Bounds on Traffic Grooming in Star and Tree Networks *

Rudra Dutta, George N. Rouskas
Department of Computer Science, North Carolina State University, Raleigh, NC 27695-7534
{rdutta,rouskas}@csc.ncsu.edu

September 18, 2001

Abstract

We consider the problem of grooming traffic in WDM star and tree networks to minimize equipment cost. We prove an important result regarding wavelength assignment in both topologies. We present a series of lower and upper bounds on the optimal solutions for both star and tree networks. The bounds allow us to evaluate a set of heuristics we also develop.

1 Introduction

Wavelength division multiplexing (WDM) technology has the potential to satisfy the ever-increasing bandwidth needs of network users on a sustained basis. In WDM networks, nodes are euipped with *optical cross-connects (OXCs)*, devices which can optically switch a signal on a wavelength from any input port to any output port, making it possible to establish *lightpath* connections between any pair of network nodes. The set of lightpaths defines a *logical topology*, and the problem arises of designing logical topologies that optimize a performance measure of interest for a set of traffic demands. This problem has been studied extensively; for a detailed discussion see [1]. Typically, the objective has been to minimize the number of wavelengths, to optimize a network-wide metric (such as delay or congestion), or a combination of the two.

With the deployment of commercial WDM systems, it has become apparent that the cost of network components, especially line terminating equipment (LTE), is the dominant cost and is a more meaningful metric to optimize than, say, the number of wavelengths. Since the data rates at which each wavelength operates continue to increase, it is clear that a number of independent traffic components must be multiplexed in order to efficiently utilize the wavelength capacity. *Traffic grooming* [2] refers to the techniques used to combine lower speed components onto wavelengths in order to minimize cost. Given the widespread use of SONET/SDH networks, early work has focused on ring topologies [3]-[9].

In this paper we consider the problem of traffic grooming in tree networks, using star networks as building blocks. Despite their simplicity, these topologies are important in their own right: star networks arise in the interconnection of LANs or MANs with a wide area backbone [10], while passive optical networks (PONs) [11, 12] and cable TV networks (which are increasingly used for high-speed Internet access) are based on a tree topology. Our work can thus be applied directly to these environments. Also, we may tackle the traffic grooming problem in mesh networks by decomposing them into stars, trees, and paths. While such a decomposition is outside the scope of this paper, the study of star and tree networks can provide insight into the general problem.

Traffic grooming remains NP-hard even for star or tree networks, and our focus is on developing bounds and heuristics. Our main results are as follows. First, we prove that in WDM networks with a star or tree topology, wavelength assignment can be performed in polynomial

*This work was supported by the NSF under grant NCR-9701113.

time for any feasible logical topology, thus it can be eliminated simplifying the problem. For stars, we obtain a sequence of lower and upper bounds which permit a tradeoff between the quality of the solution and the computational requirements; the sequence of upper bounds yields an approximation algorithm for the traffic grooming problem. For trees, we also present lower and upper bounds. For both topologies we present results which indicate that the performance of simple greedy heuristics is quite good.

In Section 2 we define the traffic grooming problem and present the results on wavelength assignment. In Section 3 we develop upper and lower bounds on the optimal solution as well as heuristics for tree networks. We present numerical results in Section 4, and we conclude in Section 5.

2 Problem Definition

2.1 Star Network

We consider a network in the form of a star $\mathcal{S}$ with $N+1$ nodes. There is a single *hub* node which is connected to every other node by a physical link. The N nodes other than the hub are numbered from 1 to N in some arbitrary order, and the hub node is numbered 0. Each physical link consists of a fiber in each direction, and each fiber can carry W wavelengths. The traffic demands between each pair (s,d) of nodes are given in the matrix $T = [t^{(sd)}]$. The network carries traffic at rates that are a multiple of some basic rate (e.g., OC-3). We let C denote the capacity of each wavelength expressed in units of this basic rate. Parameter C is also known as the *granularity* of traffic. For example, if each wavelength runs at a rate of OC-48 and the basic rate is OC-3, then $C=16$. Each quantity $t^{(sd)} \in \{0,1,2,\cdots\}$ is also expressed in terms of the basic rate, and it denotes the number of traffic units that originate at node s and terminate at node d.

