Intelligent shared-segment protection

Massimo Tornatore\textsuperscript{a,c,*}, Matteo Carcagni\textsuperscript{d}, Canhui (Sam) Ou\textsuperscript{b}, Biswanath Mukherjee\textsuperscript{c}, Achille Pattavina\textsuperscript{a}

\textsuperscript{a}Department of Electronics and Information, Politecnico di Milano, Via Ponzio 34–35, 20121 Milan, Italy
\textsuperscript{b}AT&T Services, Inc. 2600 Camino Ramon, San Ramon, CA, USA
\textsuperscript{c}Department of Computer Science, University of California, Davis, CA 95616, USA
\textsuperscript{d}Create-Net, Via alla Cascata 56/C Povo, 38100 Trento, Italy

\textbf{A R T I C L E   I N F O}

Article history:
Received 1 June 2007
Received in revised form 5 December 2007
Accepted 6 February 2008
Available online 16 March 2008

Keywords:
Optical network
WDM
Control plane
Dynamic traffic
Holding-time
Shared-segment protection

\textbf{A B S T R A C T}

Progress in network technologies and protocols is paving the road towards flexible optical transport networks, in which dynamic leasable circuits could be set up and released on a short-term basis according to customers’ requirements.

Recently, new solutions for automatized network management promise to allow customers to specify the terms of the service level agreement (SLA) to be guaranteed (with different price range) by the service provider. In this paper we consider that these service level specifications (SLS), since they are now made available on-demand during the connection request, could be exploited to retrieve useful information able to improve the routing efficiency.

In particular, we propose to exploit the knowledge of connection holding time, among the other SLS, to develop a novel intelligent approach for \textit{shared-segment protection} (SSP). We will exploit the knowledge of the holding-time of connection requests to minimize resource overbuild due to backup capacity and hence to achieve resource-usage efficiency.

For a typical US nationwide network, we compare our two proposed holding-time aware approaches to the respective two holding-time unaware approaches: both of them, even in their holding-time unaware version, have been shown to be very efficient solutions for shared-segment protection. Nonetheless, we have obtained additional savings on resource overbuild of up to 11\% for practical scenarios exploiting holding-time knowledge.

© 2008 Published by Elsevier B.V.

1. Introduction

Optical networks provide a transport infrastructure with very high capacity, thanks to wavelength-division-multiplexing (WDM) technology. The huge bandwidth of WDM also requires efficient survivability mechanisms, because the failure of a network element (usually a node or a link) can cause a large amount of data loss [1]; a highly available WDM layer is crucial to enable quality-of-service sensitive applications over it. Many existing transport networks use SONET rings: while ring-structured protection schemes are resilient to any single node or link failure, they typically rely on excessive capacity redundancy. So, recently, new techniques have been proposed to efficiently deal with this problem in mesh networks [2]. Among them, shared-segment protection (SSP) is a promising candidate because of its desirable resource efficiency, which is achieved by effectively sharing backup resources [3].

In order to efficiently accommodate the capacity needed for protection, new challenges, but also opportunities, are offered by evolution of the prevalently static traffic towards a more dynamic traffic paradigm. As a matter of fact, so far optical transport networks have been supporting connections which are provided and leased for long period of time, e.g., weeks or months.
Many new applications are emerging with requirements of large bandwidth over relatively short periods of time: let us consider for example important sport or social event video distribution or the massive data transfer for backup or storage purposes; or, in the case of research and public networks, e.g., remote control of scientific apparatus in research labs (microscopes or tele-medicine) or the interconnection of radioscopes for real-time cross-correlation of cosmic observation. It is widely agreed that this set of applications, which are still in their preliminary phase, will keep on growing, including new and always more end-user-oriented large-bandwidth services.

Technology and the bandwidth market are developing to provide the flexible platform the new applications are asking for: new agile optical crossconnects (OXC) are emerging to create mesh-structured optical WDM backbone networks; and, in order to manage and control dynamic WDM networks, new control protocols have been proposed: ASON/ASTN and GMPLS are protocol-independent, control-plane architectures (standardized by ITU and IETF, respectively).

