Cross-Layer Path Computation for Dynamic Traffic Grooming in Mesh WDM Optical Networks

Feng He and Chunsheng Xin

Dept. of Computer Science
Norfolk State University
Norfolk, VA 23504
kmhefeng@hotmail.com cxin@nsu.edu

Technical Report #NSUCS-2004-009

Abstract—**Traffic grooming in wavelength division multiplexing (WDM) optical networks is an operation to consolidate multiple low-speed traffic streams onto high-speed lightpaths. Traffic grooming of dynamic traffic over a dynamic logical topology, where lightpaths (logical links) are dynamically set up and torn down, is called *dynamic traffic grooming*, where a major task is to find a path (a list of adjacent lightpaths) in the dynamic logical topology to carry an incoming call. Dynamic traffic grooming is a significant issue in WDM optical networks due to the customer's need of the "bandwidth-on-demand" service. This paper studies path computation in dynamic traffic grooming, using various link cost functions. We propose a path computation approach utilizing the resource usage information in both logical and physical layers. We have evaluated the performance of this cross layer path computation approach, compared with the approach using the resource usage information in the logical layer only. The numerical results show that the cross layer path computation approach can achieve 10% – 20% lower blocking probability.**

I. INTRODUCTION

The wavelength division multiplexing (WDM) optical network is composed of optical nodes and optical fiber links, typically forming a mesh topology. Due to the huge transmission bandwidth of the optical fiber, the WDM optical network is typically used as the transport infrastructure for other data networks, e.g., Internet Protocol (IP) networks and Asynchronous Transfer Mode (ATM) networks. The major role of WDM optical networks is to interconnect these client networks and serves as a national or international backbone to transport client traffic. The current WDM optical networks utilize wavelength routing to provision optical connections, called *lightpaths*, to interconnect client networks. The lightpath occupies one dedicated wavelength on each traversed link in the WDM optical network, and can be either unidirectional or bidirectional. One lightpath interconnects two client nodes (e.g., IP routers, ATM switches) across a WDM optical network, and is treated as a *logical link* by the two client nodes to transports traffic from the source client node to the destination.

The client traffic is assumed consisting of traffic flows, or *calls*, in this paper, which can be ATM virtual path/channels (VP/VCs), or Label Switched Paths (LSPs) in IP networks with the Multi-Protocol Label Switching technology. There are two significant issues to transport client calls using lightpaths. First, due to the customer's need for "bandwidth-on-demand" service, client calls are typically dynamic. Therefore, the WDM optical network needs to be able to dynamically establish lightpaths to accommodate the increasing bandwidth requirement by dynamically arriving client calls, and dynamically tear down lightpaths to release wavelength resource when client calls are terminated. This dynamic lightpath provisioning can be performed using the GMPLS control plane developed in the IETF. The second issue in transporting client calls using lightpaths is that client calls have heterogeneous and arbitrary data rates, generally much smaller than the lightpath capacity. The lightpath bandwidth is at the granularity of one wavelength, typically 10 Gbps or higher, and the data rate of a client call is typically from a few Mbps to several 100 Mbps. To address this issue, client calls are consolidated onto lightpaths in a cost-effective way to achieve high lightpath utilization. More specifically, for each incoming call, we need to find a path consisting of logical links in the dynamic logical topology to carry it. The path to carry a client call must reserve dedicated bandwidth for this call on each traversed logical link. While searching for a path, the grooming algorithm may request the WDM optical network to set up a logical link in real time. This process of finding paths to consolidate client calls over the logical topology is called *traffic grooming*. Fig. 1 illustrates the traffic grooming in a WDM optical network. The calls arriving at client node C and destined to node D are carried over two paths. The logical link between node C and B is a lightpath traversing three optical nodes in the WDM optical network.

The traffic grooming problem originates from SONET/WDM rings, where the traffic is static, i.e., traffic streams are given in advance and permanent, and the bandwidth of each traffic stream is fixed. In the mesh WDM optical network (simply called optical network hereafter), although the client traffic may be static, it is typically dynamic, since the dominant client traffic are MPLS/IP LSPs and ATM VP/VCs, which are inherently dynamic traffic flows. We call the grooming of dynamic traffic as *dynamic traffic grooming*. The fundamental issue of dynamic traffic grooming is to find a path (with possible dynamic setup of lightpaths) with enough residue bandwidth for each incoming call. It is not possible to design an optimal algorithm to achieve optimal performance, since the client traffic is dynamic. As such heuristics algorithms are generally developed for this purpose. The most used heuristic is the shortest path algorithm, similar to the routing algorithms.

