ARTICLE IN PRESS

Optical Switching and Networking ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Optical Switching and Networking

journal homepage: www.elsevier.com/locate/osn

Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks

Juliana de Santi [a], André C. Drummond [b], Nelson L.S. da Fonseca [a,*], Xiaomin Chen [c]

[a] Institute of Computing, University of Campinas, Av Albert Einstein 1251, 13083-852 Campinas, SP, Brazil
[b] Department of Computer Science, University of Brasília, Brazil
[c] Institute of Computer and Network Engineering, Technische Universität Carolo-Wilhelmina zu Braunschweig, Germany

ARTICLE INFO

Article history:
Received 23 May 2014
Received in revised form
16 October 2014
Accepted 22 October 2014

Keywords:
Traffic grooming
Holding time awareness
Load balancing
Multipath routing
WDM

ABSTRACT

This paper investigates approaches for the traffic grooming problem that consider connection holding-times and bandwidth availability. Moreover, solutions can indicate the splitting of connections into two or more sub-streams by multipath routing and fine-tuned by traffic grooming to utilize network resources better. Algorithms are proposed and the results of simulations using a variety of realistic scenarios indicate that the proposed algorithms significantly reduce the blocking of connection requests yet promote a fair distribution of the network resources in relation to the state-of-the-art solutions.

1. Introduction

Bandwidth can be allocated in Wavelength Division Multiplexing (WDM) networks only in fixed amounts, rarely matching the diversity of bandwidth demands of applications. Most Internet Protocol (IP) streams require less bandwidth than the capacity of a wavelength (sub-wavelength), typically one order of magnitude less, which has motivated the adoption of traffic grooming techniques to aggregate several small flows on a single wavelength. Various approaches have been employed to groom such subwavelength streams on existing lightpaths in an attempt to promote low blocking of requests. One of these is load balancing, which can be used to avoid bottlenecks, leaving a larger number of lightpaths with residual bandwidth to be potentially allocated to incoming requests. Another consideration is the duration of the connection (holding-time) with information about it generally available in the Service Level Agreement (SLA) of requests for the establishment of connections. This information can be used to determine the wavelength to host a stream. When the duration of connections are known, the existing lightpaths can be selected by considering their lifetime (LT) to accommodate new connection requests, with the aim of increasing the availability of wavelengths to serve future connection requests.

Moreover, multimedia applications employing new high-resolution digital media formats (i.e., 4K and 8K formats) [1] and those of e-Science [2] often can generate streams which demand bandwidths larger than the capacity of a single wavelength (superwavelength). To accommodate such demands, the traffic must be split and transmitted on more than one wavelength. This division employs multipath routing which determines the routes for the sub-stream; this is usually

* Corresponding author. Tel.: +55 19 3521 5878; fax: +55 19 35215847.
E-mail addresses: santi@ic.unicamp.br (J. de Santi), andred@unb.br (A.C. Drummond), nfonseca@ic.unicamp.br (N.L.S. da Fonseca), chen@ida.ing.tu-bs.de (X. Chen).

http://dx.doi.org/10.1016/j.osn.2014.10.002
1573-4277/

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002

groomed with other streams carried on a given wavelength. Actually, a single superwavelength stream can be carried over multiple parallel routes (multipath routing), if the sum of the residual bandwidths of all paths connecting source to destination is at least equal to the bandwidth demands of the original stream. Fig. 1 illustrates how the ability to groom and split traffic streams can be employed to exploit the residual capacity, thus accommodating both subwavelength and superwavelength requests. It shows a connection request (R) for a typical 3D HDTV stream with a bandwidth requirement of 6 Gbps and a scenario in which no single path can accommodate the request. However, the total residual capacity of two established lightpaths, i.e., P1 and P2, is 7 Gbps, which can accommodate the sub-streams of the original stream.

This paper aims to investigate various approaches used in the holding-time-aware dynamic traffic grooming problem, especially focusing on load balancing over parallel paths with a residual capacity, a problem which has not yet been addressed in the literature. We propose a novel heuristic algorithm which can effectively balance the network load by utilizing residual capacity, referred as the *HTBalancing algorithm*. Moreover, we propose an Integer Linear Programming (ILP) formulation to provide optimal aggregation of the residual capacity of parallel routes when both subwavelength and superwavelength demands are involved, referred as *MPHTBalancing*. We also propose an approximation of the MPHTBalancing algorithm based on the linear relaxation technique which is a heuristic that aggregates both subwavelength and superwavelength demands and reduces the time required to solve the proposed ILP.

Numerical results show that the HTBalancing algorithm significantly reduces the blocking of connection requests while promoting a fair distribution of resources among source destination pairs. Moreover, the MPHTBalancing algorithm balances the load and reduces blocking for scenarios with both subwavelength and superwavelength demands as well as produces low differential delay values. A thorough evaluation considering a variety of scenarios was conducted. Results show that the proposed heuristic for the MPHTBalancing algorithm is able to reduce the execution time with only minor impact on the quality of the solutions, except for low loads.

The remainder of this paper is organized as follows. Section 2 gives an overview of related work. Section 3 introduces the HTBalancing and MPHTBalancing algorithms as well as a heuristic for the MPHTBalancing algorithm. Section 4 presents a numerical evaluation of the proposed algorithms. Section 5 concludes the paper.

2. Related work

Information on the duration of connections (holding-time) has been exploited for the development of traffic grooming algorithms in an attempt to satisfy customer expectations while increasing the capability of connection provisioning and the ability to efficiently use network resources [3,4].

The seminal work in [5] employs information on the duration of the connections to solve the dynamic traffic grooming problem and introduces the Holding-Time-Aware (HTA) algorithm, which tries to match the lightpath lifetime and the duration of the incoming connections to define the lightpath to which the connection should be aggregated. The goal is the minimization of the number of new lightpaths created, as well as the lengthening of the lifetime of existing lightpaths. In [6], an extension of the HTA algorithm was proposed to balance the distribution of connection requests among network links. The algorithm Adaptively Weighted Links for Holding-Time-Aware (AWLHTA) [7] extended the work in [5,6]. It aims at promoting load balancing and avoiding the creation of network bottlenecks. Routing and allocation decisions are based on link weight values, calculated by considering the connection duration, the network load and the available resources (wavelengths and transceivers). The evaluation of AWLHTA considers restrict scenarios that include a specific topology, wavelength conversion and a low number of wavelengths by fiber. Such scenarios are too specific

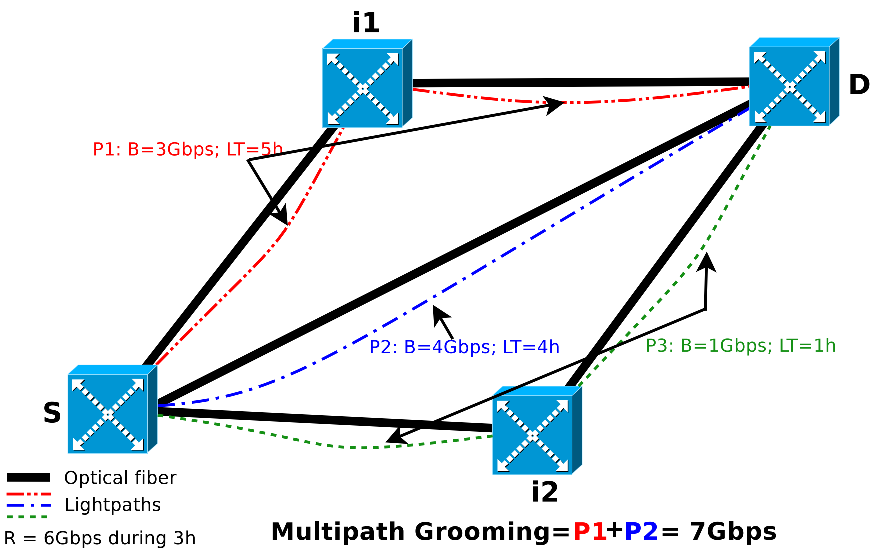

Fig. 1. Multipath grooming example. R requesting 6 Gbps is accepted by traffic grooming on the residual capacity on routes P_1 and P_2.

and results derived for these scenarios are limited and, therefore, cannot be employed for generalizing any conclusion about the performance of the algorithm. Different from the investigation in [7], the network topologies and the scenario employed in the present paper are realistic and widely used in the literature.

Information on holding time has also been used jointly with other types of time information to efficiently fulfill request demands. The work in [8] employs a sliding window model to meet the demands. Such a combination of information was also employed in [9], which considers the static provisioning problem as a mixed partitioned coloring problem. The matching of holding-time and starting time flexibility was considered in [10,11] in an attempt to improve the scheduling of connections on WDM links. In [12], the impact of flexibility of the starting time of connections on traffic aggregation was investigated under the effects of physical impairments. In [13], algorithms were proposed to groom batches of incoming requests by employing time information contained in the Service Level Agreement (SLA).

Traffic grooming based on the duration of connections has also been used to minimize energy consumption [14–16] and to support multicast requests [17]. A brief survey of connection provisioning in Optical Circuit Switched networks that considers information about timing is presented in [18].

Another issue that has been intensively investigated is the use of multiple routes for the establishment of connections. In [19], multipath routing was proposed to cope with superwavelength requests. This was the first formulation using multipath routing and the wavelength assignment problem was based on an ILP formulation. The solution significantly decreased the blocking of requests while increasing the availability of connections to accommodate incoming requests.

The work in [20] considered inter-domain multipath routing with an aggregated virtual topology for efficient resource usage, and scalable provisioning. It also investigated load balancing by splitting the traffic into real-time streaming and bulk data transfer applications.

In [21], multipath routing was extended to the Carrier Ethernet Networks. Two end-to-end multipath routing algorithms and a per-domain multipath routing that considers the requirements of buffering for packet reordering were introduced.

Motivated by the need for efficient connection provisioning and resource utilization, multipath routing has been employed to groom requests. The work in [22] investigated the impact of uneven splitting of connections on the multipath traffic grooming problem. In [23], an analysis of IP traffic grooming in different architectures considering the impact of traffic splitting and multipath routing was realized. In [24], incoming requests are aggregated in multiple existing lightpaths of dynamic optical circuit switching.

In earlier work [6], we proposed an algorithm, referred to as *HTBalancing*, to balance the network load among existing lightpaths based on the duration of requests. In this paper, we extend this approach to show the benefits of utilizing multiple parallel routes for holding-time-aware traffic grooming, especially in dynamic scenarios with mixed demands of sub- and superwavelength connections.

