scenarios. In the first scenario, referred to as the Static Infrastructure, we assume that no capacity is added, or freed by demand churn, from the moment the demands are routed the first time, and the moment the re-optimization is executed. In the second scenario, referred to as the *Growing Infrastructure*, we assume that as the network grows and demands are routed, new links or new channels on existing links are added to the network, creating opportunities for improvement of existing demands.

The case of network Net-A was treated as a growing infrastructure only, since this is the context in which this network has been built.

C. Static Infrastructure

Tables II and III summarize the results for the partial and the complete re-optimization respectively when the infrastructure of the network remains static throughout the experimentation. The tables show the quantities measured before and after re-optimization. For each scenario, the same network and routed demands are used for partial and for complete re-optimizations. The number of ports in Table II consists of ports used for the

TABLE III
COMPLETE RE-OPTIMIZATION—STATIC INFRASTRUCTURE

Scenario name	*Total (primary and backup) port count*			*Backup port count*			*Avg backup hops*		*Max backup hops*	
	before	*after*	*%save*	*before*	*after*	*%save*	*before*	*after*	*before*	*after*
Net-B	5088	4994	2%	2520	2502	0.71%	5.83	5.72	11	10
Net-C	4640	4450	4%	2340	2174	5.48%	4.61	5.20	15	16
Net-D	1112	1036	7%	504	470	6.75%	4.37	4.25	11	13

TABLE IV
CHARACTERISTICS OF NETWORKS USED IN EXPERIMENTS

Network	*Number of nodes*	*Number of Links*	*Number of Demands*
Net-A.1	48	75	$Q \approx 100$
Net-A.2	48	77	$1.06 \times Q$
Net-A.3	48	79	$1.42 \times Q$
Net-B	25	30	290
Net-C	45	75	570
Net-D	60	90	195

TABLE V
PARTIAL RE-OPTIMIZATION—GROWING INFRASTRUCTURE

Scenario name	*Backup port count*				*Average backup hops*		*Max backup hops*	
	before	*after*	*%save*	*%save of total ports*	*before*	*after*	*before*	*after*
Net-A.1	224	208	7%	4%	7.1	5.24	20	11
Net-A.2	310	214	31%	19%	5.9	4.88	15	10
Net-A.3	332	242	27%	15%	4.8	3.57	13	9
Net-B	2986	2868	4%	2%	6.33	6.02	10	10
Net-C	2576	2294	11%	5.5%	5.07	4.73	15	15
Net-D	550	520	5%	2%	5.10	4.13	16	13

TABLE VI
COMPLETE RE-OPTIMIZATION—GROWING INFRASTRUCTURE

Scenario name	*Total (primary and backup) port count*			*Average backup hops*		*Max backup hops*	
	before	*after*	*%save*	*before*	*after*	*before*	*after*
Net-A.1	400	382	5%	7.1	5.43	20	10
Net-A.2	494	372	25%	5.9	4.82	15	10
Net-A.3	612	486	21%	4.8	4.05	13	9
Net-B	5306	4970	6%	6.33	6.02	10	11
Net-C	5134	4564	11%	5.07	4.80	15	14
Net-D	1326	1030	22%	5.10	4.61	16	13

protection channels only, since the working channels remain the same. The number of ports in Table III consists of all the ports in the network, used for primary and protection channels. We observe that the partial re-optimization saves up to 3% of the total number of ports, and complete re-optimization up to 7%. The complete re-optimization offers the most cost efficient alternative, but most of the improvement is realizable using the partial re-optimization algorithm, without service interruption.

We also observe that the protection path latency tend to be slightly longer in complete re-optimization than in partial re-optimization. Although counterintuitive at first, this is actually an expected outcome in the case of shared mesh protected networks, because the complete re-optimization algorithm explores a wider solution space in which backup paths are slightly longer on the average than in partial re-optimization. This effect can be mitigated if necessary by increasing the value of ϵ, as indicated earlier in the description of the cost model.

