J Netw Syst Manage (2012) 20:34–55
DOI 10.1007/s10922-011-9210-4

Multilayer Protection with Availability Guarantees in Optical WDM Networks

Massimo Tornatore · Diego Lucerna · Biswanath Mukherjee · Achille Pattavina

Received: 15 September 2010 / Revised: 30 August 2011 / Accepted: 12 September 2011 /
Published online: 23 September 2011

Abstract Survivability is a key concern in modern network design. This paper investigates the problem of survivable dynamic connection provisioning in general telecom backbone networks, that are mesh structured. We assume differentiated services where connections may have different availability requirements, so they may be provisioned differently with protection (if needed) based on their availability requirements and current network state. The problem of effectively provisioning differentiated-service requests, that has been widely investigated for connections routed at the physical layer, assumes peculiar features if we consider sub-wavelength requests *at the logical layer* that have to be protected (or more generically, whose availability target has to be guaranteed), but also have to be groomed for an efficient use of network resources. An integrated multilayer approach is necessary that considers requirements and grooming of connections at the logical layer as well as their routing and availability at the physical layer. Joint availability-guaranteed routing and traffic grooming may lead to a negative interaction, since the objective of the first problem (guaranteeing a given level of availability to the connections) clashes with the objective of the other problem (minimizing resource consumption). For a multilayer WDM mesh network, we propose new multilayer routing strategies that perform effective availability-guaranteed grooming of sub-wavelength connections. These strategies jointly

M. Tornatore (✉) · D. Lucerna · A. Pattavina
Department of Electronics and Information, Politecnico di Milano, 20121 Milan, Italy
e-mail: tornator@elet.polimi.it

D. Lucerna
e-mail: lucerna@elet.polimi.it

A. Pattavina
e-mail: pattavina@elet.polimi.it

B. Mukherjee
Department of Computer Science, University of California, Davis, CA 95616, USA
e-mail: mukherje@cs.ucdavis.edu

considers connection availability satisfaction and resource optimization and are developed under two different practical hypotheses: *guaranteed* target, i.e., a connection is routed only if its availability target is satisfied, and *best-effort* target, a connection is always routed and, when the availability target cannot be guaranteed, the path with the best possible availability is provisioned. Numerical results are reported and discussed for the two approaches mentioned above. In both cases, the results show high effectiveness of our provisioning strategy.

Keywords Multilayer protection · Shared protection · Availability · Traffic grooming

1 Introduction

avelength-division multiplexing (WDM) technology enables the provisioning of a huge amount of bandwidth in optical transport networks. Today's WDM systems typically transport 10 Gbps (OC-192) over a single wavelength. Systems at 40 Gbps (OC-768) are also being introduced. Given the high bit rates transported over optical WDM networks, efficient survivability mechanisms are needed to avoid a failure of a network element that may cause a large amount of data loss. This may cause a significant loss of revenue for the customer, to be reclaimed from the Service Provider (SP).

On the other hand, as WDM technology matures, there exists a large gap between the capacity of a WDM channel (as mentioned before, up to OC-768) and the bandwidth requirement of a typical connection request (e.g., STS-1, OC-3, OC-12, etc.). Traffic grooming refers to the problem of efficiently multiplexing a set of low-speed connection requests onto high-capacity channels and intelligently switching them at intermediate nodes [1]. Optical switching elements such as Optical Cross Connects (OXCs) and Reconfigurable Optical Add-Drop Multiplexers (R-OADMs), traditionally devoted to fiber or wavelength switching, if properly interfaced with electronic routers (either IP-MPLS routers or SDH, ATM switches) can allow to efficiently (de)multiplex and switch the low-speed connection requests onto high-capacity channels. Multilayer switches also enable the capability of applying protection either at the electronic or at the optical layer, leading to an optimization of backup resources. Past research on traffic grooming has already explored possible solutions for multilayer protection, where the entity to be protected can be either the lightpath, at the optical layer, or the connection, at the electronic layer [2–4].

However, while grooming low-speed connection requests onto high-capacity channels, provisioning the bandwidth required by the connection (while efficiently exploiting the network capacity) should not be the only target to satisfy. In fact, in today's bandwidth markets, for each connection request an availability target (and the associated penalties to be paid if it is not met) is usually specified in Service Level Agreements (SLAs) stipulated between the SP and its customers. Hence, the SP should be able to provide different levels of protection, according to the required availability level, and manage its resources while pursuing the twofold goal to avoid penalties and to increase the profit, i.e., maximizing the number of connections

(or bandwidth) provisioned. Multilayer protection techniques supporting differentiated services have also been proposed to achieve this goal [5–9].

In this work, we present novel availability-guaranteed provisioning frameworks, generically referred to as Availability Guaranteed Protection-at-Connection (AGPAC), to route sub-wavelength connections with dedicated or shared multilayer protection and wisely manage the availability targets. Our grooming approach is availability-aware; this feature allows to solve some peculiar interactions between availability-guaranteed provisioning and traffic grooming, which, if neglected, will compromise the effectiveness of availability-agnostic approaches.

We have developed our approaches under two different practical hypotheses: *guaranteed SLA*, i.e., a connection is routed only if a path that satisfies its availability target can be found; and *best-effort*, i.e., a connection is always routed in the attempt to satisfy its SLA target, but, when the availability target can not be guaranteed, the path with the best possible availability is provisioned. For a typical backbone network, we obtained significant savings on both the availability satisfaction rate and on the resource consumption compared to traditional multilayer protection schemes.

The rest of this paper is organized as follows. Section 2 overviews some previous work on the topic. Section 3 introduces the problem of multilayer protection with availability guarantees. In Sect. 4, new availability-aware multilayer provisioning algorithms are proposed, called AGPAC and AGPAC-, in both the "guaranteed SLA" and "best-effort" cases. Section 5 evaluates by simulation the performance of AGPAC compared to traditional algorithms. Section 6 draws the conclusion of the paper. An "Appendix" is finally dedicated to explain our choice for a specific analytical formulation for availability evaluation in case of protection with shared backup resources.

2 Related Work and Background

The problem of multilayer provisioning for differentiated services has been receiving a lot of attention in recent years.

In [5] two classes of service are considered, called Fully Protected (FP) and Best-Effort Protected (BEP). FP is protected at the WDM layer, while BEP exploits the large unused amount of bandwidth in the IP backbone to restore connectivity, directly at the IP layer, after a failure. Authors in [6] jointly consider grooming and protection in WDM networks; they assume that incoming connections require a working path with minimally-guaranteed availability and a protection path with minimally-guaranteed bandwidth and they minimize the resource consumption. In [7], multilayer methods for survivability are compared in terms of resource utilization and configuration costs. Zhu et al. [8] investigates IP-over-WDM protection with differentiated services and intelligent sharing of backup resources to optimize network utilization. Rai et al. [9] proposes to protect two high QoS classes at the optical layer and a third lower-priority class is served with best-effort quality at the IP layer. An exhaustive survey on resilience differentiation in communication networks can be found in [10]. Authors in [11] classify a different class of services

according to the required availability targets; they propose an analytical formulation to evaluate availability under the various protection techniques and propose a framework for differentiated availability-aware routing at the WDM layer. The concept of availability-aware routing is based on the idea that different links may have different availability profiles and so routing decisions should be also based on the link availability values. A significant amount of research has been devoted in the last few years to the concept of availability-aware routing. We refer the reader to [11] for a more detailed overview on availability-aware routing.

