Traffic Grooming in Optical Networks: Decomposition and Partial Linear Programming (LP) Relaxation

Hui Wang and George N. Rouskas

***Abstract*—We consider the traffic grooming problem, a fundamental network design problem in optical networks. We review a typical integer linear program formulation considered in the literature, and we identify two challenges related to this formulation in terms of scalability and wavelength fragmentation. We then propose a new (to our knowledge) solution approach that decomposes the traffic grooming problem into two subproblems that are solved sequentially: 1) the virtual topology and traffic routing (VTTR) subproblem, which does not take into account physical topology constraints, and 2) the routing and wavelength assignment subproblem, which reconciles the virtual topology determined by VTTR with the physical topology. The decomposition is exact when the network is not wavelength limited. We also propose an algorithm that uses a partial linear programming relaxation technique driven by lightpath utilization information to solve the VTTR subproblem efficiently. Our approach delivers a desirable tradeoff between running time and quality of the final solution.**

***Index Terms*—Linear programming; Optimization; Traffic grooming.**

I. Introduction

In the modern world, communication services delivered via the Internet touch all of society and affect all aspects of human life. To accommodate the exponential growth of demand in communications, an infrastructure that can support ever-increasing amounts of traffic is highly needed. Optical networks have been commonly used as the backbone infrastructure of Internet services, since they deliver high performance in terms of both throughput and quality of service (QoS). The physical structure of an optical network consists of a set of nodes and optical fibers that interconnect them. With the help of wavelength division multiplexing (WDM) technology, it is possible to transmit traffic on different wavelengths within the same optical fiber simultaneously. The data rate of a single wavelength can be up to 40 Gbps, while higher rates are becoming commercially available. Therefore, the capacity of each wavelength can be significantly higher than the magnitude of individual traffic demands.

The concept of traffic grooming was introduced in the mid-1990s to address the gap between the channel capacity and individual traffic demands in optical networks. The key idea is to aggregate individual traffic requests onto wavelengths so as to improve bandwidth utilization across the network and minimize the use of network resources. Many variants of traffic grooming have been studied in the literature. Online versions of the problem target network environments in which traffic demands arrive in real time. Since future demands are not known in advance, the main objective of online problems is to minimize blocking probability or maximize throughput. Heuristics for solving online traffic grooming problems have been proposed in [1–4].

Offline traffic grooming is a fundamental network design problem that has been shown to be NP-hard [5]. Network design problems are typically formulated as integer linear programs (ILPs) and assume the existence of a traffic matrix representing the demands between node pairs. Basic ILP formulations of the problem are available in [6] and [7]. Typically, the objective is to minimize the total network cost while satisfying all demands (e.g., as in [8,9]) or to maximize the total revenue by satisfying as many traffic demands as possible given certain capacity (wavelength) constraints (e.g., as in [7]). Since electronic equipment that terminates lightpaths represents a large fraction of the overall network cost, the number of lightpaths established to carry the traffic demands is usually taken as the metric to minimize [9]. Other cost functions have also been considered, including the electronic switching cost of grooming traffic between lightpaths at intermediate switches [10] and power consumption in optical networks [11].

One essential concern about the ILP formulations is that they are solvable only for small network topologies [11]. For larger topologies representative of deployed networks, the ILP formulation cannot be solved to optimality within a reasonable amount of time (e.g., a few hours). Therefore, the offline problem has been addressed either using heuristic algorithms [12,13] or by manipulating the ILP formulation using decomposition or column generation techniques. In [2], the original ILP is decomposed into two simpler ILPs: one that addresses only the traffic routing and lightpath routing subproblems and is solved first, and another that addresses the wavelength assignment

Manuscript received February 12, 2013; revised May 17, 2013; accepted June 3, 2013; published July 15, 2013 (Doc. ID 185321).

The authors are with Operations Research and the Department of Computer Science, North Carolina State University, Raleigh, North Carolina 27695-8206, USA (e-mail: rouskas@ncsu.edu).

http://dx.doi.org/10.1364/JOCN.5.000825

problem only and takes as input the solution of the first ILP. In [14], a multi-level decomposition method is introduced to address the multi-layered routing and multi-rate connection characteristics of traffic grooming. In [15], the objective is to design a ring network that is able to satisfy any request graph with the maximum degree at most δ. The cases of $\delta = 2$ and $\delta = 3$ were solved by graph decomposition.

Column generation techniques were developed in [16,17]. Given that the main difficulty in solving the problem has to do with selecting from among an exponential number of possible paths to route the traffic demands, a heuristic algorithm using column generation for a path-based formulation of the problem was developed in [16]. The key idea was to generate an optimal subset of paths efficiently. A hierarchical optimization method was proposed in [17]. The method first deals with the grooming and routing decisions using column generation to find the dual bounds and a rounding heuristic to find an integral solution; wavelength assignment was then carried out in a second step.

Existing approaches to obtaining optimal solutions to the traffic grooming problem face serious scalability challenges as well as wavelength fragmentation issues; we discuss these challenges in Subsections II.A and II.B, respectively. The lack of scalability of optimal methods makes it difficult to characterize the performance of heuristic algorithms in realistic topologies and severely limits the application of "what-if" analysis to explore the sensitivity of network design decisions to forecast traffic demands, capital and operational cost assumptions, and service price structures.

In this paper, we develop a new decomposition algorithm and partial linear programming (LP) relaxation technique for the traffic grooming problem. Unlike existing decomposition techniques (e.g., the one in [2]), our approach achieves a significant speed-up because 1) it decouples the virtual topology and traffic routing (VTTR) aspects of grooming from the underlying network topology, and 2) by virtue of this decoupling, the key integer variables are not binary, and hence, effective LP relaxation methods may be applied. Also, in contrast to column generation [16,17], a general method for solving ILPs that has been adapted to traffic grooming problems, our approach is specific to traffic grooming and capitalizes on the structure of the problem. At the same time, our technique is complementary to column generation such that the two might be employed in combination to further improve the running time; however, combining the two techniques is outside the scope of this paper. Finally, unlike in earlier decomposition methods, we prove that our approach is optimal in most practical scenarios of interest, not only in terms of the primary objective of number of lightpaths but also in terms of the number of wavelengths needed to color these lightpaths.

The rest of the paper is organized as follows. In Section II, we present the basic but general ILP formulation for the traffic grooming problem that is the starting point of our work, and we discuss its complexity and limitations. In Section III, we introduce a new decomposition of the problem into a VTTR subproblem and a routing and wavelength assignment (RWA) subproblem. Then, in Section IV, we present an algorithm that uses lightpath utilization information to formulate a partial LP relaxation so as to solve the VTTR subproblem. We present numerical results in Section V, and we conclude the paper in Section VI.

II. Basic ILP Formulation and Challenges

Consider a connected graph $G = (\mathcal{N}, \mathcal{L})$, where $\mathcal{N}$ denotes the set of nodes and $\mathcal{L}$ denotes the set of directed links (arcs) in the network. We define $N = |\mathcal{N}|$ and $L = |\mathcal{L}|$ as the number of nodes and links, respectively. Each directed link l consists of an optical fiber that may support W distinct wavelengths indexed as $1, 2, 3, \ldots, W$. Let $T = [t_{sd}]$ denote the traffic demand matrix, where t_{sd} is a nonnegative integer representing the traffic demand units to be established from source node s to destination node d. In general, traffic demands may be asymmetric, i.e., $t_{sd} \neq t_{ds}$. We also make the assumption that $t_{ss} = 0, \forall\, s$. Finally, we denote C as the capacity of a single wavelength channel in terms of traffic demand units.