New architectures and routines for user-controlled on-demand optical circuit provisioning [4,5] based on automatic or Web-based interfaces at the the management plane (MP) [6] will enable the on-line specification of the SLA terms to be guaranteed (with different price range) by the service provider. In this paper we consider that these service level specifications (SLS) coming from the MP could be exploited by the control plane to retrieve useful information able to improve the efficiency of the routing algorithms. Note that both ASON and GMPLS use distributed real-time signaling which allow configuring connections automatically and, in conjunction with appropriately extended versions of OSPF-TE, can take charge of the distribution of this information coming from the MP.

More specifically, we propose to exploit the knowledge of connection-holding-time, among the other SLS, to develop a novel intelligent approach for dynamic provisioning of shared-segment-protected connections in optical mesh networks employing WDM or, equivalently, MPLS networks with shared-segmented-protected bandwidth guaranteed tunnels [7].

In particular, shared backup-channel capacity reservation can achieve significant advantage by exploiting the additional information associated with connection durations. This opportunity has been already exploited for shared-path protection in [8] achieving significant improvements in network resource utilization.

We have chosen to compare our approach to generalized segment protection (GSP) [9] and auxiliary-graph-based segment protection (AGBSP) [10], which have been shown to be very efficient for shared-segment protection, but they are holding-time unaware. For a typical US nationwide network, we obtained savings on resource overbuilt of up to 11% for various practical scenarios.

The rest of this paper is organized as follows. Section 2 discusses some fundamental issues on shared-segment protection and formally states the problem. Section 3 introduces two known approaches for shared-segment protection, called GSP and AGBSP. In Section 4, a new connection-holding-time aware link-cost assignment is discussed: this new methodology, named provisioning by holding-time opportunity (PHOTO) is applied to the standard algorithms GSP and AGBSP, giving origin to the new algorithms Ph-GSP and Ph-AGBSP. Section 5 evaluates by simulations the performance of Ph-GSP/AGBSP compared to the GSP/AGBSP algorithms. Section 6 concludes the paper.

2. Shared-segment protection

Various forms of segment protection have been proposed in Refs. [11,3,9,12]. The common idea of these approaches is to divide a working path (WP) into several working segments (WSs) and to protect each WS with a node/link-disjoint backup segment (BS). When a failure occurs, only the affected WS is switched to its BS, and the other WSs are unaware of the failure. In addition, in shared-segment protection (SSP), two BSs can share backup wavelength links as long as their WSs do not traverse the same link.

SSP can be classified as overlap SSP, if the WSs are allowed to overlap on some links, and no-overlap SSP, if WSs are strictly link-disjoint (see Fig. 1). SSP has a number of advantages compared to path protection. The end-to-end protection entity is a segment in segment protection as opposed to a path in path protection. Since a segment is typically shorter than a path in terms of hop count, segment protection is expected to have shorter protection-switching time and the probability of two working segments sharing the same risk is typically lower than the probability of two working paths sharing the same risk [13]. Segment protection can have better backup sharing compared to shared-path protection: in fact, segment protection has more flexibility in routing compared to path protection since path protection is a special case of segment protection in which every path has exactly one segment. Finally, segment protection is able to provide a higher availability degree with respect to the classical shared path protection: as a matter of fact, since all the overlapped segments (e.g., segment $ws_2$ in Fig. 1b) is protected by two backup segments ($bs_1$ and $bs_2$), SSP allows us to recover a larger number of double faults than shared path protection [14].

Depending on the strategy used to partition a WP, segment protection can be classified into predetermined partitioning, postdetermined partitioning, and integrated partitioning [10]. In this paper, we analyze postdetermined and integrated partitioning strategies since they have been demonstrated to provide the highest flexibility and performance.

![Fig. 1. No-overlap (a) and overlap (b) SSP.](image-url)
3. Efficient algorithms for segment shared protection without holding-time awareness

In this section we formulate the SSP routing problem and describe some recent approaches to solve it. In the next section we will see how to introduce the knowledge of connection holding-time to improve these algorithms.

3.1. Problem statement

We consider dynamic traffic. Upon the arrival of a new connection request, the network management system needs to compute a working path \( l_w \) and a list of backup segments \( \{l_b^k\} \), which divide the working path into overlapping segments \( l_b^k \) such that \( l_w \) and \( l_b^k \) are node-/link-disjoint. New backup segments \( \{l_b^k\} \) can share wavelength links with existing backup segments as well as among themselves. Unfortunately, it is NP-hard to determine if there exists an eligible solution: in [15], authors proved the NP-completeness of the existence version of shared-path protection, which is a special case of segment protection with the number of segments being one. As a result, we need to resort to heuristics.