3. Holding time aware traffic grooming

The problem of dynamic holding time aware traffic grooming involves a network (physical as well as virtual) configuration and the dynamic arrival of requests for the establishment of connections. Connection requests must be provisioned immediately. Requests are represented by $d_i(s, d, b, H)$ in which s represents the source, d the destination and b the units (OC) of bandwidth required during H units of time.

This section introduces an algorithm for dynamic holding time aware traffic grooming. The algorithm balances the load among network paths, considering the duration of the connections as well as the bandwidth availability. This algorithm is employed in two different scenarios. In Section 3.1, subwavelength connections and single path routing are considered. In Section 3.2, a mix of subwavelength and superwavelength connections, as well as multipath routing, is considered. In Section 3.4, a relaxed version of the load balancing algorithm is proposed to accelerate the solution of the problem.

3.1. HTBalancing algorithm

Information about the duration of a connection can be used to determine the ending time of the connection [3], and the lifetime of a lightpath is determined by the last ending time of the connections using this lightpath. Moreover, such information can be used to decide whether or not an incoming connection should be accepted. The aim is the maximization of the utilization of lightpaths and the minimization of the blocking probability for future demands [5,25,26]. The main idea of the approach is to use the minimum possible number of new lightpaths, which will also involve a reduction in the length of the lifetime of existing lightpaths.

The holding-time-aware traffic grooming algorithm (HTA) [5] determines the lightpath on which a connection will be groomed by calculating the cost related to extension of the lifetime of these lightpaths, an operation which involves taking into consideration both the duration of the connection and the lifetime of the lightpaths, as follows [5]:

$$c\_ht(p_i) = \begin{cases} hp_i \times H & \text{if new path} \quad \text{(a)} \\ hp_i \times \epsilon & \text{if } LT_i \geq H \quad \text{(b)} \\ hp_i \times \epsilon + hp_i \times \Delta_t & \text{if } LT_i < H \quad \text{(c)} \end{cases} \tag{1}$$

where

- $c\_ht(p_i)$ is the cost, which considers the extension of the lifetime, of the ith lightpath;
- H is the holding-time of the request;
- LT_i is the lifetime of the ith lightpath;
- $\Delta_t = H - LT_i$;
- hp_i is the number of hops along the ith lightpath.
- $\epsilon = 10^{-5}$ (a small constant) [5];

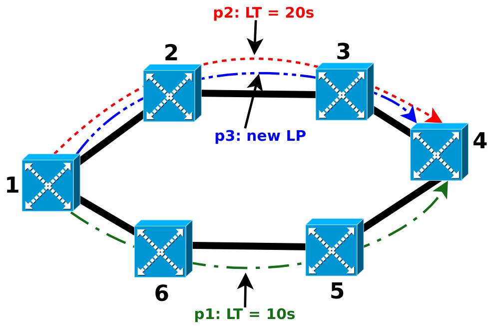

Fig. 2. Example of lightpath cost calculation based on holding-time of connection

Eq. (1a) defines the cost of establishing a new lightpath. In this case, it is necessary to pay for the lightpath establishment and for the transmission of the connection during H units of time using the links hp_i of the lightpath. The longer the lightpath, the greater is the cost of utilization.

Already established lightpaths cost less than do those yet to be created, since there is an overlap between the lifetime of these lightpaths and the duration of incoming connection. A complete overlap in relation to the holding-time of the connection implies a low cost for the establishment of the request on that lightpath (Eq. (1b)). On the other hand, when connection will last longer than the lifetime of the established lightpath (Eq. (1c)), the lifetime of the lightpath must be extended by Δ_t units of time to accommodate the request.

Fig. 2 illustrates the computation of the cost of connection establishment. An arriving request $d_1(1,4,2,30)$ requests 2 Mbps of bandwidth for 30 s (holding-time) for the source–destination pair (1–4). The paths p_1, p_2 and p_3 have, respectively, 4 Mbps, 6 Mbps and 1 Gbps and all are candidates to support d_1. Paths p_1 and p_2 are already established and have lifetimes of 10 s and 20 s, respectively. One possibility would be grooming the request d_1 on the lightpath p_1, which would imply extending its lifetime by 20 s. The cost of this provisioning is the cost of the utilization of three links $1\rightarrow6$, $6\rightarrow5$ and $5\rightarrow4$ multiplied by the 20 s ($3\times20=60$). Another possibility would be grooming on p_2. Although this would require the extension of 10 s, the cost of this utilization would be the cost of the use of the links of p_2 ($1\rightarrow2$, $2\rightarrow3$, $3\rightarrow4$) multiplied by the additional time needed ($3\times10=30$). The third option, the creation of p_3, would imply on the cost of its establishment using the links $1\rightarrow2$, $2\rightarrow3$ and $3\rightarrow4$ for 30 s is ($3\times30=90$). Thus, the lightpath which would minimize the cost of accepting request d_1 is p_2.

Associating low costs with lightpaths having longer lifetimes may exhaust the capacity of these lightpaths, however, due to unbalanced traffic. Fig. 3 illustrates the consequences of unbalanced traffic distribution. (S_1,D_1), (S_2,D_2) and (S_3,D_3) are source–destination pairs; the link $6\rightarrow7$ has two units of bandwidth available and all the other paths have three units. At time t_0, a new request $d_1(S_3,D_3,2,45)$ arrives. The lightpaths p_1 (routes 1, 6, 7, 4) and p_2 (routes 1, 2, 3, 4) are already established, with lifetimes of 40 s and 30 s, respectively. Considering only the holding time as a criterion for the calculation of the lightpath cost (Eq. (1)), lightpath p_1 would be chosen to support the connection d_1, since its lifetime is the longest. This decision would exhaust the available capacity of the link $6\rightarrow7$ and, as a consequence, all incoming connections for the (S_1,D_1) and (S_2,D_2) source–destination pairs would be blocked since the link $6\rightarrow7$ is part of a critical path.

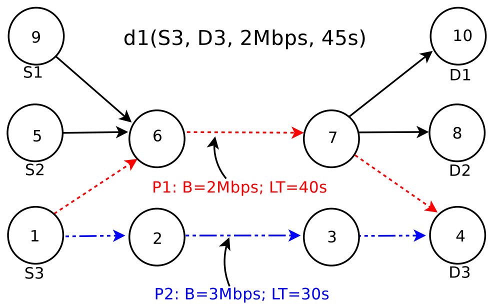

Fig. 3. Illustration of unbalanced network load.

To avoid this situation, we propose an algorithm called Holding-Time-Aware-with-Traffic-Balancing (HTBalancing), which balances the incoming connections among existing lightpaths so that the blocking of future demands can be minimized, while distributing it fairly among source destination pairs.

In order to promote network balancing, the HTBalancing algorithm aggregates connections with a known duration, taking into account a cost function that considers both the extension of the lifetime of the lightpath and the bandwidth availability of the lightpaths. This cost function is given by

$$C(p_i) = (c\_ht(p_i) \times \alpha) + \left(\frac{B(p_i) - b(p_i)}{B(p_i)} \times (1-\alpha)\right) \tag{2}$$

where

- $C(p_i)$ is the cost of utilization of lightpath p_i;
- α determines the weight applied to the factor extension of the lifetime of the lightpath in the cost function;
- b is the available bandwidth in p_i;
- B is the total bandwidth of p_i;
- $(1-\alpha)$ determines the weight of bandwidth utilization.

The first term of Eq. (2) considers the holding-time criterion in the selection of lightpaths and the second term the utilization of the lightpaths, thus, implying that lightpaths with small bandwidth availability will have a higher cost and discouraging their utilization to avoid the creation of bottlenecks. Lightpaths with large bandwidth availability, however, have a lower cost and are thus preferable for serving incoming requests. In Section 4, the impact of the usage of different α values is evaluated.

The HTBalancing algorithm is presented in Algorithm 1. Upon arrival of a connection request $d_j(s,d,b,H)$, the auxiliary graph G is constructed by including lightpaths p_i with residual capacity ($b(p_i)$) equal or greater than the required bandwidth b (Line 1). The lightpath p_i can be an already established lightpath or a new lightpath. The cost of the utilization of lightpaths in G is then computed (Lines 2–7). To avoid the creation of unnecessary lightpaths, the existing bandwidth availability is evaluated. In Line 4 of the algorithm, the cost of existing connections is accounted and, in Line 7 the cost of those lightpaths that can potentially be created is considered.

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002

Algorithm 1. *HTB*alancing.

Require: Demand $d_j(s,d,b,H)$ requesting b bandwidth units during H units of time between (s,d)
Ensure: Feasible path between s and d to aggregate d_j
1: Construct the auxiliary graph G with $p_i | b(p_i) \geq b$
2: **for all** (candidate lightpaths in G **do**
3: **if** p_i is already established **then**
4: $C(p_i) \Leftarrow (c\_ht(p_i) \times \alpha) + \left(\frac{B(p_i) - b(p_i)}{B(p_i)} \times (1-\alpha)\right)$ (Eq. (2))
5: **else**
6: **if** p_i is a new lightpath **then**
7: $C(p_i) \Leftarrow (c\_ht(p_i) \times \alpha) + (1 \times (1-\alpha))$
8: **end if**
9: **end if**
10: **end for**
11: Apply Shortest-path to determine the lowest cost lightpath

The cost of each feasible optical lightpath is calculated, and the labels of the edges representing the network weight values $C(p_i)$ are updated. Then, a traditional Shortest-path algorithm (Line 8) is employed to determine the lightpath with lowest cost.

The computational complexity of the HTBalancing algorithm can be analyzed as follows. The construction of the auxiliary graph involves $O(V^2)$ operations, where V is the number of OXC[1] in the network. Line 2 of the HTBalancing algorithm is executed $|E| = |W|$ times, since the computation of the cost is applied to each candidate lightpath (edge(E)) in the graph, i.e., all wavelengths (W) can be visited in the worst situation. Thus, the complexity of the computation of the cost is $O(E) = O(W)$. To find the lightpath with lowest cost (Line 8), the Dijkstra algorithm is used. Since the complexity of the Dijkstra algorithm is $O(W + V \log V)$ the computational complexity of the HTBalacing algorithm is $O(V^2 + W + W + V \log V) = O(V^2)$.