We are interested in minimizing the total number of lightpaths used in the network; such an objective minimizes the use of critical resources and provides ample flexibility for future expansion of the network. Hence, we consider the following minimization problem that we refer to as traffic grooming (TG).

Problem 2.1 (TG): Given graph G, number of wavelengths W, wavelength capacity C, and traffic demand matrix T, establish the minimum number of lightpaths to carry all traffic demands.

Let us define the following sets of decision variables:

- t_{ij}^{sd}: integer variable that indicates the amount of traffic, as a multiple of unit demand, from node s to node d carried on lightpaths from node i to node j;
- b_{ij}: integer variable that indicates the number of lightpaths from node i to node j;
- b_{ij}^{l}: integer variable that indicates the number of lightpaths from node i to node j that traverse link l; and
- $c_{ij}^{l,w}$: binary variable that indicates whether a lightpath from node i to node j uses wavelength w on link l.

Let us further denote the set of links going out of and coming into node n as L_n^+ and L_n^-, respectively. With these notations, the TG problem can be formulated as the following ILP, adapted from [6]:

$$\text{minimize:} \sum_{i,j \in \mathcal{N}} b_{ij}, \tag{1}$$

subject to the VTTR constraints

$$\sum_{s,d} t_{ij}^{sd} \leq b_{ij} C, \qquad i,j \in \mathcal{N}, \tag{2}$$

$$\sum_j t_{ij}^{sd} - \sum_j t_{ji}^{sd} = 0, \qquad i \in \mathcal{N}\backslash\{s,d\}, s,d \in \mathcal{N}, \tag{3}$$

$$\sum_j t_{sj}^{sd} = t^{sd}, \qquad s,d \in \mathcal{N}, \tag{4}$$

$$\sum_j t_{js}^{sd} = 0, \qquad s,d \in \mathcal{N}, \tag{5}$$

$$\sum_j t_{dj}^{sd} = 0, \qquad s,d \in \mathcal{N}, \tag{6}$$

$$\sum_j t_{jd}^{sd} = t^{sd}, \qquad s,d \in \mathcal{N}, \tag{7}$$

the lightpath routing constraints

$$\sum_{l \in L_n^+} b_{ij}^l - \sum_{l \in L_n^-} b_{ij}^l = 0, \qquad n \in \mathcal{N}\backslash\{i,j\}, i,j \in \mathcal{N}, \tag{8}$$

$$\sum_{l \in L_i^+} b_{ij}^l = b_{ij}, \qquad i,j \in \mathcal{N}, \tag{9}$$

$$\sum_{l \in \mathcal{L}_i^-} b_{ij}^l = 0, \qquad i,j \in \mathcal{N}, \tag{10}$$

$$\sum_{l \in L_j^+} b_{ij}^l = 0, \qquad i,j \in \mathcal{N}, \tag{11}$$

$$\sum_{l \in L_j^-} b_{ij}^l = b_{ij}, \qquad i,j \in \mathcal{N}, \tag{12}$$

the wavelength assignment constraints

$$\sum_w c_{ij}^{w,l} = b_{ij}^l, \qquad i,j \in \mathcal{N}, l \in \mathcal{L}, \tag{13}$$

$$\sum_{i,j} c_{ij}^{w,l} \le 1, \quad \forall\, w, l \in \mathcal{L}, \tag{14}$$

$$\sum_{l \in L_n^+} c_{ij}^{w,l} = \sum_{l \in L_n^-} c_{ij}^{w,l}, \qquad n \in \mathcal{N}\backslash\{i,j\}, i,j \in \mathcal{N}, \quad \forall\, w, \tag{15}$$

$$\sum_{l \in L_i^+} c_{ij}^{w,l} \le b_{ij}, \qquad i,j \in \mathcal{N}, \quad \forall\, w, \tag{16}$$

$$\sum_{l \in L_i^-} c_{ij}^{w,l} = 0, \qquad i,j \in \mathcal{N}, \quad \forall\, w, \tag{17}$$

$$\sum_{l \in L_j^+} c_{ij}^{w,l} = 0, \qquad i,j \in \mathcal{N}, \quad \forall\, w, \tag{18}$$

$$\sum_{l \in L_j^-} c_{ij}^{w,l} \le b_{ij}, \qquad i,j \in \mathcal{N}, \quad \forall\, w, \tag{19}$$

and the integrality constraints

$$t_{ij}^{sd}, b_{ij}, b_{ij}^l : \text{integer}; \qquad c_{ij}^{l,w} : 0, 1; \qquad w = 1, 2, \ldots, W. \tag{20}$$

The VTTR constraints [Eqs. (2)–(7)] determine the lightpaths to be established (i.e., the virtual topology of the network) and the routing of traffic demands on the virtual topology. The capacity constraint [Eq. (2)] ensures that a sufficient number of lightpaths is established between each node pair. Equations (3)–(7) are multi-commodity flow equations that find the route on the virtual topology of lightpaths for each traffic demand.

The routing constraints [Eqs. (8)–(12)] are multi-commodity flow equations that determine the physical route for each lightpath, where each lightpath corresponds to a single commodity. Equation (8) ensures that the number of incoming lightpaths is equal to the number of outgoing lightpaths at any intermediate node. Equations (9), (10) and (11), (12) are the lightpath constraints at the origin node and sink node, respectively, of each lightpath.

The wavelength assignment constraints [Eqs. (13)–(19)] enforce the two wavelength constraints: 1) Eq. (14) represents the distinct wavelength constraint that guarantees that each wavelength may only be used once on any link, i.e., no two lightpaths sharing a common link may use the same wavelength, and 2) multi-commodity flow equations [Eqs. (15)–(19)] represent the wavelength continuity constraint by ensuring that each link on the same lightpath is assigned the same wavelength. Finally, Eq. (13) ensures that each lightpath will only use one wavelength.

The above formulation and most formulations studied in the literature that use link-related variables suffer from two main challenges: scalability and wavelength fragmentation. We discuss each of these challenges in the next two subsections, respectively.

A. Challenge: Scalability

The scalability of the formulation depends directly on its size, which, in turn, is determined by the number of variables and constraints. The above formulation consists of $N^2(N-1)^2$ integer variables $\{t_{ij}^{sd}\}$, $N(N-1)$ integer variables $\{b_{ij}\}$, $N(N-1)L$ integer variables $\{b_{ij}^l\}$, and $N(N-1)LW$ binary variables $\{c_{ij}^{l,w}\}$, for a total of $O(N^4 + N^2LW)$ variables. Also, there are $O(N^3)$ routing constraints, $O(N^3W)$ wavelength constraints, and $O(N^3)$ grooming constraints, for a total of $O(N^3W)$ constraints in the formulation. Given that current technology may support up to $W = 100$ wavelengths per link, it becomes clear that the ILP formulation can be applied directly only to very small networks. In our experience [11], this ILP formulation may take tens of hours to solve, using commodity hardware, on networks with as few as a dozen nodes.

