

# Distributed Computation of Shared Backup Path in Mesh Optical Networks Using Probabilistic Methods

Eric Bouillet, *Member, IEEE*, Jean-François Labourdette, *Senior Member, IEEE*

**Abstract**—In this paper we assess the benefits of using statistical techniques to ascertain the shareability of protection channels when computing shared mesh restored lightpaths in optical mesh networks. These optical networks support wavelength conversion everywhere as a byproduct of the electronic nature of the switching in the OEO optical cross-connect used. Current deterministic approaches require a detailed level of information proportional to the number of active lightpaths. Although this is not an issue for good size networks in the foreseeable future, these approaches are not practicable for distributed route computation involving larger networks. On the other hand, distributed approaches that do not make use of shareability information require a significant amount of additional capacity compared to a centralized approach with access to complete shareability information. With the proposed approach we show that even with less information, independent of the amount of traffic demand, it is possible to predict the shareability of protection channels with remarkable accuracy. In addition, we propose a local distributed channel assignment scheme that is used in conjunction with our distributed route computation proposal to assign shared channels when provisioning the backup path. This channel assignment scheme can also be used to further optimize capacity usage in individual links upon certain events or at regular intervals. Experiments are provided, that demonstrate that our approach yields faster computation times with no significant penalty in terms of capacity usage than a centralized approach using complete information.

**Index Terms**—Optical networks, Optical Switching, Mesh Protection, Probabilistic Algorithm, Performance Analysis.

## I. INTRODUCTION

Dense Wavelength Division Multiplexed (DWDM) mesh network infrastructures that switch optical connections (lightpaths) using intelligent optical cross-connects (OXC) are emerging as the technology of choice for data networks[1]. In these architectures a single piece of equipment is capable of transferring tens of terabits per second. This equipment is continuously exposed to multifarious risks of breakdown, either due to human-induced mishaps, or to equipment malfunctions. In order to guarantee service persistence in such circumstances it is common for a carrier to reserve spare bandwidth on alternate paths, so that a service affected by a failure along its primary lightpath can be rapidly restored using the reserved bandwidth. Among the possible schemes for provisioning backup paths, dedicated mesh protection and shared mesh restoration seem to be the most appropriate approaches in the context of DWDM networks[2], [3], [4], [5], [6].

In dedicated mesh protection, as shown in Figure 1, the lightpath provisioning algorithm computes and establishes simultaneously the primaries and their protection paths. During normal operation mode, both paths carry the optical signal and the egress selects the best copy of the two. The concept of Shared Risk Group (SRG) was introduced to select the paths so that they will not be affected by a single failure[7], [8]. An SRG expresses the relationship that associates optical lines (or channels) with a single failure. As shown in Figure 2, an SRG may consist of all the optical lines in a single fiber, or the optical lines through all the fibers wrapped in the same cable, or all the optical lines traversing the same conduit. Since a fiber can traverse several conduits, an optical line may belong to several SRGs. It suffices that a primary and its backup path are SRG disjoint to ensure that at least one path survives any single failure affecting all the optical lines in an SRG.

As in dedicated protection, shared mesh restored paths are predefined, except that the cross-connections along the paths are not created until a failure occurs. During normal operation modes the spare channels reserved for protection are not used. We refer to such channels as reserved (for restoration) channels. Since the capacity is only "soft reserved", the same channel can be shared to protect multiple lightpaths. There is a condition though that two backup lightpaths may share a reserved channel only if their respective primaries are mutually SRG disjoint, so that a failure does not interrupt both primary paths. Two lightpaths, or their protection, are said to be mutually compatible, if they are not affected by the same failure. If not, they are "conflicting". Figure 3 (for normal mode) and Figure 4 (for restoration mode) illustrate an example of shared mesh restoration. The network consists of four client nodes ( $A$  to  $D$ ) and two demands ( $AB$  and  $CD$ ) accommodated across an eight node optical network ( $S$  to  $Z$ ). The dashed lines represent channels reserved for protection. Using the routing of Figure 3, demands  $AB$  and  $CD$  are compatible with respect to SRG-failures and thus their protection share a single optical line in link  $S-T$ , one less than would be required in dedicated protection. Upon failure as depicted in Figure 4, the egress and egress nodes of the disconnected paths ( $X$  and  $Z$ ) emit a request to the switches along the protection paths ( $S$  and  $T$ ) to establish the cross-connections for that path. Once the cross-connections are established, each ingress and egress node restores the connection to the new path. This architecture requires fewer resources than in dedicated protection, but the restoration involves more processing to signal and establish the cross-connections along the restoration path.

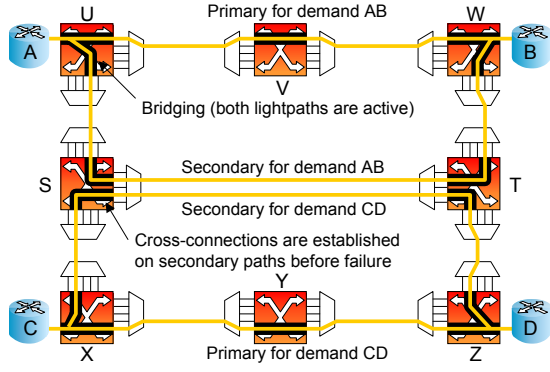


Fig. 1. Dedicated (1+1) mesh protection

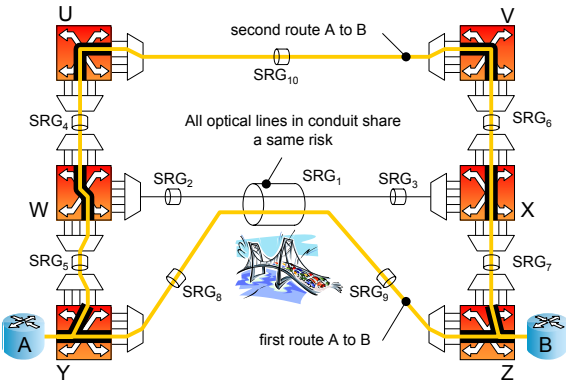


Fig. 2. Example of non-trivial SRG

There are two different policies to assign channels to protection paths[9], [10]. A failure independent strategy assigns the protection channel at the time of provisioning before failures occur. A failure dependent strategy (channel pooling) assigns the protection channels along a pre-computed route after failure occurrence and relies on the restoration signaling mechanism to select the channels on each link along the route from a pool of reserved channels [11]. A proper spare channel-provisioning scheme reserves enough channels on each link so that all lightpaths can be restored for every type of single failure. Figure 5 illustrates this with an example of a failure dependent shared-mesh restoration strategy. The example consists of three demands ( $AB$ ,  $AC$ , and  $BD$ ), routed across a six-node optical network in such a way that every combination of primary lightpath pairs, but not all three primary lightpaths at once, can fail simultaneously. If a failure dependent strategy is used, only two channels need to be reserved on the protection path, since at most two lightpaths will fail simultaneously. If however a failure independent strategy is used, we must reserve three protection channels on link  $(Y, Z)$  in order to accommodate all failure scenarios affecting links  $(U, V)$ ,  $(V, W)$  or  $(V, X)$ , even though at most two of the three channels will be used at any time. The reason for this is that link  $(Y, Z)$  is traversed by the protection paths of all three demands, for each of which a protection channel