3.2. MPHTBalancing algorithm

The stream of superwavelength requests can be carried on multiple lightpaths using different wavelength on the same fiber link, i.e., using single path routing. The same stream could be carried out using different fiber links and multiple routes by employing *multipath routing* which can also be used to decrease the blocking caused by bandwidth unavailability in a single route. Indeed, the mechanism introduced in [19] was one of the very first attempts to use multipath routing to cope with superwavelength requests.

However, the lightpaths established to compose a multipath solution may not occupy the full capacity of the reserved wavelength, leading to inefficient utilization of resources. For instance, if a connection request demands a bandwidth equivalent to the capacity of one and a half wavelength, two wavelengths are allocated, leading to only 75% utilization of the optical circuits. Moreover, most of the connections in operational optical networks require bandwidths smaller than the capacity of a wavelength, i.e., they require granularity of subwavelengths which leaves plenty of available capacity on the established lightpaths. If it is impossible to find a path with enough capacity even for subwavelength connections, the overall residual subwavelength capacity can be used to create a multipath connection from source to destination for these requests.

Since multipath routing is essential for the provisioning of superwavelength connections, but not necessarily for the promotion of efficient utilization of optical resources, a novel algorithm is proposed to balance the load in multipath routing by using dynamic traffic grooming. In this approach, information about bandwidth availability and holding time awareness are employed to avoid the formation of bottlenecks. A new MPHTBalancing (MPHTB) algorithm was thus designed to furnish efficient bandwidth provisioning for both superwavelength and subwavelength requests.

The network used as input for the ILP formulation is represented by a graph $G(V,E)$, where V is the set of nodes and E is the set of fiber links. A capacity $c_{i,j}$, given by the sum of bandwidth in the available wavelengths ($W_{i,j}$) in the fiber link, is associated with each link $e \in E$. A loop free path p in G is defined as a list of nodes $(s, v_1, \ldots, v_n, d)$, where s, d, $v_i \in V$ and (s, v_i), (v_i, v_{i+1}), (v_n, d) $\in E$, s and d are the source and destination, respectively. The virtual topology, which describes the already established lightpaths and their available bandwidths, is given as input to the ILP.

The proposed algorithm tries to find multiple routes (multipaths) so that a requested connection can be provided at a minimum cost. To formulate the ILP, the following notation is defined:

- $F_{p,i,j,w} \in \{0,1\}$: a binary variable with a value of one if the wavelength w in the link i,j is used by the path p; otherwise it is zero;
- $X_{p/lp,w} \in \{0,1\}$: a binary variable with a value of one if the wavelength w is used by a lightpath; otherwise the value is zero. This lightpath can be a new lightpath p or an already established lightpath lp;
- λ: wavelength capacity;

The following values are also furnished as input to the ILP:

- C_{lp}: the cost of an already established lightpath lp, defined by Eq. (2);
- B_{lp}: the available bandwidth in the already established lightpath lp;
- b: the bandwidth demand of a request expressed as the number of the OC carrier required (A "3" will refer to the carrier OC-3; a "12", to the carrier OC-12);
- H: connection holding time;
- α, a constant used to calculate the cost of a lightpath to be established.

Algorithm 2. *MPHTB*alancing (ILP).

$$\text{Min} \sum_{lp,w} X_{lp,w} \times C_{lp} + \sum_{p,w} \left(X_{p,w} \times (1-\alpha) + \sum_{i,j} F_{p,i,j,w} \times H \times \alpha \right) \quad (3)$$

$$\sum_i F_{p,i,j,w} - \sum_k F_{p,j,k,w} = \begin{cases} -X_{p,w} & \text{if } i = s \\ X_{p,w} & \text{if } i = d \ \forall j,w,p \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$\sum_{lp,w} (X_{lp,w} \times B_{lp}) + \sum_{p,w} X_{p,w} \times \lambda \geq b \quad (5)$$

[1] Optical Cross-Connect.

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002

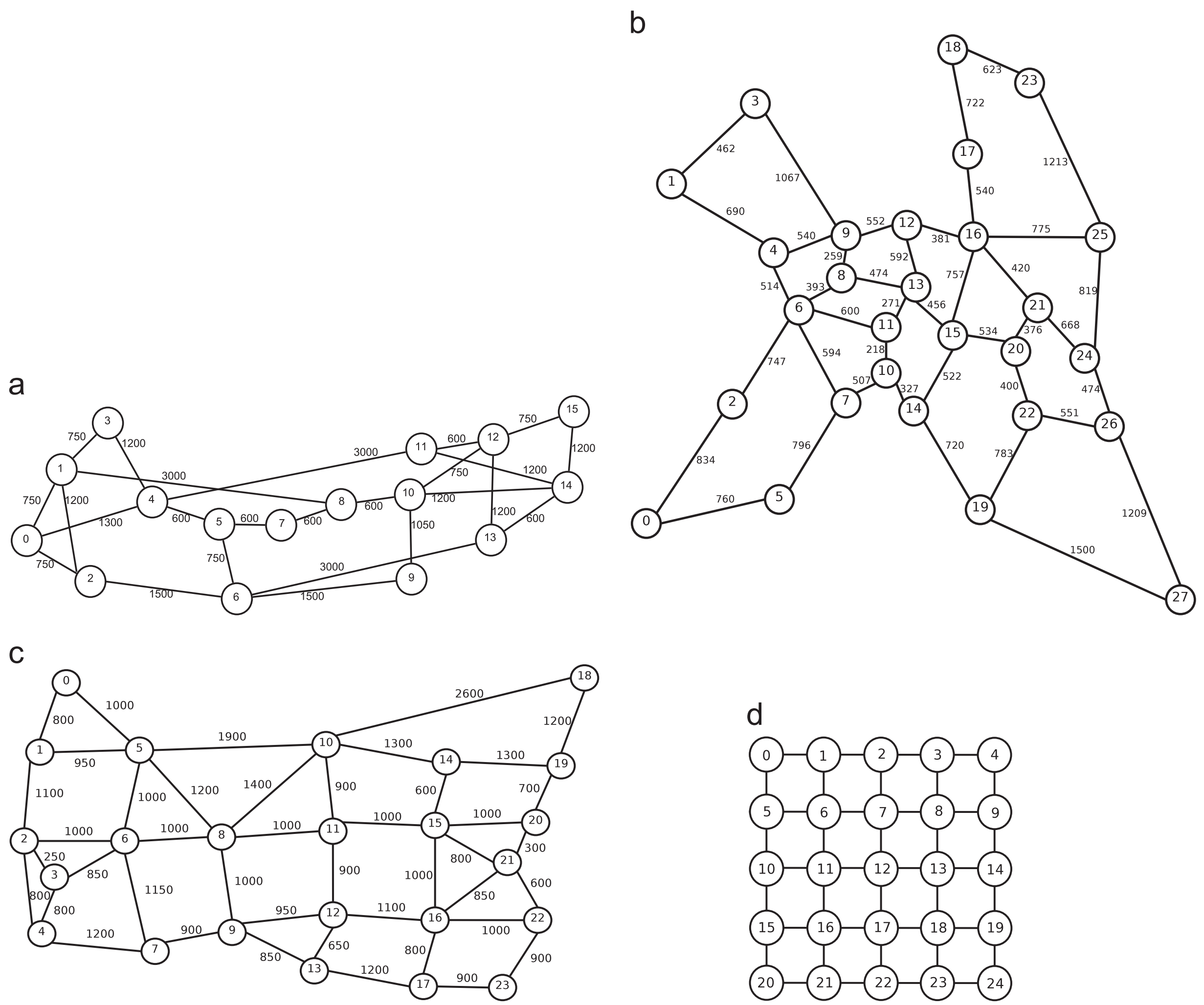

Fig. 4. Networks. (a) NSF, (b) Pan-European, (c) USA, and (d) Grid 5×5.

$$\sum_p F_{p,i,j,w} \le 1 \quad \forall \quad w,i,j \tag{6}$$

where p and $lp \in P$; $w = [1, 2, \dots W_{i,j}]$, $i,j \in E$.

The objective function (Eq. (3)) tries to find paths between a source s and a destination d with the minimum costs, taking into consideration the connection holding time and the bandwidth available on the paths. The first term of the objective function accounts for already established lightpaths with costs defined by Eq. (2), and the second term considers the cost of new lightpaths. For new lightpaths, the cost is computed by replacing $(B-b)/B$ in Eq. (2) with 1 and $c\_ht(p_i)$ with Eq. (1), i.e., $c\_ht(p_i) = hp_i \times H$, which results in a cost of $C(p_i) = ((1-\alpha) + (hp_i \times H \times \alpha))$; hp_i is the sum of $F_{p,i,j,w}$.

The flow conservation constraint (Eq. (4)) ensures that the amount of incoming data is equal to that of the outgoing data for all nodes selected to establish the lightpath s–d. Eq. (5) provides the capacity constraint that ensures that the sum of available bandwidth of the already established lightpaths (first term) plus that of the lightpaths suggested to be created (second term) must be at least equal to the bandwidth request (b) of the incoming connection.

Wavelength continuity in the formulation is assured by the use of the same index w for all variables. Distinct wavelength constraints given in Eq. (6) define that all lightpaths using the same link i–j must allocate distinct wavelengths [27].

In order to assess the benefits of the multipath formulation, a singlepath ILP was also proposed. Since the original HTBalancing algorithm could not handle connections with bandwidth requests greater than the capacity of a single wavelength, an ILP version of HTBalancing algorithm has been proposed, which is a singlepath counterpart of the MPHTBalancing. This new version of HTBalancing allows the allocation of multiple lightpaths since it must follow the same route. The ILP version of HTBalancing is presented in Algorithm 3. Notation is defined below:

- $F_{i,j,w} \in \{0,1\}$: a binary variable with a value of 1 if wavelength w in the link i,j is used; otherwise, it is zero;
- $T_w \in \{0,1\}$: a binary variable with a value of 1 if the wavelength w is used in the solution; otherwise it is zero;

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002

- $X_{lp} \in \{0,1\}$: a binary variable with a value of 1 if the already established route lp is used in the solution; otherwise it is zero;
- C_{lp}: the cost of the already established route lp;
- B_{lp}: the available bandwidth in the already established route lp;
- Θ: the number of wavelengths necessary (b/λ) to meet a requested demand;
- b, H, λ and α as defined for Algorithm 2.