Dynamic traffic grooming has recently attracted quite a few research efforts in the literature. Heuristics have been proposed in [1] – [3], focusing on selecting logical links for setup to accommodate an incoming call, in case that there is no path of existing logical links with enough residue bandwidth. The performance analysis of dynamic traffic grooming in different class of grooming algorithms has been conducted in [4] – [7]. In this paper, we focus on computing paths of existing logical links, using various cost functions, to carry incoming calls. We assume that in case there is no path with enough residue bandwidth to carry an incoming call, a new logical link is requested to be set up between the source and destination client nodes of this incoming call. Note that in [1] – [3], a logical link may be requested to be set up between client nodes other than the source and destination nodes of the incoming call. Nevertheless, this requires that the optical network has a centralized control, or a request needs to be sent from the source/destination node to another node to set up a logical link there. This requirement may not be practical, or desirable for many optical networks. Therefore, we do not consider this case in this study.

The rest of the paper is organized as follows. Section II describes the network models in dynamic traffic grooming. Section III introduces path computation utilizing only the logical layer information, or cross layer information. Section IV presents numerical results, and Section V concludes the paper.

II. NETWORK MODELS OF DYNAMIC TRAFFIC GROOMING

Traffic grooming is a process involved in both physical and logical topologies. The physical topology consists of client nodes (more specifically the border client nodes in client networks), optical nodes, and fiber links. Fig. 1 illustrates a physical topology (excluding the lightpaths). The optical network consisting of optical nodes and fiber links is used as a high-speed core to interconnect client nodes (in different client networks), and to establish lightpaths between them. A lightpath is treated as a logical link by the two end client nodes for traffic transport, and the lightpaths established among client nodes form a *logical topology*, illustrated in Fig. 2. The client traffic transport (among all client nodes) is logically performed over this logical topology. Note that since the logical topology is dynamic, Fig. 2 actually shows a logical topology at a specific time. The dashed line in Fig. 2 indicates a potential logical link that can be setup between client nodes A and D (although currently there is no logical link between them). There may be multiple logical links between two client nodes, since multiple lightpaths can be set up between them.

Dynamic traffic grooming is essentially selecting a path over the (dynamic) logical topology to carry a client call, i.e., a client call routing problem over the logical topology. The traffic grooming algorithm is similar to a routing algorithm. The difference is that in dynamic traffic grooming, the bandwidth is provisioned on-demand.

III. PATH COMPUTATION IN DYNAMIC TRAFFIC GROOMING

As discussed in the above section, traffic grooming is essentially a client call routing problem in the logical topology. Therefore the path computation can be simply conducted using the resource usage information in the logical layer. Specifically, the logical link cost can be based on the used bandwidth. We call this approach as *logical layer path computation* (LogPac). Nevertheless, different from the routing problem, dynamic traffic grooming is a process involved in two layers. Therefore we propose to utilize resource usage information of both logical and physical layers in path computation for dynamic traffic grooming. We call this approach as *cross layer path computation* (CrosPac).

A. Logical Layer Path Computation (LogPac)

Similar to the route computation in the routing protocol, LogPac can simply use the shortest path algorithm (e.g., Dijkstra's algorithm) to compute a shortest path based on some link cost function. We examine three cost functions for logical links based on the logical layer resource usage information. Let Λ be a logical link (i.e., a lightpath) in the logical topology. Let $\gamma(\Lambda)$ and $\eta(\Lambda)$ denote the capacity and used bandwidth of logical link Λ, respectively. The cost of Λ in LogPac is denoted as $C_l(\Lambda)$. We define the different type of cost functions for logical link Λ.