A network is represented as a weighted, directed graph \( G = (V, E, C) \), where \( V \) is the set of nodes, \( E \) is the set of unidirectional fibers (referred to as links), \( C : E \rightarrow R^+ \) is a function that maps the elements in \( E \) to positive real numbers representing the link costs, and \( \lambda : E \rightarrow Z^+ \) specifies the number of wavelengths on each link (where \( Z^+ \) denotes the set of positive integers). \( \lambda_e \) denotes the number of free wavelengths on link \( e \in E \).

A conflict set is associated with a link to identify the sharing potential between backup segments. The conflict set \( \mathcal{C}_e \) for link \( e \) defines the set of nodes traversed by such working segments whose backup segments utilize wavelengths on link \( e \). The conflict set \( \mathcal{C}_e \) for link \( e \) can be represented as an integer set \( \{v^e_u | e \in E, 0 \leq v^e_u \leq \lambda(e)\} \), where \( v^e_u \) specifies the number of working paths that traverse node \( u \) and are protected by link \( e \) (i.e., their corresponding backup paths traverse link \( e \)). The number of wavelengths to be reserved for backup paths on link \( e \) is thus \( v_e = \max_{u \in \mathcal{C}_e} \{v^e_u\} \). Clearly, the union of the conflict sets for all the links aggregates the per-segment-based information, and the size of the conflict set depends only on the number of links, not on the number of segments.

3.2. GSP: a postdetermined partitioning solution

GSP is a practical heuristic which, upon the arrival of a new connection request, dynamically divides a judiciously-selected working path into multiple overlapped working segments and computes a backup segment for each working segment while accommodating backup sharing. It has been proposed in [9]: with respect to previous approaches that partition the working path in a fixed manner [3], GSP is able to flexibly choose the segments to be protected by extending an idea first proposed in [16] to incorporate backup sharing.

While the details of the algorithm can be found in [9], we provide here only the basic steps of the GSP algorithm needed to understand the role played by the introduction of the holding-time knowledge in the upgrade of the algorithm:

1. select candidate working paths: compute up to \( K \) admissible minimal-cost paths \( l_w = \{l_w^k | 1 < k < K\} \) in \( G \) based on Yen’s K-shortest-paths algorithm.
2. compute backup segments for each candidate working path \( l_w^k \) in \( L_w \) as follows:
   a. define new link-cost function \( C(e) \) for \( e \in E \):
   \[
   C(e) := \begin{cases} 
   +\infty & \text{if } e \in l_w^k \lor (\lambda^e_0 = 0 \land (\forall u \in l_w^k, v^e_u = v^e_0)) \cr
   C(e) & \text{otherwise} 
   \end{cases}
   \]
   b. transform the original graph \( G = (V, E, C, \lambda) \) in \( G = (V, E, C, \lambda) \) according to a set of rules reported in [9]
   c. compute a least-cost \( l_w^k \) in \( G \)
   d. map \( l_w^k \) in \( G \) and decompose \( l_w^k \) into a list of backup segments \( \{l_b^k\} \)
3. select the pair \( \{l_w^k, \{l_b^k\}\} \) of minimal cost.

Let us focus on the link-cost assignment for backup-segments routing in Step 2(a). Once a working path is fixed, we need to set an appropriate link-cost assignment to route the corresponding backup path. By means of the conflict set, we know the amount of backup capacity reserved on link \( e \): if the entire backup capacity is needed to protect the failure of a link along the candidate working path (i.e., \( \forall u \in l_w^k, v^e_u = v^e_0 \)), then the link has to be considered as not shareable. In this case, if there is free capacity (\( \lambda^e_0 > 0 \)), then a new wavelength can be allocated and link cost is the full cost \( C(e) \). Otherwise, if \( \lambda^e_0 = 0 \), the link is not usable and its cost is set to infinite. Finally, if there are backup channels other than those reserved to protect a link along the working path (condition expressed by “if \( \forall u (u \neq s, d) \in l_w^k, v^e_u < v^e_0 \)”, then the link is shareable and we do not allocate a new channel on it. So, the cost is set to a low value by multiplying \( C(e) \) with a small constant \( \epsilon \).

