Availability Target Redefinition for Dynamic Connections in WDM Networks with Shared Path Protection

Diego Lucerna*, Massimo Tornatore†, Biswanath Mukherjee†, Achille Pattavina*

*Department of Electronics and Information, Politecnico di Milano, Via Bonzanigo 34-35, 20121 Milan, Italy
†Department of Computer Science, University of California, Davis, CA 95616, USA

Email: {lucerna,pattavina}@elet.polimi.it {tornator,mukherje}@cs.ucdavis.edu

Abstract—Recently, new solutions for automatized management in optical networks promise to allow customers to specify on-demand the terms of the Service Level Agreement (SLA) to be guaranteed by the Service Provider (SP). For a WDM mesh network that provides shared-path-protection we exploit the knowledge, among the other Service Level Specifications (SLS), of the connection holding time to dynamically manage the availability provided by the network to the connection during its lifetime. We monitor the actual availability provided to the connection, considering, e.g., the actual downtime experienced by the connection, and, every time the network status changes (typically, for a fault occurrence, or a connection departure or arrival), we evaluate the maximal availability that can be provided to the connection based on our knowledge of all the predictable network state changes, i.e., the future connection departures. Since some of the connection will be ahead of their stipulated availability (credit), while other connections will be behind their availability requirements (debit), we propose a mechanism that allows to “trade” availability “credits” and “debts”, by increasing or decreasing the shareability level of the backup capacity. So, we introduce an holding-time-aware approach which permits to manage in a more flexible manner the availability provided during the connection lifetimes. For a typical wavelength-convertible US nationwide network, our approach obtains significative savings on Blocking Probability (BP), while reducing the penalties due to SLA violations.

Index Terms—Optical network, WDM, SLA, availability, holding time, shared protection

I. 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. Various protection and restoration mechanisms for WDM mesh networks have been proposed [1]: among them, Shared-Path Protection (SPP) is a promising candidate, because of its desirable resource efficiency [2].

Recently, many new applications are emerging with requirements of large bandwidth over relatively short and predictable periods of time: let us consider, e.g., video distribution of important sport or social event, or the massive data transfer for backup or storage purposes. In order to efficiently accommodate the backup capacity, new challenges, but also opportunities, are offered by network evolution of the prevalently static traffic towards a more dynamic traffic paradigm.

Technology and bandwidth market are developing to provide the flexible platform the new applications are asking for. New architectures and routines for user-controlled on-demand optical circuit provisioning [3] will enable the on-line specification of the SLA terms to be guaranteed (with different price range) by the service provider (SP)1. In other words, the users are able to specify the QoS terms [3] of their connection requests, e.g., the availability or the holding-time (HT).

Specifically, the availability target plays a key role in SLAs stipulated between the SP and its customers. The SP should be able to provide an appropriate level of service availability 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. In particular, when adopting SPP for dynamic traffic, care should be taken in monitoring connection availability. Whenever a new SPP connection is routed, the SP must guarantee not only the SLA of the incoming connection, but also must check that availability targets of the existing connections are still respected despite the increased sharing of backup resources.

Typically, an SP provisions a connection only if its network is able to provide the customer a statistical availability (evaluated a-priori) that is equal or larger than the availability target. In this work, we present a novel availability-guaranteed provisioning framework that allows to manage dynamically the availability provided to a connection during its lifetime. Our approach i) leverages on the information about the evolution of connection departures (future departure imply decreased sharing and in turn more availability) and ii) is able to dynamically “trade” availability from connections ahead their

1Note that both ASON and GMPLS use distributed real-time signaling that, in conjunction with appropriately extended versions of OSPF-TE, can take charge of the distribution of this information coming from the MP.
availability target to connections behind availability target\(^2\).