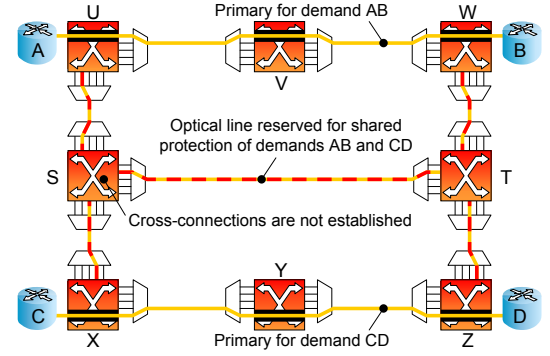


Fig. 3. Shared mesh restoration, before failure

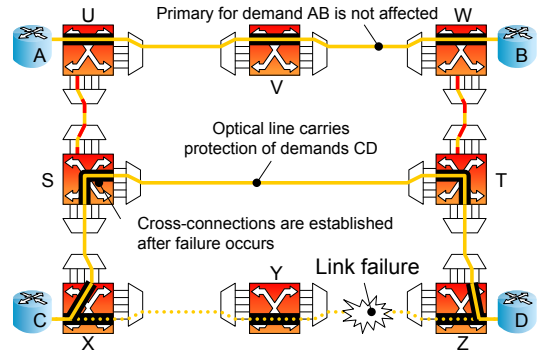


Fig. 4. Shared mesh restoration, upon failure of link Y-Z

must be pre-assigned.

Although more cost-efficient, the failure dependent approach requires additional inter-node communication to agree on the channel assignment during restoration [12], and to it we prefer the failure independent approach, which has achieved sub-200ms restoration times in large networks[13], [14]. However the gain in restoration time requires filling up the backup-to-channel lookup tables at each node during provisioning (when speed is less of an issue.) We assume that the failure independent strategy is used in the remainder of the paper, that is channels on backup paths are pre-assigned.

Now, consider the online problem of provisioning a mesh-restored lightpath using a centralized Route Computation Module (RCM), assuming a failure independent strategy. Since this problem is proved to be NP-complete if minimization of the total capacity usage (working plus protection) is sought[10], a possible approach is to enumerate a list of  $K$  minimum cost primary paths and for every one of them compute the corresponding minimum cost restoration path and reserve the channels along that path. The RCM then returns the pair of paths with the lowest combined cost. The cost of a pair is the cost of the channels along both paths, excluding the cost of (preexisting) shareable reserved channels along the backup path. Given a primary path, we compute the minimum cost backup-path by:

- (i) setting the cost of the links (SRGs) traversed by the

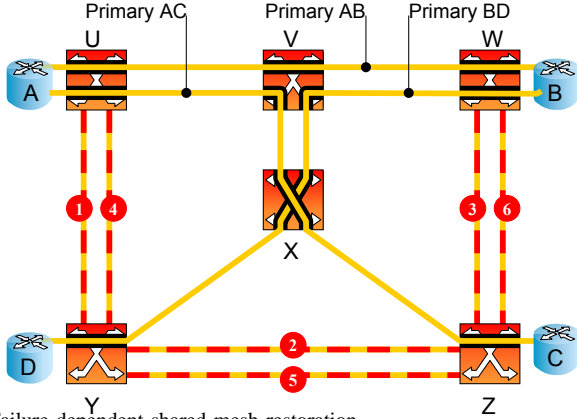


Fig. 5. Failure dependent shared mesh restoration

- primary path to  $\infty$ ,
- (ii) setting the cost of links with shareable channels to a constant  $\epsilon \ll 1$ ,
- (iii) run a shortest path algorithm using the modified link cost metric.

Step (i) and (ii) respectively ensure that primary and backup paths are SRG-diverse, and that the minimum cost backup path is found using shareable reserved channels whenever possible. In the following we are interested in step (ii), which consists of identifying shareable reserved channels. We show in particular that the time-complexity of this operation, if deterministic, is proportional to the total number of reserved channels, and thus does not scale well when the number of lightpaths established in the network becomes large. We then present a probabilistic approach to execute this operation with a certain probability of accuracy. We show that by trading a deterministic TRUE or FALSE statement for a PERHAPS statement with a measurable likelihood that PERHAPS is TRUE, the operation can be made independent of the number of reserved channels. The benefits of this substitution are (1) reduction of the path computation time and (2) reduction of the amount of information necessary to compute the paths, with no penalty or small penalty in terms of capacity efficiency. The probabilistic approach computes the restoration paths, but, unlike the deterministic approach, it does not provide the channels along the restoration path, and this assignment must be done separately as provisioning of the path takes place on a link-by-link basis. We show in this paper that this backup channel assignment operation is tantamount to a graph-coloring problem. In particular we show how a first fit based assignment can be easily improved using a graph-coloring algorithm. The result is a routing architecture that is more comparable to the failure dependent channel allocation policy in terms of computation and information complexity, but still maintain the restoration latency of the failure independent strategy by pre-allocating the reserved channels to the restoration paths.

In this work we assume that provisioning would be based on a distributed topology update and signaling approaches, using protocols such as GMPLS [15], [16], and possibly proposed

extensions to the signaling messages [17]. However this is not covered here but is the subject of ongoing research [18]. Without loss of generality we assume that all the lightpaths are bi-directional. We consider the case of Optical-Electronic-Optical (OEO) optical cross-connects only. All the OXC in the network terminate the optical signal and convert it in the electrical domain where switching, monitoring, control and signal regeneration functions are performed. In particular wavelength conversion, as opposed to all optical switches, is available everywhere in the network and is not an issue here. The case of all-optical networks implies a different set of constraints [19], [20]. It requires protocols and algorithms that are different than the one described below, and some of our claims regarding to the restoration performances of the considered protection strategies may not apply to this type of networks.

The paper is organized as follows. Section II analyzes the complexity of the deterministic approach to identify shareable channels. Section III describes the details of the probabilistic approach. Section IV describes an algorithm to compute mesh restored paths using the probabilistic approach. Since this approach does not provide the channels along the restoration path, section V describes a distributed algorithm to optimize this channel assignment separately. Section VI compares the results for realistic topologies using both probabilistic and deterministic based algorithms, and Section VII concludes this paper.

## II. COMPLEXITY OF DETERMINISTIC APPROACH

In this section we compare the complexity, expressed in terms of processing time, and amount of information required by the algorithm, when determining a protection path in a failure dependent and a failure independent strategy. In both strategies we assume that the primary lightpath is given, and that a deterministic approach is employed. Note that we measure here the complexity of computing the protection path of a new service. This time should not be confounded with the restoration latency, which is the delay required to restore all the services on the pre-computed protection paths when failures occur.

### A. Complexity of the failure dependent strategy

The failure dependent strategy only requires that sufficient protection channels be reserved on each link so that in any failure event all the affected protection paths can be accommodated. Suppose that every link maintains an associative array indicating for each SRG the number of times the SRG is traversed by a primary lightpath whose corresponding protection path traverses the link. The maximum value in this associative array is thus the maximum number of protection paths that would be concurrently activated on this link in a worst-case scenario. Therefore, if protection channels are assigned during restoration, a sufficient condition to guarantee full recovery of a single SRG failure is that each link of the network must have enough channels to accommodate its respective maximum number of concurrently activated protection paths.

