Dynamic Traffic Grooming in WDM Mesh Networks Using a Novel Graph Model

Hongyue Zhu†, Hui Zang‡, Keyao Zhu†, and Biswanath Mukherjee†
† Department of Computer Science, University of California, Davis, CA 95616, USA
Tel: +1-530-752-5129, Fax: +1-530-752-4767, Email: {zhuh, zhuk, mukherje}@cs.ucdavis.edu
‡ Sprint Advanced Technology Laboratories, Burlingame, CA 94010, USA
Tel: +1-650-375-4423, Fax: +1-650-375-4079, Email: hzang@sprintlabs.com

***Abstract*—We employ a new, generic graph model for dynamic traffic grooming in WDM mesh networks. The novelty of this model is that, by only manipulating the edges of an auxiliary graph created by the model and the weights of these edges, the model can achieve various objectives using different grooming policies, while taking into account various constraints. Based on the auxiliary graph, we develop a dynamic traffic-grooming algorithm. Different grooming policies can be implemented by different weight functions assigned to the edges in the auxiliary graph. We propose four fixed grooming policies and an Adaptive Grooming Policy (AGP), and our results show that AGP outperforms the fixed grooming policies.**

I. INTRODUCTION

As wavelength-division multiplexing (WDM) technology continues to mature, there exists a large gap between the capacity of a WDM channel (e.g., OC-48, or OC-192, or OC-768) and the bandwidth requirement of a typical connection request (e.g., STS-1, OC-3, OC-12, etc.). Traffic grooming refers to the problem of efficiently multiplexing a set of low-speed connection requests onto high-capacity channels and intelligently switching them at intermediate nodes. Past research on traffic grooming have focused mainly on SONET/WDM ring networks [1], [2]. As optical WDM backbone networks migrate from rings to meshes, traffic grooming on WDM mesh networks becomes an important research problem [3], [4], [5].

The traffic-grooming problem can be formulated as follows [3]. Given a network configuration and a set of connection requests with different bandwidth granularities, we need to determine how to set up lightpaths to satisfy the connection requests. Because of the sub-wavelength granularity of the requests, one or more connections can be multiplexed on the same lightpath.

In dynamic traffic grooming, the connection requests arrive one at a time with different starting time and holding period. The objective is to minimize the network resources used for each request, which implicitly attempts to minimize the overall blocking probability.

When a connection request arrives, the network operator should determine the following: (1) Should this connection be routed on the current set of lightpaths, i.e., virtual topology, if it is possible to do so? (2) How to change the virtual topology to accommodate the connection? Different decisions on these questions reflect the intentions of the network operator, and referred to as *grooming policies*. By using different grooming policies, a network operator can achieve various objectives. As the network state changes, the objective may also need to change. Dynamically evolving the grooming policy according to the network state is a challenge for traffic grooming.

The WDM backbone network is expected to emerge as a multi-vendor, heterogeneous mesh network. The nodes in the network may have full, partial, or no wavelength-conversion capability. Meanwhile, different node architectures may have full grooming, partial grooming (grooming with some constraints), or no grooming capabilities. These scenarios should be considered when solving the traffic-grooming problem.

In this work, we employ a novel graph model for dynamic traffic grooming in a WDM mesh network. The model consists of an auxiliary graph and a dynamic traffic-grooming algorithm. Various network constraints, such as the number of transceivers at each node, the number of wavelengths on each fiber-link, wavelength-conversion capabilities and grooming capabilities of each node, are represented by different edges in the auxiliary graph. An on-line grooming algorithm is developed based on the auxiliary graph, which applies a shortest-path routing algorithm to the auxiliary graph for each traffic demand and updates the auxiliary graph accordingly. This model can achieve various objectives under different grooming policies. Instead of designing a route-computation algorithm for each grooming policy, simple shortest-path route-computation algorithms can be used to achieve various objectives by carefully choosing the weight functions for the edges in the auxiliary graph. We propose four different fixed grooming policies and an Adaptive Grooming Policy (AGP), which dynamically adjusts the weight functions of the edges in the auxiliary graph according to the current network state, and compare the performances of the grooming policies.

In [5], we first proposed this novel graph model and studied static traffic grooming using this model. In our present work, we use this graph model to solve the dynamic traffic-grooming problem and evaluate the performance of different fixed grooming policies under dynamic environment. Another contribution of our present work is the new AGP proposal.