A clear message emerging from the previous body of literature is that, in order to achieve service differentiation, it is crucial to leverage on different protection schemes at the physical layer, but, at the same time, the availability target must be guaranteed *at the logical layer*, where connections are characterized by sub-wavelength bandwidth requirements. It follows that a comprehensive approach to combine traffic grooming and availability guarantee is necessary to address this problem.

Yao and Ramamurthy [12] is one of the first attempts to deal with the problem of protected traffic grooming with availability guarantees. The authors propose three possible options: a baseline approach with no protection, grooming protected at the link level and grooming protected at the lightpath level. Nonetheless, in [12] no notion of availability awareness is applied (i.e., availability is evaluated a-posteriori after the routing phase, not during the path computation, and the information of link availability is not exploited), so the only option to provide higher availability consists in adding a backup path(s) (either at the lightpath or at the link level). In our work, our proposed algorithm also explores solutions at minimal availability (instead of only solutions with minimal resource occupation) and this feature allows us (1) to find more reliable primary paths (even if they may require more resources) that may not require any backup path (so, in conclusion, paths that are more resource efficient) and (2) to avoid unreliable long multi-hop paths that, while maximizing grooming, tend to penalize the overall availability.

In Ref. [13], traffic grooming is investigated in combination with partial protection, i.e., the possibility to provide protection only to a subset of the entire path. This is a powerful tool to provide a differentiated quality of service to connections. Also in this case, the paper does not apply availability awareness. Moreover, the considered metric here is reliability, instead of availability, which leads to a different analysis and results. Finally, in Ref. [14] traffic grooming is associated with shared sub-path protection, a protection technique that allows the primary paths to be divided into a set of sub-paths to be protected separately: no notion of availability is applied here.

In this paper, we argue that, in the specific problem of protected traffic grooming with availability guarantees, the notion of availability-aware routing not only enables superior availability performance, but it also can result in more resource efficient solutions. E.g., sometimes the shortest path is not the most-available path, and the utilization of a simple shortest path algorithm may require the adoption of a backup path, that could be avoided if we calculated the most-available primary path.

In conclusion in this work we investigate the specific interaction of availability-aware routing and traffic grooming, a problem that, to the best of our knowledge,

has never been considered in previous works. While our approaches apply to both wavelength-continuous and wavelength-convertible networks, we hereafter assume, without loss of generality, that the network has full wavelength-conversion capability.

3 Multi-Layer Protection with Differentiated Services

The objectives of this section are (i) to formally state the problem of sub-wavelength connection routing in a multilayer scenario with availability guarantee and (ii) to show that traffic grooming and routing with availability guarantees may interact in a negative manner, leading to inefficient usage of capacity and missed availability targets, thereby motivating the introduction of a comprehensive framework to combine the two aspects.

3.1 Problem Statement

Let us introduce some notations. A network is represented as a weighted, directed graph $G = (V, E, C, \lambda, P)$, where V is the set of nodes, E is the set of unidirectional fibers (referred to as links), $C : E \rightarrow R^+$ is the cost function for each link (where R^+ denotes the set of positive real numbers), $\lambda : E \rightarrow Z^+$ specifies the number of wavelengths on each link (Z^+ where denotes the set of positive integers), and $P : V \rightarrow Z^+$ specifies the number of grooming ports at each node. A dynamic connection request c is represented as a 6-tuple $c = \langle s, d, B, t_a, t_h, SLA \rangle$ that specifies the source node, the destination node, the bandwidth requirement, the arrival time, the holding time, and the availability requirement, respectively.

We now formally state the dynamic connection-provisioning problem as follows: Given the current network state (that includes the network topology as a weighted graph G, existing lightpath connection information, route each connection request with respect to its bandwidth and availability requirement (by using unprotected, shared or dedicated protected routing) while minimizing the incremental cost in terms of the total costs of the working and backup paths under the assumptions that existing connections cannot be disturbed and no information about future arrivals is available at the time of provisioning the current connection.

3.2 Protection-At-Connection (PAC)

As discussed in the previous sections, multilayer protection techniques exist both for multilayer dedicated and shared path protection, that identify the *connection* as the entity to be path-protected at the logical layer (as a difference with the *lightpath* that is the entity usually protected at the optical layer). In Algorithm 1, we briefly describe two existing approaches, the PAC [3] (Protection-at-Connection) and SPAC [2] (Separated Protection-at-Connection) algorithms, that provide dedicated and shared protection at the connection level, respectively. For the definition of Shared Risk Link Group (SRLG), please see [2].

Algorithm 1 Protection At Connection (PAC) and Separate Protection At Connection (SPAC)

Input: $G = (V, E, C, \lambda, P), c = \langle s, d, B, t_h, t_a \rangle$, utilization of network channels and ports.

Output: A working path l_w and a protection path l_b (in SPAC, l_b can share lightpaths with other backup routes, while in PAC, dedicated lightpaths need to be reserved for l_b); NULL if one or both paths are not found.

1. Construct the current-network-state graph G_s based on the present utilization of channels and ports, according to the auxiliary graph in [15]: links associated to existing lightpaths have small costs in order to promote their utilization.
2. Compute a min-cost path l_w on G_s from node s to node d; return NULL if l_w is not found.
3. Compute a path l_b SRLG-disjoint to l_w from node s to node d:

   –PAC: l_b is a minimal-cost route on $G_s \backslash l_w$ with the same link-cost assignment utilized at Step 1.

   –SPAC: l_b is a minimal-cost route on $G_s \backslash l_w$ with a link-cost assignment which promotes the sharing potential between backup paths [3].

   return NULL if l_b is not found.
4. Return the paths l_w and l_b.

We consider a dynamic scenario in which connections are offered to the network at random instants of time, then stay in the network for a given amount of time and then they leave, releasing the resources dedicated to them. The network is multilayer, consisting of an optical layer and an electronic layer where electronic tributary signals have to be multiplexed and routed over a dynamic overlay logical topology composed by lightpaths. Our objective now is to assign, to each connection c, a level of protection that is sufficient to guarantee its SLA, instead of providing the same level of protection indiscriminately to all the connections, as it happens in PAC or SPAC.

Let us consider a traditional traffic grooming availability-agnostic approach, which is designed to aggregate protected traffic, with the objective to minimize the resource utilization, e.g., as in [15]. A traditional traffic grooming approach will primarily search for an admissible routing that exploits the lightpaths that have enough residual capacity to support the incoming connection; new lightpaths, that require additional transceivers and wavelengths, are created only if the residual logical capacity cannot accommodate the incoming request.

The attempt to stimulate traffic aggregation as much as possible may lead to long primary paths, characterized by scarce availability. This behavior is intrinsically related to the cost assignment aimed at maximizing the utilization of residual capacity in existing lightpaths, e.g., as in Step 1 of Algorithm 1. As a consequence, a scarcely-available primary path may lead to the necessity of a backup path to catch the SLA target. Instead, note that a wiser (i.e., more available) choice of the primary path may have avoided the need for a backup path and the same connection could have been served (more efficiently) in an unprotected manner. The resource saving achieved by choosing a long multi-hop primary path is then wasted due to necessity of a (maybe unnecessary) backup path. In conclusion, a strategy aimed at minimizing the network resource utilization, if applied under an availability-guarantee constraint, may not lead to a resource efficient solution.

Alternatively, one can resort to availability-aware routing approaches, that are able to give priority to paths with higher availability. Unfortunately, it has been demonstrated that routing connections with the mere objective of maximizing availability leads to an unacceptable waste of resources (see [16]). So, a solution able to manage the delicate interaction between grooming and availability-guaranteed routing is needed.