D. Growing Infrastructure

The next set of experiments cover the case of the growing infrastructure. Here demands are routed online over time, while new links and capacity on existing links are added simultaneously, creating a more realistic dynamic than the previous exercise. For the case of Net-B, Net-C, and Net-D, we first route the demand after removing a link selected empirically by inspection. In particular, we favor a link that exhibits an apparent impact on the network connectivity, but is not essential to protect all the demands. For instance in the example of Fig. 8, a good candidate link for removal would be $\{E, F\}$ or $\{O, N\}$. The link is then reinserted to simulate a capacity upgrade, and the demand routed in the first step is re-optimized in the upgraded network. Net-A being a real network, the first two steps were performed in a more imbricated way during the construction of the network. For instance, Table IV indicates that the size of Net-A increased from 75 to 79 links over the study period. Other affecting factors are the policy used to add new channels with respect to demand growth, and the pattern of channel unavailability caused by maintenance or failure conditions. Table V and Table VI summarize the results for the partial and the complete re-optimization respectively when the infrastructure of the network evolves in time. As before, the tables show the quantities measured before and after re-optimization. Note that the savings for Net-A are substantial. Unlike the static scenarios where channel availability is not an issue and assumed to be unlimited during the initial routing, the demands of this network have been routed while new channels were being added later, or while existing channels or links may have been unavailable for maintenance reasons, thus creating opportunities for optimization. The latter is the most realistic mode of operation, and the most likely to occur. Worth noticing for this scenario, is the reduction in protection path latency measured as the average number of channels traversed by the protection paths, which decreases from 7.1 to 5.24 hops for the partial re-optimization of Net-A.

Note that depending on the scenario, the difference in terms of performance between partial re-optimization and complete re-optimization is more or less pronounced. This can be due to a combination of factors, such as the demand set, or the network topology. Most importantly, it is how network capacity and demand growth are achieved. In the case of Net-A the two occur simultaneously and this case should be considered separately. In the case of Net-B, Net-C, and Net-D, it depends on the link that is added to simulate growth. We can illustrate this with a network constituted of two regions $R1$ and $R2$ only connected by two fibers $F1$ and $F2$. Any pair of demands between $R1$ and $R2$ cannot share backup capacity in $F1$ or $F2$, because in this part of the network either their working paths are not disjoint, or their backups are disjoint. If we add a third fiber $F3$ between $R1$ and $R2$, and re-optimize the demands between the two regions, the new capacity can be used to enable sharing across the backup paths. However, in order to get the most benefits of it, it may be necessary to re-optimize the working paths as well. For instance, if the working paths are all routed on $F1$ by the initial

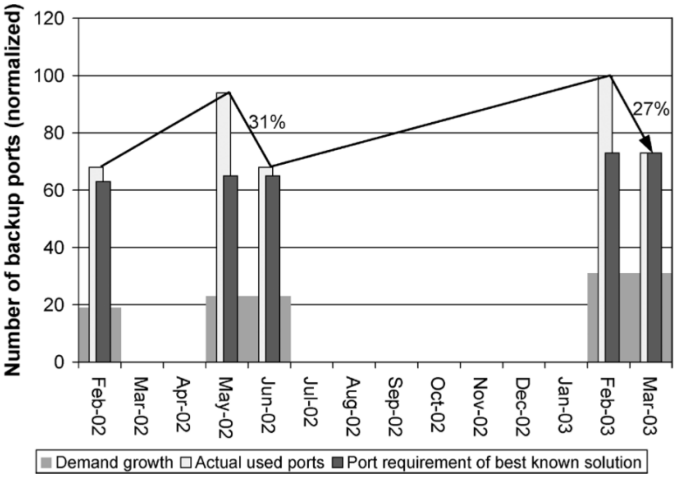

Fig. 9. Network growth over a period of 14 months, and effect of partial re-optimization on number of ports reserved for backup (normalized).

online-routing, then the addition of fiber $F3$ followed by a partial re-optimization does not enable capacity sharing between the backups, because all the working paths have a common point of failure on fiber $F1$. On the other hand, a complete re-optimization that distributes the working paths on the three fibers would allow sharing.

E. Network Dynamic

We had the opportunity to observe the effect of re-optimization on Net-A at different intervals over a 14 months period. Our measurements shown in Fig. 9 demonstrate that the network imitates the behavior illustrated in Fig. 2. To conclude the study, we performed over this period two actual backup re-optimizations of the network, saving each time the operator 27% to 31% of the number of backup ports by re-routing a subset of the backups. The corresponding savings in total number of primary and backup ports were respectively 15% to 19%. The freed capacity could be reused to carry new demands.