Actually, the state of any given link (i.e., if the link is shareable or not) may change during the holding-time of the incoming connection: e.g., a change may occur when an existing connection departs or a new connection arrives. So, a link, which is in shareable state with respect to the incoming connection at its arrival instant, may become not-shareable, since the deallocation of backup capacity due to connection departures has changed the values of the conflict vector of that link.

3.3. AGBSP: an integrated partitioning solution

To maximize backup sharing and provide flexibility at the same time, integrated partitioning first computes a possible backup segment set for a predetermined possible working segment set and then dynamically selects a best subset. In [10], authors propose a new shared-segment protection algorithm utilizing an integrated partitioning strategy. This algorithm obtains a global optimal
segmentation of a WP by dynamically selecting a best BS subset to protect the entire WP from all possible backup segments. It is worth noting that, to the best of our knowledge, AGBSP yields less blocking probability and resource overbuild ratio than any other previous existing segment protection algorithms.

The AGBSP algorithm is specified in detail in [10]. As done for GSP, we will try here to catch only the guidelines of the AGBSP procedure, aiming at clarify to role played by the holding-time knowledge in the upgraded algorithm.

First of all, let us introduce some basic concepts to be used in the following algorithm description. The possible working segment set (PWSS) for a given WP is a set containing all possible WSSs of the WP.

Correspondingly, there is a possible backup segment set (PBSS) in which each of the elements is a BS protecting a WS in PWSS. A valid working segment set (VWSS) is an eligible set that partitions a given WP into several contiguous or overlapping WSSs, and each of these WSSs is protected by a link-disjoint BS. A VWSS is a subset of a PWSS and is the algorithm solution. Correspondingly, there is a valid backup segment set (VBSS) containing the valid BSs.

Now, the main steps of the algorithm are as in the following:

1. Compute a minimal cost WP \( l_w \).
2. Construct PWSS by enumerating all possible WSSs of \( l_w \).
3. Seek a BS for each element in the PWSS to construct the PBSS, as follows:
   (a) define new link-cost function \( C(e) \) for \( e \in E \) as in the step 2(a) of GSP algorithm
   (b) according to new cost function, compute a least-cost BS for each WS in PWSS
4. Construct an auxiliary graph based on \( l_w \) and the PBSS, so that it contains all the admissible alternatives segment protected paths composed by WS and BS in PWSS and PBSS.
5. Apply the shortest-path algorithm to the auxiliary graph to determine the VWSS and VBSS.

Again a key-point of the algorithm consists in the link-cost assignment for backup-segments routing. The same consideration discussed in the previous subsection hold also in this case: once a working path is fixed, we set an opportune link-cost assignment to route the corresponding backup path, trying to encourage the resource sharing and this is made possible by means of the conflict set. And again, the state of a given link (i.e., if the link is shareable or not) will be likely to change during the holding-time of the incoming connection: in the next section we will see how to keep track of the evolution of the link state to propose a more-efficient link-cost assignment.

4. Provisioning by holding-time opportunity (PHOTO) methodology

In this study, we have assumed that information on future connections may not be known in advance (this further information would lead to the static scheduling problem). Nevertheless, we could exploit at least the information about the departure events, which is simply retrievable from the knowledge of the connection-holding-time. So, we could modify the link-cost assignment to try to capture the future degree of “shareability” of a given link.

By introducing holding-time-awareness in a routing algorithm, the usual assignment of link costs can be improved. Traditional approaches try to minimize the amount of additional wavelengths used by the new incoming connection on the basis of current network state. Our proposal consists of minimizing not only the (current) additional capacity, but also the cost of additional wavelengths multiplied by the estimated time interval during which this additional wavelength has been provisioned for the incoming connection. This cost assignment will improve the resource overbuild associated with backup paths, because the optimization metric has been evolved from wavelength mileage to wavelength mileage times time.

4.1. PHOTO Link-Cost Assignment

In order to follow step-by-step the changes in the link states, we introduce the new symbols \( v_e(\Delta t_k) \), \( v^c_e(\Delta t_k) \) and \( C(e, \Delta t_k) \), which express the values of \( v_e \), \( v^c_e \) and \( C(e) \), respectively, in the interval of time \( \Delta t_k \).