- *Hop count*: $C_l(\Lambda)=1$.
- *Used bandwidth* (BW): $C_l(\Lambda)=\eta(\Lambda)$.
- *Normalized used bandwidth* (NBW): $C_l(\Lambda)=\left\lceil \frac{\eta(\Lambda)N}{\gamma(\Lambda)} \right\rceil$, where N is a small normalization parameter, e.g., 5. The use of this cost function is because the above cost function "BW" may have a large variation.
- General type: $C_l(\Lambda)=f(\eta(\Lambda),\gamma(\Lambda))$, where function $f(\bullet)$ is a strictly increasing function, such as, $f(\eta(\Lambda))=\frac{1}{\gamma(\Lambda)-\eta(\Lambda)}$, $f(\eta(\Lambda))=\log(\eta(\Lambda))$, etc.

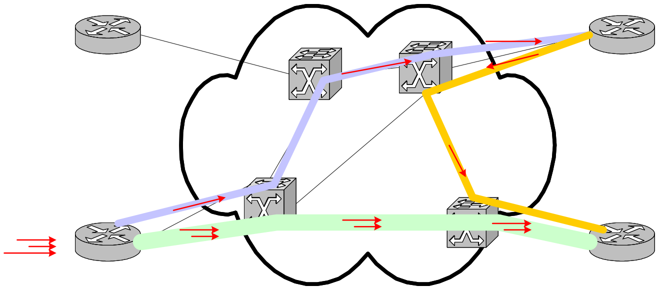


Fig. 1 Traffic grooming illustration

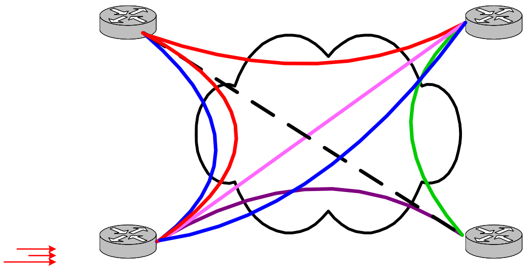


Fig. 2 Logical topology

In this paper, we will only examine the first three cost functions for LogPac.

B. *Cross Layer Path Computation (CrosPac)*

The CrosPac utilizes the neighboring information among client nodes in the logical layer as a connectivity matrix, and the wavelength usage by a logical link in the physical layer as the basic cost of this logical link, because we think that in traffic grooming, the number of wavelengths used by a logical link in the physical layer is even more important than the used bandwidth on the logical link. Certainly the bandwidth usage on a logical link can be also part of the cost function. We examine two cost functions, one using the wavelength usage only, and the other mixing both the wavelength usage by a logical link and bandwidth usage on the logical link. With these two cost functions, a shortest path algorithm can be utilized to compute a path for an incoming call.

Let $C_p(\Lambda)$ denote the cost of logical link Λ in CrosPac. Let $W(\Lambda)$ be the number of wavelengths used by Λ inside the optical network (excluding the wavelengths on user-network interface (UNI) links). The two cost functions are defined as follows:

- *Wavelength usage* (Wave): $C_p(\Lambda) = W(\Lambda)$.
- *Mixed resource usage* (Mix): $C_p(\Lambda) = W(\Lambda) + g(C_l(\Lambda))$, where $C_l(\Lambda)$ is the cost function used in LogPac, and $g(x)$ is a function to normalize x into [0, 1] to give a bias in $C_p(\Lambda)$.

In the "Mix" cost function, we may alternately define $C_p(\Lambda) = C_l(\Lambda) + g(W(\Lambda))$, and in this case, $C_l(\Lambda)$ should be in a small range to avoid the second term having no effect on $C_p(\Lambda)$.

We further propose a CrosPac scheme that first computes a set of shortest paths using the "Wave" cost function and then select one path with maximum residue bandwidth. We call this scheme as CrosPac with maximum residue bandwidth (MRB) or CrosPac–MRB. The residue bandwidth of a path $\Pi = (\Lambda_1, \cdots, \Lambda_J)$ is the minimum residue bandwidth of its links, i.e., $\min_j \{\gamma(\Lambda_j) - \eta(\Lambda_j)\}$.

Let c denote the path cost of the shortest path. Let Δ be a small number. In CrosPac–MRB, we first compute a set of paths whose costs are less than $c + \Delta$. These paths are the shortest, second shortest, …, and K^{th} shortest paths, with K as an unknown. In the literature, the Yen's algorithm [8] is efficient to compute K shortest paths. Since K is not known in advance in our computation, we make a small change to this algorithm. The Yen's algorithm iteratively computes the shortest, second shortest, …, i^{th} shortest path, until the K^{th} shortest path is obtained and then returns all K shortest paths. We have simply modified this algorithm to keep computing paths until the path cost of the current path is larger than $c + \Delta$. The efficiency of the modified algorithm is the same as the original algorithm. The CrosPac–MRB can be described as follows.

