Chapter 1

A PRACTICAL AND COST-EFFECTIVE APPROACH TO EFFICIENT TRAFFIC GROOMING IN WDM MESH NETWORKS*

Harsha V. Madhyastha
Department of Computer Science and Engineering
University of Washington
Seattle, WA 98105, USA
harsha@cs.washington.edu

C. Siva Ram Murthy
Department of Computer Science and Engineering
Indian Institute of Technology, Madras
Chennai 600036, India
murthy@iitm.ac.in

Abstract In this chapter, we present a new scheme for traffic grooming in WDM mesh networks. We put forward a new node architecture which brings together all the three qualities desired - 1. practical feasibility, 2. cost-effectiveness and 3. efficient grooming capability. None of the models considered so far in the literature have managed to satisfy all three criteria. We achieve these three ideals by considering a combination of groomers at multiple traffic granularities. We also present an algorithm for efficient traffic grooming with this new architecture. We justify the need for this new algorithm by imposing our node architecture on existing algorithms and comparing with them through a wide range of simulations.

Keywords: WDM optical mesh network, Grooming architecture, Traffic grooming, Integer linear Programming, Logical topology

*This work was supported by the Department of Science and Technology, New Delhi, India.

1. Introduction

The advent of Wavelength Division Multiplexed (WDM) Optical Networks has made it possible for each physical link to carry traffic of the order of Tbps. Several wavelengths can be multiplexed on the same fiber with each wavelength capable of carrying traffic up to 10 Gbps. Yet, individual traffic demands are still of the order of Mbps. This requires efficient grouping of individual connections onto the same wavelength as dedicating a unique wavelength for each demand will lead to huge wastage of bandwidth. Intelligent grouping is also required because each wavelength has to be dropped at the source and destination of each of the connections assigned to it. Dropping a wavelength at any node involves conversion from optical to electronic domain, and the equipment for performing this is the main contributor towards the cost of the network. This grouping of connections and assigning wavelengths to these groups, so as to optimize on some objective such as throughput or network cost, is termed as "traffic grooming".

Traffic grooming in WDM optical networks has been the focus of research in much of the recent work. As the traffic grooming problem is known to be NP-hard for arbitrary traffic [7], most of the work has been limited to domains with constraints on traffic or physical topology. Almost all of the work has only looked at the traffic grooming problem in SONET/WDM rings. Initial research focused on uniform traffic and used circle construction techniques to minimize the number of wavelengths as well as the number of Add-Drop Multiplexers (ADMs) [11, 10]. Improving on this, [1] and [2] tried to minimize the overall network cost, considering parameters such as maximum number of physical hops, though they too concentrated on rings and laid emphasis mainly on uniform traffic. Non-uniform traffic, albeit with the constraint that the total traffic added or dropped at any node is lesser than some threshold, was handled for the first time in [5] and [6]. The first attempt to handle arbitrary traffic, though restricted to SONET rings, is seen in [9].

The recent surge in the industry to use mesh networks instead of SONET rings has breathed life into research on traffic grooming in mesh networks as well. The initial work in this direction, trying to minimize the number of transceivers by designing a suitable virtual topology, is found in [4]. Its main shortcoming is that it does not consider any limitations on the physical topology such as number of wavelengths. Grooming in survivable WDM mesh networks, assuming single-link failure, was considered in [8]; but it addressed the issue of dynamic grooming. The same problem was also addressed in [12], though without considering the survivability aspect. Dynamic grooming is the problem of routing and assigning wavelengths for a new demand, given the current state of the network, whereas in static grooming the traffic demands are known apriori and all of them have to be assigned routes and wavelengths to minimize required resources (wavelengths and grooming ports). Static grooming can also be viewed from the angle of maximizing the throughput given the constraints on resources. This problem has been addressed in the context of mesh networks in [13]. It outlines two different node architectures - MPLS/IP and SONET/WDM. It propounds that the former is more cost-effective and an ILP formulation for grooming with this architecture is presented. It also proposes two heuristics for traffic grooming with the MPLS/IP node architecture in a WDM mesh network. The main drawback in this work is that it assumes unlimited grooming capability (ability to switch traffic among streams) at each node as multiplexing is done in software in the MPLS/IP architecture. This is not a practical assumption as routing each packet by examining its header involves a large overhead, which makes the setup incapable of handling the large bandwidth of an optical WDM link. Hence, full-scale grooming, *i.e.*, as much multiplexing ability as required, at extremely fine granularities is not practically feasible. (Grooming at fine granularities involves switching streams which carry very low traffic while grooming at coarse granularities can only switch higher-rate traffic streams.) On the other hand, the SONET/WDM architecture also has its own limitations. Here, the switching cost of the groomer is proportional to the square of the number of ports on it (the number of ports on a groomer is the number of streams it is capable of grooming). So, though using the SONET/WDM architecture will lead to lesser grooming equipment cost as the number of traffic streams it can groom is limited, the overall cost will be high as grooming can only be done at coarse granularities, which will lead to the need for a greater number of wavelengths. In short, though MPLS/IP is efficient, it is infeasible and not cost-effective because of high processing overhead and SONET/WDM, though feasible, is neither efficient nor cost-effective because of grooming at coarse levels and the high switching cost involved.

In this chapter, we propose a new node architecture which does away with the shortcomings of the above two architectures and combines their advantages to achieve the right combination of feasibility, efficiency and cost-effectiveness. We do so by having groomers at multiple granularities at each node. The concept of using a multi-layer node architecture was also considered in [3]. But, it failed to identify the full potential of grooming at multiple levels. In the architecture used in [3], any add-drop traffic has to pass through the complete hierarchy from bottom to top. Due to this dependence between levels, switching cost benefits are obtained only at the intermediate nodes of lightpaths. Our node architecture ensures saving in switching cost at all nodes as the groomers at different layers are completely independent, which makes use of the true strength of grooming at multiple granularities.

The remaining part of the chapter is organized as follows. A detailed description of the node architecture we propose is given in Section 1.2. The exact specification of the traffic grooming problem we attempt to solve in this chapter is clearly stated in Section 1.3. To find the solution to this problem, we first give an ILP formulation in Section 1.4 and then propose a heuristic algorithm in Section 1.5. The working of our heuristic is illustrated in Section 1.6 with the help of an example and in Section 1.7, we present the results of the various simulations performed to study the performance of our heuristic. Finally in Section 1.8, we conclude and provide directions for future work.

2. Node Architecture

Our proposed novel node architecture involves the use of two groomers - one at a coarse level and the other at a finer level of granularity, which we call the higher level and lower level groomer, respectively. To make this setup practically feasible, unlike the MPLS/IP architecture, we work with the practical assumption that the number of ports on the lower level groomer is limited. Though limited, the capability to groom at finer levels helps in efficient grooming by reducing the number of required wavelengths compared to that possible with the higher level groomer alone. The additional cost of the lower level groomer is more than offset by the decrease in infrastructure cost due to fewer wavelengths. In addition to the coarse and fine granularity groomers, our node architecture also makes use of a mapper, which has negligible cost as it does no processing; it just multiplexes/demultiplexes the add/drop traffic assuming best possible packing of the lower level streams into the higher level streams. Its low cost is due to the fact that it does not perform any switching.

The mixed groomer node architecture we present is shown in Fig. 1.1. This architecture is a very generic one and can be used on any hierarchy of traffic streams, for example, OC-48/OC-12/OC-3 or STM-16/STM-4/STM-1 or STM-1/VC-3/VC-12. From now on, for the sake of convenience, we will refer to a wavelength as OC-48, a higher level stream as OC-12 and a lower level stream as OC-3. So, in the node architecture shown, the OC-48s that need to be groomed are converted from optical to electronic form by the Receiver Array (RX) and fed as input to the higher level groomer. The function performed by the higher

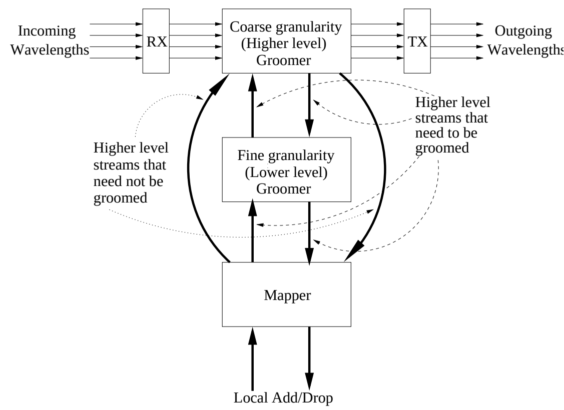


Figure 1.1. Mixed groomer node architecture

level groomer is to switch OC-12s among the different OC-48s it receives as input. The OC-12 groomer also receives OC-12s which do not need to be groomed (because they might be completely packed with OC-3s setup between the same source-destination pair), padded up to OC-48s, as input from the mapper. The mapper can also be implemented such that every OC-12, in which all the OC-3s on it are between the same source-destination pair, can be directed from the mapper to the OC-12 groomer. However, doing so when the OC-12 is not completely packed entails higher implementation complexity (as detection of padded up OC-3s is required). Hence, in our proposal, we only require the mapper to redirect OC-12s completely packed with OC-3s between the same source-destination pair to the OC-12 groomer.