Let us define \( \Delta t_k \) first. Let us suppose, without loss of generality, that the remaining holding-times (at instant \( t_k \)) of the \( n \) connections already in the network are ordered so that \( h_{i,t} < h_{i,t+1} \), \( i = 1, 2, \ldots, n \). Let us suppose also that, for a certain index \( m \), \( h_{m-1} < t_k \), but \( h_{m} \geq t_k \); so we can set \( h_{m} = t_k \). As a consequence, \( \tau = \{ h_1, h_2, \ldots, h_{m-1} \} \) will indicate the departure events before incoming connection departure \( t_k \) and \( \Delta t_k = t_k - \tau \), \( \tau \) expresses the time interval between two departures. Conflict set \( v_e(\Delta t_k) \), \( v^c_e(\Delta t_k) \), and associated cost \( C(e, \Delta t_k) \) will be updated according to the \( k \)th connection departure. In other words, we have divided the holding time \( t_k \) into a series of time intervals \( \Delta \tau \) which express the distance between two departures. The cost of link \( e \) during the interval \( \Delta t_k \) will be re-evaluated after having considered the departure of the \( k \)th connection, according to the following scheme:

\[
C(e, \Delta t_k) := \begin{cases} 
+\infty & \text{if } e \in E_w \land (\nexists j \in E_w : v^c_j(\Delta t_k) = v^c_j(\Delta t_{k-1})), \\
\epsilon \times C(e) & \text{if } \forall u \in E_w : v^c_u(\Delta t_k) < v^c_u(\Delta t_{k-1}), \\ 
C(e) & \text{otherwise.}
\end{cases}
\]

It is worth noting that (1) all of the previously-cited parameters are evaluated at the instant of arrival of the incoming connection, (2) future arrivals are not known, and (3) departure events after \( t_k + \tau \) can be neglected. Overall cost \( C(e) \) in PHOTO will be evaluated by considering each connection departure and by summing the cost contribution due to any time interval in the following manner:

\( v_e \) will be equal to \( v_e(\Delta t_1) \) (where \( \Delta t_1 = t_1 - \tau = h_{1,t} - 0 \)).
Hence, the new cost function to be minimized is not only the wavelength utilization at the arrival instant, but a more meaningful estimation, which considers the wavelength usage along the entire holding period.

4.2. A numerical example

As an example, we refer to the network in Fig. 2a where the GSP algorithm is running: the two connections \( r_1 \) and \( r_2 \) already exist at instant \( t_1^1, t_1^2 = 0 \). These connections are characterized by holding-times \( t_1^1 = 30, t_1^2 = 20 \), working routes along link \( h^1 = (b, z), h^2 = (m, l, x) \) and backup routes along \( p^1 = (b, n, z), p^2 = (m, l, x) \) respectively.

At instant \( t_2^1 = 10 \), connection \( r_1 \) with duration \( t_2^1 = 30 \) has to be provisioned also. Its working route is set on the path \( (j, k, n, x) \) (Fig. 2a).

When, at time \( t_3^2 \), a new connection \( r_3 \) is required to be set up between node \( j \) and node \( x \), the remaining holding times for the two previous connections are \( t_3^1 = 20 \) and \( t_3^2 = 10 \). Clearly, \( t_3^1 < t_3^2 \) and routing \( r_3 \)'s backup path along links used by the backup of \( r_2 \) would lead to a longer period of backup resource sharing than choosing a path along the links used by \( r_1 \)'s backup path.