B. Challenge: Wavelength Fragmentation

Now let us turn our attention to a second issue with the basic formulation. Note that the objective of the ILP is a function that depends only on the number of lightpaths and is independent of the number of wavelengths used to color the lightpaths. Specifically, the number of available wavelengths W is part of the input to the problem, and the only constraint is to color the lightpaths using no more than W distinct colors. Consequently, the ILP solver will not make any attempt to minimize the number of wavelengths as long as the wavelength constraints are satisfied. In other words, the ILP solver will stop as soon as it has determined a solution with the minimum number of lightpaths, without checking whether there exists another solution with the same objective value that uses fewer wavelengths. Therefore, the number of wavelengths in the final solution may be higher than necessary, and as a result, a critical network resource may be severely fragmented.

To illustrate how serious this issue can be, we used the above ILP formulation to solve five problem instances on a six-node ring network, each with a different traffic matrix. We solved each instance three times, each time providing as input a different value for the number W of available wavelengths, namely, $W = 5$, 10, 30. All instances can be solved with fewer than $W = 5$ wavelengths (as we will show later), and hence all three solutions we obtained for a given instance used the same number of lightpaths.

Figure 1 plots the number of wavelengths used by each of the three optimal solutions to the five problem instances. It is clear that the number of wavelengths used in the solution is an increasing function of the number W of available wavelengths input to the formulation. In particular, the optimal solution obtained with a high input value of W may use more than three times as many wavelengths as the optimal solution obtained with a low input value of W. Given that the minimum number of wavelengths required for a feasible solution is not known *a priori*, the designer may be tempted to use a high input value for W (as long as that many wavelengths are indeed available) to ensure that the ILP solver will find a solution. Such an approach will result in severe fragmentation of wavelengths, one of the critical resources in optical networks, increasing the cost of deployment and limiting future expansion of the network. We emphasize that all ILP formulations in which

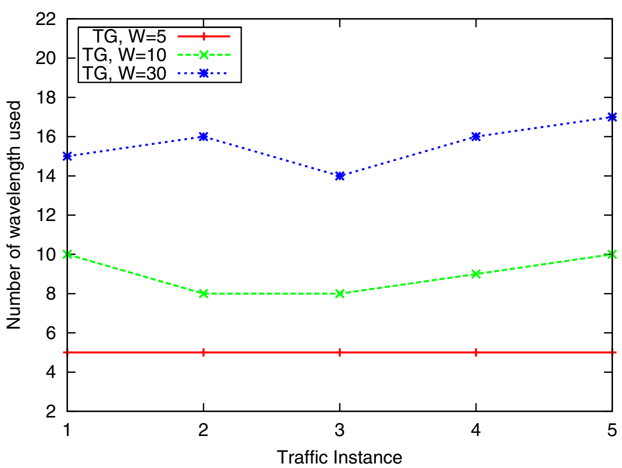

Fig. 1. Wavelength usage comparison for five traffic instances, ring network with $N = 6$ nodes.

the number W of wavelengths is taken as a constraint face similar fragmentation challenges.

We note that it is possible to take into account the wavelength use in the objective function. This can be achieved in two ways, both of which have certain disadvantages compared to the decomposition approach that we will describe in the next section. First, it is possible to minimize the weighted sum of the number of lightpaths and the number of wavelengths. However, because of wavelength reuse, the number of lightpaths in a medium-to-large network is likely to be much larger (an order of magnitude or more) than the number of wavelengths, and furthermore, the relative numbers depend on the traffic demands. (Consider, for instance, a unidirectional 10-node ring network. If the traffic matrix consists of demands between adjacent nodes only, then a single wavelength would be sufficient to establish the 10 lightpaths. On the other hand, if all demands are from each node to the node diametrically opposite in the ring, then five wavelengths would be necessary to establish the 10 lightpaths. While this is certainly an extreme case, it illustrates the two points we mention above.) Therefore, it would be difficult for the network designer to select appropriate weights in advance and independently of the traffic demands. Second, it would be possible to attempt to minimize both the number of lightpaths and the number of wavelengths simultaneously. However, such multiobjective optimization would lead to a set of Pareto optimal solutions, it is difficult to carry out, and to the best of our knowledge, it has not been carried out in the context of traffic grooming.

III. New Decomposition of Traffic Grooming

We decompose the TG problem defined earlier into two subproblems, the VTTR subproblem, and the RWA subproblem.

A. Virtual Topology and Traffic Routing

The VTTR subproblem is defined as follows:

Definition 3.1 (VTTR): Given the number N of nodes in the graph G of TG, the wavelength capacity C, and traffic demand matrix T, establish the minimum number of lightpaths to carry all trafic demands.

The VTTR subproblem can be expressed as minimizing the objective function [Eq. (1)] under the VTTR constraints [Eqs. (2)–(7)]. The output of the problem is a set of lightpaths $\{b_{ij}\}$, as well as the routing $\{t_{ij}^{sd}\}$ of the traffic demands $\{t_{sd}\}$ over these lightpaths. Since the traffic routing subproblem of VTTR, i.e., routing of demands on a given set of lightpaths, is NP-hard [5], VTTR itself is NP-hard.

Note that the VTTR problem does not take as input the network graph G, only the traffic demand matrix T (and, hence, the number of nodes, N). Consequently, the output of the problem is simply the set of lightpaths to be established, but *not* the (physical) paths that these lightpaths take in the network. Therefore, the VTTR subproblem is very different from the *GR* subproblem in the traffic grooming decomposition studied in [2]: the *GR* subproblem takes as input the network graph G and determines not only the set of lightpaths but also their (physical) paths (but not wavelengths) in the network.

The VTTR subproblem is similar in concept to the virtual topology problem studied in the context of multihop broadcast-and-select (BAS) networks [18]. In multihop BAS networks it is not possible to establish direct connections between every pair of nodes, and hence some traffic demands may need to be routed via intermediate nodes—just as traffic demands need to be routed over multiple lightpaths in our problem. Furthermore, just as BAS networks do not impose any physical topology constraints on the formation of direct connections due to their all-to-all broadcast nature, the VTTR subproblem does not impose any physical topology constraints on the formation of the virtual topology of lightpaths. The virtual topology determined by VTTR will be reconciled with the physical topology using the second subproblem, as we discuss shortly.

Ignoring the physical topology constraints in the definition, the VTTR subproblem has two major benefits. First, the running time for finding an optimal solution depends only on the size (i.e., number N of nodes) of the network, not its topology. Hence, the running time of a problem instance with a given demand matrix T would be identical for a sparse ring network and a dense mesh network of the same size. Second, the problem formulation includes the integer variables $\{b_{ij}\}$ and $\{t_{ij}^{sd}\}$, but it does not include any binary variables. Therefore, it is possible to employ partial LP relaxation techniques so as to reduce the time required to find solutions that are close to the optimal one; we describe an algorithm that uses such techniques in the following subsection.

B. Routing and Wavelength Assignment

The RWA problem is one of selecting a path and wavelength for each lightpath, subject to capacity and wavelength constraints.

Definition 3.2 (RWA): Given the graph G of TG and the set of lightpath demands $\{b_{ij}\}$ determined by the solution to VTTR, route the lightpaths on the physical topology of G and assign a wavelength to each lightpath so as to minimize the number of distinct wavelengths required.