If there is also a need to switch OC-3s among the OC-12s, then the OC-12 groomer feeds the corresponding OC-12 streams as input to the OC-3 groomer. Also, among the OC-12s generated by the mapper from the add/drop traffic, the ones which are not completely packed are routed to the OC-3 groomer. The streams between the OC-12 and the OC-3 groomer are essentially OC-48s, but only the OC-3 groomer can index the OC-3s within each OC-48 and switch them if required. The mapper receives the local add/drop traffic as input in the form of OC-3s padded up to OC-12s, and tries to pack them into OC-12s optimally. It does this by taking groups of 4 (the *groom factor* in this case) OC-12s and mapping the single OC-3s on them onto one OC-12. This has very low processing overhead as the OC-3s can be statically mapped to respective OC-12s. Since the mapper receives OC-3s padded up to OC-12s as input, if some OC-12 is assigned just one OC-3, then that OC-12 can be directly padded up to a OC-48 by the mapper and sent to the OC-12 groomer, bypassing the OC-3 groomer. The outgoing traffic from the node is converted from electronic to optical domain by the Transmitter Array (TX).

Essentially, the mixed groomer architecture can be divided into two logical units - the multiplexing/demultiplexing section (mapper) and the switching section (OC-3 and OC-12 groomers). The add/drop traffic that goes in and out of the groomer is in the form wherein each OC-3 is on a distinct OC-12. The mapper performs the task of multiplexing the OC-3s which are on the OC-12s which constitute the add traffic. This multiplexing is carried out to ensure best possible packing, *i.e.*, the OC-3s on every 4 OC-12s are multiplexed into 1 OC-12. The drop traffic is also similarly packed in the best possible manner. The mapper demultiplexes the OC-12s which constitute the drop traffic such that each of the OC-3s on these OC-12s is on a distinct OC-12. The task of switching traffic is completely handled by the OC-3 and OC-12 groomers. Since the mapper does not perform any switching, its cost is negligible in comparison with that of the groomers. The role of the OC-3 groomer is to switch OC-3s among OC-12s. Similarly, the function of the OC-12 groomer is to switch OC-12s among OC-48s. The number of switching ports taken up on the OC-3 groomer is the number of OC-12 streams it has to switch traffic amongst. Hence, from Fig. 1.1, it is clear that the number of OC-3 switching ports required is the sum of two quantities. The first being the number of OC-12s between it and the mapper. And, the second is the number of OC-12s between it and the OC-12 groomer. From the above explanation of how the mapper works, the number of OC-12 streams between the OC-3 groomer and the mapper is equal to $\lceil$ (Total add/drop traffic in terms of OC-3s)/(groom factor) $\rceil$. Similarly, the number of ports required on the OC-12 groomer is also the sum of two quantities. In this case, the first is the number of OC-48s fed as input to the groomer. The second is the number of OC-48s onto which it has to switch OC-12s, which are then fed as input to the OC-3 groomer. At the maximum, the value both these quantities take up is the number of OC-48s supported on the links incident at the node.

In our node architecture, the number of ports on the OC-3 groomer is constrained as this is a major contributor towards the cost of the setup. On the other hand, the number of ports on the OC-12 groomer can be assumed to be practically unlimited as grooming at a coarse level is comparatively inexpensive. Moreover, the number of ports required for full-scale grooming is lesser. To get an estimate of this, consider an OC-768 backbone, *i.e.*, each node in the network can handle bandwidth equivalent to OC-768. Since OC-768 is equal to 16 OC-48s, unlimited grooming capability at the OC-12 level would require 32 ports. This is because 16 ports would be required for the OC-48s on the link and another 16 for the add/drop traffic. On the other hand, since OC-768 is equivalent to 64 OC-12s, the number of ports required on the OC-3 groomer for input from the OC-12 groomer is 64. Also, 64 ports would be required for the OC-12s received from the mapper. This implies that a total of 128 ports are required on the OC-3 groomer. This quantity is 4 times as many as that on the OC-12 groomer. Since switching cost is proportional to the square of the number of ports, the switching cost at the OC-3 level is more than 16 times that at the OC-12 level[1]. On the whole, this clearly makes the cost of full-scale grooming at the OC-12 level negligible compared to that at the OC-3 level.

Let us now look at the advantages of the mixed groomer architecture over that of an OC-3 groomer or OC-12 groomer alone. If an OC-12 groomer alone is employed, it does not have the capability to switch OC-3s among OC-12s. So, the add/drop traffic in the form of OC-3s padded up to OC-12s cannot be multiplexed together. Each of these OC-12s will have to be assigned as they are to OC-48s on the link. Hence, as each OC-12 can only have one OC-3, the maximum traffic that can be supported is $\frac{1}{4}$, *i.e.*, 1/(groom factor) of the total bandwidth. On the other hand, using the OC-3 groomer alone suffers from two disadvantages. Firstly, due to the absence of the OC-12 groomer, if OC-12s among two OC-48s need to be swapped (switching at the OC-12 level), this has to be done by swapping each of the OC-3s on these OC-12s. This is costlier as switching needs to be done at a finer granularity. Also, since there is no OC-12 groomer to pick out the specific OC-12s, all the OC-12s on these OC-48s will have to be fed as input to the OC-3 groomer. More importantly, the OC-3 groomer directly receives OC-3s padded up to OC-12s as input. This implies that the number of ports consumed due to the add/drop traffic is equal to the number of OC-3s in the add/drop traffic. Note that in the mixed groomer, this number was $(\frac{1}{4})^{th}$ of the add/drop traffic because multiplexing/demultiplexing is performed by the mapper. So, the mixed groomer architecture derives its efficiency by the combination of the OC-3 and OC-12 groomers and also, maintains practical feasibility and cost-effectiveness by the constraint on the number of ports on the OC-3 groomer.

2.1 Example

The following example clearly brings to the fore the advantages of using a combination of groomers in place of having an OC-3 or OC-12 groomer alone. Consider the 6-node network shown in Fig. 1.2(a) with demands of 3, 1 and 4 OC-3s between the (source, destination) pairs

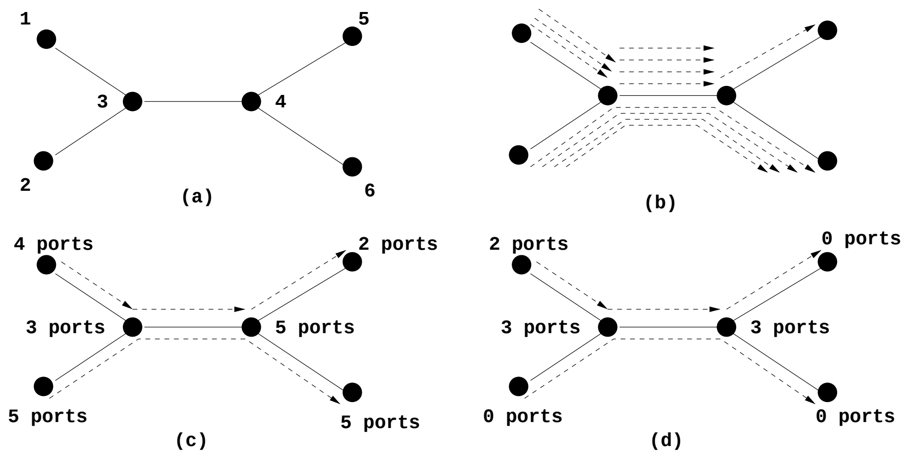


Figure 1.2. (a) Example 6-node physical topology and network state with (b) OC-12 groomer, (c) OC-3 groomer and (d) mixed groomer

(1, 4), *(3, 5)* and *(2, 6)*, respectively. As outlined above switching cost at the OC-12 level is negligible to that at the OC-3 level which only depends on the number of OC-12s on the link. So, from here on, we consider OC-12 as a wavelength. The state of the network in each of the three cases explained below is as shown in Fig. 1.2.