We denote by  $h$  the average primary path length expressed in number of links, and denote by  $m$  the number of links present in the network. We also assume that the total number of SRGs is on the order of  $O(m)$ . The condition above can thus be determined in  $O(mh)$ -time based on the information available in the associative arrays. The combined size of the arrays is on the order of  $O(m^2)$ , which is reasonable for link-state dissemination protocols such as OSPF.

### B. Complexity of the failure independent strategy

In a failure independent strategy, the protection channels must be specifically assigned to the protection paths in a way that satisfies any foreseeable combination of protection path activation. In what follows, a list of SRGs protected by a given reserved channel consists of all distinct SRGs traversed by all the primary lightpaths whose respective protection paths are assigned the reserved channel. Thus a reserved channel can be reused to protect a primary path if no SRG traversed by the primary path appears in the list of SRGs already protected by the channel. The complexity of determining reusable channels is a function the following parameters:

- $h$  denotes the average primary path length expressed in number of links,
- $h'$  denotes the average length of the backup path (usually  $h' \geq h$ .)
- $m$  denotes the number of links in the network,
- $x$  denotes the total number of protection channels reserved throughout the network.

We also assume the typical case where the average number of protected SRGs per reserved channel is on the order of  $O(m)$ .

Shareable reserved channels in the network are identified by verifying that for each reserved channel in each link that the list of SRGs protected by the channel does not intersect with the list of SRGs traversed by the primary path. Therefore, the complexity of identifying all the links with shareable reserved channels in the network is  $O(hx)$ . This complexity assumes that each reserved channel maintains a fixed length array in which each entry indicates whether an SRG is used or not. It becomes  $O(hx \log m)$  if instead a variable length list of protected SRG is used. The number of protection channels is a function of  $g$ , the number of lightpaths in the network, and can be approximated by  $x = O(gh')$ . Substituting  $O(gh')$  for  $x$ , the complexity of identifying the links with shareable channels is  $O(ghh')$ . This operation requires that the list of SRGs protected by each channel be known. The size of this information is on the order of  $O(gmh')$ . Our primary concern here is the dependence of this time and size complexity on the number of lightpaths established in the network. We thus propose to substitute this time consuming deterministic approach for a probabilistic approach whose complexity remains constant with respect to the number of lightpaths.

### III. PROBABILISTIC APPROACH

In what follows, we assume that the RCM has an up-to-date knowledge of the state of the network, which includes for each link  $(u, v)$ : (i) the number  $M$  of reserved channels

in link  $(u, v)$ , and (ii) for every SRG  $j$  in the network, the number  $n_j$  of reserved channels in link  $(u, v)$  that protect a working path traversing the SRG.

We now describe the technique used to quickly compute the probability that a reserved channel is shareable with respect to a given primary path and based on the information available to the RCM. We will first introduce a simple combinatorial problem, solve it, and show the analogy between this problem and the one that we are interested in.

#### A. A simple problem of combinations

##### The problem:

We are given  $N$  bags tagged from 1 to  $N$ , filled with marbles. Bag  $j$  ( $j \in \{1, \dots, N\}$ ) contains  $n_j$  marbles. All marbles in any given bag have the same color, but marbles in different bags have different colors, so that there is a one-to-one mapping between bags and colors. We are also given  $M$  bins tagged from 1 to  $M$ . We assume for the moment that the bins have infinite capacity. Next we empty the bags in the bins, so that not two marbles of the same bag (or same color) fall in the same bin. The questions are:

- 1) How many differentiable combinations (denoted by  $Q$ ) of marbles to bins are possible? Assume that we cannot distinguish between two marbles of the same color.
- 2) Out of all combinations computed in (1), how many of them (denoted by  $D$ ) have empty bins left?
- 3) What is the probability that at least one bin is empty? Assume that probability of occurrence is the same for all combinations computed in (1) and (2), i.e. the marbles are uniformly distributed in the bins.

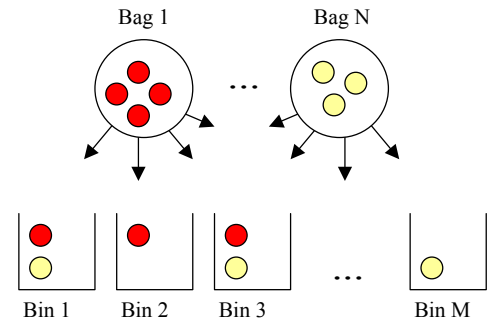


Fig. 6. Bins and bags problem

##### The answers:

In the following we use  $C(p, q) = q! / (p!(q-p)!)$  to denote the unordered combinations of  $p$  out of  $q$  elements.

- 1) First note that a solution exists if and only if  $M \geq n_j$ ,  $\forall j \in \{1, \dots, N\}$ . The marble arrangement of each bag into the  $M$  bins is not conditional to other bags' arrangements. For each bag  $j$  there are  $C(n_j, M)$  possible ways to arrange the  $n_j$  marbles into an ordered set of  $M$  bins. There are thus  $Q = \prod_{j=1}^N C(n_j, M)$  arrangements.
- 2) First note that if there exists  $j \in \{1, \dots, N\}$  such that  $M = n_j$  then there is no such combination, and the answer is  $D = 0$ . Note also that if  $M > \sum_{j=1}^N n_j$  then

we cannot fill all the bins and the answer is  $D = Q$ . If  $M = 1 + \max_{1 \leq j \leq N} n_j$ , the problem is equivalent to (1), except that now arrangements are confined to  $M - 1$  bins, the last bin being left empty. They are thus  $D = C(M - 1, M) \prod_{j=1}^N C(n_j, M - 1)$  possible arrangements. Otherwise, if  $M > 1 + \max_{1 \leq j \leq N} n_j$ , then  $D = r(M - 1)$ , where  $r(k)$  is the recursion over integers in  $k \in \{\max_{1 \leq j \leq N} n_j, \dots, M - 1\}$  such that  $r(\max_{1 \leq j \leq N} n_j - 1) = 0$ , and  $r(k) = \left[ C(k, M) \prod_{j=1}^N C(n_j, k) \right] - r(k - 1)$ . (See Appendix in Section IX)

- 3) The probability that at least one bin is empty, is equal to the ratio of the number of all the combinations with empty bins to the total number of combinations, that is:  $P = D/Q$

The exact probability that one bin is empty is:

$$P = \frac{r(M - 1)}{\prod_{j=1}^N C(n_j, M)} \quad (1)$$

$$r(k) = \left[ C(k, M) \prod_{j=1}^N C(n_j, k) \right] - r(k - 1)$$

$$r(\max_{1 \leq j \leq N} n_j - 1) = 0$$

$$k \in \{ \max_{1 \leq j \leq N} n_j, \dots, M - 1 \}$$

The estimated probability:

Note that the computation of  $D$  and  $Q$  may be tedious. We thus show here a mean to approximate this probability. First observe that the probability that at least one bin is empty is complement to the probability that all bins are non-empty. And the probability that a bin is non-empty is the complement to the probability  $p$  that this bin is empty. Although  $p$  is conditional to the probability of other bins being empty we assume that it is independent and identical for all bins. Therefore, given a bin the probability  $p$  that the bin is empty is the product of independent probabilities that all marbles of each bag are in other bins, that is  $p = \prod_{j=1}^N (1 - n_j/M)$ . Based on our observations and assumption, the probability that at least one bin is empty is  $P = 1 - (1 - p)^M = 1 - \left(1 - \prod_{j=1}^N (1 - n_j/M)\right)^M$ . The complexity of computing  $P$  (or its complement  $1 - P$ ) involves computing  $N$  products and an  $M^{th}$  power. It is realizable in  $O(N + \log M) \approx O(N)$  time.