The RWA problem is a fundamental problem in optical network design and has been studied extensively. In [19] we developed an exact ILP formulation based on maximal independent sets that solves the RWA problem in rings of size up to $N = 16$ nodes (the maximum size supported by SONET technology and hence the *de facto* maximum size of deployed ring networks) in just a few seconds, several orders of magnitude faster than earlier known solutions. We have also developed new formulations that solve the RWA problem in mesh networks up to two orders of magnitude faster than existing techniques [20,21]. Therefore, we solve the RWA subproblem using the techniques in [19–21] rather than using the corresponding part of the formulation of the TG problem in Eqs. (8)–(19).

C. Sequential Solution to the VTTR and RWA Problems

We propose to solve the TG problem by sequentially solving its two subproblems:

1) Solve the VTTR subproblem to obtain the set $\{b_{ij}\}$ of lightpaths to be established, and the routing of traffic demands $\{t_{sd}\}$ over these lightpaths.
2) Solve the RWA subproblem to find a path and wavelength for each lightpath in the set $\{b_{ij}\}$ so as to minimize the number of distinct wavelengths used in the solution.

Recall that the first step of the solution produces a set $\{b_{ij}\}$ of lightpaths that are determined only by the traffic demands and are not tied to the physical topology of the network. However, the second step routes the lightpaths over the physical links of the network, hence ensuring that the final solution is consistent with the network topology.

The following two lemmas state the properties of this sequential solution:

Lemma 3.1: Let $P_{\mathrm{TG}}^{\star}$ and $P_{\mathrm{VTTR}}^{\star}$ denote the number of lightpaths returned by the optimal solutions to the TG and VTTR problems, respectively. Then

$$P_{\mathrm{VTTR}}^{\star} \le P_{\mathrm{TG}}^{\star}. \tag{21}$$

Proof. The VTTR subproblem is a relaxed version of the original TG problem with constraints [Eqs. (8)–(19)] removed. Hence, the objective value of an optimal solution to VTTR cannot be greater than that of an optimal solution to TG. ∎

Lemma 3.2: Let $W_{\mathrm{RWA}}^{\star}$ be the number of wavelengths returned as the optimal solution to the RWA subproblem that takes as input the optimal solution $S_{\mathrm{VTTR}}^{\star}$ of the VTTR subproblem. If $W_{\mathrm{RWA}}^{\star} \le W$, where W is the number of available wavelengths given as input to the original TG problem, then $S_{\mathrm{VTTR}}^{\star}$, together with the lightpath RWA determined by the RWA subproblem, is an optimal solution to TG.

Proof. According to Lemma 3.1, the number of lightpaths in the solution $S_{\mathrm{VTTR}}^{\star}$ is such that $P_{\mathrm{VTTR}}^{\star} \le P_{\mathrm{TG}}^{\star}$. After the RWA is solved, the RWA of the lightpaths in $S_{\mathrm{VTTR}}^{\star}$ satisfy all the physical topology and wavelength assignment constraints. Hence, the final result of sequentially solving the two subproblems is also a feasible solution to the original problem TG, i.e., $P_{\mathrm{VTTR}}^{\star} \ge P_{\mathrm{TG}}^{\star}$, from which the result of the lemma follows. ∎

The practical implication of Lemma 3.2 is that whenever the network is not wavelength (bandwidth) limited, sequentially solving the VTTR and RWA subproblems will yield an optimal solution to the original TG problem that also minimizes the number of wavelengths used for the given set of lightpaths. Note also that the case of bandwidth-limited networks is not of interest in practice: in the bandwidth-limited regime, customer traffic cannot be accommodated, and hence it is unlikely that network providers would operate their networks in this regime. Instead, providers are likely to add capacity as demand increases to ensure that their networks have sufficient bandwidth to accommodate the offered traffic.

On the other hand, when the network is indeed bandwidth limited, our approach may also be useful in that it can be used by network providers to quickly determine that their network has reached its capacity limits and it is time to add capacity. Specifically, note that the VTTR subproblem yields a solution (i.e., a virtual topology of lightpaths) regardless of the number W of available wavelengths (since W is not part of the VTTR formulation). However, if the RWA algorithm requires more than W wavelengths to color these lightpaths, then it is clear that our sequential decomposition of the traffic grooming problem cannot find a feasible solution. Note that this determination is reached quickly (as indicated by numerical results we will present shortly), since only the VTTR and RWA problems are solved, not the original grooming problem. While it is possible that an optimal traffic grooming algorithm might find a feasible (and hence optimal) solution with no more than W wavelengths (e.g., by constructing a larger number of shorter lightpaths than the solution to VTTR), the fact remains that the network is likely close to its capacity limits. Instead of running an optimal algorithm (which would take a long time and would produce more lightpaths than the VTTR solution), the provider might consider the fact that the sequential decomposition approach did not produce a feasible solution as a strong indication that new additional capacity is necessary to accommodate the offered traffic.

IV. Solution Approach for VTTR

As we mentioned earlier, we have developed fast algorithms for solving the RWA subproblem in ring and mesh networks. In this section we develop a solution technique for the VTTR subproblem, an NP-hard problem for which we are not aware of any scalable solutions.

A. Partial LP Relaxation of VTTR

LP relaxation of an ILP is the problem that arises by relaxing the integrality constraints on the relevant decision variables of the original problem. Since the original ILP formulation has stronger constraints than its LP relaxation, in the case of minimization problems such as the one we consider in this work, the optimal value of the LP relaxation provides a lower bound for the original ILP formulation. Although LP relaxation sacrifices optimality, the relaxed problem can be solved as a linear program in time that may be orders of magnitude lower than the time required to solve the original ILP.

Definition 4.1 (VTTR-rlx): Given the number N of nodes in the graph G of TG, the wavelength capacity C, and traffic demand matrix T, establish the minimum number of lightpaths needed to carry all traffic demands while allowing fractional lightpaths to exist between any pair of nodes.

Let $\{b_{ij}\}$ be integer variables denoting the number of lightpaths from node i to node j in the virtual topology. VTTR-rlx can be derived from VTTR by replacing the integer variables $\{b_{ij}\}$ with nonnegative real variables $\{\bar{b}_{ij}\}$, while maintaining the integrality constraints on all other integer variables (i.e., the traffic routing variables $\{t_{ij}^{sd}\}$) in the formulation. Then, VTTR-rlx represents a *partial* LP relaxation of VTTR and can be formulated as a mixed integer linear program (MILP). Also, if $\{\bar{b}_{ij}\}$ is a feasible solution to VTTR-rlx, then $\{\lceil \bar{b}_{ij} \rceil\}$ is a feasible solution to VTTR.

We compared the VTTR and VTTR-rlx on problem instances defined on a 16-node network.[1] For the comparison, we generated traffic instances by setting each traffic demand t_{sd} as a random integer in the range $[0, t_{\max}]$. We let parameter $t_{\max} = 10$, 20, 30, 40, 50, 60, and for each value of $t_{\max}$ we generated 10 traffic matrices (i.e., problem instances) that were used to solve both VTTR and VTTR-rlx.

Table I presents the CPU time (in seconds), averaged over the 10 random instances, that CPLEX needs to solve the VTTR and VTTR-rlx problems for each value of $t_{\max}$. We imposed a 6 h limit on running time; if an instance did not complete within this time limit, we recorded the best available solution found up to that time and terminated the CPLEX process. We observe that the CPU times do not vary much across the values of $t_{\max}$, but solving the partial LP relaxation VTTR-rlx takes a fraction of a second, whereas solving the VTTR ILP takes longer than the 6 h limit we imposed.