- OC-12 groomer alone
  When only an OC-12 groomer is available at each node, there is no grooming capability at the OC-3 level at any node. So, multiple OC-3s cannot be groomed onto the same OC-12, which implies that each OC-3 has to be carried on a new OC-12. This in turn implies that the number of wavelengths required on a link is equal to the total number of OC-3s transmitted along the link. As shown in Fig. 1.2(b), though there is no switching cost at any of the nodes, the overall network cost is high due to the large number of wavelengths required to satisfy the traffic demand. In this example, atleast 8 wavelengths are required on link *(3, 4)*.

- OC-3 groomer alone
  If each node has an OC-3 groomer with full-scale grooming capability, then optimal grooming can be performed as shown in Fig. 1.2(c). But, the downside of this scheme is the high switching cost borne due to the large number of grooming ports required at each node, as shown in Fig. 1.2(c). As explained before, each add/drop OC-3 consumes a port on the OC-3 groomer and hence,

the add/drop traffic itself consumes 3, 4, 1, 3, 1 and 4 ports at nodes 1, 2, 3, 4, 5 and 6, respectively. Also, at every node, among all the OC-12s on the links incident at that node, every OC-12 that needs to be groomed consumes an OC-3 grooming port at that node. An OC-12 needs to be groomed if some OC-3s on it need to be either dropped or switched to other OC-12s. All these properties together necessitate as many as 5 OC-3 grooming ports at nodes 2, 4 and 6. So, though the number of wavelengths required is reduced from 8 to 2 in comparison with the previous case, the grooming cost introduced keeps the network cost high.

- Mixed groomer architecture (OC-12 groomer + OC-3 groomer + Mapper)
  The network state achieved with the mixed groomer node architecture (shown in Fig. 1.2(d)) clearly highlights its merits because as in the case with the OC-3 groomer alone, the number of wavelengths required is 2 but with much lower switching cost. The maximum number of ports needed at any of the nodes is 3 and three of the nodes do not even require an OC-3 groomer. The 4 OC-3s from node 2 to node 6 can be routed on the same OC-12 without consuming any OC-3 grooming ports as an OC-12 which is completely packed with OC-3s between the same (source, destination) pair directly goes from the mapper to the OC-12 groomer. Also, no ports are required for the OC-12 from node 4 to node 5 as a single OC-3 is put onto it. Lesser number of ports are also taken up at nodes 1, 3 and 4 because the mapper multiplexes/demultiplexes the add/drop traffic and hence, the number of ports consumed on the OC-3 groomer by the add/drop traffic is only $(\frac{1}{4})^{th}$ the number of add/drop OC-3s, which in this example translates into only one port at each of these nodes.

This example shows that our mixed groomer node architecture brings together the beneficial features of both a coarse granularity and a fine granularity groomer, *i.e.*, lower switching cost and lesser number of wavelengths required, respectively.

3. Problem Statement

The problem we address in this chapter is that of static grooming, with the objective of maximizing throughput given the various constraints on the available resources. We account for the following resource constraints:

1 The maximum number of distinct wavelengths on which traffic can be routed on each physical link - $\mathbf{W}_{max}$. In our problem setting, this is the maximum number of OC-48s that can be carried on any link. But, since the grooming capability on the OC-12 groomer is practically unlimited, this can be equivalently seen as the maximum number of OC-12s on each link. We assume the same $\mathbf{W}_{max}$ to hold over all physical links.

2 The number of ports on the OC-3 groomer at each node - $\mathbf{P}_{max}$. This places a limit on the number of OC-12 streams that can be groomed at each node, *i.e.*, the number of OC-12s which require OC-3s on them to be either dropped at that node or switched to other OC-12s.

The parameters given as input to the static grooming problem are:

1 The number of nodes **N** in the network. Each node is assumed to have a groomer with the mixed groomer node architecture. We assume that all physical nodes have groomers of the same size.

2 The physical topology of the network is given in the form of the adjacency matrix **L**, where $\mathbf{L}_{i,j} = \mathbf{L}_{j,i} = 1$ or 0 if a link exists or does not exist between nodes **i** and **j**, respectively. This is called the *"single fiber"* scenario.

3 As our mixed groomer architecture is a generic one, we also take the groom factor **G**, *i.e.*, the ratio between the bandwidths of the higher level stream and the lower level stream as input. In our case where we consider OC-12 and OC-3, the groom factor is 4. So, the total traffic (in terms of OC-3s) that can be loaded on any physical link is $\mathbf{W}_{max} \times G$.

4 The traffic matrix **T** gives the traffic demand $\mathbf{T}_{i,j}$ with node **i** as source and node **j** as destination. In the mixed groomer architecture, a OC-12 packed with **G** OC-3s between the same (source, destination) pair is routed directly from the mapper to the OC-12 groomer (refer the previous section on Node Architecture) and so, does not take up any ports on the OC-3 groomer. As we are placing a constraint on the ports only on the OC-3 groomer, all entries in **T** (specified in units of OC-3s) are assumed to be lesser than **G**. If any entry $\mathbf{T}_{i,j}$ is greater than **G**, then $\lfloor \frac{\mathbf{T_{i,j}}}{\mathbf{G}} \rfloor$ OC-12s can be padded to OC-48s and directly put through the OC-12 groomer, leaving behind the entry $\mathbf{T}_{i,j}$ mod **G**, which is lesser than **G**. Thus, even if no restriction is placed on the traffic matrix entries, the problem

can be reduced to an equivalent one wherein each demand is lesser than **G** OC-3s.

Given the above inputs and the limitations on the infrastructure, we aim to maximize the throughput, *i.e.*, maximize the percentage of successfully routed traffic. We give an Integer Linear Programming formulation of this grooming problem in the next section. If solved, the answer to the ILP formulation gives us the optimal solution to our problem but, solving any ILP problem entails exponential complexity. So, the ILP formulation can be used only to optimally solve the grooming problem for networks with very few nodes dealing with sparse traffic matrices. As the static grooming problem is known to be NP-hard for arbitrary traffic even for ring networks [7], it is clearly NP-hard for mesh networks as well due to the increase in complexity of the problem at hand. Hence, in order to obtain solutions for large networks, we propose a heuristic algorithm for solving it near-optimally. We demonstrate the need for this algorithm by comparing its performance with the only alternatives available - the two heuristics proposed in [13]. On executing all the three algorithms on a wide variety of traffic patterns, the results clearly show that grooming with the mixed groomer architecture necessitates the use of our heuristic as it realizes much higher throughputs compared to the other two. We also demonstrate the near-optimality of our heuristic by comparing its performance with that obtained by solving the ILP formulation on small networks.

4. ILP Formulation

Our objective in solving the above problem is to determine which are the demands that can be successfully routed to maximize throughput. This problem is usually broken up into the following two sub-problems:

- Determination of the logical topology - Which are the lightpaths to be set up? The set of lightpaths are seen as the links over which connections are routed.

- Routing of individual connections - Which are the demands to be satisfied and how is each demand routed over the logical topology?

We follow the above approach in our ILP formulation, but we look at the same problem from a different angle in our heuristic. In it we take up the Routing and Wavelength Assignment (RWA) approach. We view the problem as determining the connections to be routed, assigning routes to each one of them and then allocating wavelengths for them on each physical link along their assigned route. We adhere to the logical topology approach in our ILP formulation as it facilitates easier counting of ports - each lightpath set up consumes one port each on the groomers at its source and destination.