If we apply the link-cost assignment of GSP, we obtain the two alternative candidate set of backup segments shown in Fig. 2(b) and (c): backup route \( bp1 \) along the segments \( (j, b, n) \) and \( (k, m, x) \) and backup route \( bp2 \) along the segments \( (j, b, n) \) and \( (n, z, x) \), respectively. Since the cost of a new wavelength on a link has been set to 1, while the cost of a shared wavelength is \( \epsilon \), the cost of \( bp1 \) and \( bp2 \) would be the same: \( 2 + 2 \times \epsilon \). But, as previously pointed out, the \( bp2 \) would minimize the allocated resource in the network\(^2\). Since the shared link \( (b - n) \) is used by both the routes, we can demonstrate this last assertion by simply observing the behavior of backup capacity reserved on link \( (n, z) \) and link \( (m, x) \): namely, \( bp2 \) on link \( (m, x) \) will share backup capacity with connection \( r_2 \) for a longer time than \( bp1 \) on link \( (n, z) \) with connection \( r_1 \). This implies that: if we choose \( bp1 \), an additional wavelength will be reserved on link \( (n, z) \) for a time interval equal to \( t_2^3 - t_1^2 \); if we choose the \( bp2 \), an additional wavelength will be reserved on link \( (g, h) \) for \( t_2^3 - t_1^2 \). Since \( t_2^3 - t_1^2 > t_2^3 - t_1^2 \), the \( bp1 \) requires additional capacity for the time duration \( t_2^3 - t_1^2 \), while \( bp2 \) does not require the additional capacity during this time interval. Therefore, we should exploit the information provided by the holding-time to modify the link-cost assignment.

Let us see how in detail PHOTO evaluates the cost of link \( (m, x) \) and link \( (n, z) \) by considering the evolution of cost in three different time intervals, as follows (see Fig. 3):  

- \( \Delta_{r_1} \) (from time 10 to 20): connection \( r_3 \) does not require a new wavelength on link \( (n, z) \) and link \( (m, x) \) for its backup path, because \( r_1 \) and \( r_2 \) are disjoint to \( r_3 \). So, the link cost during \( \Delta_{r_1} \) will be \( \epsilon \Delta_{r_1} C(e) \) (shareable state) for both link \( (n, z) \) and link \( (m, x) \).
- \( \Delta_{r_2} \) (from time 20 to 30): connection \( r_3 \) does not require an additional wavelength on link \( (m, x) \) for its backup path, because \( r_2 \) is still in the network, while it requires an additional wavelength on link \( (n, z) \) because \( r_1 \) has left the network. So, cost of link \( (c, d) \) during \( \Delta_{r_2} \) will be \( \epsilon \Delta_{r_2} C(e) \), while the cost of link \( (g, h) \) during \( \Delta_{r_2} \) will be \( \Delta_{r_2} C(e) \) (not-shareable state).

\(^2\) In Section 5, we will more precisely define this parameter, which is referred to as resource overbuild (RO).
5. Results

We now quantitatively evaluate the performance of our proposed PHOTO-based algorithms compared to the baseline GSP and AGBSP algorithms. We simulate a dynamic network environment with the assumptions that the connection-arrival process is Poisson and the connection-holding-time follows a negative exponential distribution.

For the illustrative results shown here, in every experiment, 10^6 connection requests are simulated; all the plotted values have a 95% confidence interval not larger than 0.5% of the plotted value. Requests are uniformly distributed among all node pairs; average connection-holding-time is normalized to unity; the cost of any link is unity; and our example network topology with 16 wavelengths per fiber is a carrier’s US nationwide backbone network topology, as shown in Fig. 4.

We employ two metrics to highlight the performance improvement achievable by PHOTO application: total channel consumption and resource overbuild.

5.1. Total channel consumption

**Total channel consumption (TCh)** is the overall number of channels needed to support all the offered traffic multiplied by the time interval these channels are actually used (note that average holding time of a connection is normalized to unity). In Fig. 5, we show TCh for a simulation experiment with 10^6 offered connections by Ph-GSP and GSP and K = 1–3. PHOTO always requires fewer channels. This number tends to decrease for increasing load, because for high loads there is a higher probability to share. Note that the increase of K from 1 to 2 leads to a remarkable savings in network resources, while increasing K from 2 to 3 yields only a slight improvement. The same considerations hold for Ph-AGBSP.

5.2. Resource overbuild

For each connection c_i with duration h_i served during our simulation, let us consider b_i the amount of backup resources consumed and w_i the amount of working re-
A figure of merit for comparing backup resource efficiency is resource overbuild, defined as the amount of wavelength channels consumed by backup paths \( B_i = \sum_c b_{ci} \cdot h_i \) over the amount of wavelength channels utilized by working paths \( W_i = \sum_c W_{ci} \cdot h_i \) [17]. Resource overbuild (RO) indicates the amount of extra resources needed for providing protection as the percentage of the amount of resources required without protection. Typically, it is desirable to have lower RO because this implies better backup sharing.