The paper is organized as follows. In Section II, we demonstrate how to construct an auxiliary graph according to the network state [5]. Based on the auxiliary graph, a dynamic traffic-grooming algorithm is proposed in Section III. Four fixed grooming policies are proposed in Section IV and their performance is compared through numerical examples in Section V. We propose AGP and compare it with the fixed grooming policies in Section V. Section VI concludes the paper.

This work has been supported by the National Science Foundation (NSF) under Grant Nos. NCR-9508239 and ANI-9805285, and by Sprint Advanced Technology Laboratories.

0-7803-7632-3/02/$17.00 ©2002 IEEE

II. CONSTRUCTION OF AN AUXILIARY GRAPH

In order to solve the dynamic-grooming problem, we first construct an auxiliary graph according to the given network configuration (This material was introduced in [5] and repeated here for completeness.)

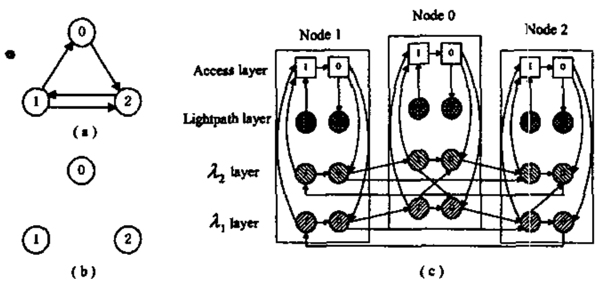

Fig. 1. (a) Physical topology of Network 1. (b) Virtual topology of Network 1. (c) Auxiliary graph of Network 1.

An illustrative example is shown in Fig. 1. In order to make the auxiliary graph clear, we choose a very simple network topology. Network 1 (Fig. 1(a)) is a three-node network with four unidirectional fiber-links, each of which has two wavelengths. All nodes are assumed to have grooming functionality. Node 0 has wavelength converters with full wavelength-conversion capability, node 1 has no wavelength converter, and node 2 has wavelength converters that can only convert λ_1 to λ_2. Initially, there is no lightpath in the network, so there is no edge in the virtual topology of Network 1, as shown in Fig. 1(b). An auxiliary graph is constructed as in Fig. 1(c).

In general, a network can be represented by a graph $G_0(V_0, E_0)$, where V_0 and E_0 are its node set and link set, respectively. We construct the corresponding auxiliary graph $G(V, E)$ with vertex set V and edge set E. Assuming that each link has W wavelengths, λ_1 through λ_W, $G(V, E)$ is constructed as follows.

Auxiliary graph G is a layered graph with $(W + 2)$ layers. Layers 1 through W denote the W *wavelength layers*, layer $(W + 1)$ is called the *lightpath layer*, and layer $(W + 2)$ is called the *access layer*, where a traffic flow starts and terminates. Each node has two ports on each layer, denoted by two vertices, an input port (a vertex marked with "I") and an output port (a vertex marked with "O"). The edges are inserted in auxiliary graph G as follows.

- *Wavelength Bypass Edges (WBE).* There is an edge from the input port to the output port on each wavelength layer at node i.
- *Grooming Edges (GrmE).* There is an edge from the input port to the output port on access layer at node i if node i has grooming capability.
- *Mux Edges (MuxE).* There is an edge from the output port on the access layer to the output port on the lightpath layer at each node.
- *Demux Edges (DmxE).* There is an edge from the input port on the lightpath layer to the input port on the access layer at each node.
- *Transmitter Edges (TxE).* There is an edge from the output port on the access layer to the output port on wavelength layer l if there are transmitters available on wavelength λ_l at node i.
- *Receiver Edges (RxE).* There is an edge from the input port on wavelength layer l to the input port on the access layer if there are receivers available on wavelength λ_l at node i.
- *Converter Edges (CvtE).* There is an edge from the input port on wavelength layer l_1 to the output port on wavelength layer l_2 at node i if wavelength λ_{l_1} can be converted to wavelength λ_{l_2} at node i.
- *Wavelength-Link Edges (WLE).* There is an edge from the output port on wavelength layer l at node i to the input port on wavelength layer l at node j if there is a physical link from node i to node j and wavelength λ_l on this link is free.
- *Lightpath Edges (LPE).* There is an edge from the output port on the lightpath layer at node i to the input port on the lightpath layer at node j if there is a lightpath from node i to node j. There is no such edge in Fig. 1(c) because there is no lightpath set up yet.