Here, we present an ILP formulation of the static grooming problem explained in the previous section. This formulation is much on the same lines as that given in [13]. The difference comes in due to the replacement of the MPLS/IP architecture with the more efficient mixed groomer architecture. As the constraint on the number of ports on the lower level groomer needs to be imposed, we need to count the number of ports assigned on the lower level groomer at each node. Hence, the logical topology section of the formulation is identical to that in [13], but the remaining sections of the formulation differ.

m and **n**	The nodes at either end of a physical link. The link (**m**, **n**) is considered to be a directed edge, outgoing from **m** and incoming into **n**. **m, n** $\in$ [1, **N**]
i and **j**	The source and destination of a lightpath, which might traverse multiple physical links. **i, j** $\in$ [1, **N**]
s and **d**	The source and destination of a routed connection, which might span multiple lightpaths. **s, d** $\in$ [1, **N**]
k	Any general node in the network. **k** $\in$ [1, **N**]
t	index of an individual OC-3 among the different OC-3s established between the same (source, destination) pair. If $\mathbf{T}_{3,5} = 6$, then 6 OC-3s with 3 as source and 5 as destination are enumerated from 1 to 6.

- The variables used in our formulation and their physical interpretation are as follows:
  - $\mathbf{V}_{i,j}^{w}$ = Number of lightpaths established on wavelength **w**, with node **i** as source and node **j** as destination.
  - $\mathbf{P}_{m,n}^{i,j,w}$ = Number of lightpaths setup between node **i** and node **j** on wavelength **w** which are routed through the physical link (**m**, **n**).
  - $\mathbf{R}_{i,j,k,w}^{s,d,t} = 1$, if the $\mathbf{t}^{th}$ OC-3 from **s** to **d** is routed through the lightpath from **i** to **j** on wavelength **w** which has its first physical hop following **i** as **k**, else it is 0. Note that the tuple (**i**, **j**, **k**, **w**) refers to a unique lightpath since at most one physical link is allowed between two nodes (from the specification of **L** in the problem definition).
  - $\mathbf{g}_{i,j,k}^{w} = 1$, if the lightpath from **i** to **j** on wavelength **w**, which has its first physical hop following **i** as **k**, needs to be groomed at the OC-3 level at nodes **i** and **j**, else it is 0.

- $\mathbf{S}^t_{s,d} = 1$, if the $\mathbf{t}^{th}$ OC-3 from **s** to **d** is established, else it is 0.

■ Objective function:

- Maximize $\sum_s \sum_d \sum_{t=1}^{T_{s,d}} \mathbf{S}^t_{s,d} \quad (\mathbf{s} \neq \mathbf{d})$

■ Constraints:

- There should be no lightpaths from node **i** to node **j**, passing through a link incoming into **i**.

$$\sum_m \mathbf{P}^{i,j,w}_{m,i} = 0 \quad \forall \mathbf{i}, \mathbf{j}, \mathbf{w} \tag{1.1}$$

- There should be no lightpaths from node **i** to node **j**, passing through a link outgoing from **j**.

$$\sum_n \mathbf{P}^{i,j,w}_{j,n} = 0 \quad \forall \mathbf{i}, \mathbf{j}, \mathbf{w} \tag{1.2}$$

- The number of lightpaths on wavelength **w** from node **i** to node **j**, passing through a link outgoing from **i**, should be equal to the number of lightpaths on wavelength **w** established from **i** to **j** in the logical topology.

$$\sum_n \mathbf{P}^{i,j,w}_{i,n} = \mathbf{V}^w_{i,j} \quad \forall \mathbf{i}, \mathbf{j}, \mathbf{w} \tag{1.3}$$

- The number of lightpaths on wavelength **w** from node **i** to node **j**, passing through a link incoming into **j**, should be equal to the number of lightpaths on wavelength **w** established from **i** to **j** in the logical topology.

$$\sum_m \mathbf{P}^{i,j,w}_{m,j} = \mathbf{V}^w_{i,j} \quad \forall \mathbf{i}, \mathbf{j}, \mathbf{w} \tag{1.4}$$

- For any node **k**, other than **i** and **j**, the number of lightpaths from **i** to **j** on wavelength **w** routed through links incoming into it should be equal to the number of lightpaths from **i** to **j** on wavelength **w** routed through links outgoing from it.

$$\sum_m \mathbf{P}^{i,j,w}_{m,k} = \sum_n \mathbf{P}^{i,j,w}_{k,n} \quad (\mathbf{k} \neq \mathbf{i}, \mathbf{j}) \quad \forall \mathbf{i}, \mathbf{j}, \mathbf{w}, \mathbf{k} \tag{1.5}$$

- A lightpath can pass through a link only if the link exists. Also, at most one lightpath on a particular wavelength can be routed through a physical link.

$$\sum_{i,j} \mathbf{P}_{m,n}^{i,j,w} \leq \mathbf{L}_{m,n} \quad \forall \mathbf{m}, \mathbf{n}, \mathbf{w} \tag{1.6}$$

  In our problem setting $\mathbf{L}_{m,n}$ is restricted to values 0 or 1 and so, the values taken by $\mathbf{P}_{m,n}^{i,j,w}$ are also restricted to 0 or 1. But, the same equation holds if multiple physical links are allowed between two nodes.

- If the $\mathbf{t}^{th}$ OC-3 from $\mathbf{s}$ to $\mathbf{d}$ is setup, then it must be routed through some lightpath originating at $\mathbf{s}$.

$$\sum_{j,k,w} \mathbf{R}_{s,j,k,w}^{s,d,t} = \mathbf{S}_{s,d}^{t} \quad (\mathbf{k} \neq \mathbf{s}) \quad \forall \mathbf{s}, \mathbf{d}, \mathbf{t} \tag{1.7}$$

- If the $\mathbf{t}^{th}$ OC-3 from $\mathbf{s}$ to $\mathbf{d}$ is setup, then it must be routed through some lightpath terminating at $\mathbf{d}$.

$$\sum_{i,k,w} \mathbf{R}_{i,d,k,w}^{s,d,t} = \mathbf{S}_{s,d}^{t} \quad (\mathbf{k} \neq \mathbf{i}) \quad \forall \mathbf{s}, \mathbf{d}, \mathbf{t} \tag{1.8}$$

- Any traffic with $\mathbf{s}$ as the source cannot be routed on a lightpath which terminates at $\mathbf{s}$.

$$\sum_{i,k,w} \mathbf{R}_{i,s,k,w}^{s,d,t} = 0 \quad (\mathbf{k} \neq \mathbf{i}) \quad \forall \mathbf{s}, \mathbf{d}, \mathbf{t} \tag{1.9}$$

- Any traffic with $\mathbf{d}$ as the destination cannot be routed on a lightpath which originates at $\mathbf{d}$.

$$\sum_{j,k,w} \mathbf{R}_{d,j,k,w}^{s,d,t} = 0 \quad (\mathbf{k} \neq \mathbf{d}) \quad \forall \mathbf{s}, \mathbf{d}, \mathbf{t} \tag{1.10}$$

- On any node $\mathbf{k}$, other than $\mathbf{s}$ and $\mathbf{d}$, if the $\mathbf{t}^{th}$ OC-3 from $\mathbf{s}$ to $\mathbf{d}$ is routed on some lightpath terminating at $\mathbf{k}$ then it must also be routed on some lightpath originating at $\mathbf{k}$.

$$\sum_{i,j,w} \mathbf{R}_{i,k,j,w}^{s,d,t} = \sum_{i,j,w} \mathbf{R}_{k,j,i,w}^{s,d,t} \quad (\mathbf{k} \neq \mathbf{s}, \mathbf{d}) \quad \forall \mathbf{s}, \mathbf{d}, \mathbf{t}, \mathbf{k} \tag{1.11}$$

- Traffic can be routed on a lightpath only if it exists and the total traffic routed on it cannot exceed the capacity of a wavelength.

$$\sum_{s,d,t} \mathbf{R}_{i,j,k,w}^{s,d,t} \leq \mathbf{G} \times \mathbf{P}_{i,k}^{i,j,w} \quad (\mathbf{k} \neq \mathbf{i}) \quad \forall \mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{w} \tag{1.12}$$

– The lightpath from **i** to **j** on wavelength **w**, which has its first physical hop after **i** as **k**, needs to be passed through the OC-3 groomer at nodes **i** and **j** if more than one OC-3 has been routed on it.

$$\sum_{s,d,t} \mathbf{R}^{s,d,t}_{i,j,k,w} \geq 2 \times \mathbf{g}^{w}_{i,j,k} \quad (\mathbf{k} \neq \mathbf{i}) \quad \forall \mathbf{i},\mathbf{j},\mathbf{k},\mathbf{w} \tag{1.13}$$

$$\sum_{s,d,t} \mathbf{R}^{s,d,t}_{i,j,k,w} \leq (\mathbf{G} - 1) \times \mathbf{g}^{w}_{i,j,k} + 1 \quad (\mathbf{k} \neq \mathbf{i}) \quad \forall \mathbf{i},\mathbf{j},\mathbf{k},\mathbf{w} \tag{1.14}$$

– The total number of lightpaths passing through the OC-3 groomer at each node must be lesser than the number of ports on the groomer.

$$\sum_{j,k,w} \mathbf{g}^{w}_{i,j,k} + \mathbf{g}^{w}_{j,i,k} \leq \mathbf{P}_{max} \quad (\mathbf{k} \neq \mathbf{i},\mathbf{j}) \quad \forall \mathbf{i} \tag{1.15}$$

Though we assume in our problem setting that the number of ports on the OC-3 groomer is same at all nodes, the above given constraint can be easily modified to handle the case where $\mathbf{P}_{max}$ varies across nodes by replacing $\mathbf{P}_{max}$ by $\mathbf{P}_i$ in the above constraint, where $\mathbf{P}_i$ is the number of ports on the OC-3 groomer at node **i**.