Fig. 6 shows that Ph-based heuristics always outperform the respective holding-time unaware heuristics when sharing of backup resources is involved. We compared in this graphic the RO performance of shared path protection (SPP) and shared-segment protection (SSP) in the case of \( K = 2 \): it is worth noting that SSP has a lower RO than SPP, since SSP has higher flexibility in finding resource-efficient backup segments compared to SSP, which is constrained to follow one single backup path from source to destination node. Furthermore, even more apparent is the RO gain achievable with integrated partitioning (AGBSP) with respect to post-determined partitioning (GSP).

Figs. 7 and 8 focus on the analysis of Ph-GSP and Ph-AGBSP. Fig. 7a shows that Ph-AGBSP has lower RO over AGBSP.

RO assumes smaller values for increasing values of offered traffic due a wider availability of shareable capacity. The percentage difference \( \left( \frac{RO_{Ph-AGBSP} - RO_{AGBSP}}{RO_{AGBSP}} \right) \times 100 \) between the two approaches is shown in Fig. 7b: the savings in RO are remarkable especially for higher values of \( K \) and light offered traffic. We obtained savings ranging from 2% to almost 10%. Note that \( K \) affects positively the RO percentage difference also because, when more than one path pair is evaluated, the working-path average length tends to increase.

Results obtained comparing Ph-GSP and GSP lead to similar conclusions: RO obtained by Ph-GSP are always smaller that the respective values of RO obtained by GSP, but this performance improvement tends to decrease with...
both. Then, percentage gain is less appreciable and varies from 2% to 7%. It is worth noting that in the analysis of the AGBSP algorithm the value of maximal value of $K$ is $K=4$, while in the case of GSP algorithm the highest value is $K=3$: we do not provide results beyond these two threshold values because a further increase in $K$ does not imply any further improvement in RO. The fact that this threshold effect arises for a larger value of $K$ in the case of AGBSP is due to the wider set of admissible solutions that AGBSP allows to investigate thanks to the integrated partitioning approach.

We can observe that PHOTO’s gain tends to decrease for high values of the arrival rate. This behavior is due to the same nature of the algorithm: Ph-GSP tries to give a suggestion on the best route for the backup path on the basis of information on holding-time associated with the existing connections and neglecting future connection arrivals. Therefore, if the arrival rate is high (e.g., arrival rate equal to 100 means that, on average, during the lifetime of a connection, 100 new connections will be offered to the network), then the quality of the estimation used by the algorithm decreases which, in turn, reduces the value of Ph-GSP/AGBSP over GSP/AGBSP.

Finally, in Fig. 9 we analyze the algorithm performance when PHOTO considers only a subset of the departing connections, instead of considering all the connections departing along the holding-time. We have drawn the percent gain in RO with respect to GSP (we have chosen the case with $K=1$) obtained by original Ph-GSP and by some Ph-GSP versions with limited knowledge of future departures (say the first $D$ departing connections, in our case 5, 10, and 20). It is not surprising that performances are strongly affected by this limitation; it is less obvious noting that gain in RO overbuild is proportional to the number of connections whose holding-time is known; this means that even if the holding-time information is available on only a subset of the connections, the proposed approach could still be used to save backup resources.

6. Conclusion and future work

In this paper we have considered how to exploit the knowledge of connection holding-time, among the other SLS, to develop a novel intelligent approach to shared-segment protection (SSP). We investigated a holding-time-aware, dynamic, connection-provisioning algorithm to improve sharing of backup resources in segment protection. We observed significant savings in protection-resource usage by employing our new approach, called PHOTO, as opposed to other efficient approaches that are holding-time-unaware, namely, GSP and AGBSP. The improvement in resource overbuild is found to be up to 11% for a US nationwide network with typical parameters. The proposed method is applicable to other contexts as well, such as MPLS networks, for bandwidth-guaranteed connections.

Acknowledgements

The work described in this paper was carried out with the support of the BONE-project (“Building the Future
Optical Network in Europe”), a Network of Excellence funded by the European Commission through the 7th ICT-Framework Programme. A preliminary version of this work has been presented at Globecom 2006.

