University of Nebraska - Lincoln
DigitalCommons@University of Nebraska - Lincoln

CSE Conference and Workshop Papers | Computer Science and Engineering, Department of

1-1-2006

Dynamic Lightpath Scheduling in Next-Generation WDM Optical Networks

Lu Shen
University of Nebraska - Lincoln, lshen@cse.unl.edu

Ajay Todimala
University of Nebraska - Lincoln

Byrav Ramamurthy
University of Nebraska - Lincoln, bramamurthy2@unl.edu

Xi Yang
University of Southern California, xyang@isi.edu

Follow this and additional works at: http://digitalcommons.unl.edu/cseconfwork

Part of the Computer Sciences Commons

Shen, Lu; Todimala, Ajay; Ramamurthy, Byrav; and Yang, Xi, "Dynamic Lightpath Scheduling in Next-Generation WDM Optical Networks" (2006). *CSE Conference and Workshop Papers.* Paper 91.
http://digitalcommons.unl.edu/cseconfwork/91

This Article is brought to you for free and open access by the Computer Science and Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in CSE Conference and Workshop Papers by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.

Proceedings INFOCOM 2006. 25th IEEE International Conference on Computer Communications.
Digital Object Identifier: 10.1109/INFOCOM.2006.31
Publication Year: 2006 , Page(s): 1 - 5

Dynamic Lightpath Scheduling in Next-Generation WDM Optical Networks

Lu Shen, Ajay Todimala and Byrav Ramamurthy
Department of Computer Science and Engineering
University of Nebraska-Lincoln
Lincoln, NE 68588-0115, U.S.A.
Email: {lshen, ajayt, byrav}@cse.unl.edu

Xi Yang
Information Sciences Institute East
University of Southern California
3811 Faifax Drive, Suite 200, Arlington, VA 22203, U.S.A.
Email: xyang@isi.edu

***Abstract*— Lightpath scheduling is an important capability in next-generation wavelength-division multiplexing (WDM) optical networks to reserve resources in advance for a specified time period while provisioning end-to-end lightpaths. In this study, we propose an approach to support dynamic lightpath scheduling in such networks. To minimize blocking probability in a network that accommodates dynamic scheduled lightpath demands (D-SLDs), resource allocation should be optimized in a dynamic manner. However, for the network users who desire deterministic services, resources must be reserved in advance and guaranteed for future use. These two objectives may be mutually incompatible. Therefore, we propose a two-phase dynamic lightpath scheduling approach to tackle this issue. The first phase is the deterministic lightpath scheduling phase. When a lightpath request arrives, the network control plane schedules a path with guaranteed resources so that the user can get a quick response with the deterministic lightpath schedule. The second phase is the lightpath re-optimization phase, in which the network control plane re-provisions some already scheduled lightpaths. Experimental results show that our proposed two-phase dynamic lightpath scheduling approach can greatly reduce WDM network blocking.**

I. Introduction

The next-generation wavelength-division multiplexing (WDM) optical networks will be a key enabler for many high-end applications, including those using Grid technologies, by provisioning end-to-end lightpaths in an on-demand manner. Unlike previous optical bandwidth consumers, end users largely control these new applications and thus the bandwidth demands come directly from the end users' requests. Such demands are usually dynamic, which means that the network operation based on the assumption of static or predictable demands will be considerably inefficient. In addition, many end users need guaranteed lightpath connections during a specified period of time in future. They usually prefer to make advance reservations for end-to-end lightpaths with predefined service durations where the starting time of the lightpath demand can be days to weeks in the future. Such an advance reservation of a lightpath is called scheduling of a lightpath and correspondingly the lightpath itself is termed as a scheduled lightpath demand (SLD) [1]. Many SLDs arrive in a dynamic manner. We distinguish between these dynamic scheduled lightpath demands (D-SLDs) as opposed to the concept of static scheduled lightpath demands (S-SLDs), where the whole set of lightpath demands is available before any actual provisioning happens in the network. Therefore, the time schedule of every S-SLD is known in advance for the network control plane. We study the problem of bandwidth allocation for D-SLDs in this paper. For the purpose of scheduling, we assume that the network time is slotted. The duration of a scheduled lightpath is measured in number of time slots. Each time slot has equal length.