5. Heuristic

If the ILP formulation given in the previous section is solved, the optimal solution to any instance of the static grooming problem we are considering can be obtained. But, since the number of variables and constraints in the formulation increases exponentially with increase in the size of the problem, practical considerations force us to take up heuristic approaches to obtain near-optimal solutions. A couple of heuristics - *Maximizing Single-Hop Traffic* (MST) and *Maximizing Resource Utilization* (MRU) - were proposed in [13]. We put forward another heuristic which is tailored to suit the mixed groomer architecture. We justify the need for this new heuristic by comparisons with those proposed in [13] which clearly show the superiority of our approach.

As outlined before, the approach we are going to follow is to determine the connections to be established and assign routes and wavelengths to them rather than build the logical topology and route the connections on it. We perform this iteratively by maintaining a partition of the set of connections, **A** and **B**, such that only those in **A** have been assigned, and in every iteration, one connection in set **B** is assigned and moved into set **A**. Since the main resource constraint limiting us from obtaining a 100% throughput is the limit on the number of ports on the OC-3 groomer, we assign the connection whose establishment would lead to the least increment in the number of OC-3 grooming ports used over all the nodes in the network. To make this decision, we need to determine for each connection in set **B** the route and corresponding wavelength assignment that would lead to the least increase in used OC-3 grooming ports among all possible route and wavelength assignments. We keep performing this iteratively until no connection can be established due to the constraints on the number of wavelengths ($\mathbf{W}_{max}$) and on the number of ports ($\mathbf{P}_{max}$) available. To evaluate the increase in ports at each stage, we determine the new lightpaths that need to be established and the old lightpaths that need to be split in order to setup the required route and wavelength assignment. Then we use the property that one OC-3 port each is taken up on the groomer at its source and destination by each lightpath carrying more than one OC-3.

Determination of the *"least-port-increase"* route and wavelength assignment for each connection would entail performing a search over all possible routes from the source to destination of that connection and over each possible wavelength assignment for each route. This search space is clearly exponential in size and as there is no possibility for pruning, the complexity involved in performing this search is exponential. First we try and reduce the search space in terms of number of routes to be examined. While considering just the shortest physical hop path from source to destination would contradict the very purpose of the search, searching over all possible routes is exponential. So, as a trade-off between complexity and optimality, we pre-determine the **k**-shortest paths for every (source, destination) pair and search over these **k** routes. **k** is a parameter which can be decreased or increased depending on whether faster execution or proximity to optimal solution is desired. Even though we have cut down on the complexity significantly by considering the **k**-shortest paths, the search remains exponential as we need to evaluate the increment in used ports over all possible wavelength assignments for each of these **k** routes. Hence, we resort to the following approach. Though $\mathbf{W}_{max}$ wavelengths are available on each physical link in the network, we start off our grooming heuristic assuming the network to have only 1 wavelength. Once the process of iteratively assigning connections stops because no more connections can be established, we increment the number of wavelengths to 2 and continue assigning connections. This process of grooming, incrementing number of wavelengths and then again grooming is performed until all $\mathbf{W}_{max}$ wavelengths have been used. This approach reduces the complexity because as the number of wavelengths is increased, the number of possible wavelength assignments does not increase by much due to the fact that the traffic on many wavelengths could have already been fully allotted on several physical links to connections assigned until then. All these modifications bring the complexity down to practical levels without adversely affecting the efficacy of the algorithm.

Until now, our heuristic neither has any look-ahead nor any adaptive component other than the property that the *"least-port-increase"* route and wavelength assignment depends on the current state of the network. To incorporate look-ahead, we modify our policy for selecting the connection to be assigned. After evaluating the least increment in ports involved in setting up each connection, we select the minimum of these and pick out the connections corresponding to this minimum. For each such connection **C**, we determine the set of connections **S** that could be added without any additional ports being consumed if **C** were to be assigned. We now find, for each **C**, the total traffic carried by **C** and by all the connections in its corresponding set **S**. The maximum value of this traffic is found and one of the connections corresponding to this maximum is assigned. This look-ahead helps us drive the search in the direction of greater throughput. The connection selection policy can be further improved by assigning the connection whose *"least-port-increase"* route has the least number of physical hops among all the connections which correspond to the maximum *"look-ahead traffic"*. The motivation behind this step is to favour lesser use of physical resources. The adaptiveness of the heuristic is further enhanced by trying to reroute the assigned connections at each stage. Once a connection is assigned, we consider each connection **C** which was assigned before this stage. We remove the connection **C** and determine its *"least-port-increase"* route and wavelength assignment in this new state of the network. If one of the following two conditions is satisfied, connection **C** is assigned to the new route, else it is put back to its old route.

- Changing the assigned route for connection **C** would lead to a decrease in the overall number of used OC-3 grooming ports in the network.

- Changing the assigned route for connection **C** would keep the number of used OC-3 grooming ports same but the rerouting would modify the state of the network to permit some connections to be added without additional increase in ports, which should not have been possible without the rerouting.

Having gone through the complete logical development of our heuristic for solving the static grooming problem, our heuristic can now be concisely put down as follows:

1 Set **NumWavs** = 1. Determine the **k**-shortest paths between every pair of nodes and store them in **Paths**. Add all desired connections (as given in **T**) to the set **B** and initialize set **A** as a null set.

2 If set **B** is empty, 100% throughput has been achieved and therefore, stop.

3 For each connection **(s, d)** in the set **B**, evaluate the increase in the number of used OC-3 ports in the network corresponding to each of the routes in **Paths(s, d)** and each corresponding feasible wavelength assignment. Using this information, determine the route and wavelength assignment which leads to the least increase in ports and store this least increase in **IncrPorts(s, d)**. If no feasible route and wavelength assignment exists, set **IncrPorts(s, d)** to ∞.

4 Find the minimum value of **IncrPorts(s, d)** for all connections **(s, d)** in the set **B**. If this minimum is ∞, skip to step 8, else store all the **(s, d)** pairs corresponding to this minimum in the set **S**.

5 For each **(s, d)** pair in **S**, determine the subset of **B** - {**(s, d)**} which can be allotted without additional consumption of ports if the connection **(s, d)** is assigned along its *"least-port-increase"* route and wavelength assignment. Sum up the traffic of connection **(s, d)** along with those in its corresponding subset and assign this value to **AddTraffic(s, d)**.

6 Find the maximum value of **AddTraffic(s, d)** for all connections **(s, d)** in the set **S**. Assign the connection **(s, d)** corresponding to this maximum along its *"least-port-increase"* route and wavelength assignment. If more than one connection takes this maximum value for **AddTraffic(s, d)**, assign any one whose route has the least number of physical hops. Move this assigned connection from set **B** to set **A**.

7 Consider each connection **(s, d)** in set **A** in increasing order of traffic. Remove the connection **(s, d)** and evaluate the decrease in the number of used OC-3 ports - **DecrPorts(s, d)**. Now, determine the *"least-port-increase"* route and wavelength assignment for **(s, d)**. If either the increase in number of ports associated with this new route is lesser than **DecrPorts(s, d)** or if the increase in number of ports is equal to **DecrPorts(s, d)** but assigning **(s, d)** to the new route would facilitate allocation of more traffic without consuming additional ports (which should not have been

Table 1.1. Example traffic matrix

Node	1	2	3	4	5	6	7	8	9
1		5	4						
2				2					
3					14				
4						2			1
5								4	
6							7	2	
7		4						2	
8	4	5							
9									

possible if the rerouting had not been done), then assign **(s, d)** to the new route. Else, put it back to the previously existing route and wavelength assignment. Go back to step 2.