We are interested in designing a *logical* topology by establishing *lightpaths* between pairs of nodes over the physical star network. We assume that no traffic component can traverse the same physical link more than once in either direction. This assumption ensures that bandwidth is not used up without bound by a single traffic component. Consequently, no node except for the hub will switch traffic, either electronically or optically. In other words, the hub is the only node which sees traffic neither originated by, nor destined to, it. The hub node is equipped with an OXC with N incoming and N outgoing ports. Thus there will be only two kinds of lightpaths in the logical topology: single-hop lightpaths which either originate at a non-hub node and terminate at the hub, or vice versa; and two-hop lightpaths that originate and terminate at non-hub nodes, and are switched optically at the hub. We assume that wavelength converters are not available, therefore, a lightpath must be assigned the same wavelength on all links along its path. We also allow for multiple lightpaths between the same source and destination nodes.

In a star topology it is straightforward to determine how much traffic will flow over each fiber for a given traffic matrix. If this aggregate traffic is more than the total bandwidth (WC) of the fiber at any of the links, the problem instance is obviously not feasible. Conversely, if the aggregate traffic originating and terminating at each non-hub node is at most WC, then the traffic can always be carried by a logical topology with W single hop lightpaths on each direction of each physical link and no two-hop lightpaths; thus the problem instance is feasible. We call this the *completely opaque* topology, since all the traffic between any two nodes must be electronically switched at the hub node.

While the completely opaque topology can be used to carry any feasible traffic demand, it has several drawbacks. First, it introduces huge demands in the processing and buffering capacity of the hub. Second, the hub node must be equipped with a number of LTE equal to NW. Since LTE are one of the most expensive components in optical networks, realizing the completely opaque topology may be prohibitively expensive. (Note that non-hub nodes must be equipped with a sufficient number of LTE to handle their own traffic, but these LTE are required regardless of the topology, opaque or transparent.) Finally, the completely opaque topology is not scalable, since upgrading the network (by adding more nodes/wavelengths per fiber) requires adding more capacity/LTE at the hub node.

In [13], we have considered the problem of traffic grooming in star networks in order to minimize the network cost. We have formulated the traffic grooming problem in stars as an integer linear program (ILP), conjectured that the problem remains sufficiently general for an exact solution to be computationally intractable, and described an approximation scheme to obtain successively better lower and upper bounds on the optimal solutions, as well as successively better heuristic solutions.

2.2 Tree Network

A star can be viewed as a special case of the more general tree topology. We extend our observations above to a physical topology in the form of a tree $\mathcal{T}$ with N nodes. We distinguish between *leaf* nodes which have degree one, and *interior* nodes which have degree greater than one. As with the star, each edge of the undirected tree consists of two fiber links, one in each direction. The number of wavelengths W, the granularity C, and the traffic matrix T have similar significance as for the star topology.

We again make the assumption that a traffic component is allowed to traverse the same physical link at most once in any direction. Therefore, a traffic component from a given source node to a given destination node will follow a unique path through the tree. Interior nodes are equipped with OXCs, and are the only nodes that switch traffic, electronically or optically. Leaf nodes are similar to non-hub nodes in the star in that they never handle any traffic other than that originated or terminated by them.

Because of the unique routing, feasibility of a problem instance can be determined by summing up the traffic flowing over each directed link. If this total exceeds the capacity WC for any fiber, then the problem instance is not feasible; otherwise, it is feasible because it can be carried by the completely opaque topology with W single-hop lightpaths over each physical link in each direction.

The problem of grooming traffic in order to minimize the total amount of electronic switching in the tree can also be formulated as an ILP. Because a star is a special case of a tree, it follows that the full virtual topology problem is intractable for the tree topology as well.

2.3 Wavelength Assignment

In general, wavelength assignment is a hard subproblem of the virtual topology problem [1]. However, in [9] we showed that wavelength assignment is simple for path networks. The two lemmas below show that wavelength assignment is straightforward for star and tree networks.

Lemma 2.1 *Consider a virtual topology on a star, in which the maximum number of lightpaths on any link is L. A wavelength assignment that uses exactly L wavelengths exists and can be obtained in time polynomial in N and L.*

Lemma 2.2 *Consider a virtual topology on a tree, in which the maximum number of lightpaths on any link is L. A wavelength assignment that uses exactly L wavelengths exists and can be obtained in time polynomial in N and L.*

The proofs of the lemmas are quite involved, but due to the page limit we cannot include them here; they can be found in [13]. The proof of Lemma 2.1 uses a corollary of Hall's matching theorem for L-regular bipartite multigraphs. The proof of Lemma 2.2 provides an algorithm which performs wavelength assignment by decomposing the tree into stars and applying the result of Lemma 2.1

The implication of Lemmas 2.1 and 2.2 is that once a feasible topology that minimizes electronic switching has been obtained for a star or tree network, assigning wavelengths to the lightpaths can be performed in polynomial time. While the traffic grooming problem remains NP-hard, the lemmas make the development of bounds and of heuristics a bit easier, since they allow us to eliminate the wavelength assignment subproblem from consideration.