Table II compares the best available solutions to VTTR obtained within the 6 h limit to the optimal solutions to VTTR-rlx, in terms of the objective value (i.e., number of lightpaths). For each row of the table (i.e., a specific value of $t_{\max}$), the values shown are averages over the corresponding ten traffic instances. However, the optimal solution to partial LP relaxation VTTR-rlx is a lower bound, but not necessarily a feasible solution to VTTR. Therefore, in the table we also present the objective value of the feasible solution obtained by rounding up the real values $\{\bar{b}_{ij}^{\star}\}$ of the optimal solution to VTTR-rlx.

From the two tables we make two important observations. First, the integrality constraints on the lightpath variables $\{b_{ij}\}$ play an important role in increasing the complexity of the branch-and-bound process of the ILP solver. Second, rounding up the real lightpath values $\{\bar{b}_{ij}^{\star}\}$ results in a large optimality gap. Based on these observations, in the next subsection we develop an iterative algorithm that strikes a good balance between running time and quality of solution.

[1] Recall that VTTR and VTTR-rlx do not take into account the physical topology of the network, only the number of nodes and traffic demands. Hence, the results we present in this section and later on are valid for any network with the stated number N nodes regardless of the underlying topology.

TABLE I
CPU Time Comparison of VTTR and VTTR-rlx, $N = 16$

t_{max}	CPU Time (s) VTTR	VTTR-rlx
10	>21,600	0.184
20	>21,600	0.199
30	>21,600	0.200
40	>21,600	0.242
50	>21,600	0.259
60	>21,600	0.188

TABLE II
Objective Value Comparison of VTTR and VTTR-rlx, $N = 16$

	Objective Value (No. of Lightpaths)		
	VTTR	VTTR-rlx	
t_{max}	(Best Available)	(Optimal)	(Rounded Up)
10	101.7	74.1	217.9
20	173.2	150.0	274.2
30	250.6	226.6	340.1
40	327.1	302.6	423.6
50	389.3	366.4	480.1
60	468.2	443.5	558.5

B. VTTR-rlx(U_l,U_h): A Utilization-Driven Partial LP Relaxation of VTTR

Consider the optimal solution $\{\bar{b}^{\star}_{ij}\}$ to the VTTR-rlx problem and the corresponding feasible solution $\{\lceil \bar{b}^{\star}_{ij} \rceil\}$ to VTTR, obtained by rounding up all the lightpath variables. Let us define U_{ij}:

$$U_{ij} = \frac{\bar{b}^{\star}_{ij}}{\lceil \bar{b}^{\star}_{ij} \rceil}, \qquad \bar{b}^{\star}_{ij} > 0. \tag{22}$$

The quantity U_{ij} represents the utilization of the lightpaths from node i to node j in the rounded-up feasible solution. When the utilization is high (i.e., U_{ij} is close to 1.0), the corresponding lightpath resources are used effectively in the solution; furthermore, rounding up the corresponding lightpath variable to obtain a feasible solution makes only a small contribution to the optimality gap. The opposite is true when the utilization of a set of lightpaths is low.

We define U_l and U_h, $0 \le U_l \le U_h \le 1$, as the low and high thresholds, respectively, on lightpath utilization. Consider a modified version of VTTR-rlx in which the following two sets of *equality constraints* are activated on the lightpath variables $\bar{b}_{ij}$ with a utilization $U_{ij} \le U_l$ or $U_{ij} \ge U_h$:

$$\bar{b}_{ij} = \lceil \bar{b}^{\star}_{ij} \rceil \quad \forall\, i,j\colon U_{ij} \ge U_h, \tag{23}$$

$$\bar{b}_{ij} = \lfloor \bar{b}^{\star}_{ij} \rfloor \quad \forall\, i,j\colon U_{ij} \le U_l. \tag{24}$$

We let VTRL-rlx(U_l, U_h) denote the modified LP relaxation of VTTR in which the variables $\bar{b}_{ij}$ are set to be equal to the floor (respectively, ceiling) of the corresponding optimal solution obtained from VTTR-rlx if $U_{ij} \le U_l$ (respectively, $U_{ij} \ge U_h$).

The key idea behind the equality constraints [Eqs. (23) and (24)] introduced in the formulation of VTRL-rlx(U_l, U_h) can be explained by using an airline analogy in which lightpaths correspond to scheduled flights. For lightpaths with low utilization, Eq. (24) forces the fractional lightpaths to zero; when solving the modified problem, the traffic carried by these fractional lightpaths will be redirected to other lightpaths. In the airline analogy, this is equivalent to canceling flights that are close to empty. Note that the deletion of lightpaths (respectively, the canceling of flights) may cause some traffic (respectively, passengers) to take a longer route to their destination; however, from the point of view of network design, this may be an acceptable tradeoff if it leads to a smaller network cost. On the other hand, when lightpaths have high utilization, the constraint [Eq. (23)] forces the fractional lightpath to become a full lightpath. As a result, the extra capacity of this new full lightpath becomes available to carry traffic that is potentially redirected by fractional lightpaths that were deleted.

The question that arises is how to determine the pair of thresholds (U_l, U_h), i.e., the specific partial LP relaxation of VTTR that provides a desired tradeoff between running time and quality of solution. To this end, we propose an iterative algorithm that uses a local search technique to select the pair (U_l, U_h). The iterative algorithm treats the integer constraints on lightpath variables $\{\bar{b}_{ij}\}$ as *lazy* constraints, activating only a subset of them at each iteration based on how they relate to the current pair of utilization thresholds.

The iterative algorithm starts by solving the partial LP relaxation VTTR-rlx in which none of the integrality constraints on $\{\bar{b}_{ij}\}$ are activated. If all lightpath variables in the optimal solution are integers, then this is a feasible (and optimal) solution to VTTR. Otherwise, the solution is examined to identify all lightpath variables with a utilization U_{ij} outside the interval $[U_l, U_h]$, and the corresponding VTRL-rlx(U_l, U_h) variant is solved. This process is repeated, increasing the threshold value U_l and decreasing U_h at each iteration, until one of the following stopping criteria is satisfied:

1) all lightpath variables in the solution are integers, and hence represent an optimal solution to VTTR;
2) the threshold pair (U_l, U_h) reaches a predetermined value; or
3) the improvement in the value of the objective function over the previous iteration is less than a predetermined minimum value δ.

A combination of the above criteria may be used, e.g., stop whenever the thresholds have reached a predetermined value or the improvement over the previous iteration is less than δ, whichever is satisfied first.

The iterative algorithm can be described by these steps:

1) Initialization: $U_l \leftarrow 0.1$; $U_h \leftarrow 0.9$.
2) Solve VTTR-rlx with no integer constraints on $\{\bar{b}_{ij}\}$ activated. Find the feasible solution by rounding up all noninteger lightpath variables and caclulate U_{ij} for all lightpath pairs using Eq. (22).
3) Solve VTTR-rlx(U_l, U_h). If the problem is infeasible, return the previous feasible solution and stop. Otherwise, find the new optimal solution, and determine the objective value of the corresponding feasible solution obtained by rounding up all noninteger lightpath variables.
4) If the stopping criterion is satisfied, return the current solution; otherwise set $U_l \leftarrow U_l + 0.1$; $U_h \leftarrow U_h - 0.1$ and repeat from Step 3.