Algorithm 3. *HTB*alancing (ILP).

$$\text{Min}\sum_{lp} X_{lp} \times C_{lp} + \sum_{w}\left(T_w \times (1-\alpha) + \sum_{i,j} F_{i,j,w} \times H \times \alpha\right) \tag{7}$$

$$\sum_{i} F_{i,j,w} - \sum_{k} F_{j,k,w} = \begin{cases} -T_w & \text{if } i=s \\ T_w & \text{if } i=d \quad \forall j,w \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

$$\sum_{lp} X_{lp} \le 1 \tag{9}$$

$$\sum_{lp} (X_{lp} \times B_{lp}) \ge \sum_{lp} (X_{lp} \times b) \tag{10}$$

$$\sum_{w} T_w = \Theta \times (1 - \sum_{lp} X_{lp}) \tag{11}$$

$$\sum_{w} F_{i,j,w} = \left\{0, \sum_{w} T_w\right\} \quad \forall i,j \tag{12}$$

$w = [1, 2, \ldots W_{i,j}],\ i,j \in E$

The objective function of the singlepath ILP formulation in Eq. (7) tries to find a single route between source s and destination d with the minimum cost. As in the multipath formulation, the cost of each lightpath in the route is defined using Eq. (2). The first term of Eq. (7) defines the cost of already established routes, while the second term denotes the cost of potential new routes. The index w in all variables ensures the wavelength continuity constraint. Eq. (8) is the singlepath counterpart of Eq. (4) of Algorithm 2 while Eqs. (10) and (11) are equivalent to Eq. (5). Since a singlepath solution must be found, only one of the already established routes (lp) can be chosen, which is specified by Eq. (9). Eq. (12) ensures that all lightpaths are established along the same route.

3.3. Computational cost

The computational complexity of the proposed ILP MPHTBalancing and ILP HTBalancing depends on the complexity involved in the solutions provided by the solver employed as well as on the number of variables and constraints of the model, as is the complexity of any ILP solution.

The MPHTBalancing ILP have three vectors of binary variables, $F_{p,i,j,w}$, $X_{p,w}$ and $X_{lp,w}$. Given that i and j refers to the network nodes, w indicates the wavelength, and p indicates one of the two possible alternative paths, the number of elements in $F_{p,i,j,w}$ is equal to $2 \times W \times N^2$. Accordingly, $X_{p,w}$ has $2 \times W$ variables. The number of elements in the vector $X_{lp,w}$ depends on the number of lightpaths established at the moment of the arrival of a request for connection establishment that arrives, which is bounded by $O(N \times W)$. Then, the number of elements of Xlp, w is $W \times O(N \times W)$. Thus, the number of binary variables of the MPHTBalancing ILP is $2 \times W \times N^2 + 2 \times W + W \times O(N \times W)$.

The number of constraints in the MPHTBalancing ILP, considering (Eqs. (4)–6), is $2 \times W \times N + 1 + W \times N^2$.

In the HTBalancing (ILP) algorithm, there are three vectors of binary variables, $F_{i,j,w}$, X_{lp} and T_w. Since i and j refer to the network nodes and w denotes the wavelength, the number of elements in $F_{i,j,w}$ is $W \times N^2$. The vector T_w has W elements and the number of elements of the vector X_{lp} cannot be defined in advance because it depends on the number of lightpaths established at the moment of the arrival of a request for connection establishment; it is bounded by $O(N \times W)$. Therefore, the number of binary variables of HTBalancing (ILP) is $W \times N^2 + W + O(N \times W)$.

The number of constraints in Eq. (12) of HTBalancing (ILP) is $3 + W \times N + N^2$.

The proposed algorithms based on ILP were implemented using the solver Xpress Optimization Suite [28] which employs a branch and cut enumeration, which is a combination of the branch and bound enumeration technique with linear programming and possible insertion of additional linear constraints. In each node of the enumeration tree, it may be required to solve a linear program that although can be solved in polynomial time, most of the ILP solvers use implementations of the simplex method which has exponential time in the worst case, but perform very well in practical instances and proved to have polynomial time on average case for many conditions [29]. The worst case complexity of the branch and bound technique for solving ILP problems is exponential [30].

3.4. Relaxation of the MPHTBalancing algorithm

A linear relaxation is a heuristic of an integer linear problem and is given by the replacement of integer values in the decision variables with real values. Linear relaxation aims to minimize the execution time for problems containing a large number of variables and generate bounds for the solution of the integer problems.

The optimal solution must approximate integer values that are valid solutions to the original problem. One widely used technique to extract the information for relaxed solutions and to reduce it to binary solutions is the randomized rounding technique [31], which converts the approximate real values with 0 or 1, represented by a defined probability. The rounding probability is given by an algorithm with random variables.

The multipath formulation (Algorithm 2) finds optimal solutions by defining binary variables $X_{p/lp,w}$ and $F_{p,i,j,w}$ which give the routes along which the requested connection should be routed. To reduce the computational demands of this ILP formulation, a heuristic algorithm employing the randomized rounding linear relaxation technique is proposed to aggregate both subwavelength and superwavelength connections. This relaxation algorithm replaces $X_{p/lp,w} \in \{0,1\}$ and $F_{p,i,j,w} \in \{0,1\}$ with $X_{p/lp,w} \in [0,1]$ and $F_{p,i,j,w} \in [0,1]$.

In the randomized rounding heuristic algorithm (Algorithm 4), the relaxation formulation is solved just once and the generated values (Line 1) are used as probability values (Line 3) to select the routes. The random drawing step (Line 4) uses a uniform probability distribution to choose the routes. The algorithm is then repeated until the minimum number of routes supplying the requested bandwidth is determined.

Algorithm 4. *MPHTB*alancing (Randomized Rounding Heuristic).

Require: Relaxation solution; Request for bandwidth (b)
Ensure: Integer solution
1: Define $X_{lp,w}$ and $F_{p,i,j,w}$ as fractional solution
2: **while** $(\sum_{chosen-p/lp} b_{p/lp} < \mathrm{b})$ **do**
3: Define $X_{lp,w}$ and $F_{p,i,j,w}$ as the probability of the selection the route p/lp as the solution
4: Select p/lp randomly based on defined probabilities
5: **end while**
6: Return the selected values

4. Numerical results

The performance evaluation of the proposed algorithms considers two distinct scenarios, one with subwavelength connections (Section 4.1) and the other with both sub- and superwavelength connections (Section 4.2). The first of scenario is used to evaluate the benefits of employing a load balancing approach when grooming traffic (HTBalancing algorithm), while the second assesses the advantages of multipath routing in networks with a balanced load (MPHTBalancing algorithm).

Simulations were performed using the WDMSim [32] simulator, with the independent replication method used to generate a confidence interval of 95% confidence level. Four topologies were used in the simulation: the NSF topology, with 16 nodes and 25 bidirectional links (Fig. 18(a)), the Pan-European topology with 28 nodes and 82 bidirectional links (Fig. 18(b)), the USA topology with 24 nodes and 43 bidirectional links (Fig. 18(c)), and the Grid 5×5 topology with 25 nodes and 40 bidirectional links (Fig. 18(d)). In all of these topologies, each fiber carries 16 wavelengths, each with a bandwidth capacity of an OC-192 carrier (10 Gbps).

4.1. Grooming of subwavelength requests

To assess the performance of the HTBalancing algorithm under a load composed of subwavelength connections, simulations were conducted and results compared to those given by the HTA algorithm [5] and by the AWLHTA algorithm [7]. The Routing and Wavelength Assignment (RWA) algorithm employed was single-hop Fixed-Alternate Routing with 5 alternative routes, with the First-Fit criterion used for wavelength assignment.

Connections arrive according to a Poisson process, and their bandwidth demands are distributed according to the following probability distribution: 6/19 for OC-3, OC-12 and OC-48; and 1/19 for OC-192 carriers [33]. Connection requests are uniformly distributed among all pairs of nodes. The holding time value follows a negative exponential distribution, with a mean of one unit. The network load, in Erlang, is defined as follows:

$$A = R \times h \times (B/\lambda) \tag{13}$$

where R is the call arrival rate, h is the call holding time, and B is the call bandwidth request normalized to the value of the wavelength capacity λ (OC-192).

Each node in the topologies considered is a partial grooming node, with 32 grooming port pairs (input, output) and no wavelength-conversion capability.

The metric used to evaluate the algorithms is the bandwidth blocking rate (BBR), i.e., the percentage of the amount of blocked traffic in relation to the total bandwidth requested during each simulation (BBR values are given in logarithmic scale). To assess the degree of fairness of the algorithms, the BBR distribution was considered for all the source–destination pairs. These distributions are shown for the NSF and grid topologies, since the results for the other two topologies fall between these two extremes.

Fig. 5 shows the BBR values as a function of the network load for different values of the threshold $\alpha = \{0.25, 0.5, 0.75\}$ employing the NSF topology and one hundred thousand requests for lightpath establishment. Low values of α give more weight to the bandwidth availability (second term) in

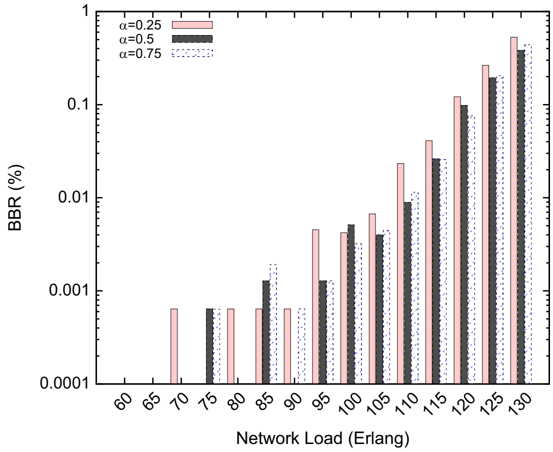

Fig. 5. Threshold analysis—BBR as a function of network load.

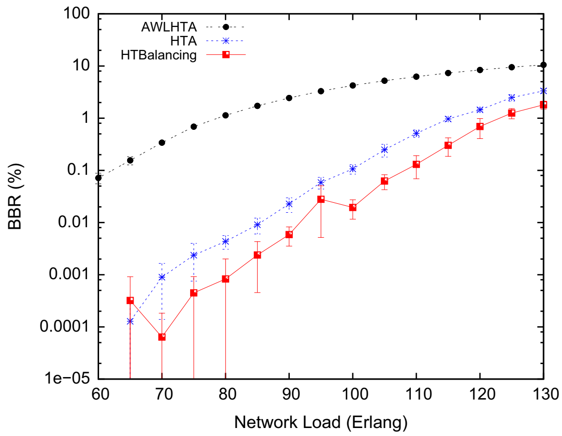

Fig. 6. BBR as a function of the network load for the NSF network.