3 Heuristics and Bounds for Tree Networks

3.1 Star Network Results

We have obtained bounds and heuristic solutions for star networks in [13]. The search space of the problem is quite large: for N non-hub nodes, each of the $N(N-1)$ traffic components may be either electronically switched at the hub or optically bypass it, and a brute-force algorithm would have to evaluate a space of $2^{N(N-1)}$ combinations. We described a search tree of partial and complete solutions to the star network problem instance, and presented an algorithm which visited each valid solution no more than once while avoiding any invalid solutions. We employed pruning techniques similar to branch-and-bound searches, as well as pruning specific to the problem, to make the search efficient. Most importantly, the incremental nature of the algorithm allows practical benefits to be obtained without completing the exhaustive search. In particular, at any intermediate stage of the generation of the full search tree, we can extract a lower bound and an upper bound on the optimal solution. The upper bound is based on a feasible (suboptimal) solution to the problem instance, providing a heuristic solution corresponding to any intermediate search tree. We defined Ψ_i and Φ_i to be the upper and lower bounds obtained, respectively, at the intermediate stages of the search tree in which the search tree is generated completely upto depth i, but not beyond, and showed that $\{\Psi_i\}$ and $\{\Phi_i\}$ are strong sequences of bounds; that is, successive bounds in each sequence are at least as strong as the previous ones, as i increases. Finally, we also showed that with the proper choice of parameters, the bounds approach each other (and the optimal, which is bracketed) with provably good characteristics. In particular, for the i-th upper and lower bound, we have:

$$\Psi_i - \Phi_i \leq \left(1 - \frac{i}{N(N-1)}\right) \left(\sum_{s,d \in \{1,\cdots,N\}, s \neq d} t^{(sd)} \right) \tag{1}$$

We also discussed a greedy algorithm that performs very well, and showed how to combine the benefits of the heuristics above with provably good characteristics and those of the greedy algorithm. The solution obtained at depth i of the search tree after enhancing $\{\Psi_i\}$ with the greedy algorithm was denoted by $\Psi_i^{(g)}$, and was shown to be at least as good as that without the greedy enhancement.

3.2 Decomposition into Star Networks

The leaf nodes of a tree network do not route traffic either electronically or optically, and we concentrate on the interior nodes. Consider an interior node p of tree $\mathcal{T}$, and the set of nodes $\{q_1, q_2, \cdots, q_n\}$ adjacent to p in $\mathcal{T}$. We define the $(n+1) \times (n+1)$ matrix $T^{(\mathcal{S}_p)} = [\tau_{ij}^{(p)}]$ as:

$$\tau_{ij}^{(p)} = \begin{cases} \sum t^{(sd)} \text{ over } (s,d) : s \neq p, d \neq p, t^{(sd)} \text{ traverses} \\ \quad \text{links } (q_i, p), (p, q_j), \forall i \neq j \in \{1, \cdots, n\} \\ \sum t^{(sp)} \text{over } s : s \neq p, t^{(sp)} \text{traverses the} \\ \quad \text{link } (q_i, p), \forall i \in \{1, \cdots, n\}, j = 0 \\ \sum t^{(pd)} \text{ over } d : d \neq p, t^{(pd)} \text{ traverses the link} \\ \quad (p, q_j), \forall j \in \{1, \cdots, n\}, i = 0 \\ 0, \quad \text{otherwise} \end{cases} \tag{2}$$

This matrix represents the traffic of the tree $\mathcal{T}$ seen from the point of view of interior node p. Now consider $T^{(\mathcal{S}_p)}$ as the traffic matrix for a star network $\mathcal{S}_p$. The hub node of this star network sees exactly the same traffic scenario as that seen by node p in the tree network, and we speak of $\mathcal{S}_p$ as being the "decomposed star network" for node p.

In the star, no node other than the hub does any electronic or optical routing. Thus, the optimal value of electronic routing for the star denotes the optimal (minimum) value of the electronic routing by the hub node of the star only. Since node p is locally in the same traffic scenario as the hub of its decomposed star network, this is the minimum amount of electronic routing that node p can perform in the tree $\mathcal{T}$ under any virtual topology and traffic grooming solution. We denote this quantity by $\phi_{\mathcal{T}}(p)$, thus $\phi_{\mathcal{T}}(p)$ is the value of electronic routing that would be obtained by solving the decomposed star $\mathcal{S}_p$ optimally. We also note that $\{\Phi_i\}$ and $\{\Psi_i^{(g)}\}$ for the star $\mathcal{S}_p$ are upper and lower bounds on its optimal electronic routing value and hence on $\phi_{\mathcal{T}}(p)$.