Let us refer to an example. Figure 1 shows a 6-node network, carrying 3 existing lightpaths. Link 3–4 has at least a free wavelength to provision a new lightpath over it. Let us assume that all the links have an availability $A_l = 0.9999$ and a new connection with an availability target $A_{SLA} = 0.9999$ requested between nodes 3 and 4. Assume lightpaths have a residual capacity sufficient to support the new incoming request. These are some possible routing choices:

(a) *Exploiting existing capacity*. The primary path is routed over the existing lightpaths. This path crosses 5 links and has an availability equal to 0.9995, less than the target SLA. So, we will need a protection path that will require the provisioning of a new lightpath over the link 3–4;
(b) *Maximizing availability*. The primary path is routed directly over link 3–4, by setting up a new lighpath on that link. We initially require an additional investment (transceivers, wavelengths), but then we do not need a protection path (since this primary path has availability equal to 0.9999 that satisfies the SLA).

This previous example shows that traditional routing policies aimed at minimizing resource utilization may not minimize the resource utilization if connections have to be routed with availability guarantees. Since various levels of availability have to be considered, our scheme will include protection, to be used whenever needed to catch the availability target, but without using it if not needed, since an abuse in protection resource may lead to an inefficient usage of network resources. To solve this problem, we will combine existing approaches and explore various paths, with the aim of choosing the best trade-off in terms of availability and resource utilization. The problem can be summarized as follows:

- *Objective Function*: maximize the number of connections in the network (equivalently, minimizing the resource utilization)

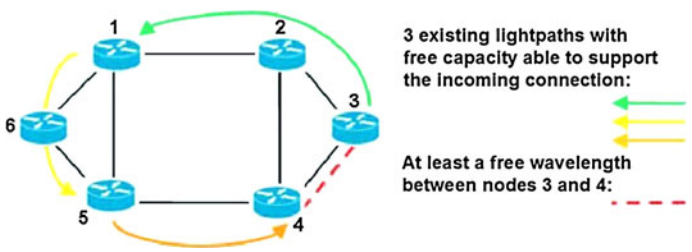


Fig. 1 Example of interaction between aggregation of traffic and availability maximization

- *Constraint*: for each connection, provision a connection with an availability greater or equal to the SLA stipulated with the customer
- *Issues*: Traffic grooming with protection (such as PAC or SPAC [2, 3]) leads to efficient resource utilization, but introduces long paths with scarce availability
- *Possible Solution*: *exploring* more routing alternatives, looking for still resource-efficient, but also SLA-complaint candidates such as highly-available paths primary paths (that may not require protection)

3.3 Availability-Aware Routing

The problem of discovering paths with high availability can be solved using availability as part of the link metric, so that paths can be evaluated not only in terms of their cost (e.g., minimizing the length or number of hops), but also considering the availability of the path itself (availability-aware routing).

If we have the availability A_e for a link e, then a possible approach is to assign to link e a cost $C_e = -Log(A_e)$. In this case, the application of the shortest-path algorithm will return the most available path [11].

Additionally, in case of a multilayer network, we have to assign an availability also to the lightpath-links. If A_e is the availability of link e, the availability of a lightpath that spans across more than one link is given by the following equation (under the hypothesis that availability is dominated by link-cut failures [11]):

$$A_{lp} = \prod_{e \in E} A_e \tag{1}$$

Given A_{lp}, an availability-aware lightpath-link cost can be defined as $C_{lp} = -Log(A_{lp})$. As for the calculation of the availability of shared- and dedicated-path-protected connections, we have used analytical formulations as in [17]. In the "Appendix" we briefly survey few relevant approaches for availability calculation in shared path protection and compare their performance, in order to justify this choice.

4 Availability-Guaranteed Protection at Connection (AGPAC)

In this section, we introduce two approaches for an availability-guaranteed provisioning of sub-wavelength connections: a complete version, referred to as AGPAC, and a simplified version of the same algorithm, called AGPAC-.

In our model, the customer can choose among different classes of services, associated with different availability targets. Connections can be unprotected, shared-path protected or dedicated-path protected. Since we are in a multilayer environment, protection is enabled at connection-level. We use the SPAC and PAC strategies as basic routines for shared and dedicated protection at connection level. For this reason, our approach is called *Availability-Guaranteed Protection-at-Connection* (AGPAC). As a first step, we identify a set of possible routing computation strategies, that return different paths with different protection degree

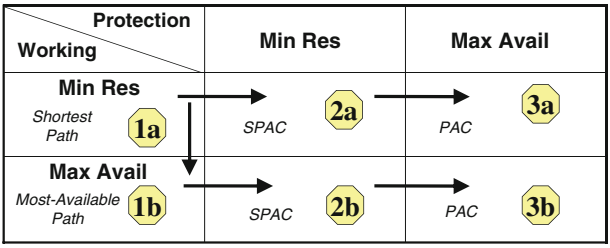


Fig. 2 Routing solutions to be explored

and resource efficiency. Then we will define a criterion to compare the various solutions and choose the best solution. In Fig. 2, we show the six routing options:

1a—W_{MinRes}: A minimal-cost primary path (W) is computed, promoting traffic aggregation (comparable to MinLP policy in [18]).
1b—$W_{MaxAvail}$: the most-reliable primary path is computed, decreasing the number of links crossed (comparable to MinTHP policy in [18]).
2a—W_{MinRes} - B_{MinRes}: If the primary path from 1*a* does not satisfy SLA ($A_{1a} < A_{SLA}$), a protection path (B) is computed with SPAC.
3a—W_{MinRes} - $B_{MaxAvail}$: If the SPAC-protected path from 2*a* does not satisfy SLA ($A_{2a} < A_{SLA}$), a protection path (B) is computed with PAC.
2b—$W_{MaxAvail}$ - B_{MinRes}: If the primary path from 1*b* does not satisfy SLA ($A_{1b} < A_{SLA}$), a protection path (B) is computed with SPAC.
3b—$W_{MaxAvail}$ - $B_{MaxAvail}$: If the SPAC-protected path from 2*b* does not satisfy SLA ($A_{2b} < A_{SLA}$), a protection path (B) is computed with PAC.

We can define 1*a*, 2*a*, 3*a* as *minimal-resource-consumption* solutions and 1*b*, 2*b*, 3*b* as *maximal-availability* solutions. In terms of cost, solution *1a* is the most efficient, while solution *3b* is evidently the least efficient. As for intermediate solutions, they can be compared two-by-two (see Fig. 3), *2a* with *1b*, *3a* with *1b*,

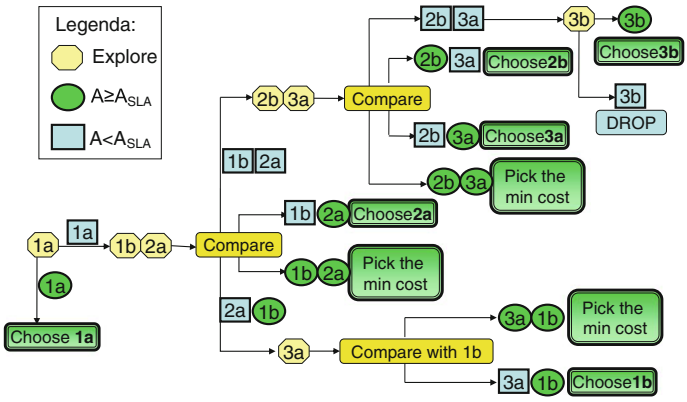


Fig. 3 Scheme for AGPAC

and *3a* with *2b*, avoiding a complete comparison among all the solutions. Our scheme works step-by-step: it computes a possible solution, evaluates its cost and admissability, compares it with a subset of other solutions and, only if necessary, it explores other solutions.