We note that at each iteration of the algorithm, a tighter partial LP relaxation of VTTR with a larger number of equality constraints is considered, generally requiring more time to solve. On the other hand, the objective value of the solution improves with each iteration. By selecting an appropriate stopping criterion, especially in terms of the threshold values on lightpath utilization, this algorithm may be designed to deliver a desirable tradeoff between running time and quality of the final solution.

V. Numerical Results

In this section we investigate the VTTR problem in terms of scalability (i.e., running time) and quality of solution. The VTTR problem and its partial LP relaxations were solved by running the IBM ILOG CPLEX 12 optimization tool on a cluster of identical compute nodes with dual Woodcrest Xeon CPUs at 2.33 GHz with a 1333 MHz memory bus, 4 GB of memory, and a 4 MB L2 cache.

Our study involves a large set of problem instances defined on several network sizes[2] with various random traffic loads. In particular, we consider networks with $N = 8$, 16, 24, and 32 nodes. In all the simulations, we set the wavelength capacity $C = 16$. For each network topology, we consider several problem instances. For each problem instance, the traffic demand matrix $T = [t_{sd}]$ is generated by drawing the (integer) traffic demands uniformly at random in the interval $[0, t_{\max}]$. The values of $t_{\max}$ we used in the simulations are 20, 30, 40, and 50. Each data point in the figures we present in this section represents the average of 10 random problem instances for the stated values of the input parameters.

Unless otherwise stated, we set the relative optimality gap to 2% for all CPLEX runs. Consequently, CPLEX terminates when it finds a solution that is within 2% of the optimal for the problem at hand, rather than continuing until the problem is solved to optimality. Later in this section we will investigate how the running time required to solve the VTTR problem is affected by this optimality gap.

[2]We remind the reader that the VTTR problem does not account for the physical topology of the network.

A. VTTR-rlx(U_l,U_h): Solution Quality

Figure 2 plots the quality of the solution of VTTR-rlx(U_l, U_h) as a function of the pair of thresholds (U_l, U_h) and for various network sizes. The quality of the solution is defined as

$$\frac{\sum_{ij} \lceil \bar{b}^{\star}_{ij} \rceil}{\sum_{ij} b^{\star}_{ij}}. \tag{25}$$

The numerator in this expression is the value of the feasible solution to VTTR obtained by rounding up the lightpath variables in the optimal solution to the modified version of the VTTR-rlx(U_l, U_h) problem for the given value of a pair of thresholds. The denominator is the value of the objective function for the optimal solution to the VTTR problem (obtained within a 2% relative optimality gap, as we explained earlier). A low value of the above expression denotes a higher quality solution.

As expected, the quality of the solution starts away from optimal for VTTR-rlx [i.e., point (0,0) in Fig. 2], since all integer lightpath variables are relaxed; the solution quality then improves as U_l increases or U_h decreases. The best result is achieved for the threshold pair $(U_l, U_h) = 0.5, 0.6$. Importantly, for all network sizes shown in Fig. 2, the solution is about 11% from the optimal one as soon as $(U_l, U_h) = (0.5, 0.6)$; this worst case occurs for $N = 8$, and the gap decreases as the network size N increases. For the 32-node network, the gap is as small as 3%. Note that this pair of values for (U_l, U_h) ensures that no wavelength is underutilized (i.e., it is filled to 50% at minimum) while also leaving some room to accommodate future demands without necessarily setting up additional lightpaths.

B. VTTR-rlx(U_l,U_h): Scalability

Figure 3 plots the CPU time it takes to solve the VTTR-rlx(U_l, U_h) problem as a function of the thresholds U_l and U_h; note that the pair (0,0) in the figure corresponds to the solution of the relaxed problem VTTR-rlx. The figure

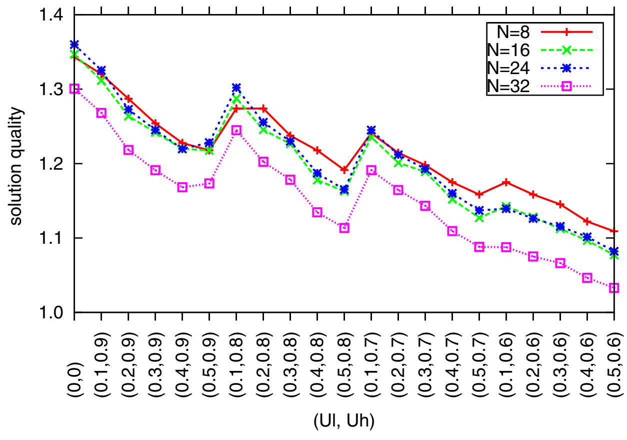

Fig. 2. Solution quality as a function of (U_l, U_h), $t_{\max} = 30$.

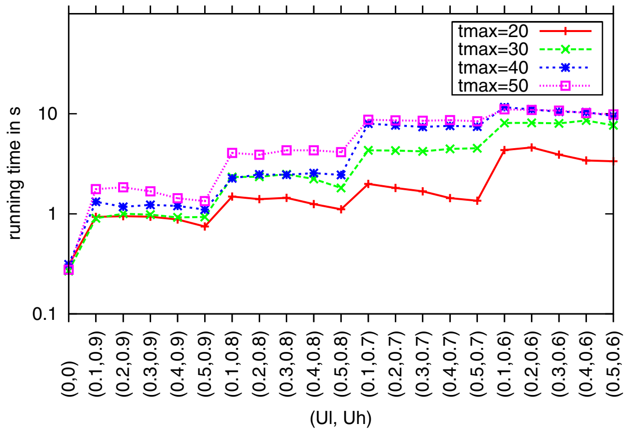

Fig. 3. CPU time as a function of (U_l, U_h), $N = 16$.

plots results for networks with $N = 16$ nodes and various values of $t_{\max}$. As expected, the running time generally increases as U_l increases or U_h decreases, since in both cases equality constraints are imposed on a larger number of lightpath variables. We also see that the running time is not significantly affected by the value of parameter $t_{\max}$. When $t_{\max}$ increases from 20 to 50, the running time only increases by a few seconds. This shows that the algorithm is effective across a range of traffic loads.

Based on the last observation, for the simulations in the remainder of this section we have fixed the value of $t_{\max} = 30$; with this value of $t_{\max}$, the average size of demands between any source–destination pair is close to the capacity $C = 16$ of a wavelength. Results for other values of $t_{\max}$ exhibit the same behavior and are omitted.

Figure 4 plots the CPU time to solve VTTR-rlx(U_l, U_h) as a function of U_l. We plot results for a 24-node network, as they are representative of the running time trends for other network sizes. As shown in the figure, when U_h is kept fixed, the running time does not change significantly as a function of the lower threshold value U_l. However, as U_h decreases, the running time increases significantly. In other words, the value of the high threshold U_h has a stronger influence on the performance of the algorithm with respect to running time.

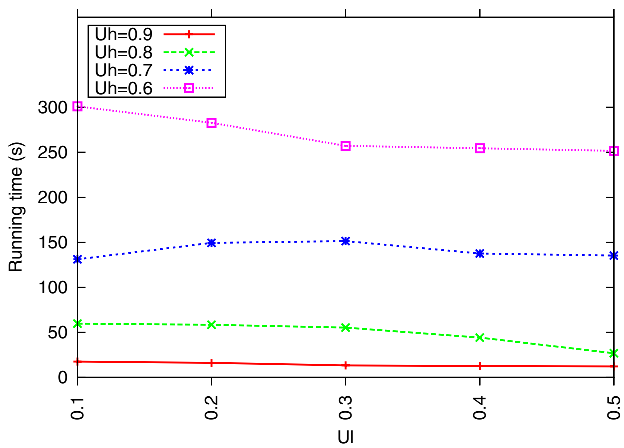

Fig. 4. CPU time as a function of U_l, $N = 24$, $t_{\max=30}$.