VI. Conclusion

In this paper we have presented a re-optimization algorithm to re-arrange shared mesh protected lightpaths. The proposed algorithm allows for two types of re-optimization. A complete re-optimization algorithm that re-routes both primary and backup paths, and a partial re-optimization algorithm that re-routes the backup paths only. Re-routing backup paths only is a sub-optimal but attractive alternative that avoids any service interruption. Our experiments indicate that the complete re-optimization achieves a 3% to 5% savings in the cost of the network, and most of the improvement can be achieved by way of the partial re-optimization alone. Re-optimization would only occur at certain intervals (every few weeks or couple of months) or upon certain events such as when new links are added, or when the potential savings from re-optimization exceeds some threshold. We also prove that although none of the proposed algorithms is guaranteed to find the minimum cost solution, there exist cases where the re-optimization algorithm can achieve the optimum result, while there are no demand sequences for which a sequential routing algorithm can.

Appendix

In this Appendix we give the ILP formulation for the routing problem of shared mesh protected demands in optical mesh networks. We distinguish between two schemes to allot channels to protection paths as discussed earlier in the introduction. There are "pre-assigned channels" and "pooling." We precise the differences between the two schemes in the formulation where applicable.

For each demand, the ILP formulation selects the primary from a prescribed set of K possible paths, whereas backups are determined according to the node-arc flow conservation equation. Although this implementation may leads to sub-optimal solutions if the set of possible primary paths is not appropriately constructed, it nevertheless allows us to pin down the primary paths for partial re-optimizations, and has the advantage of being less complex and faster to solve.

A. ILP Description

Inputs:

- N: set of nodes
- E: set of undirected edges
- D: set of demands
- R^d: set of K candidate paths for demand d, d member of D.
- R_i^d: ith primary path for demand d, $1 \leq i \leq K$
- C_e: Network link cost of link e, e member of E
- F: set of all failure scenarios (set of all links)

Variables

- P_k^d: binary, equal to 1 if the kth route is chosen as the primary path for demand d
- N_i^d: binary, equal to 1 if node i appears in the backup path of demand d
- B_e^d: binary, equal to 1 if backup path of demand d uses link e, e member of E

— With pre-assigned channels (assuming fixed link capacities): We define $B_{e,c}^d$ to be 1 if backup path of demand d uses channel c, $B_e^d = \sum_c B_{e,c}^d$.

- W_e: capacity on link e, e member of E (upper-bounded if pre-assigned channels.)
- $Z_{e,f}^d$: binary, equal to 1 if backup path of demand d uses link e upon failure scenario f, e member of E, f belongs to the set of all links of all primary paths for demand d.

— With pre-assigned channels (assuming fixed link capacities): We define $Z_{e,f,c}^d$ to be 1 if backup path of demand d uses channel c upon failure scenario f, $Z_{e,f}^d = \sum_c Z_{e,f,c}^d$

Objective: Minimize the total cost

$$\sum_{e \in E} C_e W_e.$$

Constraints: One primary path is chosen for each demand

$$\sum_{i \in R^d} P_i^d \quad \forall d.$$

Primary and backup paths must be diverse

$$B_e^d + P_k^d \leq 1 \quad \forall d \in D \quad \forall k \in R^d \quad \forall e \in R_k^d.$$

Flow conservation equations that determine the backup path

$$\sum_{e:e=(i,j),e\in E} B_e^d - \sum_{j:(i,j)\in E} N_j^d = \begin{cases} 1, & \text{if } i \in \{s_d, t_d\} \quad \forall d \in D,\ \forall i \in N \\ 0, & \text{otherwise.} \end{cases}$$

The constraint on the link capacity must be satisfied

$$\sum_{d\in D} \sum_{i:e\in R_i^d} P_i^d + \sum_{d\in D} Z_{e,f}^d \le W_e \quad \forall e \in E \quad \forall f \ne e.$$

Constraints on the sharing variables

$$Z_{e,f}^d \le B_e^d \quad \forall d \in D \quad \forall e \in E \quad \forall f \ne e$$
$$Z_{e,f}^d \ge P_k^d + B_e^d - 1 \; \forall d \in D \quad \forall k \in R^d \quad \forall f \in P_k^d \quad \forall e \ne f.$$