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002

Eq. (2), while high values of α give more weight to the lightpath extension (first term). The value of $\alpha=0.25$ leads to the greatest blocking. For values of $\alpha \geq 0.5$, the impact on the BBR values is not significant. Moreover, there is a slight variation in the results for these ranges of values. The BBR values for $\alpha=0.5$ are slightly greater than those for $\alpha=0.75$ for loads of 100 and 120 Erlangs. For other loads, the BBR values produced by $\alpha=0.5$ are slightly smaller than those produced when $\alpha=0.75$. These results show that giving more weight to the bandwidth availability leads to less blocking. Therefore, it is recommended to set the value of α to at least 0.5 in order to reduce the BBR values. The threshold value $\alpha=0.5$ will be used in the rest of this paper.

The BBR values for the NSF topology as a function of the load are presented in Fig. 6. The BBR produced by HTBalancing is lower than that given by the HTA algorithm. The BBR value produced by the HTA under loads of 70 Erlangs is 14 times greater than that produced by HTBalancing. For higher loads, the differences between the BBR values decreases but that of HTA is still 1.8 times greater for loads of 130 Erlangs. The lower node connectivity in NSF topology leads to the creation of bottlenecks, even under low loads. The HTA algorithm tries to minimize the additional lifetime imposed on existing lightpaths by incoming requests which tends to concentrate the traffic on few lightpaths, although this leads to bottlenecks. The HTBalancing algorithm, on the other hand, prioritizes the distribution of the load to avoid bottlenecks, which leads to lower blocking ratio. In the evaluation reported in [7], the AWLHTA is able to block a lower number of requests than the HTA and HTBalancing algorithms. However, the BBR produced by the AWLHTA algorithm, in the experiments reported here, is three and four orders of magnitude higher than those produced by HTA and HTBalancing, respectively. Under loads lower than 70 Erlangs, the AWLHTA algorithm produces BBR of the order of 10^{-1} which is very high. The difference in BBR decreases with the load increase. However, the difference between those produced by AWLHTA and by HTBalancing is one order of magnitude under loads of 130 Erlangs. The equations used by AWLHTA to calculate the link weight consider full wavelength conversion and, consequently, the AWLHTA algorithm does not produce good performance in the scenarios used in the present paper, which employ no wavelength conversion. Furthermore, the configuration of AWLHTA parameters depends both on the number of nodes in the network and on the load, which does not favor its use in realistic settings. Moreover, the topology used in [7] is too restrictive and does not allow a thorough evaluation as done in this section.

Fig. 7 shows the per pair BBR distribution and the mean BBR value for a single simulation for a load of 115 Erlangs. The straight line represents the HTA algorithm mean BBR value. For the same pairs, the HTA algorithm generated BBR up to 3.7 times greater than its own mean BBR value, and up to 12 times greater than the mean BBR value produced by HTBalancing. Such difference shows the importance of balancing the network load.

Fig. 8 shows the BBR values for the Pan-European network, which contains cycles as that formed by nodes 6, 2, 0, 5, 7, 6. Such cycles make difficult the search for alternative routes with the HTBalancing algorithm. For loads of 70 Erlangs, the BBR values generated by the HTA algorithm are almost 2 orders of magnitude higher than those resulting from the use of the HTBalancing algorithm.

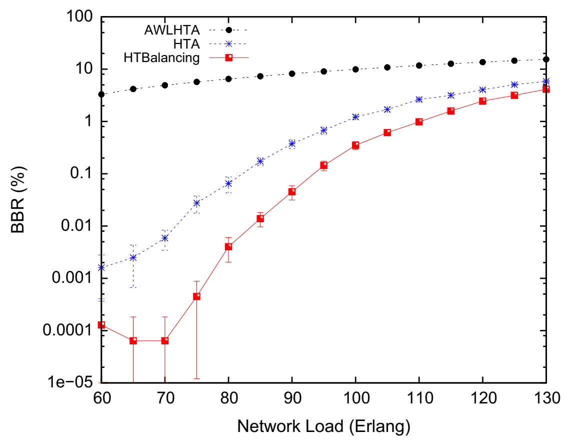

Fig. 8. BBR as a function of the network load for the Pan-European network.

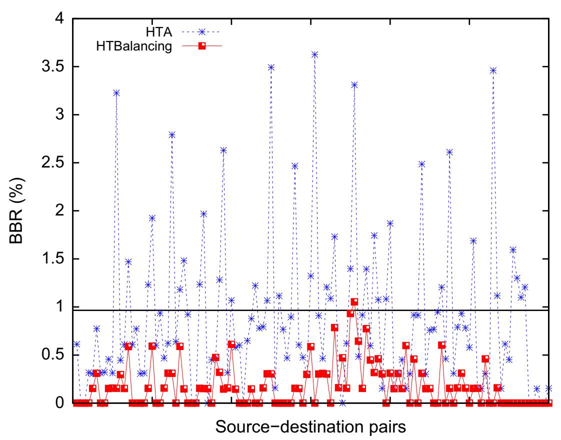

Fig. 7. BBR of each source–destination pair for a load of 115 Erlangs (NSF).

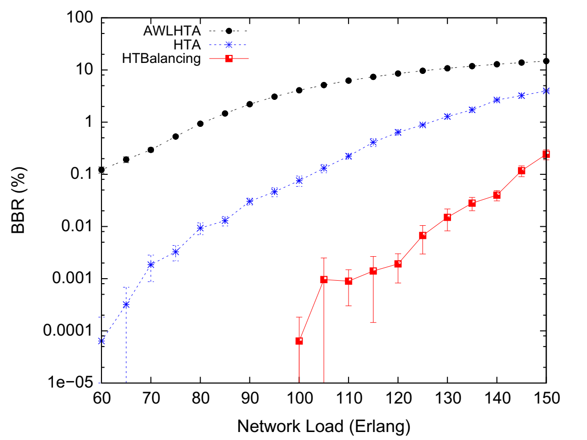

Fig. 9. BBR as a function of the network load for the USA network.

Under loads of 130 Erlangs, the HTA algorithm generated BBR values 1.2 times greater than those produced by HTBalancing. These values are influenced by the cycles presented in the Pan-European topology. Such cycles also impact the BBR produced by AWLHTA. Under loads of 60 Erlangs, the BBR values are almost ten times greater than those generated for the NSF topology (Fig. 6). The choice of the link weight function and the unavailability of wavelength conversion in the simulated network leads the AWLHTA to produce bottlenecks which are critical in the presence of cycles, i.e., when there is no alternative path between source and destination in the cycle.

Fig. 9 shows the results for the USA network. In this scenario, the differences between the BBR from HTBalancing and that resulting from the HTA algorithm are even greater than those for the NSF topology. This difference is highly influenced by characteristics of the USA topology, which has a high degree of node connectivity, thus providing various alternative paths, an option which helps avoid the creation of bottlenecks. For loads as low as 100 Erlangs, the HTBalancing approach does not produce any blocking, although the HTA algorithm produces a BBR greater than 10^{-2}. Under loads of 100 Erlangs, the BBR produced by the HTBalancing algorithm is almost 2 orders of magnitude lower than those produced by the HTA algorithm. The minimum difference in BBR for the two algorithms is for loads of 150 Erlangs, when the HTA algorithm gives a BBR value 16.3 times greater than does the HTBalancing algorithm. The AWLHTA also takes advantage of the higher numbers of available paths in the USA topology which makes the BBR produced by AWLHTA to reach 10^{-1} only under loads of 90 Erlangs. The difference between the BBR values produced by AWLHTA and those given by HTA and HTBalancing is three orders of magnitude and four orders of magnitude, respectively. Under loads of 150 Erlangs, the difference between the BBR produced by the AWLHTA and HTBalancing is more than 2 orders of magnitude. Despite the existence of parallel lightpaths in the USA topology, the link weight used by the AWLHTA algorithm is not able to suggest the establishment of alternative lighthpaths.

Fig. 10 shows the BBR for the Grid network. The BBR values given by the HTBalancing algorithm are null for loads lower than 95 Erlangs, while the HTA algorithm starts blocking for a load of 70 Erlangs. For loads of 95 Erlangs, the HTA algorithm leads to BBR values 1.5 orders of magnitude greater than those obtained with HTBalancing. Although the difference in BBR decreases with the load increase, the HTA algorithm produces BBR values which are 11.5 times greater than those resulting from the use of the HTBalancing algorithm for loads of 150 Erlangs. The BBR given by AWLHTA for the Grid topology is similar to that for the USA topology since both topologies provide several alternative routes to establish the requested connections. The difference between the BBR given by AWLHTA and HTA or HTBalancing can be of four orders of magnitude. The symmetry of the Grid topology favors the creation of multiple concurrent paths which favors lower levels of blocking.

Fig. 11 presents the per pair BBR distribution and the mean values for a single simulation for a load of 135 Erlangs on the Grid topology. For some source destination pairs, the BBR value given by the HTA algorithm was up to 3.7 times larger than the mean BBR value (depicted by the straight line in the graphic) and up to 31 times larger than the mean BBR resulting from the use of the HTBalancing algorithm. The Grid network has greater connectivity than the other three topologies, so the HTBalancing algorithm is especially effective in avoiding network bottlenecks. As can be seen in the distribution of blocking, the BBR values produced by the HTBalancing algorithm were distributed more uniformly among the source destination pairs than were those resulting from the use of the HTA algorithm, showing that the HTBalancing promotes fairness.