The estimated probability that one bin is empty is:

$$P = 1 - \left(1 - \prod_{j=1}^N \left(1 - \frac{n_j}{M}\right)\right)^M \quad (2)$$

*B. Analogy with SRG arrangement into a set of reserved channels*

Assume that the  $M$  bins of the problem presented in III-A are the reserved channels in a given link. And assume that the  $N$  bags represent a list of  $N$  SRGs traversed by the primary path for which a reserved channel is sought. The  $n_j$  marbles denote the number of times each SRG of the list is protected (through pre-established paths) by the reserved channel set.

Evidently the same SRG cannot be protected multiple times by the same reserved channel otherwise contention would exist through their respective primaries if the SRG fail. This restriction is expressed in the problem formulation by the fact that two identical marbles (same SRG) cannot fall into the same bin (reserved channel). Thus, the problem above deals with computing the probability that there is at least one shareable reserved channel, i.e. a reserved channel that does not contain any of the  $N$  SRGs. We have shown that this probability is approximated in  $O(N)$  time, where  $N$  is the number of SRGs on the primary path. Typically  $N$  is the average path length  $h$ . Therefore, the time complexity of identifying all the links with shareable reserved channels in the network is  $O(hm)$ . This complexity is to be compared with  $O(ghh')$  of the deterministic approach.

Remember that in the computation of these probabilities we have made two simplifying assumptions: (i) the probability of a reserved channel being shareable is pair-wise independent of other reserved channels, and (ii) SRGs are uniformly distributed across reserved channels. The effect of the first assumption is easy to quantify by way of simulations (see Section VI.) The effect of the second assumption on the other hand is subtler because it depends on the policy used for allocating reserved channels. For instance a "First Fit" or "Max Fit" policy tends to pack (protect) more SRGs in some reserved channels than others within the same link. As it turns out, a First Fit policy increases the probability that a reserved channel is available compared to a uniformly randomized allocation.

#### IV. PROBABILISTIC ROUTING ALGORITHM WITH PARTIAL INFORMATION

We describe here in details an algorithm that implements the probabilistic approach, and compare it with the equivalent deterministic algorithm.

Given: a topology represented as a graph  $G(V, E)$  where vertices represent optical cross-connects (OXC) and links represent fiber strands between OXCs. A network state database, that indicates for each link the number of channels available, the number of reserved channels, and the number of times each SRG in the network is protected by a reserved channel in that link. The latter information is stored into an array. The array's indices correspond to SRGs and each entry in the array counts the number of reserved channel cross-connections that would occur in the link if the corresponding SRG fails.

Input: a pair of nodes  $A-Z$

Output: a pair of bi-directional lightpaths from  $A$  to  $Z$ , primary and secondary with minimum cost, excluding restoration channels that are shared with pre-established backup paths.

Algorithm ComputeRoute:

- 1) Compute  $k$ -shortest paths. Sort the paths by length and denominate them  $w_1$  to  $w_k$ .
- 2) Set  $S = \emptyset$
- 3) For each shortest path  $w_i$ , do:

- a) To each link that shares an SRG with  $w_i$  or has neither available channel nor reserved channel, assign infinite weight
- b) For each link without a reserved channel, set weight to cost of link
- c) For each link with reserved channel, set weight to cost of link times the probability that no reserved channel is shareable (by way of the approach presented earlier in section III, equation 2 of this document.)
- d) Compute the shortest path  $s_i$  using the metric defined in parts 3a to 3c, and set  $S \leftarrow S + \{w_i, s_i\}$
- 4) Select the minimum cost path pair  $\{w_k, s_k\} \in S$ .
- 5) If no path can be found in 4, return NO\_PATH, otherwise return  $\{w_k, s_k\}$ .

Upon a request to provision a new service, the ComputeRoute procedure is invoked to compute a "probably" most cost-efficient route. This route is not guaranteed to be feasible however: because of its probabilistic nature, or because the information used to compute the route could be outdated. Feasibility is thus verified during path setup, when cross-connections are created along the path. Backup channels are assigned locally, using a "first-fit" approach, or using some local optimization algorithm for instance (as described later in Section V). If the local channel assignment procedure cannot find a shareable backup channel, it creates one from the pool of available channels or fails if the pool of available channels is exhausted. If path setup fails, the signaling clears the partially created cross-connections, and returns an error message identifying the set of links that are unable to satisfy the request. The returned links are removed from the network, and a crank-back mechanism invokes the ComputeRoute procedure on the reduced network to compute an alternate route. This iteration is repeated up to a maximum number of crank-backs before giving up, or until a route is successfully setup. In Section VI, we present results that assume that backup channels are assigned on a first-fit basis during path setup signaling, and are re-optimized at the end of the simulation using the local channel assignment optimization described later in Section V. We make this assumption only for experimental purposes; in a distributed environment one could use the channel assignment optimization during path setup. We also assume that channels are always available (uncapacitated case) and therefore crank-back is not required. This is a realistic assumption since network deployment activities are usually planned to maintain network utilization below 70%.

The algorithm is self-explanatory. It differs from the deterministic algorithm only in step 3b and 3c. In the deterministic algorithm the weight of a link is set to the link cost times  $\epsilon \ll 1$  if it contains a shareable reserved channel and link cost if it does not. In the probabilistic algorithm this weight is replaced by the cost of the link times the probability that no reserved channel is shareable in the link. Note that the deterministic approach requires additional information to compute the routes. In particular it needs to know whether each SRG is protected or not for every reserved channel. Whereas in the probabilistic approach, only the number of times an SRG is protected in every link by any reserved channels of

that link needs to be known. Finally, note that we separated lightpath provisioning from routing, and channel assignment is performed in a distributed way after the lightpaths are selected by the RCM. The objective of the RCM is to compute the paths so that sharing is maximized during channel assignment. Even though a link may be erroneously tagged as having a shareable channel during path computation, the channel assignment procedure during path setup will guarantee that they are no sharing violation. In order to guarantee this, the scheme used for channel assignment requires the same information as for the deterministic approach, however this information can be distributed across the nodes in the network: it suffices that each node maintains a local database of all the reserved channels terminating into it. We address this backup channel assignment problem in the next section.

## V. LOCALLY OPTIMIZED CHANNEL SELECTION

### A. Shared mesh protection provisioning using vertex coloring

We use the term SRG to indicate a group of optical equipment that share a common risk of failure. Recall that two mesh restored protection paths are "compatible" and may share a protection channel if their respective primary paths are SRG disjoint. Otherwise they are said to be "conflicting". Although only single SRG failures are considered here, the description of the algorithm can easily be transposed to protect against node failure as well: replace SRG by node where it applies. Given a group of protection paths traversing a common link, the problem is to assign the minimum number of protection channels to the paths in the link in accordance to the rules of sharing. Typical online provisioning algorithm assigns protection channels on a first-come first-serve basis and reserve new channels when sharing is not possible with present protection channels. In this approach the number of protection channels depends ultimately on the order of arrival of the protection paths. Since the order cannot be determined in advance, an optimization algorithm must be invoked at regular intervals to reassign the channels. In this section, we show that finding the optimum assignment is equivalent to solving a vertex-coloring problem.