Algorithm : CrosPac–MRB

Input: an incoming call.
Output: a path to carry the incoming call.

1. *Use the modified Yen's algorithm to find a set of* K *shortest paths,* $\Pi_1, \cdots, \Pi_K$*, with the path costs less than* $c + \Delta$ *, based on the "Wave" cost function.*
2. *Denote the residue bandwidth of* $\Pi_1, \cdots, \Pi_K$ *as* $\mathrm{P}_1, \cdots, \mathrm{P}_K$.

3. *Select path* Π_k *such that* $\mathrm{P}_k = \min_j \{\mathrm{P}_j\}$.

Alternatively, we may first compute a set of shortest paths using the logical layer cost functions and then select a path based on the wavelength usage of each path in the physical layer. Nevertheless, this study will not examine this approach.

IV. NUMERICAL RESULTS

The sample network is the 14-node NSFNET as illustrated in Fig. 3. We assume each optical node is attached by one client node. There is one bi-directional fiber link between two neighboring optical nodes and every fiber contains 16 wavelengths. Client calls are generated following Poisson arrival and exponential service. The offered load between each pair of client nodes is assumed uniformly distributed, generated as $b*(1+\chi)$, where χ denotes a uniform random variable in [0, 1], and b is a scale number to control the amount of offered load. In the following discussion, the “offered load” refers to b. We assume that the logical link capacity is 100 and the call data rate is arbitrary between 1 and 100. We have studied two types of client call data rate distributions: uniform distribution (evenly distributed), and log normal distribution (unevenly distributed) with parameters $\mu = 2.83258$ and $\sigma = 1$ (which means that the data rate of an incoming call will be OC-3 with the highest probability, in the case that the lightpath capacity is OC-48). The shortest path routing for lightpaths and first-fit wavelength assignment are used in the optical network. In CrosPac–MRB, the allowed path cost difference (Δ) is set as 2. The LogPac using the cost function “Hop Count”, “BW”, “NBW” (defined in the preceding section), will be called as LogPac–HopCount, LogPac–BW, and LogPac–NBW, respectively, in the following discussion. Similarly, the CrosPac using the cost function “Wave” and “Mix” will be called as CrosPac–Wave, and CrosPac–Mix, respectively. In LogPac–NBW, the normalization parameter N is assumed as 5.

Table I presents the average blocking probabilities while the call data rate distribution is log normal and there is no wavelength conversion in the optical network. Fig. 4 plots both the average blocking probability and throughput, after normalization using the numerical results of LogPac–BW as basis, i.e., other results have been divided by the corresponding quantity of LogPac–BW. We can see that among LogPac schemes, the LogPac–BW performs approximately best, with the LogPac–HopCount as the worst due to its unawareness of bandwidth utilization. Furthermore, the CrosPac schemes clearly perform better than LogPac schemes, although while the offered load is very small or large, the performance of both approaches is approximately the same. The blocking probabilities of CrosPac schemes are approximately 10% – 20% lower than these of LogPac schemes, which is a significant