3.3 Lower Bounds

Since the amount of electronic routing performed by the different interior nodes of the tree are disjoint quantities, the quantity $\sum_p \phi_{\mathcal{T}}(p)$ is a lower bound on the amount of total electronic routing performed in any virtual topology, and hence on the optimal value of electronic routing in the tree. However, solving the decomposed star network may not be practically possible for an internal tree node p. In other words, the quantity $\phi_{\mathcal{T}}(p)$ may not be readily available. However, using our method of successively better approximations for stars, we can obtain as good a lower bound on $\phi_{\mathcal{T}}(p)$ as the computation required makes practical. Let us denote the actual lower bound used on $\phi_{\mathcal{T}}(p)$ by $\phi_{\mathcal{T}}^{(i)}(p)$. This quantity is chosen out of the quantities $\{\Phi_i\}$ for the decomposed star network of the node p. The closeness with which we approach $\phi_{\mathcal{T}}(p)$ may be different for every p. Now the quantity $\sum_p \phi_{\mathcal{T}}^{(i)}(p)$ is still a lower bound on the optimal electronic routing for tree $\mathcal{T}$, though a weaker one. However, such a bound requires much less computation to determine. Thus, we can consider the lower bounds $\{\sum_p \phi_{\mathcal{T}}^{(i)}(p)\}$ to be a sequence of bounds on the optimal electronic routing for the tree network, where each $\phi_{\mathcal{T}}^{(i)}(p) = \Phi_i$ for the decomposed star of p is an increasingly better lower bound on $\phi_{\mathcal{T}}(p)$. Since $\{\Phi_i\}$ for the star is a strong sequence of bounds, so is the sequence of bounds for the tree that we have described.

3.4 Upper Bounds and Heuristic Solutions

We now show how to obtain a feasible solution to the tree network using the solutions for the star network. We call an interior node of the tree network opaque if it routes all traffic electronically. (Conversely, if a node performs optical routing without any restriction other than traffic and wavelength constraints, we call it a *transparent* node.) As with the star, we can create a feasible virtual topology in which no node routes any traffic optically. All traffic at all interior nodes is routed electronically, creating a *completely opaque* topology as before. Since this is a feasible topology, the amount of electronic routing performed in this topology is an upper bound on the optimal; in fact, it is the loosest such bound because there is no virtual topology in which more electronic routing will need to be performed. Let the amount of electronic routing an interior node p does as an opaque node in the tree be $\psi_{\mathcal{T}}(p)$. Then, the completely opaque upper bound is given by $\Psi_{\mathcal{T}} = \sum_p \psi_{\mathcal{T}}(p)$.

However, realistically we would like to use the optical routing capability of the nodes and create a solution to the tree network in which the amount of electronic routing to be performed is reduced from the maximum at least at some nodes. Recall that $\phi_{\mathcal{T}}(p)$ is the minimum amount of electronic routing node p can do locally. However, to attain this value, the traffic to/from other nodes from/to node p must be groomed according to the optimal solution to $\mathcal{S}^{(p)}$. For two interior nodes p and q which are adjacent, it will not in general be possible to simultaneously attain $\phi_{\mathcal{T}}(p)$ and $\phi_{\mathcal{T}}(q)$ as electronic routing values, because the optimal solutions of the two decomposed stars will in general require the same traffic component in the tree to be differently groomed. For this reason, the lower bound we derived in the last section will in general be unattainable.

To examine what combinations of star decompositions may nevertheless be useful in creating feasible solutions for the tree network, consider a decomposed star network for an interior node p. The hub node corresponds to p, whereas the other nodes of the star correspond to the nodes of the tree that are adjacent to p in $\mathcal{T}$. Some of these nodes may be leaf nodes of the tree, in which case the solution to the decomposed star may be transferred to the tree without any change. However, in general some of the non-hub nodes of the star will be other interior nodes of the tree, and will have their own star decompositions. To create a feasible solution to the tree, we must adopt some method of reconciling the star solutions for adjacent interior nodes of the tree. Below we propose two methods of doing this.