Our method is based on the hypothesis that a dynamic connection request specifies its duration in the SLA [4]. When a new connection has to be routed, the availability status of existing connections in the network is re-evaluated by considering their past holding time and remaining holding time and its availability target can be redefined: e.g., if a connection has not been affected by failures, it can be considered in “debit” of availability with respect to the SP and its availability target can be opportunistically decreased by the SP, as long as the original SLA target provided to the customer is still respected. On the other hand, if a connection has undergone an outage period and it is getting close to its maximal acceptable downtime, then it can be considered in “credit” of availability with respect to the SP, and the availability target will be increased by the SP in the attempt to match the original stipulated SLA target. In other words, our approach dynamically transfers availability from connections that have an availability debit to connections in credit of availability by allowing more or less sharing of backup resources for those connections. Moreover, we exploit also the knowledge of the future departures of connections from the network. Under this assumption, even if the SP can not guarantee the a-priori statistical availability of the connections, the SP can decide to accept the new incoming connection request, based on the future credit of availability provided by departing connections.

Even if the methodology proposed in this paper can be applied to any SPP algorithm presented in literature, here we show the effectiveness of our approach through simulation experiments using the algorithm Availability-Guaranteed Provisioning algorithm (AGP) presented in [2].

The rest of the paper is organized as follows. Section II overviews the background work on the availability-guaranteed SPP strategies. In section III we formally state the SPP problem. In section IV a rigorous approach to evaluate the availability in a SPP network scenario is presented. Section V describes a holding-time-agnostic, yet efficient, solution for SPP, called AGP. In section VI a new Holding-Time-Aware approach (HTA) to manage dynamically the availability targets is presented for the SPP problem. Section VII evaluates by simulations the performance of our methodology HTA applied on AGP algorithm compared with the AGP basic approach. In Section VIII, we draw some conclusions.

II. PRIOR WORK

We consider the problem of dynamic lightpath provisioning with Shared-Path Protection (SPP). Under the dynamic provisioning scenario, a network management system needs to compute two link-disjoint paths, a dedicated working path and a shared backup path, for an incoming lightpath request based on the current network state. Connection routing usually relies on applying two-step methodologies (e.g., [5]), which allow to compute separately the working and the backup paths by shortest-path algorithms that minimize the total link costs. Generally, link weights are assigned according to optimized metrics such as fiber distance, hop count, link load, etc.

In [6], [7], meaningful link-cost assignment approaches have been proposed to increase the sharing of resources which are already reserved by the backup paths of other working connections, instead of reserving new resources. Reference [4] shows that in SPP backup capacity can be decreased by exploiting the holding-time information of connections which have already been provisioned in the network.

However, these approaches do not take into consideration that incoming connections may have different availability requirements. The works in [2] and [8] propose new routing algorithms which support service differentiation under static and dynamic traffic conditions, respectively. The primary objective is to route connections that comply with their target availability. A secondary objective is to minimize resource usage. In [9], it is shown that, if the cost of a link is defined as a function of its availability, finding a shortest path traversing these links becomes equivalent to finding the Most-Reliable Path (MRP). The work in [10] also investigates SPP with guaranteed availability requirements in a dynamic environment and uses a matrix-based approach for availability analysis. Different availability analysis methods can be found in previous studies [9], [11], [12], which estimate the availability of a path or a path pair for a connection request before provisioning resources. In [13], the authors show that, beside the classical methods to estimate the long-term statistical availability of a connection, it is crucial to determine a safety factor on the total outage time stipulated with a customer on any finite-term contract. Ref. [14] provides an analytical analysis to compute the safe margin considering both the number of failures and failure repair time as random variables.

To the best of our knowledge, [15] is the first work where connection-availability and connection holding time are jointly considered. However, neither the outage history of connections nor the connections holding time information are exploited to dynamically manage the availability targets during connections’ lifetime. Our previous work [16] represents the first step in the development of the complete framework proposed here: the main difference is that, while only the “debit” case is considered in [16], here availability can be both borrowed and returned to the network under the form of debits and credits, making the approach much more efficient and dynamic.

III. NOTATION AND PROBLEM STATEMENT

We first define the notations and then formally state the SPP problem. A network is represented as a weighted, directed graph \( G = (V, E, C, W_e) \), 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 \( W_e : E \rightarrow Z^*_+ \) specifies the number of wavelengths on a generic link \( e \) (where \( Z^*_+ \) denotes the set of positive integers).
We use \( W^e \) to denote the number of free wavelengths on link \( e \in E \). We denote the set of existing lightpaths in the network at any time by \( L = \{ (l^w, l^b, s_d, t_a, t_b) \} \), where the quadruple \( (l^w, l^b, s_d, t_a, t_b) \) specifies the working path, the backup path, the arrival and the holding time for the \( i^{th} \) lightpath.