Each edge in the auxiliary graph G has a property tuple $P(c, w)$ associated with it, where c denotes the capacity of this edge and w denotes its weight. For a wavelength-link edge, its capacity is the capacity of the corresponding wavelength on the corresponding link. For a lightpath edge, its capacity is the residual capacity of the corresponding lightpath. For all the other types of edges, we set the capacity to ∞. The weights of edges can reflect the cost of each network element (transceiver, wavelength-link, wavelength converter, etc.), and/or a certain grooming policy. The weights can either be fixed, or be adjusted according to the current network state. A fixed weight assignment reflects a fixed grooming policy, while an adjustable weight assignment reflects an adaptive grooming policy.

From the above procedure, it should be clear that the auxiliary graph reflects the current state of the network, and the network can be heterogeneous, with different nodes having different resources and capabilities.

III. DYNAMIC TRAFFIC-GROOMING ALGORITHM

Based on the auxiliary graph, we develop a dynamic-grooming algorithm. The input includes the initial network state and a set of traffic demands, each of which has different arrival and departure time, and can be represented by $T(s, d, g, m)$, where s and d are the source and destination nodes, respectively; g is the granularity of the traffic demand, for instance, OC-48; and m is the amount of the traffic in units of g. The algorithm works as follows.

- Initialization:
  Construct the corresponding auxiliary graph G according to the initial network state.
- When a connection request T arrives:
  - Step 1 Compute the shortest path p from the output port on the access layer of the source to the input port on the access layer of the destination of T on graph G. If such a path does not exist, block the traffic demand; otherwise, continue with the following steps.

- Step 2 If p contains wavelength-link edges, set up one or more lightpaths going through the corresponding wavelength-links.
- Step 3 Route T along the pre-existing lightpaths in p and/or lightpaths newly set up according to p.
- Step 4 Update graph G as follows:
  - For each newly setup lightpath, a lightpath edge from the output port of the starting node of the lightpath to the input port of the ending node of the lightpath is added on the lightpath layer.
  - The wavelength-link edges used by the lightpath are removed from the corresponding wavelength layers.
  - If there is no more transmitter/receiver available at node i on wavelength λ_l, the corresponding transmitter/receiver edge will be removed from G, i.e., this node cannot source/sink a lightpath on wavelength λ_l any more and can only be bypassed by a lightpath.
  - If there is no more wavelength converter which can convert wavelength λ_{l_1} to wavelength λ_{l_2} available at node i, the converter edge will be removed from G.
  - Update the property tuple $P(c, w)$. of the edges. For the lightpaths carrying the traffic T, the capacities of the corresponding lightpath edges are decreased by the amount of the carried traffic. How to update the weights of the edges in the graph is determined by the grooming policies, which will be addressed in Section IV.

- When a connection terminates:
  - Step 1 Remove the traffic from the network.
  - Step 2 Tear down all the lightpaths that do not carry any traffic.
  - Step 3 Update graph G by applying the reverse of the update method used in Step 4 above. We omit the details for terseness.

In the algorithm, a connection request is routed according to the shortest path from the source to the destination in the auxiliary graph. Therefore, the weights of the edges in the auxiliary graph will determine how to carry a connection in the network. Different weight settings will achieve different objectives and network performance.

IV. GROOMING POLICIES

A grooming policy determines how to carry the traffic in a certain situation. In general, for a traffic demand $T(s, d, g, m)$ in a network, there are four possible operations that can be used to carry the traffic without altering the existing lightpaths. Note that we do not consider reconfiguring existing lightpaths because, then, the traffic on the network would be interrupted.

- *Operation 1*: Route the traffic onto an existing lightpath directly connecting the source s and the destination d.
- *Operation 2*: Route the traffic through multiple existing lightpaths.
- *Operation 3*: Set up a new lightpath directly between the source s and the destination d and route the traffic onto this lightpath. Using this operation, we set up only one lightpath if the amount of the traffic is less than or equal to the capacity of the lightpath.
- *Operation 4*: Set up one or more lightpaths that do not directly connect source s and destination d, and route the traffic onto these lightpaths and/or some existing lightpaths. Using this operation, we need to set up at least one new lightpath. However, since some existing lightpaths may be utilized, the number of wavelength-links used to set up the new lightpaths could be less than the number of wavelength-links needed to set up a lightpath directly connecting source s and destination d.