References


Matteo Carcagni received the M.S (laurea) degree in telecommunications engineering from Politecnico di Milano, Italy in 2006. He is a Ph.D. candidate in telecommunications engineering at the International Doctorate School in Information and Communication Technology at the University of Trento, Italy under the supervision of Prof. Imrich Chlamtac. His research interests focus on survivable transparent optical networks and his current activities include the development of enhanced GMPLS control plane architectures.

Canhui (Sam) Ou received the B.S. degree from Peking University, Beijing, China, in 2000, and the M.S. degree from the University of California, Davis, in 2001. He is working toward the Ph.D. degree in the Computer Science Department, University of California, Davis. He worked as a Summer Intern at Sprint Advanced Technology Laboratories, Burlingame, CA, during summer 2001 and 2002. His research interests include design and analysis of wavelength-routed WDM networks with focus on survivability and traffic grooming.

Biswa Nath Mukherjee received the B.Tech. (Hons) degree from Indian Institute of Technology, Kharagpur (India) in 1980 and the Ph.D. degree from University of Washington, Seattle, in June 1987. At Washington, he held a GTE Teaching Fellowship and a General Electric Foundation Fellowship. In July 1987, he joined the University of California, Davis, where he has been Professor of Computer Science since July 1995 (and currently holds the Child Family Endowed Chair Professorship), and served as Chairman of the Department of Computer Science during September 1997 to June 2000. He is winner of the 2004 Distinguished Graduate Mentoring Award at UC Davis. Two PhD Dissertations (by Dr. Laxman Sahasrabuddhe and Dr. Keyao Zhu), which were supervised by Professor Mukherjee, were winners of the 2000 and 2004 UC Davis College of Engineering Distinguished Dissertation Awards. To date, he has graduated 30 Ph.D. students. Currently, he supervises the research of nearly 25 scholars, mainly PhD students and visiting research scientists in his laboratory. He is co-winner of Best Paper Awards presented at the 1991 and the 1994 National Computer Security Conferences, and at the Optical Networks Symposium of the IEEE Globecom 2007 conference. He serves or has served on the editorial boards of the IEEE/ACM Transactions on Networking, IEEE Network, ACM/Baltzer Wireless Information Networks (WINET), Journal of High-Speed Networks, Photonic Network Communications, Optical Network Magazine, and Optical Switching and Networking. He served as Editor-at-Large for optical networking and communications for the IEEE Communications Society; as the Technical Program Chair of the IEEE INFOCOM’96 conference; and as Chairman of the IEEE Communication Society’s Optical Networking Technical Committee (ONTC) during 2003–05. He is author of the textbook “Optical WDM Networks” published by McGraw Hill in 1997, a book which received the Association of American Publishers, Inc.’s 1997 Honorable Mention in Computer Science. He was a Member of the Board of Directors of iLocks, Inc., a Silicon Valley startup company during 2005–07. He has consulted for and served on the Technical Advisory Board (TAB) of a number of startup companies in optical networking. His
current TAB appointments include: Teknovus, Intelligent Fiber Optic Systems, and LookAhead Decisions Inc. (LDI). He is a Fellow of the IEEE. His research interests include lightwave networks, network security, and wireless networks.

Achille Pattavina (M’85/SM’93) received the Dr. Eng. degree in Electronic Engineering from University “La Sapienza” of Rome (Italy) in 1977. He was with the same University until 1991 when he moved to “Politecnico di Milano”, Milan (Italy), where he is now Full Professor. He has been author of more than 100 papers in the area of Communications Networks published in leading international journals and conference proceedings. He has been guest or co-guest editor of special issues on switching architectures in IEEE and non-IEEE journals. He has been engaged in many research activities, including European Union funded projects. He has authored two books, Switching Theory, Architectures and Performance in Broadband ATM Networks (New York: Wiley, 1998) and Communication Networks (McGraw-Hill, 1st ed. 2002, 2nd ed. 2007, in Italian). He has been Editor for Switching Architecture Performance of the IEEE Transactions on Communications since 1994 and Editor-in-Chief of the European Transactions on Telecommunications since 2001. He is a Senior Member of the IEEE Communications Society. His current research interests are in the area of optical switching and networking, traffic modeling and multilayer network design.