The allocation of protection channels is tantamount to a vertex-coloring problem: given the set of all restoration paths that intersect on a given link, represent every path as a vertex, and connect with an edge every pair of vertices whose corresponding paths are conflicting. Assign a distinctive color to each protection channel, and allot a protection channel to each path, that is color the vertices. Clearly, two vertices cannot be allotted the same color if they are connected by an edge, since the corresponding restoration paths are conflicting and cannot share a channel. The objective is to minimize the number of protection channels (respectively number of colors) required to accommodate all backup paths (respectively color all vertices), while avoiding conflicts.

This problem is known to be NP-hard, however there are many heuristics that can be used to compute sub-optimal solutions. A vertex-coloring algorithm that offers a good tradeoff between quality and runtime complexity is DSATUR[21].

1) *Example:* Consider the example of Figure 7 below (7a through 7d.) The figure illustrates five lightpaths  $\{AD, CD, BC, AC, BD\}$  and their protections, routed in a four-node ring network. All the protections traverse link  $C-D$ . The demands are provisioned following the sequence indicated in Table 7b. If we use a typical online shared-mesh protection provisioning, and apply the graph representation presented earlier to  $C-D$ , we obtain the "coloring" shown in Figure 7c. Even though a single failure in this example affects at most three primaries, this coloring consumes four colors, indicating that four protection channels are required. An optimized coloring yields the solution shown in Figure 7d, which consumes only three colors. Comparing figures 7c and 7d, we observe that a new channel ( $R$ ) should have been allotted to the protection path of demand ( $BC$ ) instead of sharing channel ( $B$ ) with the protection of demand ( $AD$ ). This solution however is not considered because not optimal when the third demand is being provisioned (that is demands  $\{AD, CD\}$  are routed and request for demand ( $BC$ ) has not yet arrived) since at that time it would consumes three channels ( $B, G, R$ ) instead of two ( $B, G$ ).

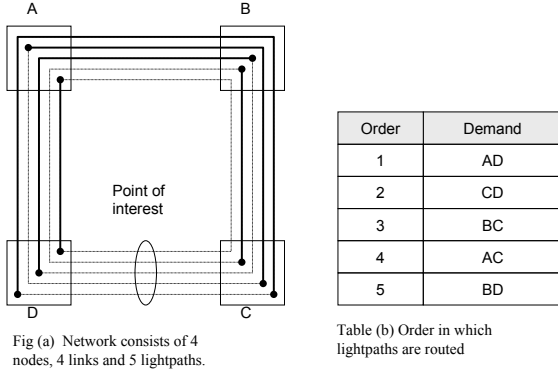


Fig (a) Network consists of 4 nodes, 4 links and 5 lightpaths.

Table (b) Order in which lightpaths are routed

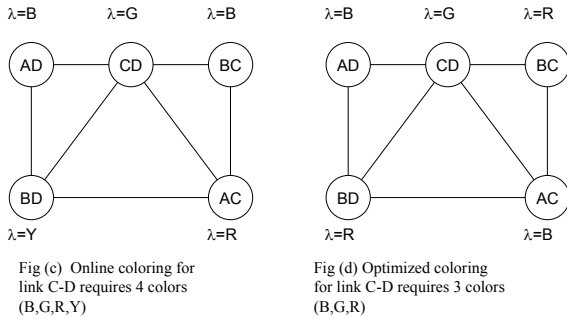


Fig (c) Online coloring for link C-D requires 4 colors (B,G,R,Y)

Fig (d) Optimized coloring for link C-D requires 3 colors (B,G,R)

Fig. 7. Example of sub-optimal (first-fit) and optimal local protection channel assignment.

### B. Implementation and Applications

With the probabilistic routing algorithm, the protection channels are not determined by the routing algorithm, and they thus need to be determined by the optimized channel assignment procedure each time a lightpath is being provisioned and signaled. Furthermore, the optimized channel reassignment

can be a low priority program thread running in background upon certain events, or at regular intervals. The information necessary to accomplish this task is available locally in every OXC and independent of non-adjacent OXCs. Thus each OXC can run a copy of the algorithm in a distributed manner, locally and independently of other OXCs. A change in the allocation of a protection channel needs only to be propagated to its end-points. Since protection channels are "booked" and actually not cross-connected until a restoration occurs, the task amounts to no more than modifying and exchanging sharing databases between pairs of adjacent nodes. For every OXC-pair connected by at least one optical line, the OXC with highest IP address is delegated to perform the task.

A byproduct of the optimized channel reassignment is that it can also be used to migrate the protection paths of mesh dedicated protections to shared mesh protections if desired. By changing their protection type to shared mesh protections, we allow the thread to apply the channel reassignment optimization to these services. The algorithm does not optimize the routes of the backup paths however, and the resulting solution is thus not as efficient as a re-optimization algorithm that reroutes the backup paths to maximize sharing[22].

Finally the channel re-optimization procedure closes an advantage gap of the failure dependent strategy over the failure independent strategy in the case of multiple failure scenarios. Since the protection channels are pre-assigned in the failure independent case, there is a higher probability that two services affected by two distinct failures contend for the same protection channel, even if there are parallel protection channels available[23], [24]. Re-provisioning mechanisms that compute protection paths on the fly when the planned protection fails would mitigate this problem, but are not covered here[25]. Then again, a background channel re-optimization process would detect the prospect for such contentions after the first failure, and re-assign the channels to eliminate them.

## VI. EXPERIMENTS AND RESULTS

### A. Accuracy and Distributions of Probability Functions

For the experiments presented in this section we used both the deterministic and probabilistic implementations of the algorithm. A great care was taken in optimizing the deterministic implementation for speed. The probabilistic code was then derived from the deterministic code by modifying step 3c as described above. We use the same first-fit channel assignment algorithm in both implementations in order to isolate and limit our measurements to the effects of using a probabilistic approach. The benefits of a local channel assignment optimization are measured separately in Section VI-C.

In the following we first measure the quality of the estimated probability that a link contains a shareable reserved channel given the information on the number of times each SRG traversed by the primary path is restored in that link. The experiment consists of simulating a large number of arbitrary instances of the problem presented in Section III-A. For each instance of the problem, we use a Monte-Carlo method[26] to generate several millions of randomly

selected arrangements, and compute the ratio of generated combinations with available reserved channels to the total number of generated combinations (i.e. estimate equation 1, and equation 2 computed in Section III-A). We then compare the difference between each experimental probability and the corresponding exact and approximate probabilities obtained by computation. The results are shown in Figure 8 and Figure 9. Figure 8 demonstrates the error distribution of the exact probabilities minus experimental probabilities obtained over the range of problem instances. The simulation exhibits an accuracy within 0.01 of the exact probability, and a closer look even indicates that 70% of the time the difference is within  $5 \times 10^{-4}$ . In comparison, we observe in Figure 9 that the estimate probability has a tendency to underestimate the experimental probability, but it is accurate within 0.05 for 85% of the time, which is quite remarkable given the simplicity of the computation. Note that this difference was expected due to the independence assumption made in the determination of equation 2.