We associate a link vector \([6]\) with each link in the network, to identify the sharing potential between backup paths. The link vector \( \nu_e \) for link \( e \) can be represented as an integer set, \( \{ \nu^e_e | \nu^e_e \in E, 0 \leq \nu^e_e \leq W^e \} \), where \( \nu^e_e \) specifies the number of working lightpaths that traverse link \( e \) and are protected by link \( e \) (i.e., their corresponding backup lightpaths traverse link \( e \)). Through such a simple data structure, the link vector captures the necessary information on the sharing potential offered by each link. The number of wavelengths which need to be reserved for backup lightpaths on link \( e \) is thus \( \nu^e_e = \max_{e' \in E} \{ \nu^e_e \} \). Therefore, using the link vector, we can simply reserve \( \nu^e_e \) wavelengths on link \( e \) as backup wavelengths.

Based on the information contained in \( \nu_e \), an SPP provisioning procedure has to find two Shared-Link-Risk-Group (SLRG)-disjoint paths for the incoming request \( (l^w, l^b, s_d, t_a, t_b) \), characterized respectively by its working path, backup path, source node, destination node, arrival time and duration. We recall that, under this assumption, the working and backup lightpaths, \( l^w \) and \( l^b \), of a new incoming connection must satisfy the following constraints with respect to the other existing lightpaths \( l^w \) and \( l^b \) as follows:

- **C.1** \( l^w \) and \( l^b \) are link disjoint.
- **C.2** \( l^w \) and \( l^w \) do not utilize the same wavelength on any common link they traverse.
- **C.3** \( l^w \) does not share any wavelength with \( l^b \) on any common link they traverse.
- **C.4** \( l^b \) and \( l^b \) can share a wavelength on a common link only if \( l^w \) and \( l^w \) are link disjoint.

If two routes \( l^w \) and \( l^b \) are found in respect to the constraints C.1-C.4, the connection is provisioned and the link vector along the routes are update.

In Section V, we will describe how this SPP provisioning process can be upgraded to enforce availability targets. Meanwhile, in Section IV a rigorous approach to evaluate the availability of an SPP connection is shown.

### IV. Availability Evaluation for a SPP Connection

In general, availability is the probability of a repairable system to be in an operating state. Failures and down states occur, but maintenance or repair action always return the system to an operating state. The basic equation for the availability of a system with constant failure rate \( \lambda \) and repair rate \( \mu \) is:

\[
A = \frac{\text{MTTR}}{\text{MTBF} + \text{MTTR}} = \frac{\mu}{\mu + \lambda}, \tag{1}
\]

where \( A \) is the availability, \( \text{MTBF} = 1/\lambda \) is the mean time between two consecutive failures, and \( \text{MTTR} = 1/\mu \) is the mean time to repair [11]. According to [17] we consider failure-immune nodes and we focus only on link failures. Then, let \( A_e \) denotes the availability of the link \( e \).

The availability of the working and the backup paths can be individually computed. Let \( l^w \) and \( l^b \) denote the set of links used by working path and backup path, respectively. Then, the availabilities of the paths are given by the following equations:

\[
A_{l^w} = \prod_{e \in l^w} A_e, \tag{2}
\]

\[
A_{l^b} = \prod_{e \in l^b} A_e. \tag{3}
\]

Since SPP provides 100% restorability on single failures, let us consider the effects of double failures on a connection to evaluate the availability [18]: the two additional parameters \( z_e^{e', e''} \) and \( D_e^{e', e''} \) will allow us to identify the links that cause a resource conflict with links owing to the working path under the assumption that (i) two concurrent failures are affecting the network and (ii) one out of these two failures is affecting the working path. \( z_e^{e', e''} \) represents the number of working paths that cross the two links \( e' \) and \( e'' \), whose backup path contains \( e \). \( D_e^{e', e''} \) denotes the number of wavelengths that would be required on link \( e \) in order to fully restore the traffic on link \( e \) even if \( e' \) and \( e'' \) fail:

\[
D_e^{e', e''} = v_e^{e'} + v_e^{e''} - z_e^{e', e''}. \tag{4}
\]