Each operation has certain prerequisites for it to be applied. For instance, if there is no lightpath between the source and the destination that can accommodate the traffic, then Operation 1 cannot be used. In some situations, all the operations are applicable, while in other situations, only some of them are. If none of them can be applied, the traffic must be blocked without reconfiguring the existing lightpaths.

In a situation where multiple operations can be applied, how to choose the appropriate operation is a matter of the grooming policy. By combining the various operations in different priority order, we can achieve different grooming policies.

Below, we present four different grooming policies.

- *Minimize the Number of Traffic Hops on the Virtual Topology (MinTHV).*
  We first use Operation 1. If Operation 1 fails, we always try to set up a lightpath from s to d and route the traffic onto this lightpath (Operation 3). Only when such a direct lightpath cannot be set up, we use multi-hop grooming by either Operation 2 or Operation 4, and choose the one with fewer hops on the virtual topology (number of lightpaths). This policy chooses the route with the fewest lightpaths for a connection.
- *Minimize the Number of Traffic Hops on the Physical Topology (MinTHP).*
  We compare the number of wavelength-links used by all the four operations and choose the one with the fewest wavelength-links.
- *Minimize the Number of Lightpaths (MinLP).*
  This policy is similar to MinTHV but it tries to set up the minimal number of *new* lightpaths to carry the traffic. Operation 1 is attempted first. If it fails, we try to route the traffic using multiple existing lightpaths (Operation 2). If Operation 2 also fails, we try to set up one lightpath with the minimal number of wavelength-links either by Operation 3 or Operation 4. If such a lightpath is not feasible, we go with Operation 4 and set up two or more lightpaths.
- *Minimize the Number of Wavelength-Links (MinWL).*
  This policy is similar to MinTHP but it tries to consume the minimum number of *extra* wavelength-links, i.e., wavelength-links not being used by any lightpaths for now, to carry the traffic. The difference between MinLP and MinWL is that, if both Operations 1 and 2 fail, MinWL compares the number of wavelength-links used by Operations 3 and 4, and chooses the one requiring fewer

wavelength-links; MinLP, on the other hand, compares the number of lightpaths used by Operations 3 and 4, chooses the one requiring fewer lightpaths, and uses the number of wavelength-links for tie-breaking.

Our graph model can easily implement these grooming policies by applying different weight-assignment functions. We call an edge in an auxiliary graph the dominant edge if this edge satisfies the following condition: if a path p_1 in the graph contains more of this kind of edges than another path p_2, then the weight of p_1 is always larger than that of p_2. Here, the weight of a path is the summation of the weights of the edges it traverses. To achieve MinTHV, we just need to make GrmEs the dominant edges. To achieve MinLP, we should make TxEs and RxEs the dominant edges. To achieve MinWL, WLEs should be the dominant edges. In all the above three policies, the weight of the LPEs is fixed and less than that of WLEs. To achieve MinTHP, we also make WLEs the dominant edges. In addition, a LPE is considered as a concatenation of WLEs whose corresponding wavelength-links are used by the lightpath represented by the LPE in the auxiliary graph, and the weight of a LPE is the summation of the weight of those WLEs.

In dynamic grooming, the network state changes as connection requests come and go. To achieve good performance, the grooming policy should be adjusted according to the current network state. For instance, if transceivers are becoming the more scarce resource, we should make full use of existing lightpaths to accommodate the new traffic and avoid setting up new lightpaths. This requirement can be easily satisfied by modifying the weights of edges according to the current network state. This capability of easily adjusting grooming policies makes the graph model very suitable for dynamic traffic grooming.