4.1.1. *Synthesis of the results*

For all investigated topologies, the employment of the HTBalancing algorithm balanced the traffic for subwavelength demands, moreover, the accepted connections were fairly distributed among the S-D nodes. For the NSF and Pan-European topologies, the algorithm was less effective due to the reduced node connectivity, which limited the number of alternative paths that could be used to establish lightpaths. On the other hand, the use of HTBalancing algorithm for highly connected topologies such as the USA and the Grid topologies can take advantage of traffic

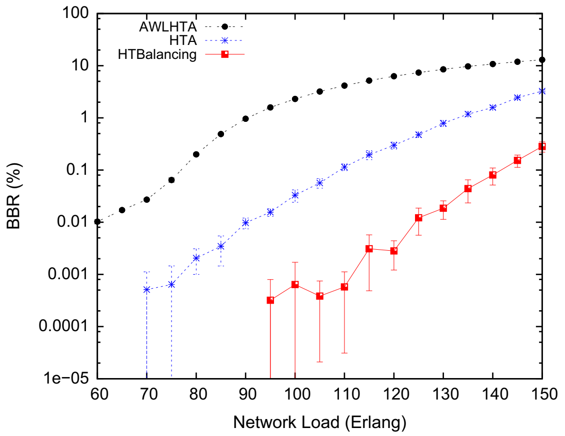

Fig. 10. BBR as a function of the network load for the Grid network.

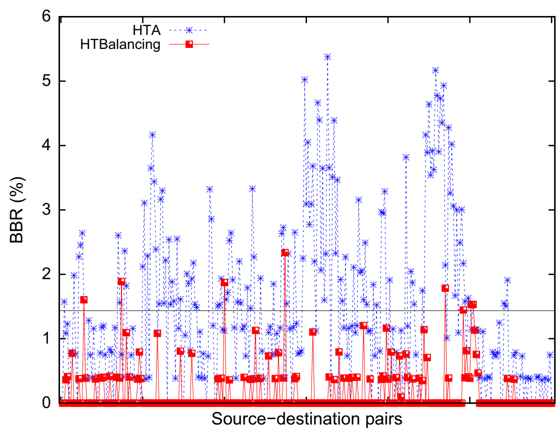

Fig. 11. BBR of each source–destination pair for a load of 135 Erlangs (Grid).

balance, thus leading to lower levels of blocking. For all the topologies considered, the HTBalancing algorithm surpasses the state-of-the-art solution by a few orders of magnitude. Moreover, the per pair BBR results show that the single path HTBalancing algorithm is able to reduce the overall blocking drastically, while limiting the BBR differences among pairs, thus leading to a fairer solution.

4.2. Grooming of subwavelength and superwavelength requests

In this section, a mixture of traffic flows generated by standard IP applications (subwavelength) and highly demanding multimedia applications (superwavelength) is considered in the simulations. The assessment of the dynamic traffic grooming used with multipath routing was made possible by the use of the balanced multipath solution (MPHTBalancing) compared to its single path counterpart (HTBalancing).

The algorithms as well as the relaxed version received as input information about the call $(s; d; b; H)$ and about the state of the network for each of the incoming requests. Both optimal and relaxed algorithms were solved using the Xpress-MP Suite tool [28]. When a request was accepted, the network state was updated. Each node in the topologies considered is a full grooming optical switch, with no wavelength conversion capability.

Connections which are established as single-hop lightpaths arrive according to a Poisson process. Low bandwidth demands are distributed according to the following probability distribution: 9/45 for OC-1; 8/45 OC-3; 7/45 for OC-12; 6/45 for OC-48. Superwavelength connection requests are distributed according to the following probability distribution: 5/45 for OC-211; 4/45 for OC-403; 3/45 for OC-595; 2/45 for OC-787; 1/45 for OC-979. The connection requests are uniformly distributed among all pairs of nodes. The holding time follows a negative exponential distribution with a mean of one unit.

Although the differential delay (DD) among the lightpaths was not considered in the ILP formulation, it was measured during the simulation to evaluate the proposed algorithm. Ideally, this should be reduced to minimize the space for storing the packets. The DD was computed as follows [34]:

$$DD = \sum_{p \in P} (d_{\tilde{p}} - d_p) \tag{14}$$

where DD is the differential delay of the set of the lightpaths (P) selected by the ILP solution; $d_{\tilde{p}}$ is the lightpath with the greatest delay in the set P.

Fig. 12 presents the BBR values for the NSF topology as a function of load. For low loads, the HTBalancing algorithm produces blocking, while employing the MPHTBalancing algorithm leads to blocking only for loads under 40 Erlangs. Moreover, bandwidth blocking ratio is 145 times higher than that resulting from the use of the MPHTBalancing algorithm. Under loads of 120 Erlangs, this difference decreases to 1.7 times. Due to the low node connectivity in the NSF networks, a concentration of traffic leads to the creation of bottlenecks even under low loads. This shows the importance of an algorithm which aggregates the connections of distinct routes, as this maximizes the chances of accepting calls.

Fig. 13 shows the per pair BBR distribution and the mean BBR value for a single simulation for a load of 90 Erlangs on the NSF topology. For some pairs, the HTBalancing algorithm produced BBR values up to three times greater than the mean BBR value (depicted by the straight line in Fig. 13), and up to 11.2 times greater than the mean BBR value resulting from the use of the MPHTBalancing algorithm. These results show that the use of multiple lightpaths in only one route for the aggregation of connections leads to the concentration of traffic and the inevitable creation of bottlenecks. This is especially critical for the NSF topology, given the low degree of node connectivity, which shows that distributing the traffic among different routes is fundamental in improving fairness among source–destination connection pairs.

Fig. 14 shows the BBR values for the Pan European network as a function of the load. For loads of 30 Erlangs, the HTBalancing algorithm generates a bandwidth blocking rate 12.6 times greater than that produced by the MPHTBalancing algorithm. For loads of 120 Erlangs, this difference decreases to 1.4 times. The existence of cycles in the Pan

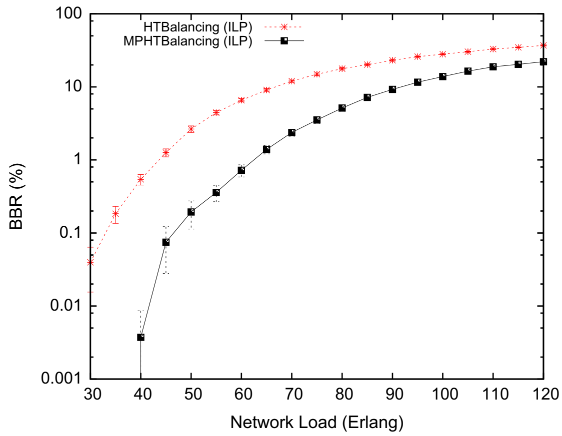

Fig. 12. BBR as a function of the network load for the NSF network.

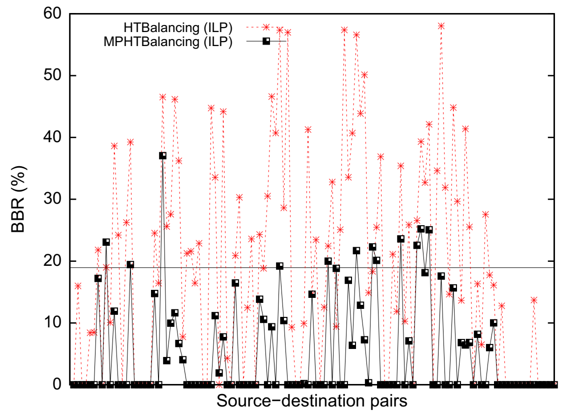

Fig. 13. BBR distribution of each source–destination pair for a load of 90 Erlangs (NSF).

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002

European network makes it difficult to find alternative routes. However, the MPHTBalancing algorithm showed greater capability in providing the required connections than did the HTBalancing algorithm. The results show the importance of using the MPHTBalancing algorithm to maximize the chances of connections being accepted.

Fig. 15 shows the BBR values for the USA topology as a function of the load. Again for low loads, the HTBalancing solution generates blocks for requests even under low loads, while the MPHTBalancing algorithm produces blocking only after loads reaching 55 Erlangs. Under loads of 55 Erlangs, the employment of the HTBalancing algorithm generates BBR values 89 times greater than those given by the MPHTBalancing algorithm. For loads of 120 Erlangs, the MPHTBalancing algorithm promotes a reduction in BBR of 2.1 times in relation to its single path HTBalancing counterpart. The difference in BBR values between the HTBalancing algorithm and the MPHTBalancing algorithm is even greater than that found for the NSF network, due to the capability of the MPHTBalancing algorithm to explore the high node connectivity present in the USA topology.

Fig. 16 presents the BBR values for the Grid topology as a function of the load. Even for low loads, the solution based on the HTBalancing algorithm produces blocking, while that employing MPHTBalancing as an aggregation algorithm only results in blocking requests for loads greater than 45 Erlangs. The BBR values produced by the MPHTBalancing algorithm are 162 times lower than those resulting from use of the HTBalancing algorithm under loads of 50 Erlangs, and 1.8 times lower for loads of 120 Erlangs. As in the USA topology, the Grid network is characterized by the existence of many alternative routes, and this is exploited by the MPHTBalancing algorithm to find better lightpaths, thus avoiding the creation of bottlenecks and maximizing the acceptance of high-speed connections.

Fig. 17 shows the per pair BBR distribution and the mean BBR value for a single simulation for a load of 90 Erlangs for the Grid network. The straight line reflects the mean BBR value generated by the HTBalancing algorithm. As in the topologies previously investigated, the variation in BBR values around the mean value is greater when the HTBalancing algorithm is employed for traffic grooming. For some pairs, the HTBalancing algorithm produces BBR values up to 5.1 times greater than the mean BBR value and up to 28.2 times greater than the BBR mean resulting

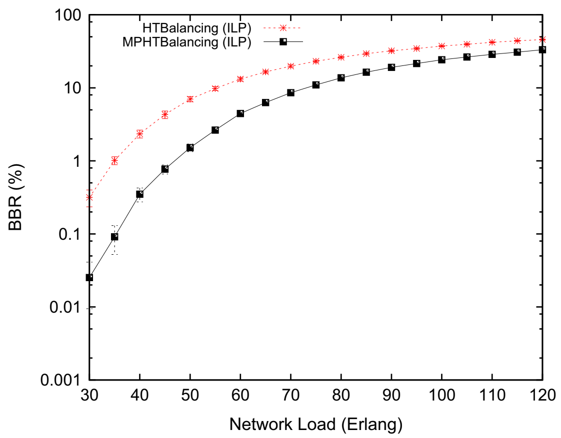

Fig. 14. BBR as a function of the network load for the Pan European network.