In Fig. 3 the step-by-step flow chart of our algorithm is represented. Note that a routing for a connection is admissible if (i) it satisfies the targeted SLA, (ii) if it does not compromise the availability of connections that have been already routed, by increasing the sharing degree of their shared backup capacity too much, and (iii) there is enough capacity in the network to support the new path. The algorithm (see Algorithm 2) starts from solution *1a*: if *1a* is admissible, then it is also the best choice and no protection is needed. Otherwise, we proceed towards solutions *1b* and *2a*, i.e., we either look for a more reliable primary path or we add a shared backup path to the path in *1a*, using SPAC. According to the admissibility of solutions *1b* and *2a*, we choose how to proceed. If both are admissible, we pick the solution with minimal costs; if solution *1b* is admissible but solution *2a* is not, *1b* is then compared to solution *3a*, that associates with path *1a* a dedicated backup path by means of PAC policy (if *3a* is admissible, we pick the one with minimal costs, otherwise we choose *1b*). If solution *2a* is admissible but solution *1b* is not, we choose *2a*, and if both are not admissible, we proceed toward solutions *2b* and *3a*. Again the evaluation of the best path is based on the admissibility of the two alternatives. If both are admissible, we pick the solution with minimal costs; if only one out of the two candidates is admissible, then there is no other choice than to pick that solution; if both are not admissible, we proceed toward solution *3b*, the last possible attempt, that provides the highest availability. If *3b* is also not admissible, the connection is dropped. This algorithm is also reported in Algorithm 2, for sake of clarity.

Algorithm 2 AGPAC (Guaranteed SLA)

Input: $G = (V, E, C, \lambda, P), c = \langle s, d, B, t_a, t_h, SLA \rangle$, utilization of network channels and ports

Output: A path l_w (with or without protection path l_b) with minimal cost, guaranteeing the required SLA; NULL if the path is not found.

1. Compute the minimal cost l_w (option 1a).
   a. if $A_{l\_w} \geq SLA$ return l_w. Else:
2. Explore options 1*b* and 2*a*:
   a. if both are admissible, choose the one with minimal cost, and return l_w (if 1*b*) or (l_w, l_b) (if 2*a*);
   b. if 1*b* is admissible and 2*a* is not, then compare 1*b* with 3*a*, choose the one with minimal cost and return l_w (if 1*b*) or (l_w, l_b) (if 3*a*);
   c. if 2*a* is admissible and 1*b* is not, then return (l_w, l_b) (of option 2*a*);
3. If 1*b* and 2*a* are not admissible, explore options 2*b* and 3*a*.
   a. if both are admissible, choose the one with minimal cost and return (l_w, l_b);
   b. if 3*a* is admissible and 2*b* is not, then return (l_w, l_b) (of option 3*a*);
   c. if 2*b* is admissible and 3*a* is not, then return (l_w, l_b) (of option 2*b*);
4. If 2*b* and 3*a* are not admissible, explore options 3*b*.
   a. If 3b is admissible, return (l_w, l_b) (of option 3b)

else return NULL.

4.1 AGPAC-

AGPAC explores a wide spectrum of possible solutions, and this may lead to excessive complexity. We propose here a simplified version of AGPAC, referred to as AGPAC-, that exploits only solutions of type *a* (namely, *1a*, *2a* and *3a*). In Algorithm 3, we formally describe AGPAC-; in Fig. 4 we draw a flow chart of the simplified algorithm.

4.2 The "Best-Effort" Version for AGPAC and AGPAC-

The previous version of AGPAC and AGPAC- are referred to as "Guaranteed SLA", because connections are not routed (i.e., they are dropped) if no path among the six options that are explored satisfies the SLA target. The rationale behind "Guaranteed SLA" approaches is to avoid that the SP excessively increases excessively the risk of paying the penalty for dissatisfying the customer requiring the connection.[1]

So, the choice of dropping or accepting a connection strictly depends on the penalty that the SP has to pay when the SLA is violated. In other words if we define

$$R = \frac{penalty}{service\_cost}$$

as the ratio between the penalty and the service cost, it is easy to see that for $R \rightarrow 1$ it is desirable to use a guaranteed SLA cost (i.e., accept a connection only if its availability satisfies the SLA target) especially at high loads, when the provisioning of a connection with a high chance of paying a penalty may hinder the provisioning of a connection that satisfies the SLA. On the contrary for $R \rightarrow 0$ it may result in a higher revenue a strategy that simply accepts all connections (best effort), since the penalty to be paid is negligible or, at least, very low.

These are the reasons why we now briefly introduce algorithms for the so-called *best effort* case, i.e. the case in which a connection is always routed in the attempt to satisfy its SLA target. However, even when the availability target cannot be guaranteed, the path with the best possible availability is still provisioned.

The modification to evolve AGPAC and AGPAC- from "guaranteed SLA" to "best effort" concerns the way the last possible option (solution 3b, the one with highest availability) is managed. When this option is still not enough to catch the availability target, the connection is still routed (using the option 3b). In Fig. 5, for

[1] Unfortunately, guaranteeing that a path has a path availability larger than target availability does not ensure that the SLA will not be violated. In literature, various works have dealt with the concept of risk of an SLA violation [19, 20], also called interval availability [21], which provides a metric to evaluate how likely a connection with a given availability will violate its SLA target over a prefixed time period. The theoretical formulations to represent the risk of SLA violation focus mainly on unprotected and/or dedicated protected services and do not consider traffic grooming due to the computational complexity of this problem; furthermore only recently consideration for service availability has been introduced directly in routing problems, and only for unprotected routing [22]. For all of these reasons, in this paper we focus on the availability metric, leaving as a possible extension of this work the investigation of how to include the risk of an SLA violation in our problem.

Algorithm 3 AGPAC- (Guaranteed SLA)

Input: $G = (V, E, C, \lambda, P), c = \langle s, d, B, t_a, t_h, SLA \rangle$, utilization of network channels and ports.

Output: A path l_w (with or without protection path l_b) with minimal cost, guaranteeing the required SLA; NULL if the path is not found.

1. Compute the minimal cost l_w (option 1a).
   a. if $A_{l\_w} \geq SLA$ return l_w; else:
2. Explore option 2*a*:
   a. if 2*a* is admissible, then return (l_w, l_b); else:
3. Explore option 3*a*.
   a. if 3*a* is admissible, then return (l_w, l_b);
   else return NULL.

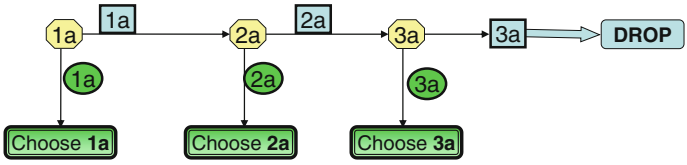
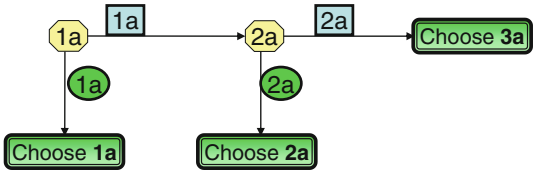


Fig. 4 Scheme for AGPAC-

Fig. 5 Scheme for "best effort" AGPAC-



sake of brevity, we report the modified flow chart of AGPAC- for the best effort case, as opposed to the chart of a guaranteed-SLA AGPAC- in Fig. 4.