V. NUMERICAL EXAMPLES

We compare the performance of different grooming policies on the network topology shown in Fig. 2, which has 19 nodes and 31 links. All the nodes have grooming capability but no wavelength-conversion capability. Each link is bidirectional with $W = 16$ wavelengths in each direction, and each wavelength has a capacity of OC-192. The traffic arrival is a Poisson process and the connection holding time is exponentially distributed (whose average value is normalized to unity in our studies reported here). The traffic is uniformly distributed among all node pairs. There are four types of connection requests: OC-3, OC-12, OC-48, and OC-192, and the proportion of the number of these connections is 6:6:6:1. For a connection request $T(s, d, g, m)$, m, the amount of the traffic in unit of g, is uniformly distributed between 1 and 32, 1 and 16, 1 and 8, and 1 and 2 for OC-3, OC-12, OC-48, and OC-192 types of connections, respectively. We simulate 100,000 connection requests to obtain the network performance under a certain scenario and a grooming policy. We ran our simulation experiments on a Linux PC with a 1.5-GHz Pentium IV processor and 512-MB memory. Each data point reported in the illustrations in this section took between 6-9 minutes of running time on this computer.

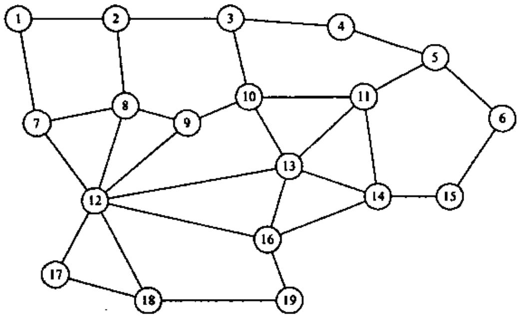

Fig. 2. A 19-node telecom network.

TABLE I

AVERAGE UTILIZATION OF WAVELENGTH-LINKS AND TRANSCEIVERS WHEN W=16 AND L=300 ERLANGS.

	$Tx = 16$		$Tx = 32$		$Tx = 40$	
	U_W	U_{Tx}	U_W	U_{Tx}	U_W	U_{Tx}
MinTHV	0.7819	0.9858	0.8878	0.7264	0.8905	0.5884
MinTHP	0.5674	0.9901	0.7354	0.8165	0.7361	0.6910
MinLP	0.7403	0.9807	0.8890	0.8007	0.8918	0.6651
MinWL	0.6201	0.9859	0.8133	0.8683	0.8120	0.7825

Table I shows the average utilization of wavelength-links (U_W) and the average utilization of transceivers (U_{Tx}) when the network load L is 300 Erlangs. When each node has only 16 transceivers, the utilization of transceivers is very high since they are the more constrained resources. When there are 32 transceivers at each node, the utilizations of both transceivers and wavelengths are quite balanced and high as well. If there are 40 transceivers at each node, we have relatively more transceivers; hence, wavelength-links become the more constrained resources, so the utilization of the transceivers is relatively lower.

Figure 3 shows the network performance under different grooming policies with $Tx = 32$ transceivers at each node. We observe that, as the network load increases, the perçentage of blocked traffic also increases, but different grooming policies have different blocking probabilities. Grooming policy MinTHV performs best, followed by MinTHP, MinLP, and MinWL in sequence.

When we alter the network configuration by adding more transceivers at each node, the performance of each of the policies also changes, as shown in Fig. 4. In this scenario, each node has 40 transceivers instead of 32. We observe that, now, MinTHP outperforms MinTHV and achieves the best results, and MinLP becomes the poorest-performing policy. This is because, in this network configuration, there are relatively more transceivers in the network so that wavelength-links become the more constrained resources. Recall that MinTHP utilizes wavelength-links more efficiently than other policies; hence, it performs the best in this case.

Another observation from both Figs. 3 and 4 is that MinTHV and MinTHP always perform better than MinLP and MinWL in terms of percentage of blocked traffic. This is because MinTHV and MinTHP examine the *overall* resource requirement of a given connection, while MinLP and MinWL only consider the new lightpaths to be set up or the extra wavelength-links used by these new lightpaths while setting up the connection. There-

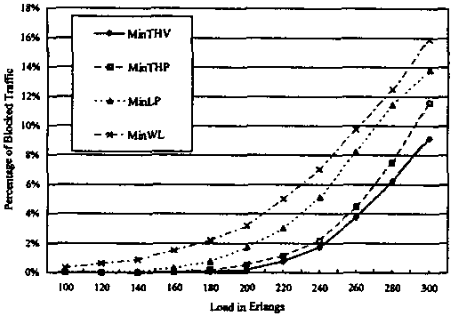

Fig. 3. Percentage of blocked traffic when $Tx = 32$.