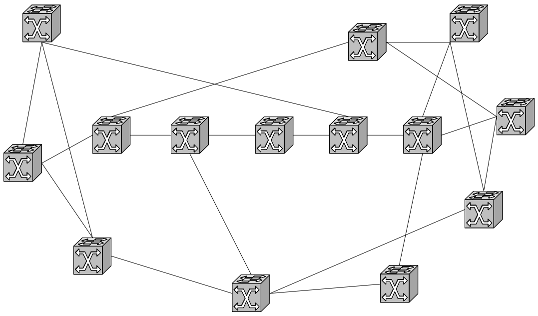
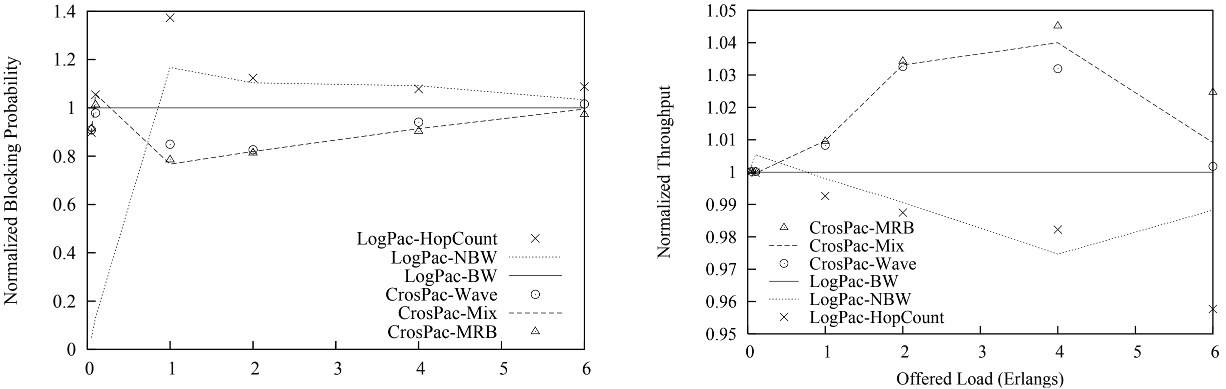


Fig. 3 The 14-node NSFNET

Table I Average blocking probability with call data rate distribution as log normal, and no wavelength conversion

Offered load	0.05	0.1	1	2	4	6
LogPac–BW	0.0023	0.0065	0.011	0.053	0.133	0.237
LogPac–NBW	0.00011	0.00086	0.013	0.058	0.146	0.244
LogPac–HopCount	0.0021	0.0068	0.015	0.059	0.144	0.257
CrosPac–Mix	0.0021	0.0068	0.0084	0.043	0.122	0.235
CrosPac–Wave	0.0021	0.0063	0.0093	0.043	0.125	0.24
CrosPac–MRB	0.0021	0.0066	0.0086	0.042	0.12	0.22



(a) (b)

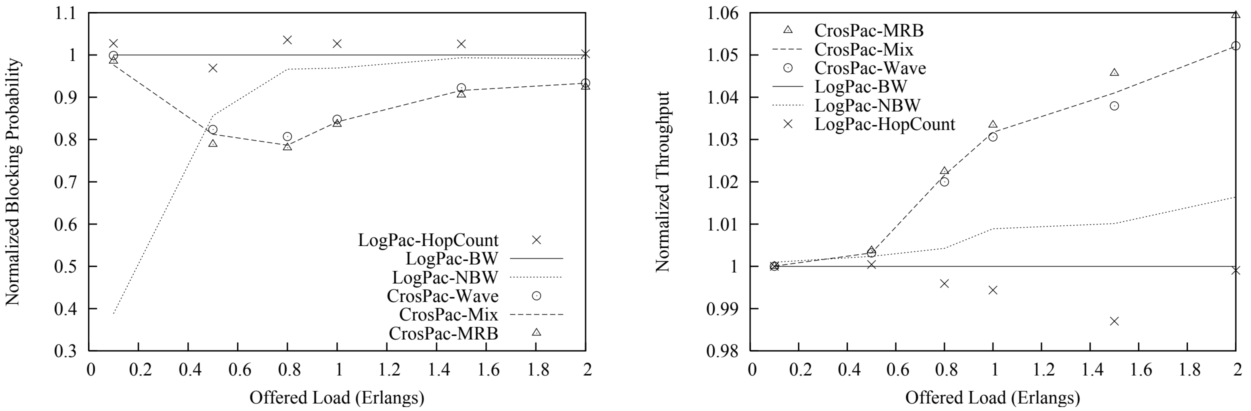
Fig. 4 Numerical results with call data rate distribution as log normal, and no wavelength conversion, (a) blocking probability, (b) throughput

Table II Throughput with call data rate distribution as uniform, and with wavelength conversion