3.4.1 Solution with Opaque Nodes

An opaque node electronically routes all traffic that passes through it. While this is wasteful in terms of electronic routing, an opaque node optically terminates and originates all traffic, so that the traffic components can be rearranged and reassigned to lightpaths arbitrarily. It is easy to see that the conflict between star solutions to adjacent interior nodes does not arise if the decomposed star for one of the interior nodes is solved optimally while the other one is left as an opaque node. In other words, if we interpose at least one opaque node between every two transparent nodes of the tree (for which we solve the decomposed star optimally), then there is no problem in combining the corresponding star solutions.

In such a solution, each node p performs either $\phi_{\mathcal{T}}(p)$ amount of electronic routing (the best possible), or $\psi_{\mathcal{T}}(p)$ (the worst). For the best topology which utilizes a combination of transparent and opaque nodes, we would like to choose the nodes such that we get greatest benefit in terms of electronic routing. Ideally, we would like to find the set of nodes N_t to be designated as transparent nodes, (composed of pairwise non-adjacent interior nodes) such that $\sum_{p \in R}(\psi_{\mathcal{T}}(p) - \phi_{\mathcal{T}}(p))$ is maximized. However, this is equivalent to finding a maximal independent set in a graph, which is NP-complete [14]. An efficient way to pick N_t is to utilize the level ordering of the tree $\mathcal{T}$. Designate any interior node r as the root of the tree. We partition the interior nodes of the tree into two sets, N_0 and N_1, such than N_0 contains all interior nodes which are at even depth of the tree from the root r (including r itself), and N_1 contains all interior nodes at odd depth. Now either of the sets N_0 and N_1 may be used as the set N_t of transparent nodes, and the other as the set of opaque nodes. Since every adjacent node to a node $p \in N_0$ is from N_1 and vice versa, it is obvious that any choice for the root r will yield the same two sets N_0 and N_1, albeit possibly exchanged.

This may seem like a quite arbitrary method of determining N_t, but actually this is a fairly good approximation algorithm. To see this, consider that if N_0 is designated to be N_t, each node p_t in N_0 will perform electronic routing to the amount of $\phi_{\mathcal{T}}(p_t)$ only, representing a benefit of $\sum_{p_t \in N_0}(\psi_{\mathcal{T}}(p_t) - \phi_{\mathcal{T}}(p_t))$ for the whole set N_0. We call this the "benefit" of the set of nodes N_0 and denote it by $B(N_0)$. We define $B(N_1)$ in a similar manner. Now the completely opaque solution incurs electronic routing to the amount of $\Psi_{\mathcal{T}}$ as defined above, and the lower bound we derived in Section 3.3 (possibly unattainable) shows that the maximum saving in electronic routing for the entire tree is bounded by $\sum_p(\psi_{\mathcal{T}}(p) - \phi_{\mathcal{T}}(p)) = B(N_0) + B(N_1)$. By choosing N_t to be the set among N_0 and N_1 with the larger benefit, we are guaranteed to approach the optimal at least by 50%.

3.4.2 Solution with Semi-Opaque Nodes

While we obtain a good solution by solving the tree with alernating opaque and transparent nodes, we may miss opportunities to groom traffic at the opaque nodes. Also, the amount of electronic routing is distributed unevenly, maximum possible for some nodes and minimum possible for others. We now propose an approach which does not share these characteristics. The key observation is that if we impose the solution obtained from a decomposed star for a node p in the tree, another interior node q adjacent to p needs to be opaque in the solution to the tree *with respect to* p, but not necessarily to other nodes.

Designate an interior node r as the root of the tree. Let r be a transparent node, and impose the solution of the decomposed star $\mathcal{S}_r$ on the tree network. For each child p of r, create a decomposed star after the fashion of Section 3.2, with the following difference: every traffic component to and from the non-hub node corresponding to r to any other non-hub node q is constrained to be electronically routed at the hub node. A traffic component in the tree which would normally be represented by a traffic component from q to r in the star decomposition for p is now represented by two traffic components, each of the same magnitude as the original, one from q to the hub p and another from p to r. In the optimal solution to such a star, there will be no lightpaths formed to/from r that pass optically through p, because there is no traffic for such a lightpath to carry. At the same time, the traffic scenario locally seen by node p has been preserved, under the assumption that the traffic to/from node r cannot be optically routed. We call this the star decomposition of node p constrained by node r, and we refer to node p as a *semi-opaque* node. The optimal solution to such a star can be implemented without any conflict with the optimal solution for the decomposed star for r in the tree network. Similarly, we would create star decompositions for each child q of p constrained by p, and so on down the tree. All the constrained star networks will be consistent, that is, it will be possible to implement the solutions in the tree network without conflict.