Figure 5 plots the running time as a function of the network size N. For this simulation, we set $(U_l, U_h) = (0.5, 0.6)$, the value that achieves the solution of highest quality (i.e., the smallest number of lightpaths), as we observed earlier. As we can see, it is possible to obtain a feasible solution to the VTTR problem for network sizes up to $N = 32$ within 1 h without sacrificing much in terms of optimality (as Fig. 2 indicates). Such problem instances are impossible to solve by directly using the original ILP formulation for the VTTR problem.

Finally, Fig. 6 compares the running time as a function of network size of three methods for solving the VTTR problem:

1) solving VTTR to optimality;
2) solving VTTR with a 2% relative optimality gap; and

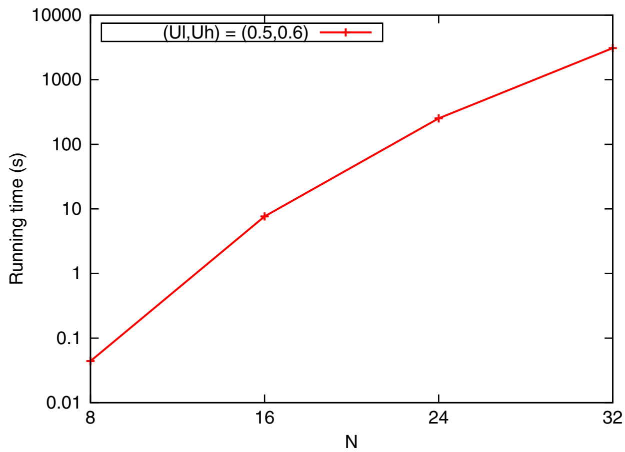

Fig. 5. CPU time as a function of N, $(U_l, U_h) = (0.5, 0.6)$, $t_{\max} = 30$.

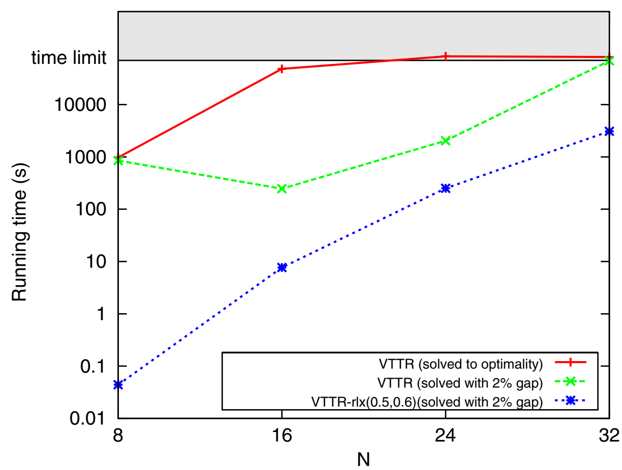

Fig. 6. CPU time comparison as a function of network size N, $t_{\max} = 30$.

3) solving VTTR-rlx(0.5, 0.6) with a 2% relative optimality gap.

All simulations are allowed to run for as long as 24 h. As we can see, the running time of the first method (i.e., directly solving the ILP formulation for the VTTR problem to optimality) is extremely long: within the 24 h time limit, this method can only solve network sizes up to 16 nodes. Setting the optimality gap to 2% (i.e., using the second method listed above) reduces the running time of finding a solution significantly, but even in this case it is not possible to solve 32-node networks within the 24 h time limit. Notice that with the second method, the running time for the 8-node network is longer than the time it takes to solve the 16-node network. This drop in running time when the network size increases from 8 to 16 can be explained by making the observation that with a fixed value of $t_{\max}$ ($= 30$ for these results), as the network size grows, so does the offered traffic and the optimal objective value (i.e., number of lightpaths). Hence, CPLEX may take a shorter time to reach a solution that is 2% from the optimal solution as the network size increases, as the absolute difference from the optimal solution is larger. Of course, as the network size increases further, the increase in the number of variables and constraints becomes once again the factor determining the running time—hence the increase as network size grows to 24 and beyond.

Finally, solving the modified VTTR-rlx(U_l, U_h) relaxed problem is significantly faster over all network sizes and reduces the running time by more than one order of magnitude compared to solving VTTR directly within a 2% optimality gap. In particular, the VTTR-rlx(U_l, U_h) problem can be solved on the 32-node network in about 3000 s (i.e., less than 1 h), while also obtaining a solution that is within 3% of the best solution (refer also to Fig. 2) that we were able to obtain after running the second method for 24 h. Note that in Fig. 6, we used $(U_l, U_h) = (0.5, 0.6)$ as the pair of thresholds for the algorithm; however, depending on the importance of solution quality relative to running time, other pairs of thresholds may be applied to further reduce the running time (as shown in Fig. 4).

Based on these results, we conclude that for small networks, e.g., of size between 8 and 10 nodes, the VTTR problem can be directly solved by using the MILP formulation with or without imposing a 2% optimality gap. However, for larger networks, solving the VTTR-rlx(U_l, U_h) problem we presented in Section IV is more efficient and effective. The particular pair of thresholds (U_l, U_h) to use may be fine tuned using the iterative algorithm we described in the previous section.

C. Wavelength Fragmentation

Let us now turn our attention to how sequentially solving the VTTR and RWA subproblems addresses the wavelength fragmentation challenge we discussed in Subsection II.B. Figure 7 is nearly identical to Fig. 1 in that it plots the number of wavelengths used by solutions to the TG problem for five problem instances on a six-node ring network; each solution is obtained by providing the stated number W of wavelengths as input to the TG formulation. In addition, Fig. 7 also includes the number of wavelengths obtained by sequentially solving the VTTR and RWA subproblems on each of the five instances. As we can see, the sequential solution uses fewer wavelengths than any of the solutions to the original TG problem. This result confirms our earlier observations that if the network is not wavelength limited, then not only is the sequential solution optimal in terms of the number of lightpaths (the objective of the TG problem), but it also minimizes the number of wavelengths required to establish these lightpaths.

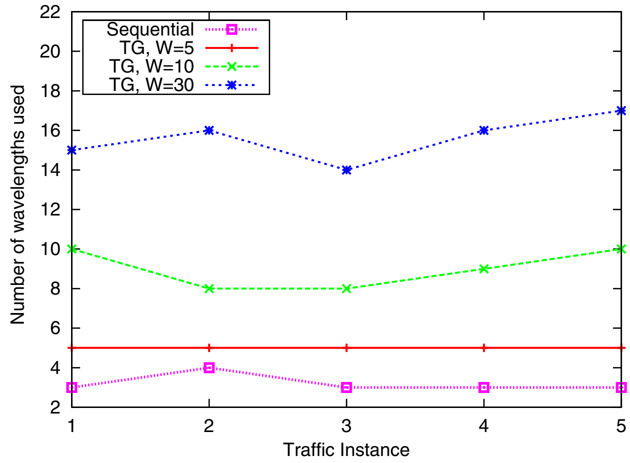

Fig. 7. Wavelength usage comparison, six-node ring network, $t_{\max} = 12$.