Offered load	0.1	0.5	0.8	1	1.5	2
LogPac–BW	0.9986	0.9862	0.9099	0.8342	0.6703	0.5535
LogPac–NBW	0.9996	0.9886	0.9137	0.8416	0.6771	0.5627
LogPac–HopCount	0.9986	0.9867	0.9062	0.8295	0.6616	0.553
CrosPac–Mix	0.9987	0.9894	0.9295	0.8607	0.6978	0.5823

improvement. The throughput improvement of the CrosPac approach compared with the LogPac approach is also acceptable, approximately 4%. The reason that the blocking probability improvement is greater than the throughput improvement is that the CrosPac approach accepts significantly more low data rates calls than the LogPac approach, which, however, does not have a profound contribution to throughput improvement. Within CrosPac schemes, the CrosPac–MRB is slightly better than the other two schemes, possibly due to its use of the residue bandwidth on the entire path instead of using the aggregate path cost.

Table II presents the throughput while the call data rate distribution is uniform, and there is wavelength conversion in the optical network. Fig. 5 plots both the average blocking probability and throughput (after normalization using the numerical results of LogPac–BW as basis) for this scenario. The numerical results from this scenario exhibit a similar property as in the scenario of log normal call data rate distribution. A minor difference is that LogPac–NBW performs slightly better than LogPac–BW. The performance improvement of the CrosPac approach compared with the LogPac approach also



(a) (b)

Fig. 5 Numerical results with call data rate distribution as uniform and with wavelength conversion, (a) blocking probability, (b) throughput

exhibits a similar feature, with a 10% – 20% lower blocking probability and 3% – 4% higher throughput, and the throughput improvement is slightly better than the previous scenario.

V. CONCLUSION

In this paper, we have performed an experimental study on dynamic traffic grooming in wavelength division multiplexing (WDM) optical networks with mesh topologies. Traffic grooming is an operation to efficiently consolidate client calls onto lightpaths, which form a logical topology. Dynamic traffic grooming in WDM optical networks is essentially a client call routing problem over the logical topology, and thus a critical issue is to find a path with enough residue bandwidth to carry an incoming client call. A straightforward approach for path computation is to compute a shortest path in the logical topology utilizing the bandwidth usage information on each logical link, called *logical layer path computation* (LogPac). Due to the cross layer property of traffic grooming, we have proposed to consider both logical and physical layer resource usage information in the path computation, called *cross layer path computation* (CrosPac). We have examined various cost functions for both LogPac and CrosPac approaches, and evaluated their performance through simulations, under evenly or unevenly distributed call data rates, and with or without wavelength conversion in the optical network. The numerical results demonstrate that the CrosPac approach outperforms the LogPac approach, due to its consideration of resource usage in both layers. In the future, we will continue to examine optical network design using cross layer information. We plan to study the effect of dynamic traffic grooming on lightpath routing in the physical layer, and consider cross layer information in lightpath route computation.

REFERENCES

[1] K. Zhu and B. Mukherjee, "On-line approaches for provisioning connections of different bandwidth granularities in WDM mesh networks," in *OFC Technical Digest*, OFC 02, pp. 549 – 551

[2] H. Zhu, H. Zang, K. Zhu and B. Mukherje, "A novel generic graph model for traffic grooming in heterogeneous WDM mesh networks," *IEEE/ACM Trans. Networking*, vol. 11(2), pp. 285 – 99, Apr. 2003.

[3] C. Xin et al., "An integrated lightpath provisioning approach in mesh optical networks," in *OFC Technical Digest*, OFC 2002, pp. 547 – 549

[4] C. Xin, C. Qiao and S. Dixit, "Analysis of single-hop traffic grooming in mesh WDM optical networks," in *Proc. of SPIE*, vol. 5285, Opticomm 2003, Oct. 2003, Dallas, TX, USA, pp. 91– 101.

[5] C. Xin and C. Qiao, "Performance analysis of multi-hop traffic grooming in mesh WDM optical networks," in *Proc. IEEE IC^{3}N*, 2003, pp. 237 – 242.

[6] C. Xin, C. Qiao and S. Dixit, "Traffic grooming in the mesh WDM optical network – performance analysis," *IEEE J. Select. Area. Commun.*, to be published.

[7] C. Xin, "Blocking analysis of dynamic traffic grooming on mesh WDM optical networks," submitted to *IEEE Trans. Networking*.

[8] J. Yen, "Finding the K shortest loopless paths in a network," *Management Science*, vol 17(11), pp. 712 – 716, Jul. 1971.