In other words, if \( D_e^{e', e''} \) is larger than \( v_e^{e'} \), if \( e' \) and \( e'' \) fail, then some demands cannot be restored on \( e \), because of insufficient backup bandwidth. Now, for each connection, we can define \( S_c = \{ e'' | \exists e' \in l^w, e \in l^b \land D_e^{e', e''} > v_e^{e'} \} \) as the set of links \( e'' \) whose failure causes a conflict on link \( e \) if two concurrent failures occur on \( e' \) and \( e'' \). \( S_c \) is formed by a series of links which are all disjoint from the backup and the working path of the connection.

In summary, if \( A_{S_c} = \prod_{e \in S_c} A_e \), the following equation can be applied to evaluate the availability of a SPP connection:

\[
A \approx A_{l^w} + A_{l^b} A_{S_c} - A_{l^w} A_{l^b} A_{S_c}. \tag{5}
\]

### V. Availability-Guaranteed SPP Provisioning Algorithm

As a first step to introduce the problem of Availability-Guaranteed SPP Provisioning, let us redefine the set of the existing connections \( L = \{ (l^w, l^b, s_d, t_a, t_b, SLA_i, A_i) \} \), where the sextuple specifies the working path, the backup path, the arrival time, the holding time, the stipulated availability target specified in the SLA, and the effective availability provided to a connection. Similarly, \( l = (l_a, l_b, SLA) \) defines the arrival time, the holding time and the stipulated availability target specified in the SLA of a new incoming connection. As a difference from the traditional provisioning approach, the working path \( l^w \) and the backup path \( l^b \) of the incoming connection must satisfy two additional conditions with respect to the other existing connections in \( L \):

- **C.5** The availability target \( SLA \) of the incoming connection must be guaranteed.
C.6 The availability target $SLA_i$ of the existing connections must be guaranteed.

So, once a couple of working and backup paths $l_w$ and $l_b$ has been found, (i) the availability $A$ supplied to the incoming connection must satisfy the availability target $SLA$; In addition, note that the provisioning of a new connection may reduce the availability of other existing connections due to the increased sharing of backup capacity. So, (ii) even if the SLA of one of the existing connections gets violated due to the increased sharing, then the incoming connection is blocked (even if its own availability meets the requirement). Otherwise, if two routes $l_w$ and $l_b$ are found in respect to the constraints C.1-C.6, the connection is provisioned and the link vectors along the routes are updated.

Several algorithms have been proposed to solve the availability-guaranteed SPP provisioning problem. In Alg. 1, we describe a baseline approach, called AGP, for availability-guaranteed SPP dynamic provisioning, which is a modified version of the algorithm in [2], that we will use as a holding-time-agnostic counterpart of our approach. In AGP, a connection can be either unprotected or shared-path-protected, such that its SLA requirement is met and network resource usage is minimized. The routing algorithm follows this two-step procedure: first the MRP is computed as the working path. If the SLA target is not met, then the connection is also provided by a shared-protected backup path. To compute the backup path, a new cost function $C_e = -log(A_e \times \alpha_e)$ is applied to each link $e$, so that the path with minimum cost will be the path with maximum availability. Note that $A_e$ is the availability of link $e$, while $\alpha_e$ represents the availability product of the links which, in case of double failure, contend on link $e$ backup resources with links belonging to the working path. If the backup wavelengths on a link $e$ can be shared by the incoming connection, $\alpha_e$ is evaluated considering only the $v^*_{e'}$ wavelengths already existing on link $e$ until the new arrival. Otherwise, a link is usable but not sharable, if no existing backup $v^*_{e'}$ wavelengths can be shared on it but there is at least one free wavelength on this link. In this latter case, $\alpha_e$ is evaluated considering $e$ as formed $(v^*_{e'} + 1)$ backup wavelengths. $\alpha_e$ is rigorously defined in Eqn. 7, where $W^*_{e'}$ denotes the number of free wavelengths on link $e \in E$.