Or, with pre-assigned channels the constraint on the sharing variables is

$$Z_{e,f,c}^d \ge P_k^d + B_{e,c}^d - 1 \quad \forall d \in D \quad \forall k \in R^d \quad \forall f \in P_k^d \quad \forall e \ne f \quad \forall c.$$

REFERENCES

[1] G. Ellinas, E. Bouillet, R. Ramamurthy, J.-F. Labourdette, S. Chaudhuri, and K. Bala, "Routing and restoration architectures in mesh optical networks," *Opt. Netw. Mag.*, vol. 4, no. 1, pp. 91–106, Jan.–Feb. 2003.

[2] T. Stern and K. Bala, *Multiwavelength Optical Networks, A Layered Approach*. Reading, MA: Addision Wesley, 1999.

[3] J.-F. Labourdette, E. Bouillet, R. Ramamurthy, G. Ellinas, S. Chaudhuri, and K. Bala, "Routing strategies for capacity-efficient and fast-restorable mesh optical networks," *Photonic Network Communications (Special Issue on Routing, Protection and Restoration Strategies and Algorithms for WDM Optical Networks)*, vol. 4, no. 3/4, pp. 219–235, Jun.–Dec. 2002.

[4] G. Ash, K. Chan, and J.-F. Labourdette, "Analysis and design of fully shared networks," in *14th Int. Teletraffic Congr.*, Antibes, Juan-Les-Pins, France, Jun. 1994, pp. 1311–1320.

[5] Y. Xiong and L. Mason, "Restoration strategies and spare capacity requirements in self-healing atm networks," *IEEE/ACM Trans. Netw.*, vol. 7, no. 1, pp. 98–110, Feb. 1999.

[6] B. Doshi *et al.*, "Optical network design and restoration," *Bell Labs Tech. J.*, vol. 4, no. 1, pp. 58–84, Jan./Mar. 1999.

[7] R. Doverspike *et al.*, "Fast restoration in a mesh network of optical cross-connects," in *Proc. Optical Fiber Communication Conf. (OFC)*, San Diego, CA, Feb. 1999, pp. 170–172.

[8] R. Iraschko, M. MacGregor, and W. Grover, "Optimal capacity placement for path restoration in STM or ATM mesh-survivable networks," *IEEE/ACM Trans. Netw.*, vol. 6, no. 3, pp. 325–336, Jun. 1998.

[9] E. Bouillet, P. Mishra, J.-F. Labourdette, K. Perlove, and S. French, "Lightpath re-optimization in mesh optical networks," presented at the 7th Eur. Conf. Networks and Optical Communications (NOC), Darmstadt, Germany, Jun. 2002.

[10] J. Doucette and W. D. Grover, "Comparison of mesh protection and restoration schemes and the dependency on graph connectivity," presented at the 3rd Int. Workshop Design of Reliable Communication Networks, Budapest, Hungary, Oct. 2001.

[11] E. Bouillet, J. Labourdette, G. Ellinas, R. Ramamurthy, and S. Chaudhuri, "Local optimization of shared backup channels in optical mesh networks," presented at the Optical Fiber Communication Conf. (OFC), Atlanta, GA, Mar. 2003.

[12] T. Lakshman and M. Kodialam, "Dynamic routing of bandwidth guaranteed tunnels with restoration," in *Proc. IEEE INFOCOM*, Tel Aviv, Israel, Mar. 2000, pp. 902–911.

[13] **<PLEASE PROVIDE PAGE NUMBERS.>** S. Samieian, Z. Bogdanowicz, R. Ramamurthy, S. Chaudhuri, and K. Bala, "Capacity requirements of ring, 1 + 1 and mesh configurations," *Lightw. Mag.*, Aug. 2001.

[14] R. Ramamurthy, Z. Bogdanowicz, S. Samieian, D. Saha, B. Rajagopalan, S. Sengupta, S. Chaudhuri, and K. Bala, "Capacity performance of dynamic provisioning in optical networks," *IEEE J. Lightw. Technol.*, vol. 19, no. 1, pp. 40–48, Jan. 2001.

[15] A. Akyamaç *et al.*, "Ring speed restoration and optical core mesh networks," presented at the 7th Eur. Conf. Networks and Optical Communications (NOC), Darsmstadt, Germany, Jun. 2002.