8 If **NumWavs** $< \mathbf{W}_{max}$, then increment **NumWavs** and go back to step 3, else stop.

6. Illustrative Example

To offer a better understanding of our heuristic and to show a glimpse of how our heuristic outperforms those proposed in [13], we consider an example. The 9-node network considered has a physical topology as shown in Fig. 1.3. In this example, we take the groom factor $\mathbf{G} = 18$, the number of wavelengths $\mathbf{W}_{max} = 2$ and the number of ports $\mathbf{P}_{max} = 2$. The traffic matrix $\mathbf{T}$ is as shown in Table 1.1.

When the *Maximizing Single-Hop Traffic* (MST) heuristic from [13] is executed on the above example (without the $\mathbf{P}_{max}$ constraint as it does not consider the mixed groomer architecture), the logical topology shown in Fig. 1.4(a) is setup which can be established only if $\mathbf{P}_{max} \geq 5$. On routing the individual connections on this logical topology, a

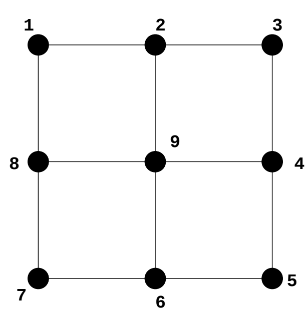


Figure 1.3. 9-node physical topology

throughput of 98% is obtained even though both the wavelengths have been utilized on most of the links. Similarly, the *Maximizing Resource Utilization* (MRU) heuristic from [13] was executed on this example and the logical topology setup was that shown in Fig. 1.4(b), which requires $\mathbf{P}_{max} \geq 3$. Though *MRU* consumes lesser grooming ports than *MST*, the throughput obtained also decreases to 67%.

On the other hand, our heuristic manages to achieve 100% throughput making use of only one wavelength and with the constraint of having only 2 ports on the OC-3 groomer at each node. The various states of the network as the execution of our heuristic progresses are shown in Fig. 1.5.

(a) Initially, the *"least-port-increase"* route for each connection is the shortest path from source to destination. As no lightpath presently exists, the **IncrPorts** value for each connection is 2 except for **(4, 9)**, whose **IncrPorts** value is 0 as a lightpath with just one OC-3 does not pass through the OC-3 groomer (refer section on Node Architecture). Hence, **(4, 9)** gets assigned.

(b) All the unassigned connections now need at least 2 ports and so, the one with maximum traffic among them - **(3, 5)** - gets assigned.

(c) Even now, the least value of **IncrPorts** is 2 and as many lightpaths have not yet been setup, no rerouting is beneficial. So, the connection with the next highest traffic - **(6, 7)** - gets established.

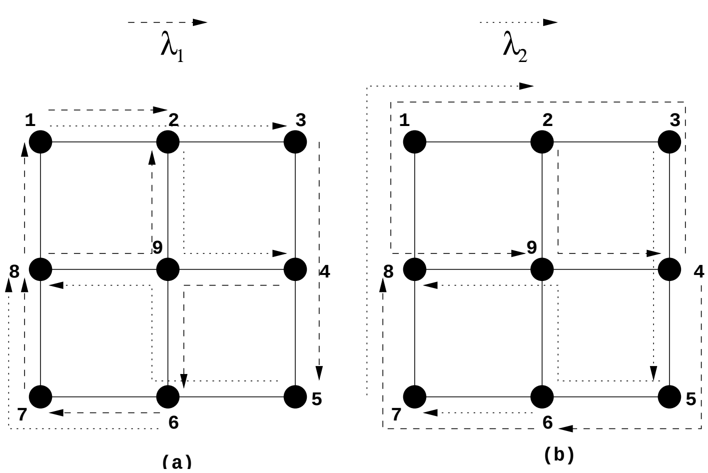


Figure 1.4. Logical topology setup by (a) MST heuristic (b) MRU heuristic

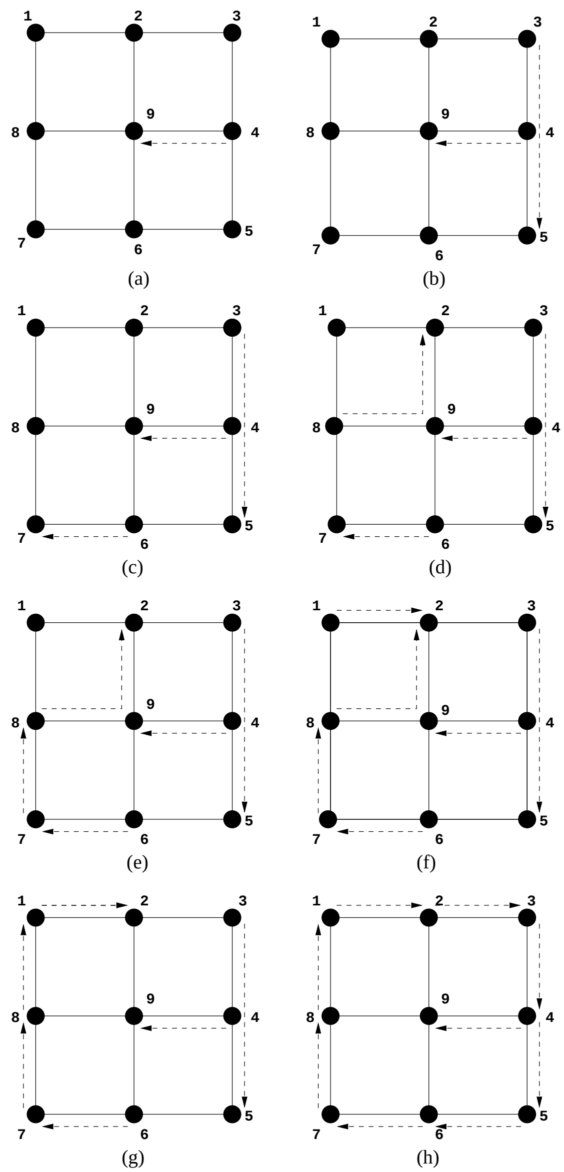


Figure 1.5. Development of logical topology in our heuristic

(d) Continuing with the trend, the connection carrying the highest traffic among all the unassigned connections - **(8, 2)** - gets assigned. One point to note at this stage is that the **AddTraffic** value for both **(7, 8)** and **(6, 8)** is 4 because if any one of them is setup, the other connection can be routed without consuming additional ports. Yet, **(8, 2)** got assigned as **T(8, 2)** $= 5 > 4$.

(e) Next, the connection**(7, 8)** gets assigned as its **IncrPorts** value is the current minimum 2 and its **AddTraffic** value is 8 (because if **(7, 8)** is setup, then the connections **(6, 8)** and **(7, 2)** can be routed without taking up more ports). After **(7, 8)** is setup, both **(6, 8)** and **(7, 2)** get assigned as the **IncrPorts** value for both is 0.

(f) Even now, the least value of **IncrPorts** is 2 and among the connections corresponding to this minimum, **(1, 2)** has the highest traffic and so, is selected for assignment.

(g) Having introduced **(1, 2)**, we can see that it is beneficial to reroute connection **(8, 2)**, changing its route from $\mathbf{8} \rightarrow \mathbf{9} \rightarrow \mathbf{2}$ to $\mathbf{8} \rightarrow \mathbf{1} \rightarrow \mathbf{2}$. The motivation behind this is that the rerouting does not increase the number of OC-3 grooming ports used on the whole but permits the connection **(8, 1)** to be routed without further usage of ports. So, the route for **(8, 2)** is changed and now, since **IncrPorts** value for **(8, 1)** is 0, it is assigned.

(h) Progressing in the same manner as described until now, the final logical topology setup is as shown in Fig. 1.5(h).

This example not only clearly outlines the working of our heuristic but also shows that it is better than both *MST* and *MRU* as a higher throughput was achieved utilizing lesser number of wavelengths and under a tighter constraint on the number of ports.