Although the solution with semi-opaque nodes is likely to distribute the electronic routing load more evenly through the tree, it is difficult to characterize the total amount of electronic routing performed and set meaningful bounds on the improvement over the completely opaque topology. We note that the choice of the root r characterizes the solution (because r is the only interior node that is transparent, the others being semi-opaque). Thus, the algorithm can be repeated with each interior node designated as a root and the best solution adopted, increasing the algorithm complexity by a linear factor.

In our solution to the tree, we require the optimal amount of electronic routing for the decomposed star (possibly constrained by another node) to be available. If the amount of computation required precludes determining the optimal value for a star network, then the best available upper bound from the sequence of bounds $\{\Psi_i\}$ or $\{\Psi_i^{(g)}\}$ may be used instead in the tree solution. For obvious reasons, the solutions obtained for the tree network will still be feasible, and the total electronic routing values will still represent upper bounds (though less tight). As tighter and tighter upper bounds for the optimal star solutions are used (at progressively greater computational costs), the upper bound obtained for the tree network will also grow tighter.

3.5 Greedy Heuristics

In this section we describe two greedy heuristics for the tree network. Because the sequence of feasible solutions we have proposed above get progressively more costly to compute, efficient greedy heuristics would be valuable; and because our solutions never form lightpaths of more than two hops, greedy heuristics have a good chance of outperforming them in specific cases.

Both heuristic algorithms start by reducing the traffic matrix as described in Section 3.4, then ordering the reduced traffic elements in descending order and attempting to optically route the traffic components in this order. The first algorithm, which we call "Greedy-A", attempts to route a traffic component optically at each intermediate node along its path. If this fails for a traffic component (because there is not sufficient bandwidth at some intermediate link to accomodate the rest of the traffic that must flow over than link, if this traffic component is given a lightpath), then we abandon that traffic component, consigning it to be electronically routed at each intermediate node, and go on to the next traffic component in the greedy ordering. The algorithm terminates when all traffic components have been examined.

The second algorithm, called "Greedy-B', does not abandon a traffic component if it cannot be optically routed at every intermediate node, but rather takes a "best-effort" approach, optically routing it whenever possible, electronically routing it otherwise. Thus, if a traffic component cannot be optically routed along its entire path, Greedy-A will leave it to be carried on single hop ligthpaths on each of the links it traverses, but Greedy-B may form some lightpaths covering part of the path. It may appear that Greedy-B makes more of an effort to optically route traffic and is guaranteed to give better results overall, but this is not true in general. When Greedy-A leaves a traffic component not optically routed at an intermediate node that Greedy-B would have routed it, some extra bandwidth remains that may lead to more efficient grooming for some traffic components later in the greedy ordering. Thus, depending on the problem instance, either algorithm might outperform the other.

4 Numerical Results

We now present a sampling of the results we have obtained for star and tree networks; for a more complete set of results the reader is referred to [13].

4.1 Star Networks

We characterize a traffic matrix for a star network by two parameters: the *loading factor* and the amount of *hub traffic*. The loading factor is the sum of all the traffic components expressed as a percentage of the total bandwidth available in the network (i.e., the total bandwidth of all the fiber links). For low values of loading, the network is underutilized; such networks are not interesting as it is likely that every traffic component can be given a lightpath. For 100% loading, only traffic components equal to C can be given a lightpath, all other traffic must be electronically switched at the hub. The interesting and most realistic operating condition is when the loading factor is just under 100%, so that opportunities for grooming exist without the problem being trivial; thus, we present results for a loading factor of 90%. The amount of hub traffic is the (average) fraction of the total traffic on each link that is accounted for by traffic to and from the hub. We present results for two values of the hub traffic, 30% and 60%.

For the results shown in Figures 1-2 we have used $W = 24$ wavelengths, $C = 16$, and the number of non-hub nodes $N = 8, 10$. The figures plot the *grooming effectiveness* of the series of upper and lower bounds against the level i of the search tree. The grooming effectiveness is the total amount electronic routing expressed as a fraction of the amount of electronic routing for the completely opaque virtual topology. Three curves are plotted, one for the series $\{\Phi_i\}$ of lower bounds, one for the series $\{\Psi_i\}$ of upper bounds, and one for the series $\{\Psi_i^{(g)}\}$ of upper bounds (denoted as "Greedy enhanced" in the figures) computed by applying the greedy algorithm to complete a mask matrix rather than the pessimistic completion used for $\{\Psi_i\}$. We observe that the sequence of upper and lower bounds do indeed converge to the optimal relatively quickly. For $N = 8, 10$, we were able to reach the optimal within a few minutes of computation on a SUN Sparc-10 workstation. In fact, the optimal is reached before all levels of the search tree are considered (for $N = 8$ the maximum number of levels is 56, and for $N = 10$ it is 90). We also observe that the series $\{\Psi_i^{(g)}\}$ of upper bounds outperforms the series $\{\Psi_i\}$.