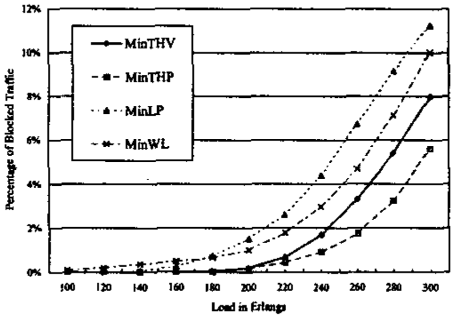

Fig. 4. Percentage of blocked traffic when $Tx = 40$.

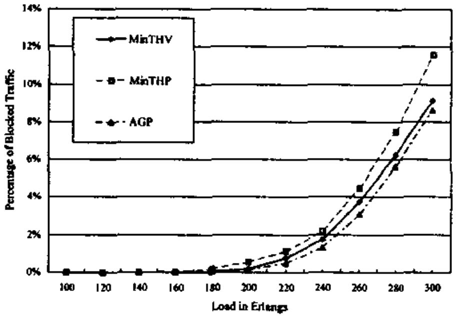

Fig. 5. Performance of AGP when $Tx = 32$.

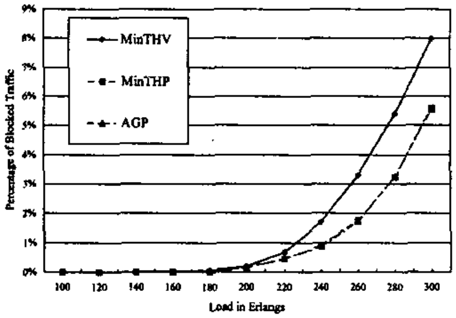

Fig. 6. Performance of AGP when $Tx = 40$.

fore, MinTHV and MinTHP are more resource-efficient.

From the above results, we can observe that different grooming policies have different performance under various network configurations, which suggests that a grooming policy should be adjusted according to the current network state.

A. Adaptive Grooming Policy (AGP)

Since MinTHV performs best when transceivers are the more constrained resources and MinTHP gives the best results when wavelength-links become more scarce resources, we try to utilize the advantages of these two grooming policies by combining them together. Here we present an Adaptive Grooming Policy (AGP) which, for each connection request, switches between MinTHV and MinTHP according to the current network state.

We use the ratio of the number of unused wavelength-links in the network to the total number of available transceivers at all nodes as an indicator of the network state. If the ratio is larger than Δ_1, MinTHV will be used to avoid setting up lightpaths since transceivers are more scarce resources at this time; if the ratio is less than Δ_2, MinTHP will be employed to try to save wavelength-links as much as possible; if the ratio is in between, the policy will not be changed.

For our numerical examples, we report results for $\Delta_1 = 1.2$ and $\Delta_2 = 1.0$. (We experimented with other combinations of these two parameters, and found these choices of values to perform the best for the network topology in Fig. 2.) Our results are shown in Figs. 5 and 6. We observe that the Adaptive Grooming Policy (AGP) achieves the best results under different network configurations.

VI. CONCLUSION

In this study, we used a novel graph model for dynamic traffic grooming in a WDM optical mesh network. We proposed four different fixed grooming policies and an Adaptive Grooming Policy (AGP). Our results show that different grooming policies have various performance under different network configurations and AGP can achieve the best performance.

REFERENCES

[1] X. Zhang and C. Qiao, "An effective and comprehensive approach for traffic grooming and wavelength assignment in SONET/WDM rings," *IEEE/ACM Transactions on Networking*, vol. 8, pp. 608–617, October 2000.

[2] O. Gerstel, R. Ramaswami, and G. H. Sasaki, "Cost-effective traffic grooming in WDM rings," *IEEE/ACM Transactions on Networking*, vol. 8, pp. 618–630, October 2000.

[3] K. Zhu and B. Mukherjee, "Traffic grooming in WDM optical mesh networks," *IEEE Journal on Selected Areas in Communications*, vol. 20, pp. 122–133, January 2002.

[4] S. Thiagarajan and A. Somani, "Traffic grooming for survivable mesh networks," in *Proc., OPTICOMM '01*, (Denver, CO), August 2001.

[5] H. Zhu, H. Zang, K. Zhu, and B. Mukherjee, "A novel, generic graph model for traffic grooming in heterogeneous WDM mesh networks," *IEEE/ACM Transactions on Networking*, accepted for publication.