4.3 Complexity Analysis

AGPAC and AGPAC- exploit as a basic procedure the principle of the heuristic SPAC in [3]. The computational complexity of SPAC is $O(E \cdot W + K \cdot N^3)$, where N is the number of network nodes, E is the number of links, W is the number of wavelengths, and K is the number of distinct paths. Specifically, the three main steps of SPAC (construction of the auxiliary graph, computation of K-shortest paths, computation of backup paths) have complexity $O(E \cdot W)$, since there can be as many as $E \cdot W$ lightpaths, $O(K \cdot N^3)$ and $O(K \cdot N^2)$, respectively. In the worst case scenario, AGPAC calls a shortest path algorithm, followed by the SPAC algorithm, followed by the PAC algorithm (PAC is the dedicated version of SPAC, i.e., it has a lower complexity), for both the objectives of minimization of availability and minimization of resource occupation. So, the AGPAC complexity is twice the

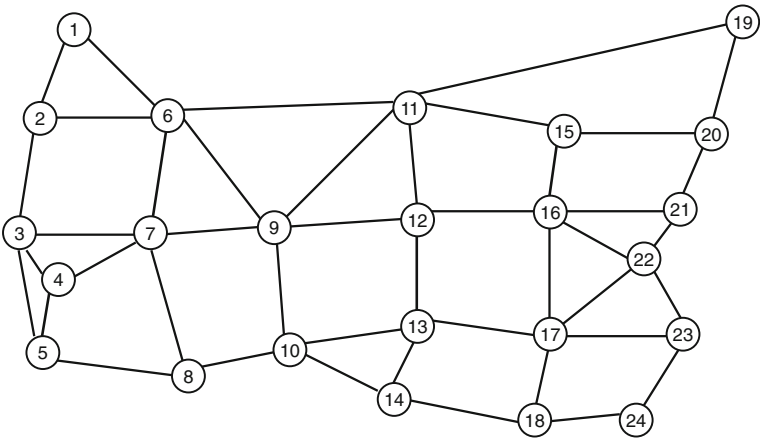


Fig. 6 A carrier's US nationwide backbone network topology

complexity of SPAC. AGPAC - follows the some procedure, but it only considers a minimization of resource occupation, so its complexity is exactly the same as SPAC.

5 Illustrative Numerical Examples

We show illustrative results for a typical US network topology (Fig. 6). Fiber links are equipped with 16 wavelengths. Connection arrivals are Poisson and uniformly distributed among all node pairs. Holding times follow a negative exponential distribution with the average normalized to unity. Once the node-pair is chosen, five types of connection requests can be chosen, OC-1 (STS-1 at 55 Mbit/s), OC-3 (STM-1 at 155 Mbit/s), OC-12 (STM-4 at 620 Mbit/s), OC-48 (STM-16 at 2.5 Gbit/s) or OC-192 (STM-64 at 10 Gbit/s, i.e., the full wavelength). Proportions among the bandwidth-request types are 300 : 20 : 6 : 4 : 1 (according to measurements on a realistic backbone network [15]). Availability requirements of the requests are uniformly distributed over the three classes {0.999, 0.9999, 0, 99999}. Link availability is randomly distributed among 2-9s, 3-9s, 4-9s, 5-9s (e.g., 3-9s means 0.999, read as three nines), in order to obtain an average link availability of 0.9999 and 0.9995.

We compare our algorithms, AGPAC and AGPAC-, to PAC and SPAC [2, 3]. We employ three metrics to evaluate our new approaches and compare them with traditional schemes: *Bandwidth-Blocking Ratio* (BBR), i.e., the percentage of bandwidth that is blocked over the total required bandwidth, *Availability Satisfaction Rate* (ASR), i.e., the percentage of connections that respect the targeted SLA and *Resource Overbuild* (RO) that indicates how many extra resources we need, on

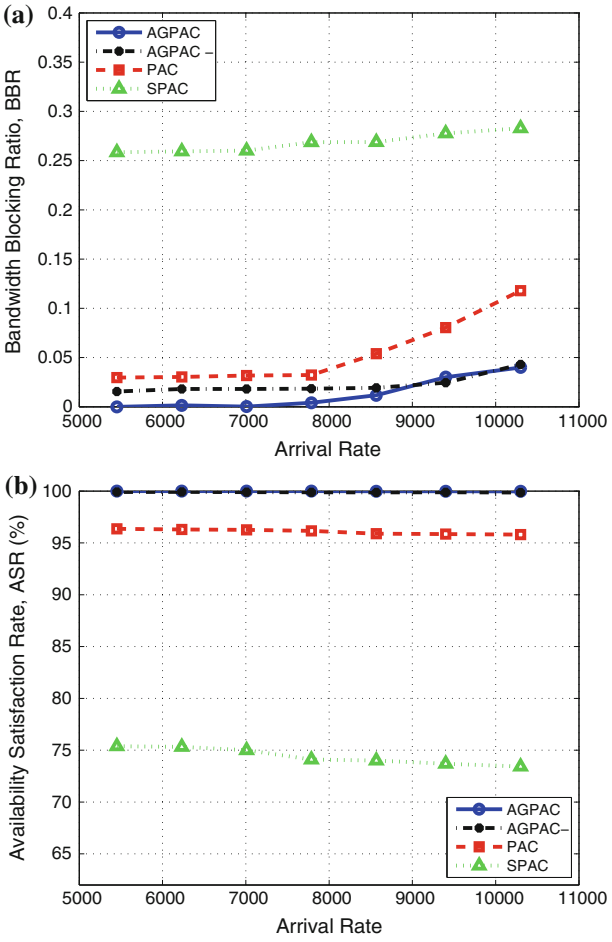


Fig. 7 Comparison of **a** BBR and **b** ASR among AGPAC, AGPAC-, PAC and SPAC (average link availability equal to 0.9995)

average, to protect connections [11]. The three above-mentioned metrics are mainly analyzed for increasing values of the *arrival rate*, expressed in Erlang.[2]

5.1 Results for Guaranteed SLA

In Fig. 7, AGPAC and AGPAC- are compared to PAC and SPAC when the average link availability is set to 0.9995. In Fig. 7a, we show the BBR comparison. AGPAC, being able to provide a customized level of protection and availability to the different connections and outperforms both SPAC and PAC in terms of BBR.

[2] Note that, since offered connections are characterized by different granularituies of the required capacity, the arrival rate is expressed in terms of OC-1 connections (e.g., each time an OC-48 is offered, that adds a term 48 to the arrival rate).

This result is expected for SPAC, since the high sharing degree of SPAC does not allow to satisfy the availability requirements of the connections, especially those requiring 5-9s target (note that the percentage of bandwidth rejected, 25% is just slightly less than the percentage of connection served requiring 5-9s, which is 33%). It is less expected that the PAC strategy, that provides dedicated protection to all the connections, is outperformed by AGPAC. This happens since the connections with very high availability requirements (5-9s) sometimes (typically when they are required among geographically distant nodes that are connected by less available paths) require availability-aware schemes to choose the most available routing in the networks. Note also that AGPAC and AGPAC- have a very similar performance, even if AGPAC has a slightly lower BBR for low loads when the possibility to use availability-aware options (the b's in Algorithm 2) is more useful. For high loads, performances converge when the network has not enough residual capacity to support the capacity-consuming availability-aware routing.

The ASR in Fig. 7b confirms the previous findings. AGPAC and AGPAC- in this case have exactly the same results, and it is clearly shown that indiscriminate protection does not help in a scenario with differentiated availability requirements, while a meaningful choice has to be carried to associate the correct level of protection according to the availability requirements.