VI. HOLDING-TIME-AWARE (HTA) AVAILABILITY TARGET REDEFINITION

AGP is an efficient holding-time-agnostic solution for the availability-guaranteed SPP provisioning problem. Let us now discuss what are the upgrades in order to obtain the corresponding holding-time-aware version and to show the benefits achievable by exploiting our new methodology.

In our holding-time-aware version of AGP, whenever a new connection is offered to the network, we can exploit a three crucial pieces of information: (i) the outage history of the connection (i.e. if an existing connection has been already subject to outages), (ii) for how long the connection is going to remain in the network, (iii) for how long the other existing connections are going to remain in the network. In the following we explain how the SP can redefine the connections’ availability target in presence of a network state change. Moreover, we present a methodology to evaluate, for a generic connection, its maximum future suppliable availability (i.e., the availability which, considering the connection departures, will be supplied if no new connections will share backup resources with that connection. Like in a feedback

The basic idea is that, with respect to a specific connection, SP may pass from situations of availability credit to conditions of availability debit, and viceversa, as long as the overall SLA target is guaranteed. There are four possible availability “transactions” that can be managed by the SP:

1) SP credit: in a classical availability-guaranteed provisioning approach, initially the SP is in credit with the customers. In fact, a connection is accepted only if the statistical estimation of the availability provided in the routing phase is larger than the stipulated availability. But topological constraints and link-availability granularity of the network force the SP to provide its customer a statistical availability level that can be sensibly higher than what is paid for.

2) Reducing SP credit: the SP has a means to provide an effective connection availability level closer to the stipulated value by decreasing step-by-step the availability target of a connection. E.g., if a connection has not been affected by failures, at the next network change
the working path is less than the availability target. Instead of providing a service outage at the working path, the SP can voluntarily provide an availability level lower than the targeted one, because it can not immediately guarantee an availability target. In both cases, the SP may take availability debit which could be extinguished by reserving more availability supply in the future (when other sharing connections will leave the network).

3) **SP debit**: two main causes can lead to an SP availability debit: one can be an outage affecting the connection and putting its SLA at violation risk; alternatively the SP can voluntarily provide an availability level lower than the targeted one, because it can not immediately guarantee an availability target. In both cases, the SP may take availability debit which could be extinguished by reserving more availability supply in the future (when other sharing connections will leave the network).

4) **Paying off SP debit**: a service interruption, if not opportunistically managed, may lead to an SLA violation. In accord to the AGP approach, at the time of an outage the SP is in availability debit only if it is guaranteed that, in the future, the required availability will be provided. In conclusion, the availability debit must be bounded by the maximum future suppiable availability for that connection.

### A. SP credit

Let us refer to the following example to show how an SP can exploit the connection-holding-time knowledge to take advantage of an availability credit. In Fig. 1, we show the state of a network (e.g. \( \forall c \in E: W_c = 8 \)) at an instant \( t_e = t_{15} = 10 \), when connection \( r_2 \) has to be provisioned between nodes \( E \) and \( F \) with \( t_{E,F}^f = 30 \). At time \( t_e = 10 \), a connection \( r_1 \) has already been routed into the network between nodes \( A \) and \( B \) at the instant \( t_e = 0 \) and it is characterized by an holding time \( t_h^i = 20 \). Both connections require an availability target \( SLA = SLA_1 = SLA_2 = 0.99 \).

In accord to the AGP approach, at the time \( t_e = 0 \), we have fixed the route of the working path of connection \( r_1 \) along the MRP (link A-B); as the second step, since the availability \( A_1 \) provided by the working path was less than the availability target \( SLA_1 \), a backup path was routed on nodes A-C-D-B (dashed line in figure) utilizing the link cost assignment in Eqn. 6. Since \( r_1 \) and \( r_2 \) share a wavelength on link C-D along their backup paths, connection \( r_2 \) is accepted because condition C.5 and C.6 are both not violated, i.e. the availability target of connection \( r_2 \) is respected and the availability target of connection \( r_1 \) is still guaranteed. We recall that backup sharing reduces the availability of a SPP connection. In this case, utilizing the traditional approaches, both of the two previous conditions are not respected, because \( A_1 = 0.98945 < SLA_1 = SLA_2 \). In this case \( r_2 \) request can be either refused or dedicated protection have to be utilized, which will produce an expensive resource consumption.