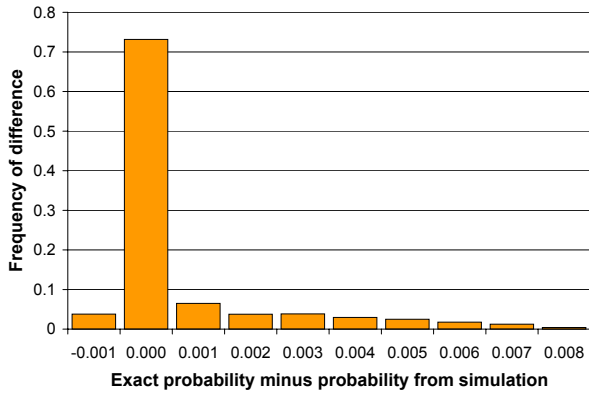


Fig. 8. Error distribution of exact sharing probability minus probability from simulation.

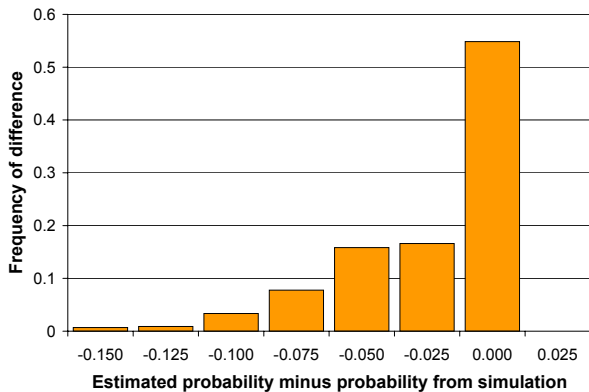


Fig. 9. Error distribution of estimated sharing probability minus probability from simulation.

### B. Comparison of deterministic versus probabilistic weight functions on real networks

In the next set of experiments we consider two scenarios inspired from real life networks. NetA is a 100-node, 137-

link network, with one unit of demand between every pair of node (4950 demands). NetB is 220-node, 300-link network, also with one unit of demand between every pair of node (24090 demands.) For the sake of simplicity we assume here that every link costs one unit of currency and corresponds to one SRG (i.e. one SRG per link and one link per SRG). We also assume that capacity is abundant, and path setup always succeeds without requiring crank-back. We then route the demands on each network using the deterministic and the probabilistic algorithms. We are interested here in the processing time to complete each algorithm, and the quality of the solutions expressed in total number of channels required (used for primaries and reserved for backups.) Tables I and II summarize the results. For NetA (respectively NetB) we observe that the probabilistic approach is 6.78 time faster (respectively 19.7 time faster) than the deterministic approach while the capacity penalty is only 2% (respectively 3%) more capacity. Also important is the amount of information the RCM needs to compute the routes. The probabilistic based RCM only requires one array per link, where each entry indicates the number of times the SRG is protected in the link by any reserved channel. For instance in the NetB problem, there are 300 such arrays (one per link) of 300 entries each (one per SRG). For comparison, the deterministic approach needs an array for each reserved channel, where each entry corresponds to an SRG and indicates whether the SRG is protected or not by the reserved channel. In the solution of the NetB problems, 213052 of the channels are reserved for protection; thus 213052 arrays of 300 entries would be required in the deterministic method.

Network	Deterministic	Probabilistic	Ratio
NetA	156	23	6.78:1
NetB	9885	501	19.7:1

TABLE I  
TIME TO COMPLETE IN SECONDS.

Network	Deterministic	Probabilistic	Ratio
NetA	61312	62716	100:102
NetB	520771	536343	100:103

TABLE II  
USAGE AS TOTAL NUMBER OF CHANNELS.

For the same set of experiments, Figure 10 plots the distributions of sharing probabilities as computed in step 3c of the probabilistic algorithm during the provisioning of the demand in NetA and NetB. The distributions are similar and show that 70% of the time (77% in NetB) it was possible to predict almost certainly whether there would be a shareable reserved channel (probability 0.0 that a link does not have shareable channel, 48% of the instances for NetA, and 57% of the instances for NetB) or not (probability 1.0, 22% of the instances for NetA, and 20% of the instances for NetB.)

The last experiment of this section, we compare the deterministic and the probabilistic-based algorithms with a distributed algorithm that was first described in[27]. The third



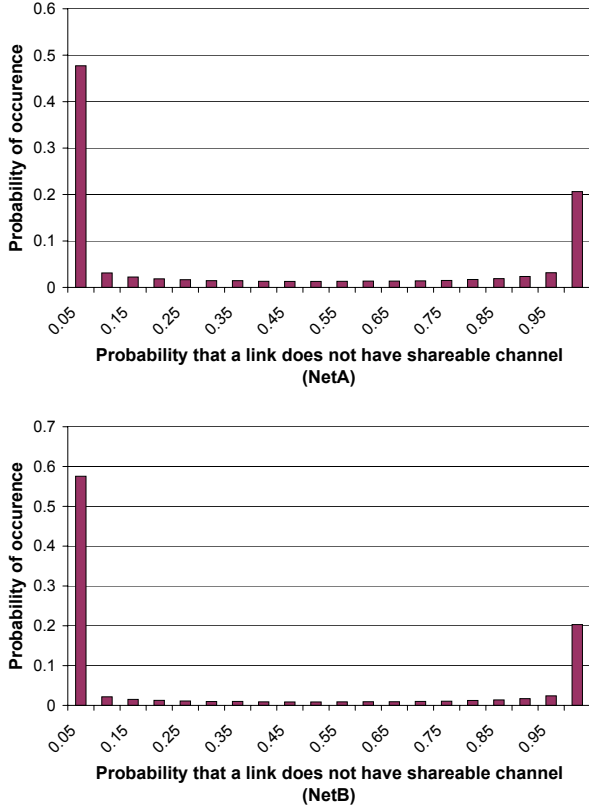


Fig. 10. Distribution of sharing probabilities (NetA, top and NetB, bottom).

algorithm computes a pair of disjoint paths based on the topological information without trying to share existing protection channels, for which we assume that the information is not globally available. As for the probabilistic approach, the assignment of the protection channels is done locally during path-setup signaling when the paths are provisioned. For this comparison we experiment the three algorithms, centralized, probabilistic, and shortest disjoint paths, on three real carrier networks with realistic demands, NetC, NetD, and NetE. NetC is a 17-node, 26-link network, NetD is a 45-node, 77-link network, and NetE is a 50-node, 88-link network. The results, presented in Table III, indicate that the probabilistic approach is comparable to the deterministic approach. In comparison, the third approach, which ignores the possibility of sharing existing protection channels, performs relatively poorly, and requires from 8% to 17% more channels than the other two algorithms.

Network	Deterministic	Probabilistic	Shortest Disjoint Paths
NetC	3884	3908 (100.6%)	4217 (108.6%)
NetD	897	901 (100.5%)	1055 (117.5%)
NetE	1194	1237 (103.6%)	1391 (116.5%)

TABLE III

COMPARISON OF TOTAL NUMBER OF CHANNELS REQUIRED FOR DETERMINISTIC, PROBABILISTIC AND DISTRIBUTED (SHORTEST DISJOINT PATHS) APPROACHES. PERCENTAGES ARE RELATIVE TO DETERMINISTIC APPROACH.