7. Simulations and Results

In this section, we present the results of various simulations that we have conducted. These simulations can be broadly classified into three groups, based on their objectives:

1 Comparison with the optimal solution obtained by solving the ILP formulation

2 Demonstrating the efficiency of our heuristic

3 Comparison of our heuristic with *MST* and *MRU* heuristics

In our simulations, we used the 6-node network and the 15-node network, whose physical topologies are as given in [13]. These two networks were used for comparison of our heuristic's solution with that yielded by

Table 1.2. Comparison of throughput with ILP and MST

$\mathbf{W}_{max}$	**G**	ILP	Our heuristic	MST
1	6	42%	35%	26%
2	6	64%	50%	47%
3	6	76%	71%	70%
4	6	88%	86%	80%
1	7	47%	40%	26%
2	7	65%	64%	48%
3	7	91%	84%	71%
4	7	95%	93%	81%
1	8	51%	45%	26%
2	8	68%	65%	50%
3	8	91%	89%	73%
4	8	100%	96%	82%

the ILP formulation, and the *MST* and *MRU* heuristics, respectively. The traffic matrices used for these simulations were obtained by generating each demand as a uniformly distributed random number in the range 0 to 5. As our problem formulation requires each traffic demand to be lesser than the groom factor **G**, the value of **G** is taken to be greater than 5 in all our simulations.

7.1 Comparison with ILP

Solving the ILP formulation we presented in Section IV gives the optimal solution for a particular instance of the static grooming problem. Hence, to demonstrate the near-optimality of our heuristic we compared the solution it provided with that obtained on solving the corresponding ILP formulation. We also determined the results given by the *MST* and *MRU* heuristics to show that our heuristic's solution is much nearer to the optimum. Since solving the ILP formulation entails very high complexity, these simulations could be carried out only on the 6-node network given in [13].

In these simulations, the number of wavelengths was varied from 1 to 4, with the groom factor varying from 6 to 8 for each wavelength. For each combination of number of wavelengths and groom factor, the *MST* heuristic was executed and the maximum of groomer ports used among all nodes was determined, as this is the size of the groomer required. Using this groomer size, the corresponding throughput yielded by our heuristic and by solving the ILP formulation were obtained. These simulations were again repeated with the *MRU* heuristic. The results of these are shown in Table 1.2 and Table 1.3, from which it can be clearly observed that the solution yielded by our heuristic is very close to the optimal solution in most of the cases. It is also seen that the through-

Table 1.3. Comparison of throughput with ILP and MRU

$\mathbf{W}_{max}$	**G**	ILP	Our heuristic	MRU
1	6	42%	35%	23%
2	6	67%	56%	50%
3	6	76%	71%	63%
4	6	91%	89%	81%
1	7	47%	40%	24%
2	7	70%	68%	53%
3	7	91%	84%	68%
4	7	96%	93%	87%
1	8	51%	45%	28%
2	8	78%	75%	54%
3	8	91%	89%	74%
4	8	100%	100%	92%

put given by the *MST* and *MRU* heuristics is lesser than that given by our heuristic in all the cases considered. The better performance of our heuristic is due to our approach of routing connections over the *"least-port-increase"* route and the adaptability incorporated through rerouting of connections.

Over and above this, we also used our ILP formulation to demonstrate the benefits of using our mixed groomer node architecture in place of a coarse granularity or fine granularity groomer alone. For the case of a coarse granularity groomer, the only change required in the formulation is to set the value of $\mathbf{P}_{max}$ to 0, whereas for a fine granularity groomer, we need to set $\mathbf{P}_{max}$ to ∞ and add variables to count the actual number of ports used. The same methodology as that used in the comparison of our heuristic with the optimal solution was used here too. Not only was the throughput obtained in the 3 cases (coarse groomer, fine groomer and mixed groomer) measured but the number of ports consumed in the fine groomer case was also determined. The value of $\mathbf{P}_{max}$ was taken to be 5 for the mixed groomer. The results of these comparisons are shown in Table 1.4. We observe that in all the considered instances, the throughput yielded by utilizing the mixed groomer node architecture is much higher than that given by the coarse groomer. Though higher throughput is achieved with the fine granularity groomer, in most of the cases this increase is insignificant in comparison with the associated increase in network cost (increase in number of groomer ports required). Though this could not be verified due to the high complexity involved in solving ILP formulations, we believe that utilizing the fine groomer in larger networks with more dense traffic will entail even higher increase in network cost without much advantage in the throughput compared to the mixed groomer case. These results re-emphasize the need for our mixed groomer node architecture.

Table 1.4. Comparison of throughput of mixed groomer with coarse groomer and fine groomer

$\mathbf{W}_{max}$	**G**	Coarse groomer	Fine groomer		Mixed groomer
			Throughput	No. of ports	
1	6	10%	41%	3	41%
2	6	11%	63%	6	62%
3	6	12%	78%	7	76%
4	6	15%	97%	11	86%
1	7	10%	47%	3	47%
2	7	14%	74%	6	71%
3	7	15%	85%	7	77%
4	7	20%	97%	10	90%
1	8	10%	51%	3	51%
2	8	15%	79%	6	74%
3	8	16%	95%	7	82%
4	8	22%	98%	8	93%

7.2 Efficiency of Our Heuristic

Any good grooming algorithm must satisfy the basic condition that the throughput must increase with increase in available resources. In our problem setting, the two basic resources are:

1. Total bandwidth available in the network - measured in terms of number of wavelengths $\mathbf{W}_{max}$ and groom factor $\mathbf{G}$.

2. Grooming capacity of the groomer - measured in terms of number of ports $\mathbf{P}_{max}$ on the OC-3 groomer in the mixed groomer architecture.

To show that our heuristic yields higher throughput with greater available bandwidth, we fixed the number of wavelengths at 10 and the num-

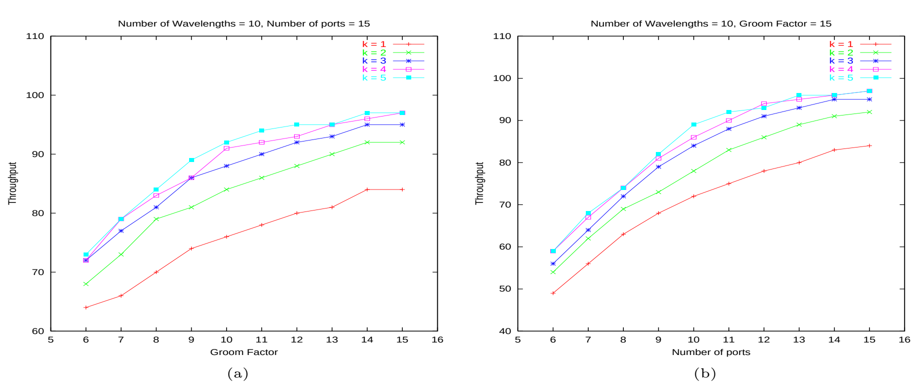


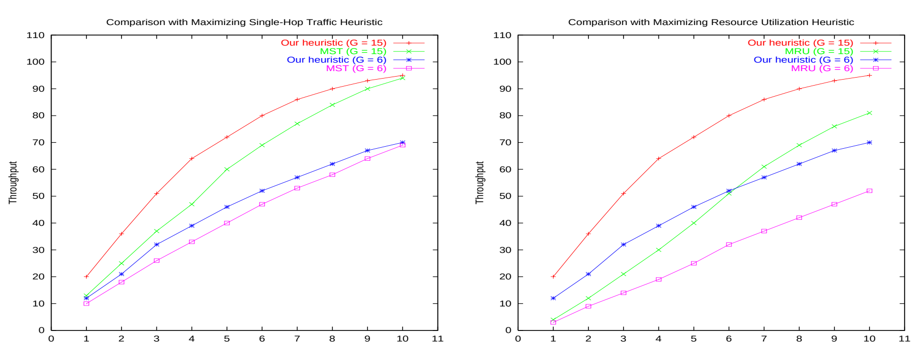
Figure 1.6. Increasing throughput with (a) increasing groom factor and (b) increasing number of ports

ber of OC-3 grooming ports at 15, and then increased the groom factor from 6 to 15. Similarly, to demonstrate increase in throughput with greater grooming capability, we fixed the number of wavelengths at 10 and the groom factor at 15, and then increased the limit on the number of OC-3 ports from 6 to 15. Both these simulations were repeated for values of **k** (where **k**-shortest paths were used for determining *"least-port-increase"* route) varying from 1 to 5. The graphs in Fig. 1.6(a) and Fig. 1.6(b) not only show increase in throughput as desired but also demonstrate that the performance of the heuristic saturates even with the small values of **k** considered. So, even though the heuristic considers only the **k**-shortest paths, the throughput is almost as good as that obtained by a comprehensive search over all routes. The property of throughput increasing with increase in number of wavelengths is shown by the results obtained in the next section.