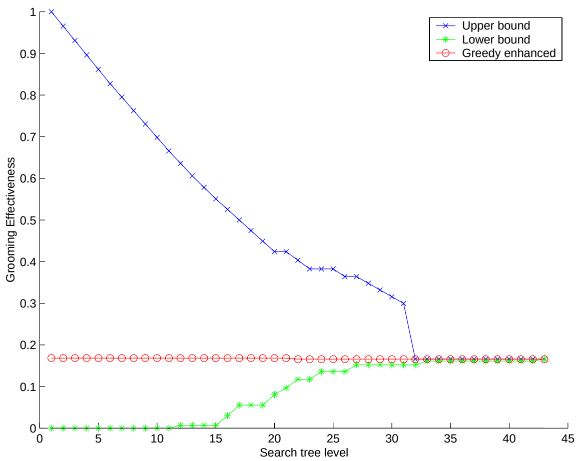

Figure 1: Star result: $N = 8$, 60% hub traffic

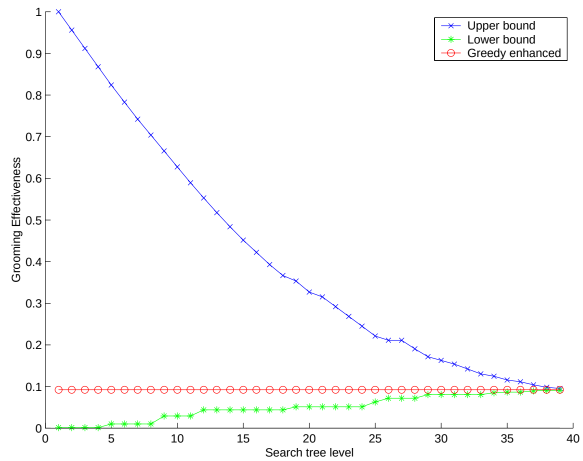

Figure 2: Star result: $N = 10$, 30% hub traffic

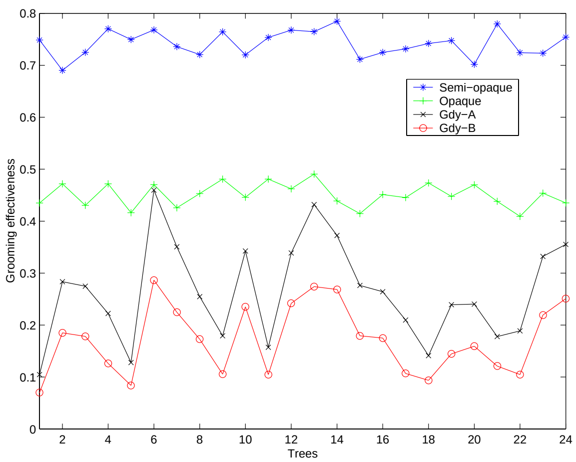

Figure 3: Ensemble of tree networks: $C = 16$

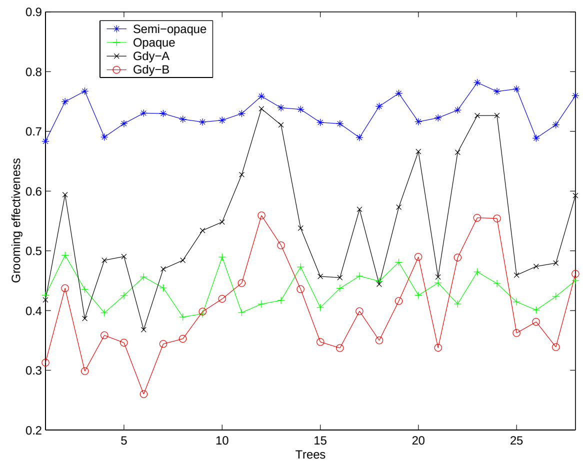

Figure 4: Ensemble of tree networks: $C = 32$

4.2 Tree Networks

We generated trees such that each interior node has between 2 and 7 adjacent nodes. Thus the number of leaf nodes (likely to represent trafic endpoints) is a large fraction of the total number of nodes. We considered matrices with high loading because these are of interest, as before, and traffic components were drawn from distributions with means inversely proportional to the path length. We used values of W between 70-200, and of C between 16-48.