### C. Benefits of locally optimized lightpath provisioning

In the next set of experiments we compare the benefits of local protection channel optimization on two realistic core mesh networks. This procedure is independent of the method used to compute the paths, and we thus use it in combination of the deterministic routing algorithm only. Network A consists of shared-mesh capable optical switches in 46 cities interconnected by 75 links and loaded with 570 lightpaths. Network B consists of 61 switches, 88 links, and 419 lightpaths. For each network, we provision all the demands in sequence using various values of demand churns. The demand churn is the amount of demand expressed in percentage of the total routed demand, which after some time is taken out of service and removed from the network to leave room for subsequent demands. The rate at which demands are removed is determined such that if the churn is  $C$ , then at the end of the simulation the network contains  $(100 - C)\%$  of the total demand, and the remaining  $C\%$  will have been routed then removed before the end of the simulation. We use a first-fit channel assignment during provisioning, and apply a local channel optimization after all the demands are routed. We measure the amount of protection channel required before and after local channel assignment optimization and report the saving in percentage of total backup capacity in 11. Our measurements indicate that as the demand churn increases, the number of protection channels that can be freed becomes substantial.

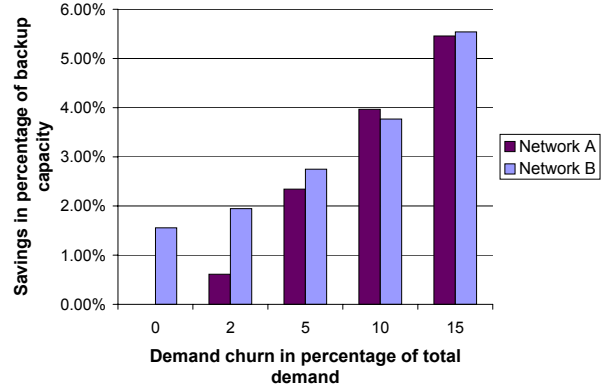


Fig. 11. Savings generated by local optimization in percentage of backup capacity as a function of demand churn.

### D. Summary

Before we conclude, we summarize with a comparison of the different approaches presented in this paper, summarized in Table IV. Four strategies are compared, each corresponding to a column in the table. They are the failure dependent strategy using channel pooling, the centralized failure independent strategy, without and with local re-optimization of the backup channels, and the distributed failure independent strategy with local re-optimization of the backup channels. We evaluate the strategies according to six performance factors, one for each row of the table.

## VII. CONCLUSION

In this paper we describe a probabilistic approach to identify shareable channels in a network when computing shared mesh-restored lightpaths. We show that a summarized information consisting of one fixed length array for every link is sufficient to compute the paths efficiently while maximizing sharing opportunities. In contrast, the deterministic approach needs one such array for every protection channels, and thus does not scale when the demand grows.

Our results demonstrate that the probabilistic approach completes the routing 6 to 20 times faster than the deterministic approach for networks ranging from 100 to 200 nodes. Although the probabilistic approach uses several orders of magnitudes less information than what is necessary for a deterministic approach, their solutions are within 2% to 3% of each other in terms of capacity usage. In fact our experiments indicate that 70% of the time this little information is sufficient to determine with certainty whether there exists a protection channel on a link that could be shared or not.

One possible and natural application of the probabilistic approach is for distributing the routing of shared mesh restored lightpaths to the optical switches. The local database of each switch may contain a summarized information that is necessary to compute the routes using the probabilistic approach. Since this information is small, it can easily be disseminated by link-state protocols, such as OSPF. Using this information each demand's ingress switch can compute a path equivalent to a path computed by a centralized deterministic algorithm with a complete view of the network's state.

This work also proposes a distributed method that rearranges the allocation of shared channels reserved for restoration, with objective to minimize the number of allotted channels. This algorithm can be implemented as an independent background process to supplement either centralized or distributed provisioning algorithms. It is effective to correct sub-optimality inherent to a first fit based provisioning, or seize on improvement opportunities that are brought forth by demand churn.

## VIII. REFERENCES

### REFERENCES

- [1] T. E. Stern and K. Bala, *Multiwavelength Optical Networks: A Layered Approach*. Reading MA: Addison Wesley, 1999.
- [2] B. Doshi *et al.*, "Optical network design and restoration," Bell-labs Technical Journal, Tech. Rep., January–March 1999.
- [3] R. Doverspike *et al.*, "Fast restoration in a mesh network of optical cross-connects," in *OFC'99*, San Diego CA, February 1999.
- [4] J. Doucette and W. Grover, "Comparison of mesh protection and restoration schemes, and the dependency on graph connectivity," in *3rd International Workshop on the Design of Reliable Communication Networks*, Budapest Hungary, October 2001.
- [5] R. Iraschko, M. MacGregor, and W. Grover, "Optimal capacity placement for path restoration in stm or atm mesh-survivable networks," *IEEE/ACM Transactions on Networking*, vol. 6, June 1998.
- [6] E. Bouillet *et al.*, "Stochastic approach to compute shared mesh restored lightpaths in optical network architectures," in *INFOCOM'02*. New York NY: IEEE, July 2002.
- [7] S. Chaudhuri, G. Hjalmtysson, and J. Yates, "Control of lightpaths in an optical network," in *Optical Internetworking Forum*, January 2000.
- [8] R. Ramamurthy *et al.*, "Capacity performance of dynamic provisioning in optical networks," *IEEE JLT*, vol. 19, January 2001.