Figure 8 reports the RO for the four approaches. SPAC always requires the minimal amount of backup capacity (except for very light loads, when chances for sharing are very low), because all the connections share backup capacity while AGPAC applies dedicated reservation of backup capacity for connections with high availability requirements. Nonetheless, the additional amount of capacity required by AGPAC is negligible, considering the large gains that have been achieved in terms of BBR and ASR. PAC, as a dedicated protection strategy, confirms to be inefficient in backup-resource utilization. Results do not vary sensibly for increasing loads.

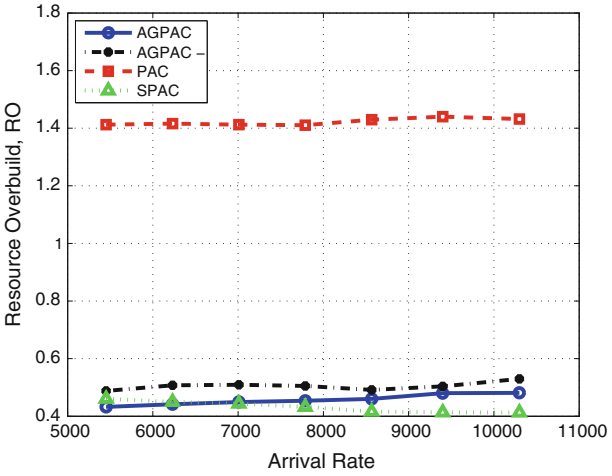


Fig. 8 RO comparison among AGPAC, AGPAC-, PAC and SPAC

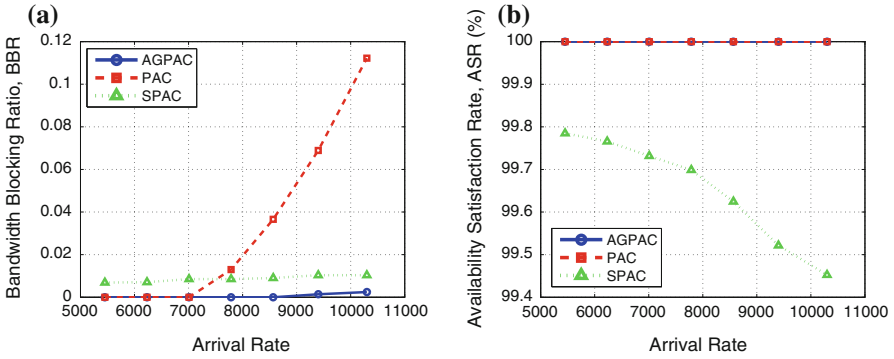


Fig. 9 Comparison of BBR among AGPAC, PAC e SPAC (average link availability equal to 0.9999)

We now consider in Fig. 9 the case in which the network is generally more available, and the average link availability is equal to 0.9999. AGPAC- is not considered here, since its results are again very close to AGPAC. Figure 9 shows the BBR for AGPAC, PAC and SPAC. Since the network is now much more available on average, AGPAC and SPAC achieve much better results: AGPAC has almost no blocking, while SPAC decreases its BBR from 25% to less than 1%. On the contrary, PAC's performance is almost unchanged, since dedicated protection is also allowed in the previous case to avoid blocking due to availability, and the main cause of blocking is again lack of resources. Figure 9b reports the ASR for the three approaches. As for ASR for the three approaches, PAC and AGPAC succeed in satisfying the SLA target of all routed connections. As for SPAC, the ASR is still very high (since the network is now very available), but does not achieve the 100% target. Results for RO are not reported, since they remain in agreement to Fig. 8.

5.2 Results for Best Effort Case

In Fig. 10, the BBR of AGPAC and AGPAC- is compared to PAC and SPAC when the average link availability is set to 0.9995. Results differ significantly compared to the guaranteed SLA case. Before providing a formal discussion of this result, it is worth mentioning that BBR assumes a different meaning here. In fact, in the previous subsection, for the guaranteed SLA, bandwidth blocking resulted from the joint effect of capacity shortage under high loads and SLA dissatisfaction. Now, in the best effort case, blocking arises only due to limited capacity reasons, but the fact that all the connections are now accepted increases the blocking due to capacity.

Hence, considering Fig. 10, we can conclude that: (i) SPAC now has the lowest BBR among all the approaches (which stays very close to zero for all the traffic load ranges in our experiment); (ii) the BBR experienced by PAC, compared to the previous case in Fig. 10, is almost the same (note that the y-axis now has a different scale), since PAC always provides a very high level of availability; (iii) AGPAC returns an intermediate performance between SPAC and PAC. A less intuitive conclusion regards the performance of AGPAC- compared to AGPAC:

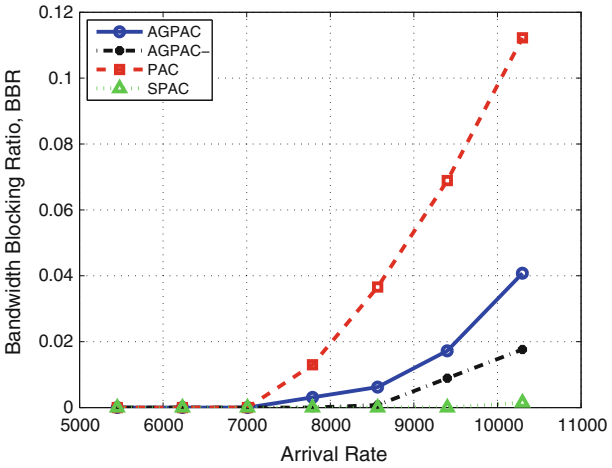


Fig. 10 Comparison of BBR among AGPAC, AGPAC-, PAC and SPAC (average link availability equal to 0.9995, best-effort case)

AGPAC- now outperforms AGPAC, because AGPAC- does not resort to any capacity-demanding availability-aware approach, and therefore it is able to save more capacity than AGPAC.

Results on the ASR are less interesting, since they basically repeat the same trends as in the guaranteed case. Even when more connections are accepted, the average availability of the connection does not significantly change (only a small decrease, due to higher traffic, can still be appreciated). Analogous considerations hold for RO, where we can again notice a general equivalence of the results compared to the guaranteed SLA case, with only a slight increase of RO due to higher accepted traffic. So, for sake of conciseness we do not report graphics for ASR and RO.

Finally, in Fig. 11 we show the percentage of connections that uses unprotected, dedicated-path-protected or shared-path-protected routing in AGPAC. The percentage of connections routed without protection does not change for an increasing arrival rate, since the connections that are routed in an unprotected manner are only those connections with 3-9s availability target and an hop distance less or equal than two hops. Shared-path-protected connections decrease for an increasing arrival rate,

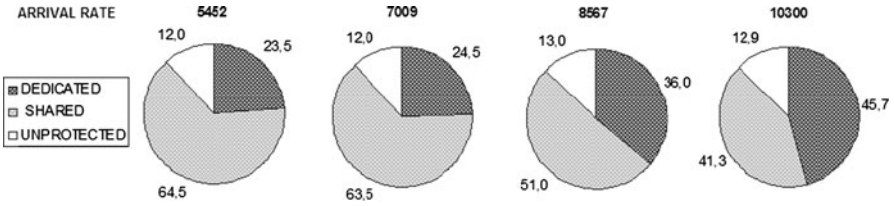


Fig. 11 Percentage of utilization of the various protection schemes in AGPAC ($A_{link} = 0.9995$) for best effort case

because, when the network is more loaded, the sharing degree increases and it is not possible to guarantee the SLA for high sharing.