VI. Concluding Remarks

We have presented a new (to our knowledge) decomposition of the traffic grooming problem into two subproblems that are solved sequentially, the VTTR problem and the classical RWA problem. The decomposition is exact when the network is not wavelength limited, and minimizes the number of wavelengths used in the solution, avoiding the wavelength fragmentation issues of typical ILP formulations of the traffic grooming problem. We also developed an algorithm based on partial LP relaxation to solve the VTTR subproblem. The threshold parameters of the algorithm may be tuned to achieve a desired tradeoff between running time and quality of the final solution for VTTR. This decomposition approach to traffic grooming scales well to both ring and mesh network topologies and enables operators to carry out extensive "what-if" analysis in optimizing their network. We believe that the decomposition can be extended to variants of the grooming problem that take into consideration QoS constraints and survivability; such extensions are the subject of ongoing research in our group.

Acknowledgments

This work was supported by the National Science Foundation under grant CNS-1113191.

References

[1] F. Farahmand, X. Huang, and J. P. Jue, "Efficient online traffic grooming algorithms in WDM mesh networks with drop-and-continue node architecture," in *Proc. Broadnets*, Oct. 2004, pp. 180–189.

[2] J. Q. Hu and B. Leida, "Traffic grooming, routing, and wavelength assignment in optical WDM mesh networks," in *Proc. IEEE INFOCOM*, Mar. 2004, pp. 495–501.

[3] S. Huang and R. Dutta, "Research problems in dynamic traffic grooming in optical networks," in *Proc. First Workshop on Traffic Grooming*, Apr. 2004.

[4] H. Zhu, H. Zang, K. Zhu, and B. Mukherjee, "A novel generic graph model for traffic grooming in heterogeneous WDM mesh networks," *IEEE/ACM Trans. Netw.*, vol. 11, no. 2, pp. 285–299, Apr. 2003.

[5] S. Huang, R. Dutta, and G. N. Rouskas, "Traffic grooming in path, star, and tree networks: Complexity, bounds, and algorithms," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 4, pp. 66–82, Apr. 2006.

[6] R. Dutta and G. N. Rouskas, "Traffic grooming in WDM networks: Past and future," *IEEE Network*, vol. 16, no. 6, pp. 46–56, Nov./Dec. 2002.

[7] K. Zhu and B. Mukherjee, "Traffic grooming in an optical WDM mesh network," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 1, pp. 122–133, Jan. 2002.

[8] A. L. Chiu and E. Modiano, "Traffic grooming algorithms for reducing electronic multiplexing costs in WDM ring networks," *J. Lightwave Technol.*, vol. 18, no. 1, pp. 2–12, Jan. 2000.

[9] V. R. Konda and T. Y. Chow, "Algorithm for traffic grooming in optical networks to minimize the number of transceivers," in *IEEE Workshop on High Performance Switching and Routing*, 2001, pp. 218–221.

[10] R. Dutta and G. N. Rouskas, "On optimal traffic grooming in WDM rings," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 1, pp. 110–121, Jan. 2002.

[11] E. Yetginer and G. N. Rouskas, "Power efficient traffic grooming in optical WDM networks," in *Proc. IEEE GLOBECOM*, Dec. 2009.

[12] B. Chen, G. N. Rouskas, and R. Dutta, "On hierarchical traffic grooming in WDM networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 5, pp. 1226–1238, Oct. 2008.

[13] K. Zhu and B. Mukherjee, "A review of traffic grooming in WDM optical networks: Architectures and challenges," *Opt. Netw. Mag.*, vol. 4, no. 2, pp. 55–64, Mar./Apr. 2003.

[14] W. Yao, G. Sahin, M. Li, and B. Ramamurthy, "Analysis of multi-hop traffic grooming in WDM mesh networks," in *Proc. Broadnets*, Boston, MA, Oct. 2005, pp. 165–174.

[15] X. Munoz and I. Sau, "Traffic grooming in unidirectional WDM rings with bounded degree request graph," in *Graph-Theoretic Concepts in Computer Science* (Lecture Notes in Computer Science, Vol. 5344). Springer, 2008, pp. 300–311.

[16] M. Dawande, R. Gupta, S. Naranpanawe, and C. Sriskandarajah, "A traffic-grooming algorithm in wavelength-routed optical networks," *INFORMS J. Comput.*, vol. 19, no. 4, pp. 565–574, 2007.

[17] B. Vignac, B. Jaumard, and F. Vanderbeck, "A hierarchical optimization approach to optical network design where traffic grooming and routing is solved by column generation," in *Proc. INOC*, Apr. 2009.

[18] J.-F. P. Labourdette and A. S. Acampora, "Logically rearrangeable multihop lightwave networks," *IEEE Trans. Commun.*, vol. 39, no. 8, pp. 1223–1230, Aug. 1991.

[19] E. Yetginer, Z. Liu, and G. N. Rouskas, "Fast exact ILP decompositions for ring RWA," *J. Opt. Commun. Netw.*, vol. 3, no. 7, pp. 577–586, July 2011.

[20] Z. Liu and G. N. Rouskas, "A fast path-based ILP formulation for offline RWA in mesh optical networks," in *Proc. IEEE GLOBECOM*, Dec. 2012.

[21] Z. Liu and G. N. Rouskas, "Link selection algorithms for link-based ILPs and applications to RWA in mesh networks," in *Proc. ONDM*, Apr. 2013.

Hui Wang is a Ph.D. degree candidate in operations research, with co-major in computer science, at North Carolina State University (NCSU). She received a B.S. degree in applied mathematics from Peking University, Beijing, China, in 2007, and master's degrees in statistics and in operations research from NCSU in 2011 and 2009, respectively. Her research interests include optical networks, network design and optimization, and performance evaluation.

George N. Rouskas is a Professor of Computer Science at North Carolina State University and an IEEE Fellow. He received a B.S. degree in computer engineering from the National Technical University of Athens (NTUA), Athens, Greece, and M.S. and Ph.D. degrees in computer science from the College of Computing, Georgia Institute of Technology, Atlanta, GA. His research interests include network architectures and protocols, optical networks, network design and optimization, and performance evaluation. He is co-editor of the book *Next-Generation Internet Architectures and Protocols* (Cambridge University, 2011), author of the book *Internet Tiered Services* (Springer, 2009), and co-editor of the book *Traffic Grooming for Optical Networks* (Springer, 2008). He is a founding co-editor-in-chief of the *Optical Switching and Networking* journal, and he has served on the editorial boards of *IEEE/ACM Transactions on Networking*, the IEEE/OSA *Journal of Optical Networking*, *Computer Networks*, and *Optical Networks*. He is the General Chair for ICCCN 2013, and he has served as TPC or general chair for several conferences, including ICCCN 2011, the IEEE GLOBECOM 2010 ONS Symposium, BROADNETS 2007, IEEE LANMAN 2004 and 2005, and IFIP NETWORKING 2004. He is the recipient of a 1997 NSF CAREER Award, the 2004 ALCOA Foundation Engineering Research Achievement Award, and the 2003 NCSU Alumni Outstanding Research Award, and he was inducted into the NCSU Academy of Outstanding Teachers in 2004. He serves as the Secretary of the IEEE Optical Networking Technical Committee (ONTC) in 2012-2013, and he served as a Distinguished Lecturer for the IEEE Communications Society in 2010-2011.