- [9] J.-F. Labourdette *et al.*, "Routing strategies for capacity-efficient and fast-restorable mesh optical networks," *Photonic Network Communications*, July–December 2002, special issue on "Routing, Protection, and Restoration Strategies and Algorithms for WDM Optical Networks (invited).
- [10] G. Ellinas *et al.*, "Routing and restoration architectures in mesh optical networks," *Optical Networks Magazine*, January–February 2003.
- [11] S. Datta *et al.*, "Efficient channel reservation for backup paths in optical mesh networks," in *GLOBECOM'01*. San Antonio TX: IEEE, November 2001.
- [12] R. Doverspike and J. Yates, "Challenges for mpls in optical network restoration," *IEEE Communication Magazine*, August 1999.
- [13] A. Akyamaç *et al.*, "Ring speed restoration and optical core mesh networks," in *NOC'02*, Darmstadt Germany, June 2002.
- [14] P. Charalambous, "Deployment of an intelligent optical network," in *OFC'03*, Atlanta GA, March 2003.
- [15] G. Li *et al.*, "Efficient distributed path selection for shared restoration connections," in *INFOCOM'02*. New York NY: IEEE, July 2002.
- [16] C. Qiao *et al.*, "Distributed partial information management (dpim) schemes for survivable networks - part i," in *INFOCOM'02*. New York NY: IEEE, July 2002.
- [17] H. Liu *et al.*, "Distributed route computation algorithms and dynamic provisioning in intelligent mesh optical networks," 2003, submitted to ICC.
- [18] H. Liu, "Ospf-te extensions in support of shared mesh restoration," IETF, Tech. Rep., October 2002, draft-liu-gmpls-ospf-restoration-00.txt.
- [19] G. Sahin and M. Azizoglu, "Optical layer survivability: Single service-class case," in *SPIE-Int. Soc. Opt. Eng. Proceedings of Spie - the International Society for Optical Engineering*, vol. 4233, USA, 2000, pp. 267–78.
- [20] —, "Optical layer survivability for single and multiple service classes," in *Journal of High Speed Networks*, vol. 10, no. 2. IOS Press, Netherlands, 2001, pp. 91–108.
- [21] D. Brélaz, "New methods to color the vertices of a graph," *Communications of the ACM*, vol. 22, no. 4, April 1979.
- [22] E. Bouillet *et al.*, "Lightpath re-optimization in mesh optical networks," in *NOC'02*, Darmstadt Germany, June 2002.
- [23] A. Akyamaç *et al.*, "Reliability of single domain versus multi domain optical mesh networks," in *NFOEC'02*, Dallas TX, September 2002.
- [24] Z. Dziong, S. Kasera, and R. Nagarajan, "Efficient capacity sharing in path restoration schemes for meshed optical networks," in *NFOEC'02*, Dallas TX, September 2002.
- [25] R. Ramamurthy, A. Akyamaç, J.-F. Labourdette, and S. Chaudhuri, "Pre-emptive reprovisioning in mesh optical networks," in *OFC'03*, Atlanta GA, March 2003.
- [26] G. Fishman, *Monte Carlo: Concepts, Algorithms, and Applications*. New York NY: Springer-Verlag, November 1995.
- [27] R. Ramamurthy *et al.*, "Comparison of centralized and distributed provisioning of lightpaths in optical networks," in *OFC'01*, Anaheim CA, March 2001.
- [28] D. Papadimitriou, "Analysis of generalized mpls-based recovery mechanisms," IETF, Tech. Rep., November 2002, draft-papadimitriou-ccamp-gmpls-recovery-analysis-03.txt.

## IX. APPENDIX

In reference to question 2 of the problem presented in III-A, we show here how to compute the number  $D$  of "non-blocking" combinations. Let  $r(\max_{1 \leq j \leq N} n_j - 1) = 0$ , and  $r(k) = \left[ C(k, M) \prod_{j=1}^N C(n_j, k) \right] - r(k-1)$ .

Case 1: if  $M = \max_{1 \leq j \leq N} n_j + 1$  then there is at most one bin empty, and the answer is the number of solutions in the remaining  $M - 1$  bins. Thus  $D = C(M - 1, M) \prod_{j=1}^N C(n_j, M - 1) = r(M - 1)$ , as expected.

Case 2: If  $M > \max_{1 \leq j \leq N} n_j + 1$ , then there may be up to  $M - \max_{1 \leq j \leq N} n_j$  empty bins. An incorrect answer would be to treat Case 2 in the same way as we treat Case 1, that is to remove 1 bin out of  $M$ , and compute all possible combinations in the  $M - 1$  remaining bins. In order to understand why this is incorrect, take the case

$M = 2 + \max_{1 \leq j \leq N} n_j$  and assume that we treat it as in Case 1. There can be up to 2 empty bins, and all combinations that have 2 empty bins will be counted twice, once for each of the two bins that is removed. Figure 12 illustrates this. The figure represents 3 bins, and one marble. If we remove one bin at the time and count the number of possible ways to place the marble in one of the two remaining bins, we observe that some combinations are equivalent. For instance, in combinations a) or b), c) or d) and e) or f) the marble respectively occupies the same position, but a different bin was removed. Therefore, although the computation in Case 1 would indicate 6 possible combinations, they are actually 3 of them. The argument presented in this simple example can be easily extended to cover the case of  $n$  marbles in  $M = n + 2$  bins. Observe now that if the number of combinations in  $(M - m)$  bins is known, then it is easy to derive from it the number of combinations in  $(M - m + 1)$  bins. Let  $r(M - m)$  denote the number of combinations in  $(M - m)$  bins, then the number of possible combinations in  $(M - m + 1)$  bins is  $C(M - m + 1, M) \prod_{j=1}^N C(n_j, M - m + 1)$  minus the number of combinations in  $(M - m)$  bins that would otherwise be counted twice, that is  $r(M - m + 1) = \left[ C(M - m + 1, M) \prod_{j=1}^N C(n_j, M - m + 1) \right] - r(M - m)$ . Replacing  $M - m + 1$  by  $k$ , one recognizes the recursion presented in III-A, part 2.

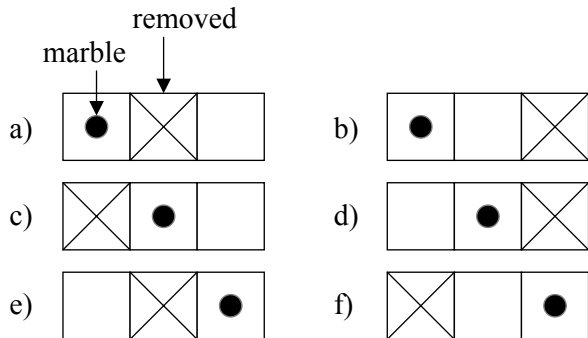


Fig. 12. Combinations of 1 marble into 2 bins (out of 3 bins)

	<i>Failure dependent strategy using channel pooling (centralized or distributed)</i>	<i>Centralized, failure independent strategy with no local re-optimization of protection channels</i>	<i>Centralized, failure independent strategy with local re-optimization of protection channels</i>	<i>Distributed, failure independent strategy, with local re-optimization of protection channels</i>
<i>Time complexity of provisioning algorithm</i>	$O(hm)$ Fast.	$O(ghh')$ Slow, but speed is not an issue for provisioning purposes.		$O(hm)$ Fast.
<i>Amount of information required by provisioning algorithm.</i>	$O(m^2)$ Minimal.	$O(gh'm)$ Large. Grows with traffic.		$O(m^2)$ Minimal. Well Adapted for link advertisement protocols such as OSPF[28], [18].
<i>Protection capacity required.</i>	Most capacity efficient of all presented strategies.	Slightly less capacity efficient than failure independent strategy with local re-optimization of protection channels.	Nearly as capacity efficient as failure dependent strategy.	Slightly less capacity efficient than centralized failure independent strategies.
<i>Restoration Latency</i>	Slow. Requires inter-node communication to agree on a protection channel assignment during failure.	Fast. Each node knows immediately which protection channels to cross-connect, because this information is predetermined and available locally.		
<i>Availability in case of multiple failures</i>	High availability. Protection channels can be assigned to accommodate multiple failures.	Prone to protection channel contentions in case of multiple failures. Although this is mitigated by re-provisioning mechanisms[23].	At least as good as failure dependent strategy (protection channels can be locally re-optimized after first failure)	
<i>Complexity of Implementation</i>	Very complex. Requires communication between adjacent nodes to agree on selected channels and to remove channel selection conflicts.	Complex.	Complex.	Complex.

$g$  = number of lightpaths,  $h$  = average primary path length,  $h'$  = average backup path length,  $m$  = total number of links in network.

TABLE IV  
COMPARISON OF THE PRESENTED ROUTE PROVISIONING ALGORITHM.