Results for when the average link availability set to 0.9995 are also not reported, because they mainly confirm the conclusion already discussed for the guaranteed SLA case.

6 Conclusion

We proposed and investigated a novel dynamic provisioning strategy with differentiated service in a multilayer mesh network, e.g., an MPLS-over-WDM mesh network. The provisioning strategy captured the essence of both availability guarantee and effective resource sharing. By extensive simulative experiments, we demonstrated that our proposed provisioning strategy, called Availability Guaranteed Protection-at-Connection (AGPAC), can guarantee a 100% Availability Satisfaction Rate (ASR) by employing a little extra resources, compared to general protection-at-connection schemes such as, PAC or SPAC, that are unaware of connection-availability constraints. AGPAC also achieved resource-usage optimization due to differentiated services. The performance of our provisioning strategy indicated that both resource-utilization optimization and connection-availability guarantee can be achieved by considering them jointly in our algorithm.

Acknowledgments A preliminary version of this work has been presented in the 13th Conference on Optical Network Design and Modeling 2009 (ONDM09). The research leading to these results has received funding from the European Communitys Seventh Framework Programme FP7/2007-2013 under grant agreement no. 247674 (STRONGEST project). The authors would like to also thank Stefano Cavallaro and Roberto Lucadello for the active collaboration in the software and algorithm developments.

Appendix: On the Availability Evaluation for Shared Protected Connections

There is a relevant body of literature developed in these last years, regarding analytical models for availability calculation for protected connections, based on the classical reliability theory [23]. In the case of dedicated protection, the evaluation of the availability of the protected connection can be easily obtained with the following expression

$$A = A_w + A_b(1 - A_w)$$

where A_w and A_b represent the availability of the working and the backup path, respectively.

On the contrary, various methodologies have been proposed to solve the problem of availability evaluation with different degree of complexity and precision. In fact, the availability calculation in the case of shared protection has been proven to be NP-complete, and we have to resort to approximations to find practical models for the availability calculation.

The key problem to be solved is how to account for the connections that are "conflicting" with the backup path of the connection under analysis (say b), i.e.

those connections that in case of failure (single or multiple) compete with b for the same shared backup capacity. Intuitively, the count of the conflicting connections becomes more and more complex for an increasing number of concurrent failures, and the corresponding number of possible conflict scenarios to be considered increases exponentially. In conclusion, it is not computationally feasible to account for all the possible scenarios.

A first approximated formulation appeared in [24], where the authors consider as "conflicting" connections all the connections that are sharing with b some capacity along their backup path (the set of these connections is referred to as *Protection Group*). In this case, the formula for the availability A_i of the generic protected connection i is given by

$$A_i \approx A_{w_i} + (1 - A_{w_i})A_{b_i} \prod_{\forall h \in PG_i (h \neq i)} A_{w_h} \tag{2}$$

where A_{w_h} is the availability of the working path of a generic connection h belonging to the protection group $\mathbf{PG_i}$ of connection i.

An extension of this model can be found in [25], where a more accurate count of the conflicting connections is proposed. The study in [25] recognizes that only some of the links along the working paths w_h in the PG_i do actually induce a conflict in case of failures, and that only those links have to be accounted for. The formula 2 is evolved in

$$A_i = A_{w_i} + Ab_i Ac_i \cdot A_{w_i} Ab_i Ac_i$$

where Ac_i is the product of the availability of all the links conflicting according to the new definition. The authors propose a data structure, called conflict vector, to evaluate the set of conflicting links c_i given a connection i. Please refer to [25] for further details.

Also in [11] and [28], the authors propose similar approaches. They use the concept of *grabbing* probability, i.e. the probability that the connection "grabs" the shared capacity in case of contention. We omit further details for these two works since they are based on a similar formulations to the one in [25], and are leading to a similar performance.

A completely different approach is proposed in [26]. The authors investigate a two-step approach: a first step to be executed once, during the network planning phase and a second step to be applied for each connection whose availability is currently evaluated. The calculation is based on a Markov chain model. An extension of this work is reported in [27], where an investigation of the upper and lower bounds of the previous analytical approach is conducted.

Finally, an extremely precise methodology is proposed in [17]. This work is the first one to apply the concept of conflicting links to each link of the working path. In other words, for each link of the working path the authors evaluate which other links would compete (or "conflict") with it in case of double failure, while previously, as in [25] and [26] conflicting links where considered as conflicting with any link of the working path. Now a set conflicting links is associated to each link of the working path and not generically to the working path. We will see that this novelty avoids a significant overestimation of connection unavailability.

For this method, each of the competing connections has a probability 0.5 to acquire the shared capacity. Formula 3 describes the calculation in [17]. For each link i on the working path, a "modified" unavailability is calculated. Consider that $Cond_{CT}$ indicates the set of links conflicting with i, i.e., those links that in case of another concurrent failure together with i would cause the loss, for the connection in exam, of a traffic quantity equal to NT, over a total amount of traffic $d_{s,t}$, then:

$$U_i^* = U_i \cdot \left[\sum_{j \notin W \cap i \in B} U_i + 0.5 \cdot \sum_{j \in Cond_{CT}} \left(U_j \cdot \frac{NT_{i,CT}^{i,j}}{d_{s,t}} \right) \right] \tag{3}$$

returns U_i^* as the modified link availability that captures the likelyhood of a conflict (note that U_i is the actual link availability). Once the U_i^* has been calculated, under the approximation of rare-events, the entire path unavailability can be expressed as:

$$U_{path} = \sum_{i=1}^{N} U_i^*$$

where N is the number of links in the working path. Please refer to [17] for further details.

It is worth mentioning that for all these works, the main approximation is that no more than two concurrent failures are explicitly taken into account (as in [25]).

In Fig. 12, we compare some of the approaches described above over a simple network topology: a single protected connection consists of a working path with two links and and a protection path with three links. The three backup links can be shared by no other connections up to ten other connections (x-axis in the Fig. 12). All the links and sharing connections have an availability equal to 0.9. Monte Carlo simulation [24] provides a benchmark for all the approaches. The approach in [17]

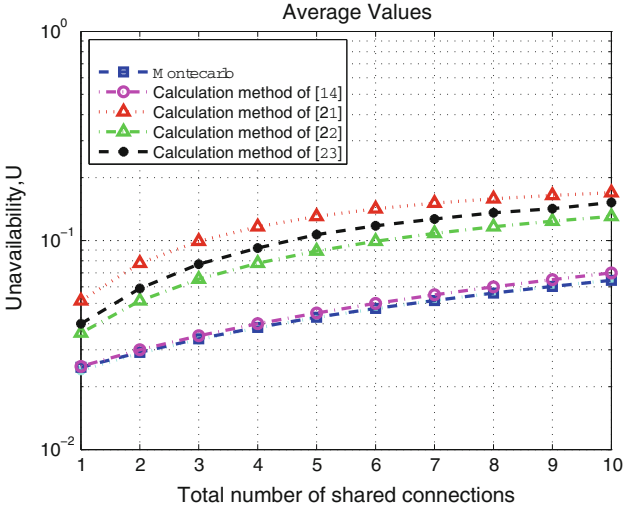


Fig. 12 Comparison of the availability estimation provided by the various availability models provided in this section (U_{link} = 0.1)

returns the estimation that is closest to the actual value of availability. So, in our simulation in Sect. 5, we decide to use the approach in [17].