[16] S. Datta, S. Sengupta, S. Biswas, and S. Datta, "Efficient channel reservation for backup paths in optical mesh networks," in *Proc. IEEE GLOBECOM*, vol. 4, San Antonio, TX, Nov. 2001, pp. 2104–2108.

[17] D. Anding, "Service offerings utilizing digital cross-connect systems-a bell south perspective," in *Int. Conf. Communications*, Philadelphia, PA, Jun. 1998, pp. 336–339.

[18] T. Noh, D. Vaman, and X. Gu, "Reconfiguration for service and self-healing in ATM networks based on virtual paths," *Comput. Netw.*, vol. 29, no. 16, pp. 1857–1867, Dec. 1997.

[19] J. A. S. Monteiro and M. Gerla, "Topological reconfiguration of atm networks," in *Proc. IEEE INFOCOM*, San Francisco, CA, Jun. 1990, pp. 207–214.

[20] A. Herschtal and M. Herzberg, "Dynamic capacity allocation and optimal rearrangement for reconfigurable networks," in *Proc. IEEE GLOBECOM*, vol. 2, Singapore, Nov. 1995, pp. 946–951.

[21] D. Medhi, "Multi-hour, multitraffic class network design for virtual path-based dynamically reconfigurable wide-area ATM networks," *IEEE/ACM Trans. Netw.*, vol. 3, no. 6, pp. 809–818, Dec. 1995.

[22] S. Datta and Z. Bogdanowicz, "Route re-optimization in optical shared mesh networks," presented at the IASTED Int. Conf. Wireless and Optical Communications, Alberta, Canada, Jul. 2002.

[23] I. Baldine and G. N. Rouskas, "Dynamic reconfiguration policies for WDM networks," in *Proc. IEEE INFOCOM*, vol. 1, New York, Mar. 1999, pp. 313–320.

[24] G. N. Rouskas and M. Ammar, "Dynamic reconfiguration in multihop WDM networks," *J. High Speed Netw.*, vol. 4, no. 3, pp. 221–238, Jun. 1995.

[25] J.-F. P. Labourdette, G. W. Hart, and A. S. Acampora, "Branch-exchange sequences for reconfiguration of lightwave networks," *IEEE Trans. Commun.*, vol. 42, no. 10, pp. 2822–2832, Oct. 1994.

[26] D. Banerjee and B. Mukherjee, "Wavelength-routed optical networks: linear formulation, resource budgeting tradeoffs, and a reconfiguration study," in *Proc. IEEE INFOCOM*, Kobe, Japan, Apr. 1997, pp. 269–276.

[27] M. Sridharan, A. K. Somani, and M. V. Salapaka, "Approaches for capacity and revenue optimization in survivable WDM networks," *J. High Speed Netw.*, vol. 10, no. 2, pp. 109–125, Aug. 2001.

[28] A. Gencata and B. Mukherjee, "Virtual-topology adaptation for WDM mesh networks under dynamic traffic," in *Proc. IEEE INFOCOM*, vol. 1, New York, Jun. 2002, pp. 48–56.

[29] D. Banerjee and B. Mukherjee, "Wavelength-routed optical networks: linear formulation, resource budgeting tradeoffs, and a reconfiguration study," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 598–607, Oct. 2000.

[30] E. Bouillet, J.-F. Labourdette, R. Ramamurthy, and S. Chaudhuri, "Enhanced algorithm cost model to control tradeoffs in provisioning shared mesh restored lightpaths," presented at the Optical Fiber Communication Conf. (OFC), Anaheim, CA, Mar. 2002.

[31] C. Qiao *et al.*, "Novel models for efficient shared-path protection," presented at the Optical Fiber Communication Conf. (OFC), Anaheim, CA, Mar. 2002.

[32] J. Doucette, W. Grover, and T. Bach, "Bi-criteria studies of mesh network restoration: Path length versus capacity tradeoffs," presented at the Optical Fiber Communication Conf. (OFC), Anaheim, CA, Mar. 2001.

[33] S. Baroni, "Routing and Wavelength Allocation in WDM Optical Networks," Ph.D. thesis, Dept. Electron. Electr. Eng., Univ. London, London, U.K., May 1998.