Figures 3 and 4 plot the grooming effectiveness of the solutions using opaque or semi-opaque nodes, and the two greedy heuristics, Greedy-A and Greedy-B. Each figure plots results for 25 tree networks, Figure 3 for $C = 16$ and Figure 4 for $C = 32$. The solution to trees which combines star networks with opaque nodes reduces electronic routing by at least half, and it consistently outperforms the solution using semi-opaque nodes. The greedy heuristics perform very well in all cases. However, since they are both based on giving full lightpaths to individual traffic components, it is not surprising that with a larger granularity of traffic to be groomed (see Figure 4 with $C = 32$), the greedy heuristics start to show less advantage over the decomposition methods, which are able to groom different traffic components into the same lightpath. That is, the extra computation required to compute the feasible solutions combining star networks is likely to be justified by a greater gain in grooming as the granularity increases and the grooming problem becomes more difficult. Greedy-B is seen to consistently outperform Greedy-A. This result may be expected since Greedy-B generally requires a significantly larger amount of computation than Greedy-A.

Overall, these results indicate that significant gains in terms of electronic routing can be achieved by appropriate traffic grooming.

5 Concluding Remarks

We have considered the traffic grooming problem in WDM star and tree topologies with the objective of minimizing the amount of network-wide electronic routing. We showed that we can eliminate the wavelength assignment subproblem and concentrate on simply finding good feasible topologies. We have obtained lower and upper bounds on the objective function for both star and tree networks, and we have presented a set of heuristics that perform well across a wide range of traffic patterns and loads. We are currently investigating the application of these results to the traffic grooming problem in general mesh topologies.

References

[1] R. Dutta and G. N. Rouskas, "A survey of virtual topology design algorithms for wavelength routed optical networks," *Optical Networks*, vol. 1, no. 1, pp. 73–89, January 2000.

[2] R. A. Patterson, E. Rolland, R. Barr, and G. W. Ester (Eds.), "Special issue on telecommunications grooming," *Optical Networks*, vol. 2, no. 3, May/June 2001.

[3] X. Zhang and C. Qiao, "On scheduling all-to-all personalized connections and cost-effective designs in WDM rings," *IEEE/ACM Trans. Networking*, vol. 7, pp. 435–445, June 1999.

[4] R. Berry and E. Modiano, "Minimizing electronic multiplexing costs in SONET/WDM rings with dynamically changing traffic," *IEEE JSAC*, vol. 18, no. 10, pp. 1961–1971, 2000.

[5] P-J. Wan, G. Calinescu, L. Liu, and P. Frieder., "Grooming of arbitrary traffic in SONET/WDM BLSRs," *IEEE JSAC*, vol. 18, no. 10, pp. 1995–2003, 2000.

[6] O. Gerstel, P. Lin, and G. Sasaki, "Wavelength assignment in a WDM ring to minimize cost of embedded SONET rings," in *IEEE INFOCOM*, 1998, pp. 94–101.

[7] O. Gerstel, R. Ramaswami, and G. Sasaki, "Cost effective traffic grooming in WDM rings," in *IEEE INFOCOM*, 1998, pp. 69–77.

[8] G. Li and R. Simha, "Efficient routing to reduce the cost of add-drop multiplexers in WDM optical ring networks," in *SPIE Conference*, 1999, vol. 3843, pp. 258–267.

[9] R. Dutta and G. N. Rouskas, "On optical traffic grooming in WDM rings," in *Proceedings of ACM SIGMETRICS 2001*. June 2001, pp. 164–174, ACM.

[10] M. Kuznetsov et al., "A next-generation optical regional access network," *IEEE Commun.*, vol. 38, no. 1, pp. 66–72, Jan 2000.

[11] U. Killat (Editor), *Access to B-ISDN via PONs*, Wiley, 1996.

[12] "Broadband optical access systems based on passive optical networks (PON)," ITU-T Recommendation G.983.1, October 1998.

[13] Rudra Dutta, *Virtual Topology Design for Traffic Grooming in WDM Networks*, Ph.D. thesis, NCSU, Raleigh, NC, August 2001.

[14] M. R. Garey and D. S. Johnson, *Computers and Intractability*, W. H. Freeman and Co., New York, 1979.