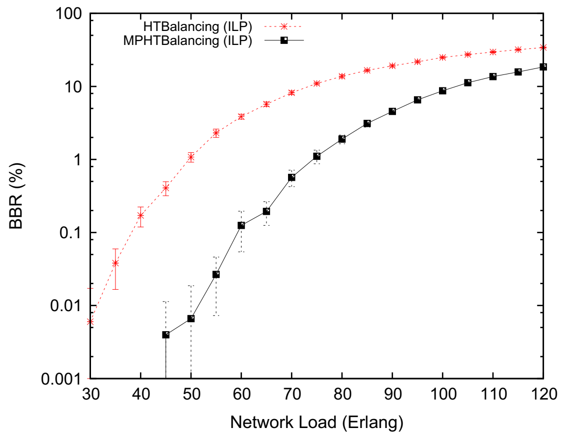

Fig. 16. BBR as a function of the network load for the Grid network.

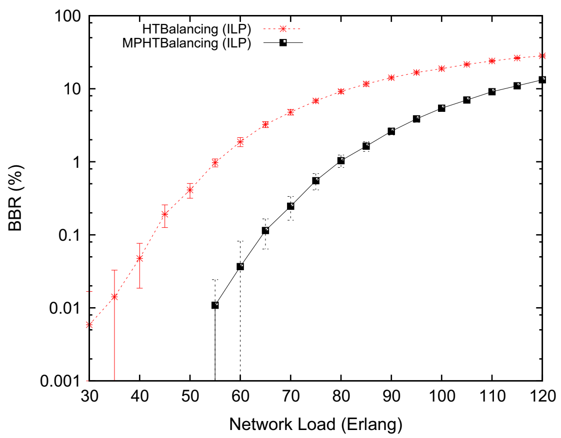

Fig. 15. BBR as a function of the network load for the USA network.

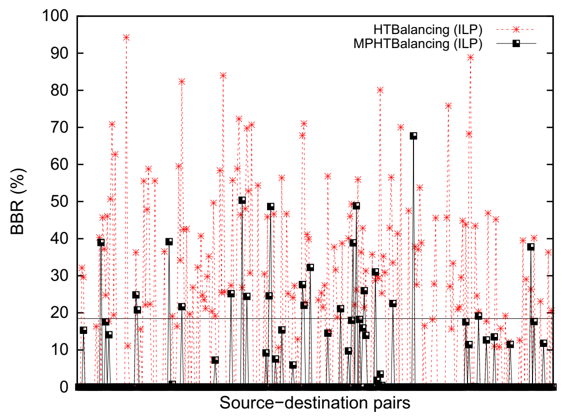

Fig. 17. BBR distribution of each source–destination pair for a load of 90 Erlangs (Grid).

from the use of the MPHTBalancing algorithm. These results emphasize the benefits of the use of lightpaths in distinct routes to promote fairness among source and destination pairs.

Fig. 18 shows the average differential delay of the lightpaths selected by the MPHTBalancing algorithm for all topologies considered (Fig. 4). The differential delay values are reduced due to the selection of short routes, as a consequence of the definition of the cost function (Eq. (2)), which considers the number of hops (hp_i) along the route. All differential delay values are below 5 ms, which is the upper limit suggested in [35]. Such results show the ability of the MPHTBalancing algorithm to limit the value of the differential delay irrespective of the topology.

4.2.1. Synthesis of the results

In the mixed traffic scenario, the use of the MPHTBalancing algorithm reduced the BBR in more than one order of magnitude when compared to that of the HTBalancing algorithm.

Moreover, the differential delay values obtained are safely below the practical limit suggested in the literature. This is mainly due to the fact that the MPHTBalancing cost function (Eq. (2)) leads to the choice of a set of shorter routes as candidate for the establishment of connection, thus reducing the variability in route length.

4.2.2. Relaxed MPHTBalancing ILP solution

The MPHTBalancing algorithm based on ILP formulation, however, involves high computational costs. In order to overcome this problem, a relaxation used as a heuristic of the MPHTBalancing algorithm was proposed in Section 3.4. This section presents the evaluation of the performance of this relaxed algorithm.

Tables 1 and 2 present a comparison between the MPHTBalancing algorithm and its relaxed version considering both sub- and superwavelength connections. The execution time values for the relaxed algorithm are presented in columns two, and six, while the execution time values for the optimal algorithm are presented in columns three and seven. The BBR values for the relaxed algorithm are shown in columns four and eight, while the BBR values for the optimal algorithm are shown in columns five and nine.

For most cases, the execution time decreases with the load increase, which is a consequence of the large number of requests of connections rejected due to the unavailability of resources (and consequently not processed by the solver). Moreover, the execution time values for the optimal algorithm grow faster than those for the relaxed algorithm. In most of the experiments, the relaxation of variables decreased the execution time of the algorithm based on ILP. For the USA topology (Table 1), the execution time

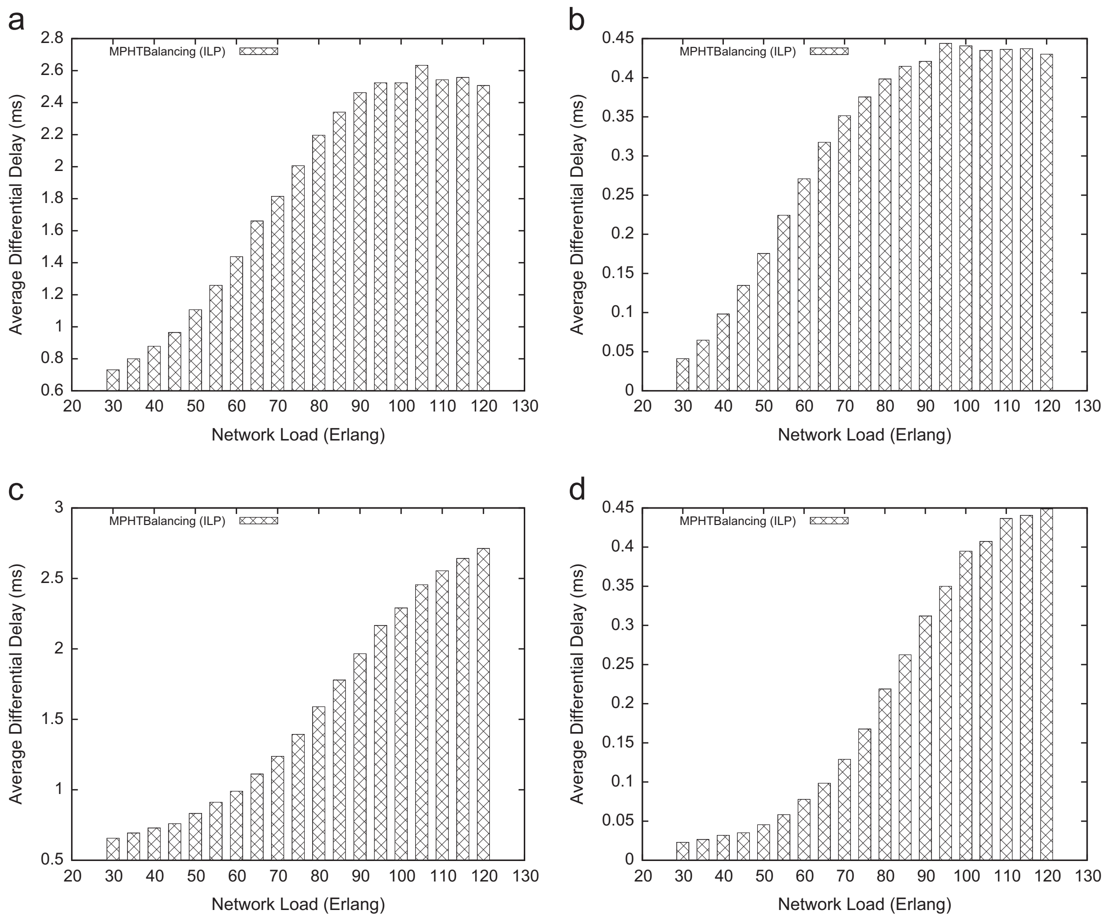

Fig. 18. Average differential delay of lightpaths as a function of the network load. (a) NSF, (b) Pan-European, (c) USA, and (d) Grid.

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002

Table 1
Comparison of MPHTBalancing and its relaxed version for USA and Grid networks.

Load (Erlang)	Approximation × Optimum							
	USA				**Grid**			
	Run time (ms)		BBR(%)		Run time (ms)		BBR (%)	
	Approx	Opt	Approx	Opt	Approx	Opt	Approx	Opt
30	125.1	179.0	0.022	0.0	143.1	191.5	0.13	0.0
40	123.8	176.2	0.048	0.0	141.4	188.6	0.26	0.01
50	121.9	171.8	0.086	0.0	139.8	184.2	0.25	0.01
60	120.1	167.1	0.12	0.04	137.0	179.1	0.37	0.14
70	118.2	162.2	0.4	0.25	135.8	173.1	0.9	0.49
80	116.2	158.6	1.09	1.03	132.2	166.1	2.44	1.95
90	114.5	153.3	2.86	2.62	129.2	159.8	5.42	4.62
100	113.0	146.1	5.52	5.41	126.9	155.8	9.72	8.67
110	111.7	142.4	9.53	9.1	125.3	150.8	14.95	13.81
120	110.0	139.4	13.42	13.24	128.0	147.8	19.2	13.82

Table 2
Comparison of MPHTBalancing and its relaxed version for NSF and Pan European networks.

Load (Erlang)	Approximation × Optimum							
	NSF				**Pan European**			
	Run time (ms)		BBR (%)		Run time (ms)		BBR (%)	
	Approx	Opt	Approx	Opt	Approx	Opt	Approx	Opt
30	71.7	83.4	0.0	0.0	166.6	216.4	0.08	0.06
40	70.0	80.0	0.03	0.004	164.4	207.9	0.54	0.39
50	68.9	78.7	0.24	0.19	160.4	199.4	1.73	1.69
60	67.6	77.0	0.72	0.72	158.3	190.9	4.55	4.55
70	66.6	76.0	2.37	2.37	156.0	183.8	8.61	8.61
80	65.5	74.0	5.25	5.14	154.2	179.8	14.07	13.9
90	64.4	72.3	9.24	9.24	152.7	176.0	19.92	19.92
100	63.9	70.4	13.84	13.84	151.8	173.2	24.11	23.93
110	63.1	69.2	18.82	18.82	150.8	170.4	29.26	29.26
120	62.8	68.3	22.36	22.12	153.1	167.9	33.53	33.53

decreased 29.4 ms for loads of 120 Erlangs and 53.9 ms for loads of 30 Erlangs. For the Grid topology (Table 1), the reduction in execution time was 19.8 ms for loads of 120 Erlangs and 48.4 ms for loads of 30 Erlangs. Due to the large number of nodes and links, the Pan European network requires the largest execution time (Table 2). In relation to the optimal algorithm for this topology, the relaxed algorithm decreased the execution time 14.8 ms and 49.8 ms for loads of 120 Erlangs and 30 Erlangs, respectively. The time required to find solutions for the NSF topology is shorter than those for other topologies (Table 2) as a consequence of its size and fast bottleneck formation. The relaxed algorithm decreases the execution time 5.5 ms for loads of 120 Erlangs and 11.7 ms for loads of 30 Erlangs.