7.3 Comparison with MST and MRU

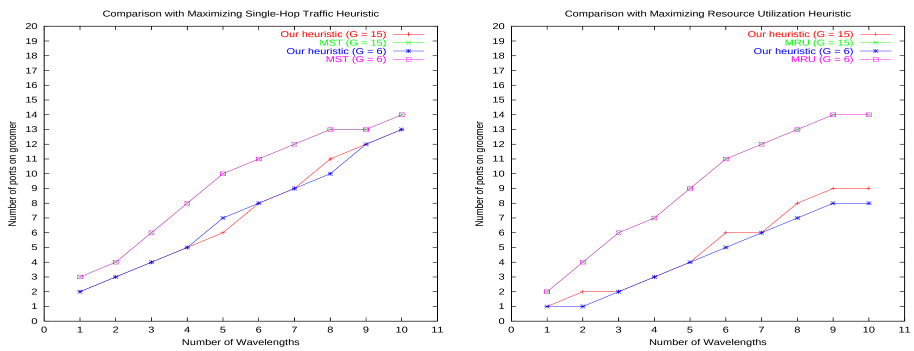
Having proposed the mixed groomer node architecture, we also presented an algorithm for grooming with this setup as we cannot expect the *Maximizing Single-Hop Traffic* (MST) and *Maximizing Resource Utilization* (MRU) heuristics to perform well in this new scenario. To justify the need for our heuristic, we performed various simulations comparing its performance with that of *MST* and *MRU*, and the results clearly highlight the superiority of our heuristic.

Since the objective of our heuristic is to generate higher throughput given the constraint on grooming capability, the obvious way of displaying better performance is by showing higher throughput under the same grooming constraints. For this purpose, during each run of the simulations, we executed the *MST* heuristic, and obtained the throughput it



(a) (b)

Figure 1.7. Comparison of throughput with (a) MST and (b) MRU



(a) (b)

Figure 1.8. Comparison of number of ports with (a) MST and (b) MRU

yields and the number of ports taken up on the groomer at each node. Here too, we evaluated the maximum ports taken up among all nodes as that would be the groomer size required at each node. Using the same groomer size, we executed our heuristic and obtained the corresponding throughput. We conducted these simulations with wavelengths varying from 1 to 10. For each wavelength, the value of the throughput was determined with the value of the groom factor as 6 and 15, in order to demonstrate better performance under both sparse and dense traffic scenarios. The results of this experiment are shown in Fig. 1.7(a). Similar comparisons were carried out with the *MRU* heuristic and the corresponding results are shown in Fig. 1.7(b). These results reflect the fact that with the same amount of resources available, our heuristic performs much more efficient grooming than the *MST* and *MRU* heuristics. This has been shown in scenarios of both dense and sparse traffic.

An alternative way of looking at our problem of obtaining better throughput under grooming constraints is to minimize the grooming capability required to obtain a specific throughput. In light of this new view, we compared our heuristic with *MST* and *MRU* heuristics in terms of number of ports required on the OC-3 groomer to obtain the same throughput. As before, we executed the *MST* heuristic and determined the throughput obtained and the groomer size required to obtain it. We then executed our heuristic repeatedly to find the minimum groomer size required to obtain a throughput greater than that obtained by the *MST* heuristic. Here too, we performed the simulations with wavelengths varying from 1 to 10 and the number of ports corresponding to each wavelength was determined with the groom factor taking the values 6 and 15. The results of these simulations are shown in Fig. 1.8(a). The *MST* heuristic was seen to consume the same number of ports for a given number of wavelengths, irrespective of the groom factor. Hence, the plots with $\mathbf{G} = 6$ and $\mathbf{G} = 15$ for the *MST* heuristic are seen to coincide in Fig. 1.8(a). Results of similar comparisons with the *MRU* heuristic are shown in Fig. 1.8(b) and here too, the plots with $\mathbf{G} = 6$ and $\mathbf{G} = 15$ for the *MRU* heuristic are seen to coincide. The results of this section again indicate the higher efficiency of our heuristic in cases of both dense and sparse traffic as it is able to generate the same throughput as that given by the *MST* and *MRU* heuristics even with far less resources at hand. All the results obtained in this section substantiate the fact that the policy of assigning connections to their *"least-port-increase"* route and rerouting them to save on resources helps our heuristic to obtain excellent results.

8. Conclusion

In this chapter, we proposed a novel node architecture for traffic grooming in WDM optical networks. We listed out the advantages of the mixed groomer architecture in comparison with the MPLS/IP and SONET/WDM node architectures, and outlined its features in the light of practical feasibility, cost-effectiveness and efficient grooming capability. We presented an ILP formulation and also proposed a heuristic for the static grooming problem with the objective of maximizing throughput. We performed a wide range of simulations to demonstrate the efficiency of our heuristic and to display better performance in comparison with the *MST* and *MRU* heuristics. The results obtained in these simulations clearly substantiate our claims.

In the future, we intend to address the issue of dynamic grooming with our mixed groomer node architecture. Also, the concept of survivability can be brought into the focus of the grooming problem, irrespective of the static or dynamic setting.

Acknowledgments

The mixed groomer node architecture was first conceived when the first author undertook a summer internship at Tejas Networks, Bangalore during the period May to July 2002. The authors thank Dr. Sarath Kumar and Dr. Kumar N. Sivarajan at Tejas Networks for their invaluable contributions through insightful discussions.

Notes

1. Though the number of ports on the OC-3 level is *exactly* 4 times that at the OC-12 level, we say switching cost is "more than" 16 times because the intrinsic cost of switching traffic streams increases as we go to finer levels of granularity.

References

[1] Gerstel, O., Lin, P., and Sasaki, G. (1999). Combined WDM and SONET network design. In *Proc. IEEE INFOCOM'99*, pages 734–743.

[2] Gerstel, O. and Ramaswami, R. (2000). Cost effective traffic grooming in WDM rings. *IEEE/ACM Transactions on Networking*, 8(5):618–630.

[3] Hiuban, G., Pérennes, S., and Syska, M. (2002). Traffic grooming in WDM networks with multi-layer-switches. In *Proc. IEEE ICC'02*.

[4] Konda, V. R. and Chow, T. (2001). Algorithm for traffic grooming in optical networks. In *Proc. IEEE HPSR'01*, pages 218–221.

[5] Modiano, E. and Berry, R. (1999). Minimizing electronic multiplexing costs for dynamic traffic in unidirectional SONET ring networks. In *Proc. IEEE ICC'99*, pages 1724–1730.

[6] Modiano, E. and Berry, R. (2000). Reducing electronic multiplexing costs in SONET/WDM rings with dynamically changing traffic. *IEEE Journal on Selected Areas in Communications*, 18(10):1961–1971.

[7] Modiano, E. and Chiu, A. (2000). Traffic grooming algorithms for reducing electronic multiplexing costs in WDM ring networks. *IEEE Journal on Lightwave Technology*, 18(1):2–12.

[8] Thiagarajan, S. and Somani, A. K. (2001). Traffic grooming for survivable WDM mesh networks. In *Proc. SPIE OptiComm'01*, pages 54–65.

[9] Wan, P. J., Calinescu, G., Liu, L., and Frieder, O. (2000). Grooming of arbitrary traffic in SONET/WDM rings. *IEEE Journal on Selected Areas in Communications*, 18(10):1995–2003.

[10] Zhang, X. and Qiao, C. (1999). On scheduling all-to-all personalized connections and cost-effective designs in WDM rings. *IEEE/ACM Transactions on Networking*, 7(3):435–445.

[11] Zhang, X. and Qiao, C. (2000). An effective and comprehensive approach to traffic grooming and wavelength assignment in SONET/WDM rings. *IEEE/ACM Transactions on Networking*, 8(5):608–617.

[12] Zhu, H., Zang, H., Zhu, K., and Mukherjee, B. (2002). Dynamic traffic grooming in WDM mesh networks using a novel graph model. In *Proc. IEEE Globecom'02*.

[13] Zhu, K. and Mukherjee, B. (2002). Traffic grooming in an optical WDM mesh network. *IEEE Journal on Selected Areas in Communications*, 20(1):122–133.