References

1. Dutta, R., Rouskas, G.: Traffic grooming in WDM mesh networks: past and future. IEEE Netw. Mag. **43**(6), 46–56 (2002)
2. Ou, C., Zhu, K., Zang, H., Zhang, J., Zhu, H., Sahasrabuddhe, L.H., Mukherjee, B.: Traffic grooming for survivable WDM networks: dedicated protection. IEEE/OSA J. Opt. Netw. **3**, 50–74 (2004)
3. Ou, C., Zhu, K., Zang, H., Sahasrabuddhe, L.H., Mukherjee, B.: Traffic grooming for survivable WDM networks: shared protection. IEEE J. Sel. Areas Commun. **21**(9), 1367–1383 (2003)
4. Yao, W., Ramamurthy, B.: Survivable traffic grooming with path protection at the connection level in WDM mesh networks. IEEE/OSA J. Lightwave Technol. **23**(10), 2846–2853 (2005)
5. Nucci, A., Taft, N., Barakat, C., Thiran, P.: Controlled use of excess backbone bandwidth for providing new services in IP-over-WDM networks. IEEE J. Sel. Areas Commun. **22**(9), 1692–1707 (2004)
6. Fang, J., Sivakumar, M., Somani, A.K., Sivalingam, K.M.: On partial protection in groomed optical WDM mesh networks. In: Proceedings of IEEE International Conference on Dependable Systems and Network (DSN) (2005)
7. Bigos, W., Cousin, B., Gosselin, S., Le Foll, M., Nakajima, H.: Survivable MPLS over optical transport networks: cost and resource usage analysis. IEEE J. Sel. Areas Commun. **25**(5), 949–962 (2007)
8. Ratnam, K., Gurusamy, M., Zhou, L.: Differentiated QoS routing of restorable sub-lambda connections in IP-over-WDM networks using a multi-layer protection approach. In: Proceedings of International Conference on Broadband Networks (BroadNets 2005), vol. 1, pp. 127–136 (2005, October)
9. Rai, S., Song, L., Cavdar, C., Andrei, D., Mukherjee, B.: A novel approach to provision differentiated services in survivable IP-over-WDM networks. Opt. Switch. Netw. **5**(2–3), 170–176 (2008)
10. Cholda, P., Mykkeltveit, A., Helvik, B.E., Wittner, O.J., Jajszczyk, A.: A survey of resilience differentiation frameworks in communication networks. IEEE Commun. Surv. Tutor. **9**(4), 32–55 (2007)
11. Song, L., Zhang, J., Mukherjee, B.: Dynamic provisioning with availability guarantee for differentiated services in survivable mesh networks. IEEE J. Sel. Areas Commun. (OCN Supplement) **25**(4), 32–44 (2007)
12. Yao, W., Ramamurthy, B.: Survivable traffic grooming with differentiated end-to-end availability guarantees in WDM mesh networks. In: Proceedings of IEEE Workshop on Local and Metropolitan Area Networks (LANMAN), pp. 87–90 (2004)
13. Xiang, B., Yu, H., Wang, S., Li, L.: A differentiated shared protection algorithm supporting traffic grooming in WDM mesh networks. In: Proceedings of International Conference on Communications, Circuits and Systems (ICCCAS), vol. 1, pp. 628–632 (2004)
14. He, R., Wen, H., Li, L., Wang, G.: Shared sub-path protection algorithm in traffic-grooming WDM mesh networks. Photonic Netw. Commun. **8**(3), 239–249 (2004)
15. Zhu, H., Zang, H., Zhu, K., Mukherjee, B.: Dynamic traffic grooming in WDM mesh networks using a novel graph model. In: Proceedings of IEEE GlobeCom '02, vol. 3, pp. 2681–2685 (2002)
16. Tornatore, M., Maier, G., Pattavina, A.: Capacity versus availability trade-offs for availability-based routing. OSA J. Opt. Netw. **5**(11), 858–869 (2006)
17. Zhou, L., Held, M., Sennhauser, U.: Connection availability analysis of shared backup path-protected mesh networks. IEEE/OSA J. Lightwave Technol. **25**(5), 1111–1119 (2007)
18. Zhu, H., Zang, H., Zhu, K., Mukherjee, B.: A novel generic graph model for traffic grooming in heterogeneous WDM mesh networks. IEEE/ACM Trans. Netw. **11**(2), 285–299 (2003)
19. Clemente, R., Bartoli, M., Bossi, M., D'Orazio, G., Cosmo, G.: Risk management in availability SLA. In: Proceedings of International Conference on the Design of Reliable Communication Networks (DRCN) (2005)
20. Zhou, L., Grover, W.: A theory for setting the safety margin on availability guarantees in an SLA. In: Proceedings of International Conference on the Design of Reliable Communication Networks (DRCN) (2005)

21. Mello, D.A.A., Quiterio, G.S., Waldman, H., Schupke, D.A.: Specification of SLA survivability requirements for optical path protected connections. In: Proceedings of Optical Fiber Communication Conference (OFC) (2006)
22. Xia, M., Tornatore, M., Martel, C., Mukherjee, B.: Risk-aware routing for optical transport networks. In: Proceedings of IEEE INFOCOM (2010, April)
23. Lewis, E.E.: In: Sons, J.W. (ed.) Introduction to Reliability Engeneering. Wiley (1987)
24. Arci, D., Maier, G., Pattavina, A., Petecchi, D., Tornatore, M.: Availability models for protection techniques in WDM networks. In: Proceedings of International Conference on the Design of Reliable Communication Networks (DRCN) (2003)
25. Tornatore, M., Lucerna, D., Song, L., Mukherjee, B., Pattavina, A.: Dynamic SLA redefinition for shared-path-protected connections with known duration. In: Proceedings of Optical Fiber Communication Conference (OFC) (2008)
26. Mello, D.A.A., Schupke, D.A., Waldman, H.: A matrix-based analytical approach to connection unavailability estimation in shared backup path protection. IEEE Commun. Lett. **9**(9), 844–846 (2005)
27. Mello, D.A.A., Waldman, H.: Analytical bounds on the unavailability of protected connections in WDM optical networks. IEEE Commun. Lett. **11**(11), 901–903 (2007)
28. Zhang, J., Zhu, K., Zhang, H., Matloff, N., Mukherjee, B.: Availability-aware provisioning strategies for differentiated protection services in wavelength—convertible WDM mesh networks. IEEE/ACM Trans. Netw. **15**(5), 1177–1190 (2004)

Author Biographies

Massimo Tornatore is assistant professor at Politecnico di Milano, where he received a PhD degree in Information Engineering in 2006, and visiting assistant professor at the University of California, Davis. He is author of about 100 conference and journal papers and his research interests include protection, energy efficiency, traffic grooming in optical WDM networks and group communication security.

Diego Lucerna received a Ph.D. in Information Engineering in 2011 from Politecnico di Milano. His research interests include switching technologies, network equipments, telematic applications and management of public and private networks. He is currently enrolled in Huawei Technologies Italia as "Customer Support Optical Engineer" for WDM, SDH and microwave systems.

Biswanath Mukherjee received BTech from IIT-Kharagpur and PhD from University of Washington. He was General/TPC Co-Chair of OFC-2011/2009, and TPC Chair of IEEE INFOCOM-96. He is Distinguished Professor at University of California, Davis; co-winner, Best Paper Awards, IEEE Globecom 2007 and 2008; author of "Optical WDM Networks"; and IEEE Fellow.

Achille Pattavina received the Dr. Eng. degree in Electronic Engineering from University La Sapienza of Rome (Italy) in 1977. Since 1991 he has been with "Politecnico di Milano", Milano (Italy), where he is Full Professor. He has been author/coauthor of more than 200 papers in the area of communications networks.