**Fig. 1.** In this simple network example, traditional availability evaluation fails to route the second connection, while the availability target redefinition succeeds.

However, we can notice that connection \( r_1 \) has not been affected by failures during its previous lifetime, from time 0 to time 10. Then, a new availability target \( SLA_i \) for \( r_1 \) can be set, taking into account that from \( t_{15} \) to \( t_e \), the connection has been served with availability \( A_i^t = 1 \) and that the connection will remain in the network from \( t_e \) to \( t_{15}^i + t_{h}^i \). In general, the availability target can be redefined in the following manner:

\[
SLA_i = \frac{SLA_1 \cdot t_{15}^i + A_i^t (t_{h}^i - t_e)}{t_{15}^i + t_{h}^i - t_e}
\]  

(8)

and a new target \( SLA_i \) can substitute the previous \( SLA_1 \) target that we were using during the check of condition C.6 in Alg.1. In this specific case, the availability target of connection \( r_1 \) will be reduced, i.e. the SP’s availability credit is decreased, given that \( r_1 \) has not been subject to any service outage. As result, \( SLA_1 = 0.98 < A_1 = 0.98945 \) and condition C.6 is now respected. However, connection \( r_2 \) could not be accepted because condition C.5 is still violated.

### B. SP debit

The availability redefinition reported in Eqn. 8 may also be utilized when the SP goes in availability debit due to network failures. In this case, applying availability redefinition allows
to increase the availability target achieving more conservative treatment of the connection and, in turn, reducing the availability debit. The SP will still accept new connections, but it will avoid that their backup paths share backup resources with existing connections which have experienced an outage.

However, as mentioned before, an SP’s availability debit may be induced voluntarily to accept more connections in the network; in the following we provide an example of a voluntarily induced availability debit. In order to follow the time evolution of of the network links’ state, we introduce the new symbols \( \nu_e(\Delta t_k), \nu_e'(\Delta t_k), z_e^{(\Delta t_i')} (\Delta t_k) \) and \( \tilde{D}_e^{(\Delta t_i')} (\Delta t_k) \) which express the values of \( \nu_e, \nu'_e, z_e \) and \( D_e \) respectively, in the time interval \( \Delta t_k \).

Let us expressly define \( \Delta t_k \) first. According to connection holding times, the \( t_i \)'s can be ordered so that \( t_i^1 + t_i^h \leq t_i^{i+1} + t_i^{h+1}, i = 1, 2, \ldots, |L| \). As a consequence, \( \tau = \{ t_0^1, \ldots, t_0^{|L|} \} = \{0, t_0^1 + t_0^1, t_0^2 + t_0^2, \ldots, t_0^{|L|} + t_0^{|L|}\} \) will indicate the departure events and \( \Delta t_k = t_{k-1} - t_k \) expresses the time interval between two departures. \( \nu_e(\Delta t_k), \nu_e'(\Delta t_k), z_e^{(\Delta t_i')} (\Delta t_k) \) and \( \tilde{D}_e^{(\Delta t_i')} (\Delta t_k) \) will be updated according to the \( k \)-th connection departure. In other words, we have divided the time into a series of intervals \( \Delta t \) which express the distance between two departures. In Fig. 2 we focus on the departure events on link C-D of the network in Fig.1, assuming that also connection \( r_2 \) has been provisioned:

![Fig. 2. Time evolution on link C-D of the network in Fig.1.](image)

- \( \Delta t_1 \) (from time 10 to time 20): backup paths of connection \( r_1 \) and connection \( r_2 \) share a wavelength on link C-D. During this time interval the provided availability \( A_1(\Delta t_1) = A_2(\Delta t_1) \) is low and equal to 0.98945.

- \( \Delta t_2 \) (from time 20 to time 40): \( r_2 \) has a dedicated resource on link C-D because connection \( r_1 \) has left the network. In this time interval the available availability \( A_2(\Delta t_2) \) is equal to 0.99152.