Under low loads, the difference in blocking between the two algorithms can be as large as it is for the GRID topology. However, as the network load increases, the BBR values given by the relaxed algorithm approximate more closely to those resulting from the optimal solution due to the reduction of the space of solutions. This behavior leads to efficient solutions since this decrease in performance occurs only when networks are under extremely low levels of blocking. Results show that the relaxed algorithm is efficient in reducing the execution time needed for the optimal algorithm, while generated acceptable solutions for most loads and topologies.

5. Conclusion

This paper provides a comprehensive investigation of the holding time aware dynamic traffic grooming problem. First, an initial strategy was developed based on knowledge of the holding-time connections and the network bandwidth availability, the HTBalancing algorithm. Extensive simulations for various topologies showed that the values of the bandwidth blocking ratio produced by the HTBalancing algorithm, which aggregates subwavelength connections, are significantly lower than those produced by the state-of-the-art solution. Furthermore, the HTBalancing algorithm is able to promote fairness among source–destination pairs.

When dealing with a more realistic traffic scenario, composed of simultaneous sub- and superwavelength connections at the same time, the results show that the MPHTBalancing algorithm produces lower bandwidth blocking ratio and a fair distribution of accepted connections, while differential delay remains at acceptable levels. Furthermore, the proposed relaxed heuristic for the MPHTBalancing algorithm is capable of reducing the execution time of the optimal algorithm, yet still generates good solutions.

Acknowledgments

This work was partially sponsored by CAPES and by CNPq under support code 141246/2009-0, and INCT FOTONICOM (Grant no. 08/57857-2).

References

[1] D. Simeonidou, D. Hunter, M. Ghandour, R. Nejabati, Optical network services for ultra high definition digital media distribution, in: Broadband Communications, Networks and Systems, 2008, pp. 165–168.

[2] I.J. Taylor, E. Deelman, D.B. Gannon, M. Shields, Workflows for e-Science: Scientific Workflows for Grids, Springer Publishing Company, New York, NY, USA, 2007.

[3] W. Fawaz, B. Daheb, O. Audouin, M. Du-Pond, G. Pujolle, Service level agreement and provisioning in optical networks, IEEE Commun. Mag. 42 (1) (2004) 36–43.

[4] E. Marilly, O. Martinot, H. Papini, D. Goderis, Service level agreements: a main challenge for next generation networks, in: The Second European Conference on Universal Multiservice Networks, 2002 (ECUMN 2002), 2002, pp. 297–304.

[5] M. Tornatore, A. Baruffaldi, H. Zhu, B. Mukherjee, A. Pattavina, Holding-time-aware dynamic traffic grooming, IEEE J. Sel. Areas Commun. 26 (3) (2008) 28–35.

[6] J. de Santi, A. Drummond, N. da Fonseca, A. Jukan, Load balancing for holding-time-aware dynamic traffic grooming, in: 2010 IEEE Global Telecommunications Conference, IEEE GLOBECOM 2010, 2010, pp. 1–5.

[7] Z. Xu, J. Huang, Z. Zhou, Z. Ding, T. Ma, J. Wang, A novel grooming algorithm with the adaptive weight and load balancing for dynamic holding-time-aware traffic in optical networks, Opt. Fiber Technol. 19 (5) (2013) 392–399.

[8] Y. Chen, S. Phibbs, A. Jaekel, Joint scheduling and virtual topology design for sliding periodic traffic, Opt. Switch. Netw. 10 (3) (2013) 186–194.

[9] Z. Liu, W. Guo, Q. Shi, W. Hu, M. Xia, Sliding scheduled lightpath provisioning by mixed partition coloring in WDM optical networks, Opt. Switch. Netw. 10 (1) (2013) 44–53.

[10] W. Su, G. Sasaki, C.-F. Su, A. Balasubramanian, Scheduling of periodic connections with flexibility, Opt. Switch. Netw. 3 (3–4) (2006) 158–172.

[11] A. Muhammad, R. Forchheimer, Reducing blocking probability in dynamic WDM networks using different schemes, in: 2011 International Conference on the Network of the Future (NOF), 2011, pp. 97–101.

[12] A. Muhammad, R. Forchheimer, L. Wonsinska, Impairment-aware dynamic provisioning in WDM networks with set-up delay tolerance and holding-time awareness, in: 2011 17th IEEE International Conference on Networks (ICON), 2011, pp. 213–218.

[13] J. de Santi, N. da Fonseca, G. Figueiredo, Algorithm for traffic grooming of batches of deadline-driven requests, in: 2013 IEEE International Conference on Communications (ICC), 2013, pp. 2267–2271.

[14] S. Zhang, D. Shen, C.-K. Chan, Energy-efficient traffic grooming in WDM networks with scheduled time traffic, J. Lightwave Technol. 29 (17) (2011) 2577–2584.

[15] Y. Chen, A. Jaekel, Energy aware resource allocation based on demand bandwidth and duration, Proc. Comput. Sci. 10 (2012) 998–1003. aNT 2012 and MobiWIS 2012.

[16] Y. Chen, A. Jaekel, Energy-aware scheduling and resource allocation for periodic traffic demands, IEEE/OSA J. Opt. Commun. Netw. 5 (4) (2013) 261–270.

[17] A. Munir, S. Tanwir, S. Zaidi, Holding time aware dynamic bandwidth allocation algorithm for emerging bandwidth on demand multicast applications, in: The Sixth International Symposium on HONET, 2009, pp. 16–21.

[18] A. Munir, S. Tanwir, S. Zaidi, Requests provisioning algorithms for dynamic optical circuit switched (DOCS) networks: A survey, in: IEEE 13th International Multitopic Conference, 2009, pp. 1–6.

[19] X. Chen, A. Jukan, A. Drummond, N. da Fonseca, A multipath routing mechanism in optical networks with extremely high bandwidth requests, in: IEEE Global Telecommunications Conference, 2009 (IEEE GLOBECOM 2009), 2009, pp. 1–6.

[20] X. Chen, A.C. Drummond, A. Jukan, N.L. da Fonseca, Multipath routing with topology aggregation for scalable inter-domain service provisioning in optical networks, Opt. Switch. Netw. 9 (4) (2012) 314–322.

[21] X. Chen, M. Chamania, A. Jukan, A. Drummond, N.L.S. Fonseca, On the benefits of multipath routing for distributed data-intensive applications with high bandwidth requirements and multidomain reach, in: IEEE CNSR, 2009, pp. 1–6.

[22] L. Padhi, S. Kartikeya, K. Sivalingam, S. Sai, Multi-path routing in optical WDM networks: Even versus uneven split bandwidth allocation, in: 2010 International Conference on Signal Processing and Communications (SPCOM), 2010, pp. 1–5.

[23] R. Shenai, K. Sivalingam, Performance study of ip and sonet grooming in optical WDM mesh networks, J. Opt. Netw. 6 (6) (2007) 568–583.

[24] M. Gillani, A. Ali, H. Ilyas, A. Sultan, Multi-path traffic grooming in docs, in: High Capacity Optical Networks and Enabling Technologies (HONET), 2011, pp. 71–75.

[25] M. Tornatore, A. Baruffaldi, H. Zhu, B. Mukherjee, A. Pattavina, Exploiting connection-holding time for an efficient dynamic traffic grooming, in: The Sixth International Workshop on DRCN, 2007, pp. 1–8.

[26] M. Tornatore, A. Baruffaldi, H. Zhu, B. Mukherjee, A. Pattavina, Dynamic traffic grooming of subwavelength connections with known duration, in: Conference on OFC/NFOEC, 2007, pp. 1–3.

[27] G.N. Rouskas, H.G. Perros, A Tutorial on Optical Networks, Advanced Lectures on Networking, Springer-Verlag, New York, 2002, pp. 155–193.

[28] FICO-Xpress, Xpress optimization suite. URL ⟨hhttp://www.dashoptimization.com/⟩, 2014.

[29] D.A. Spielman, S.-H. Teng, Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time, J. ACM 51 (3) (2004) 385–463.

[30] G. Sierksma, P.v. Dam, G.A. Tijssen, Linear and Integer Programming : Theory and Practice, Monographs and Textbooks in Pure and Applied Mathematics, Dekker, New York, 1996.

[31] P. Raghavan, C.D. Tompson, Randomized rounding: a technique for provably good algorithms and algorithmic proofs, Combinatorica 7 (4) (1987) 365–374.

[32] A. Drummond, WDMSim—optical WDM networks simulator. URL ⟨http://www.lrc.ic.unicamp.br/wdmsim⟩, 2014.

[33] H. Zhu, H. Zang, K. Zhu, B. Mukherjee, Dynamic traffic grooming in WDM mesh networks using a novel graph model, in: IEEE GLOBECOM '02, vol. 3, 2002, pp. 2681–2685.

[34] S. Ahuja, T. Korkmaz, M. Krunz, Minimizing the differential delay for virtually concatenated ethernet over sonet systems, in: Proceedings of Computer Communications and Networks, 2004 (ICCCN 2004), 2004, pp. 205–210.

[35] S. Huang, B. Mukherjee, C. Martel, Survivable multipath provisioning with differential delay constraint in telecom mesh networks, in: The 27th Conference on Computer Communications, IEEE INFOCOM 2008, 2008, pp. 191–195.

Please cite this article as: J. de Santi, et al., Holding-time-aware dynamic traffic grooming algorithms based on multipath routing for WDM optical networks, Optical Switching and Networking (2014), http://dx.doi.org/10.1016/j.osn.2014.10.002