In practical network operation, many end users require deterministic services. By a deterministic service, we mean that after submitting a request for a lightpath, a user expects a deterministic answer to whether the request can be accommodated and if so, the precise schedule information for the request. Deterministic service provisioning in the presence of D-SLDs may be inefficient in terms of network resource utilization. Unlike S-SLDs, arrivals of D-SLDs cannot be precisely predicted and thus resource allocation for D-SLDs is difficult to be optimized as a whole. Although the deterministic answer returned for a D-SLD can be optimal at the current time, the resource allocation in the network may still become sub-optimal with the arrivals of future D-SLDs. Interestingly, before a D-SLD is physically provisioned in the network, any adjustment carried out on the resources reserved for this D-SLD, e.g. rerouting and reassignment of wavelength, will not disrupt its service. Therefore, we have the opportunity to perform re-optimization for all D-SLDs scheduled to be set up in the future. In this paper, we propose that resources reserved for scheduled lightpaths be re-optimized before they are physically provisioned so that better network performance can be achieved.

We use two example applications to characterize two types of D-SLDs. The first application is to schedule a real-time, collaborative scientific experiment. A lightpath is scheduled between the data gathering and data processing facilities at a fixed start time and a fixed end time (i.e., for a fixed duration), e.g. [9:00 a.m., 10:00 a.m.] on Monday. We denote such a demand to be of the time-fixed type. In the second example, a financial institution demands a scheduled lightpath on a weekly basis to backup the huge amount of data to a data repository in a different city. The transfer takes up to one hour. The demand is specified with a loose starting time window, say [1:00 a.m., 5:00 a.m.] on Saturdays, during which any

lightpath starting time is acceptable. We denote such a demand to be of the time-window type. To acknowledge any time-window request, a deterministic schedule should be returned so that the user, application and protocol know the exact start time for transferring data. In another type of applications, a deadline is specified. For example, an ftp session for database recovery must be finished by 7:00am on Mondays. A deadline type demand can be translated into a time-window type demand by specifying the start time of the window as [Now, Deadline - Estimated duration].

In this paper, we address both time-fixed and time-window D-SLDs. We propose a two-phase dynamic lightpath scheduling approach. In the first phase, we employ a deterministic lightpath scheduling algorithm to compute the schedule for time-fixed and time-window D-SLDs. This is called the deterministic lightpath scheduling phase, in which each request is assigned a route and wavelength with a fixed time schedule or is denied. In the second phase, a re-optimization procedure is carried out to re-provision those lightpaths that have been scheduled, but are not in service yet (i.e., they have not been physically provisioned). This is called the lightpath re-optimization phase. When a blocking occurs in the deterministic lightpath scheduling phase, the re-optimization procedure can be invoked to eliminate the blocking by changing the resource allocation for some scheduled lightpaths. Note that any re-provisioned lightpath must be guaranteed a set of resources (wavelengths along the path) and its time schedule must not be changed. After a lightpath is physically set up (provisioned), none of the resources allocated to it should be changed until it is torn down (e.g. rerouting of existing lightpaths in the network is not permitted).

Previous work in [2] studied the static lightpath scheduling problem. Both integer linear programming (ILP) and heuristic approaches were used to accommodate the static scheduled lightpath demands (S-SLD). In [1], the authors proposed a traffic model for advance reservation. Several simple routing and wavelength assignment (RWA) algorithms were proposed to schedule S-SLDs. In [3], the authors presented a tabu-search meta heuristic for routing time-fixed S-SLDs with the objective of minimizing the number of wavelength-links. More recently, the authors in [4] studied the static lightpath scheduling problem. They considered both time-fixed and time-window requests. They presented an algorithm for placing the demands in the specified time window to minimize the overlapping of time among a set of demands. In [5], the authors proposed a dynamic traffic model for advance reservation. Several simple RWA algorithms were proposed to schedule the D-SLDs.