As examined in previous section, during \( \Delta t_1 \), SP can reduce its availability credit with connection \( r_1 \). Moreover, at the same time interval, SP can go voluntarily into debit with \( r_2 \) because it will be paid off during the \( r_2 \)'s residual lifetime, i.e., \( \Delta t_2 \). Likewise, the SP will be able to guarantee an overall availability of \( A_2 = 0.99083 \) and also connection \( r_2 \) will be accepted.

More generally, the state of a link can vary in time, passing, e.g., from shared to dedicated, and the availability provided to the connection could consequently change. Each of these availability contributions \( A_2(\Delta t) \) can then be weighted proportionally over each time interval according to the following equation:

\[
\tilde{A}_i = \sum_{\Delta t_k} A_i(\Delta t_k) * \Delta t_k \frac{t_i^h + t_i^h - t_c}{t_i^h + t_i^h - t_c},
\]

where the new \( \tilde{A}_i \) can substitute the previous \( A_2 \) in conditions C.5 and C.6 of Alg.1. In other words, \( \tilde{A}_i \) expresses the maximum supplied availability for the connection \( r_i \) if its backup path will be not shared with any other future incoming connections. This condition can be easily enforced by preventing the backup paths of future incoming connection from sharing backup capacity of connection \( r_i \).

Exploiting this holding-time-aware (HTA) approach [4], the SP is able to accept more connections into the network and violates less SLA availability targets by dynamically adapting them to network state evolution. Finally, note that our approach does not re-evaluate the availability status of all existing connections in the network, but it is applied only to the incoming connection and to connections that share backup resources with it. This, for common network scenarios, relevantly limits the computational complexity.

VII. ILLUSTRATIVE NUMERICAL EXAMPLES

We now quantitatively evaluate the performance of the two approaches: (1) AGP, the holding-time-unaware provisioning approach with no availability updating, and (2) HTA, the holding-time-aware provisioning approach with availability target redefinition. 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. Average connection-holding time is normalized to unity. For the illustrative results shown here, in every experiment, 10⁵ connection requests are simulated. All the plotted values have a 95% confidence interval not larger than 5% of the plotted value. Requests are uniformly distributed among all node pairs; availability requirements of the requests are uniformly distributed over the three classes \( \{0.99, 0.999, 0.9999\} \), denoted as \( C_1, C_2, C_3 \), respectively.

The example network topology with 32 wavelengths per fiber is shown in Fig. 3. The MTTR is considered constant and normalized to 0.032⁴, while MTBF is a Poisson distribution that guarantees a link availability value of 0.999.

We employ four performance metrics: Blocking Probability, SLA Success Ratio, Resource Distribution and Availability Gap.

A. Blocking Probability

The Blocking Probability (BP) indicates the ratio of the blocked connections over the offered connections to the network. Exploiting the HTA approach, the SP will be able to accept more connections into the network, because, by

3Note that, in order to pay back its debit, usually the SP needs to supply a future availability lower than the maximum future supply availability derived by Eqs.9. As a matter of fact, the debit may be dynamically compensated by iterative credit reductions if a connection has not experimented any outages.

4With an average connection holding time of 15 days, the MTTR results equal to 12 hours.
periodically redefining the availability target of a connection, more effective backup sharing is allowed in the network. Tab. I compares the BP achieved by our previous approach reported in [16] and HTA with AGP. The approach in [16] outperforms AGP, especially for high loads: e.g., at around 100 Erlangs, BP decreases from 17% to 12%. Further, the HTA approach presented in this paper further reduces the BP in respect to the previous approach reported in [16]: at around 100 Erlangs, BP decreases from 12% to 7%. The connection blocking may be due to four different causes: lack of resources, violation of condition C.5, violation of condition C.6, and violation of both conditions C.5 and C.6. Figures 4 and 5 show the impact of the various contribution to BP in the AGP and HTA approach, respectively, for increasing traffic load. Noth that in AGP approach, the inability to guarantee the SLA availability target to existing connections is the main cause of connection blocking (violation of condition C.6). As show in Fig. 5, HTA outperforms AGP, because it drastically reduces the blocking due to violation of condition C.6.