[34] Y. Liu, O. Tipper, and P. Siripongwutikorn, "Approximating optimal spare capacity allocation by successive survivable routing," in *Proc. IEEE INFOCOM*, Anchorage, AL, Apr. 2001, pp. 699–708.

[35] P. Charalambous, G. Ellinas, C. Dennis, E. Bouillet, J.-F. Labourdette, S. Chaudhuri, M. Morokhovich, and D. Shales, "Dynegy's national long-haul optical mesh network utilizing intelligent optical switches," presented at the LEOS 2002 Annual Meeting, Glasgow, Scotland, Nov. 2002.

[36] P. Charalambous *et al.*, "A national mesh network using optical cross-connect switches," presented at the Optical Fiber Communication Conf. (OFC), Atlanta, GA, Mar. 2003.

Eric Bouillet received the M.S. and Ph.D. degrees in electrical engineering from Columbia University, New York, NY. He also holds a joint degree from l'Ecole Nationale Supérieure des Télécommunications ENST Paris and EURECOM Sophia Antipolis.

He is currently with the IBM T. J. Watson Research Center, Hawthorne, NY. Before joining IBM, he was a Network Architect at Tellium, responsible for the design of optical networks and optimization of lightpath provisioning and fault restoration algorithms. Previously, he was a member of the technical staff in the Mathematical Sciences Research Center, Bell Labs, Lucent Technologies, Murray Hill, NJ, working in the areas of routing in ATM and optical networks. He has authored or co-authored several journal and conference papers and has a number of U.S. and international patents pending in the area of optical networking.

Jean-Francois Labourdette (SM'??) **<PLEASE PROVIDE YEAR(S) OF IEEE MEMBERSHIP.>** received the undergraduate degree in electrical engineering from l'Ecole Nationale Supérieure des Télécommunications, Brest, France, and the Ph.D. degree from Columbia University, New York, NY.

is Solutions Architect Manager with Verizon Enterprise Sales Group. Before joining Verizon in March 2004, he was Manager of system engineering at Tellium, responsible for all network element and network management system engineering activities. He joined Tellium in October 2000 as Manager of network routing and design, responsible for Tellium's routing architecture and algorithms, network dimensioning, and customer network design activities. Previously, he was a Manager of multiservice globalization planning at AT&T, where he was responsible for network, service, and operations planning for AT&T international data services (frame relay, ATM, and MPLS IP VPN). Earlier at AT&T, he was a System Engineer in the routing planning group, working on dynamic call routing for AT&T's switched network and facility routing and rearrangement for the AT&T transport network. He has more than 40 publications in international conferences and journals.

Dr. Labourdette was the recipient of the 1991 Eliahu I. Jury Award for best dissertation. He is a member of the Optical Society of America.

Ramu Ramamurthy received the B.Tech. degree from the Indian Institute of Technology, Madras, India, and the M.S. and Ph.D. degrees from the University of California at Davis.

He is currently a Senior Network Architect with Tellium, Oceanport, NJ, where he works on the design of algorithms and protocols for dynamic provisioning and restoration in optical networks. Prior to joining Tellium, he was a Research Scientist with Telecordia Technologies, where he worked on network control and the management of IP/WDM optical networks.

Sid Chaudhuri received the Ph.D. degree in physics from the University of Pittsburgh, PA, in 1981.

He joined AT&T Bell Labs in 1984 where he developed AT&T's Core Transport Network architecture using leading edge optical technologies—SONET, DWDM, and optical cross-connect systems. While at AT&T Laboratories he developed intelligent optical network architectures using all-optical switching technologies. He joined Tellium in 2000, where, as the Director of Network Architecture, he led the development of system architecture for optical switching systems, dynamic provisioning, fast mesh restoration, and routing algorithms for optical networks. As the Physical and Link Layer Working Group chairman of the Optical Internetworking Forum (OIF), he led the development of the first set of OIF technical agreements. He also served as the chairman and president of OIF during 2001–2002. His early initiatives to bring optical and IP technologies together to create scalable and manageable optical internetworking architectures are at the forefront of architectural activities in both the optical networking research and standards communities. He has published over 70 papers on electronic and optical properties of impurities in bulk semiconductors and quantum well structures, and optical networking.