### B. SLA Success Ratio

In Fig. 6 we show the ratio of the connections which violate the SLA target over the accepted connections to the network. Our numerical results demonstrate that the number of violations is much lower when HTA is used. For the sake of conciseness, we report only the case for the SLA Class 3, i.e., the class of connections which always require a backup path. Note that this metric may not result in a fair comparison because different schemes will accommodate different requests, but this measure does not differentiate one request from another.

### C. Resource Distribution

In order to investigate the fairness of these approaches, we now evaluate the resource distribution among different classes in AGP and HTA. Connections can be provisioned in two different ways: unprotected or shared-protected. We observe that all the connection requests in SLA Class 1 are unprotected, and all the connection requests in SLA Class 3 are shared-path protected. The proportion of shared-path-protected connection increases with the increase in traffic load. In Fig. 7 it can be seen that, with AGP approach, increasing the network load the percentage of unprotected connection for SLA Class 2 is incremented: AGP encourages the routing of connection which does not require a backup path and block the other connections, i.e. connections that can not meet its SLA target with only a provisioned working path. On the contrary, using HTA approach the percentage of unprotected connection is almost constant with different traffic loads. So, HTA is able to better distribute the backup sharing during connection lifetimes, protecting a higher number of connections.

### D. Availability Gap

The relevant BP decrement shown in Tab. I can be motivated also looking at the reduction of the gap between the stipulated

---

**TABLE I**

<table>
<thead>
<tr>
<th>Arrival Rate</th>
<th>20</th>
<th>40</th>
<th>60</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGP</td>
<td>0.526</td>
<td>4.38</td>
<td>9.041</td>
<td>14.092</td>
<td>17.677</td>
</tr>
<tr>
<td>[16]</td>
<td>0.250</td>
<td>2.567</td>
<td>5.745</td>
<td>9.034</td>
<td>12.270</td>
</tr>
<tr>
<td>HTA</td>
<td>0.151</td>
<td>1.028</td>
<td>2.6190</td>
<td>4.9650</td>
<td>7.2391</td>
</tr>
</tbody>
</table>

**Fig. 3.** A carrier’s US nationwide backbone network topology.

**Fig. 4.** Blocking Probability for AGP.

**Fig. 5.** Blocking Probability for HTA.
SLAs and the actual value of the availability $A$ provided to connections in the AGP and HTA scenario. In Tab. II, we compare the average stipulated $SLA_i$ (which is the SP objective and it would be reached if all the connections $i$ in the network would be provided exactly with $A_i = SLA_i$), with the actual average availability provided by HTA ($A_{HTA}$) and AGP ($A_{AGP}$). It can be seen that values for $A_{HTA}$ are much closer to the $SLA$ target than those of $A_{AGP}$.

### Table II

<table>
<thead>
<tr>
<th>Arrival Rate</th>
<th>Average Stipulated SLA</th>
<th>AGP</th>
<th>HTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.9963</td>
<td>0.9989</td>
<td>0.9988</td>
</tr>
<tr>
<td>40</td>
<td>0.9968</td>
<td>0.9991</td>
<td>0.9987</td>
</tr>
<tr>
<td>60</td>
<td>0.9963</td>
<td>0.9995</td>
<td>0.9982</td>
</tr>
<tr>
<td>80</td>
<td>0.9963</td>
<td>0.9989</td>
<td>0.9984</td>
</tr>
<tr>
<td>100</td>
<td>0.9963</td>
<td>0.9984</td>
<td>0.9984</td>
</tr>
</tbody>
</table>

### Fig. 6
Percentage of violated SLA targets for SLA Class 3: AGP vs. HTA.

### Fig. 7
Unprotected connections percentage for SLA Class 2: AGP vs. HTA.

## VIII. Conclusion

In this paper we have proposed a mechanism for availability-guaranteed SPP that allows to “trade” availability “credits” and “debits”, by increasing or decreasing the shareability level of the shared backup capacity. We have shown that our approach allows to obtain relevant improvements on blocking probability and reduce SLA target violations, updating the availability targets and supplying an availability level that follows the network changes.

## References