The remaining portion of this paper is organized as follows. Section II presents the network model and problem statement. Section III describes our deterministic lightpath scheduling algorithm for both time-fixed and time-window D-SLDs in the deterministic lightpath scheduling phase. Section III describes the re-optimization procedure for the lightpath re-optimization phase. Section IV presents our experimental results. Section V concludes this paper.

II. Network Model and Problem Statement

A. Network Model

In this paper, we consider WDM wavelength-routed mesh networks. Such a network consists of a set of reconfigurable optical cross-connects (OXCs) interconnected by optical fiber links. Each link has two fibers in opposite directions, while each fiber has a fixed number of wavelengths. We assume that the OXCs have no wavelength conversion capability; thus all lightpaths are subject to the wavelength continuity constraint.

We make the following operational assumptions and use these notations in our network model.

- Network time is slotted. A time slot is the minimum time unit in the network, each having an equal, fixed length. We denote the time slots starting from a given time 0 by using a sequence of $\{T_0, T_1, \ldots\}$ indices.
- The duration of any D-SLD is an integral multiple of a time slot. A lightpath can only be scheduled at the beginning of a time slot. Note that in the rest of this paper, we use the term lightpath interchangeably with D-SLD.
- Each D-SLD occupies the whole capacity of a single wavelength. The ith D-SLD in the network is denoted by $(s^i, d^i, t^i, \tau^i, l^i)$, where$s^i$ is the source node, d^i is the destination node, t^i and τ^i are the starting time slot and duration (in time slots) for scheduled lightpath, and l^i is the maximum path length in kilometers. For a time-fixed D-SLD, t^i is a fixed value. For time-window D-SLD, t^i represents a range of contiguous time slots.
- For any wavelength on any link, its availability in one time slot is independent of its availability in other time slots. However, the wavelength assigned for a specific lightpath must be the same when that lightpath spans multiple time slots.

A lightpath in the network must be in one of the following two states:

- *scheduled*: In this state, the lightpath has been scheduled and its starting time cannot be changed; however, its route and wavelength may be changed.
- *in-service*: In this state, the lightpath has been physically provisioned and its starting time, routing and wavelength assignment cannot be changed.

As discussed in Section I, once a lightpath is scheduled, its starting time cannot be changed because the requestor has been acknowledged. However, a lightpath can be rerouted for the re-optimization purpose anytime before its starting time. To avoid interrupting ongoing services, the lightpath cannot be altered when it is in service (after the scheduled starting time). Each lightpath is also subject to a maximum path length constraint, which is usually considered in practice to address physical impairments and end-to-end delay.

B. Problem Statement

We study the dynamic lightpath scheduling problem that consists of two phases, namely deterministic lightpath scheduling for phase I and lightpath re-optimization for phase II.

When a D-SLD request arrives, the system enters phase I, where the system quickly checks if the D-SLD can be scheduled with a route and wavelength being reserved at its starting time and for the requested duration. Phase II re-provisions those scheduled lightpaths to improve network performance. In theory, we can perform re-optimization after the arrival of each D-SLD, whose overhead, however, would be overwhelming. To reduce the overhead, we conduct lightpath re-optimization only if a network blocking occurs.

In the following part of this section, we present the problem statement for the deterministic lightpath scheduling and lightpath re-optimization phases, respectively. We first define some common notations. By using a graph $G(V, E)$, we represent an optical network topology, where V is a set of nodes and E is a set of links in the network. By $U[t]$ we denote the number of used wavelength-links during time slot t, and by $U_e[t]$ we denote the number of used wavelengths during time slot t on the link $e \in E$. Therefore, $U[t] = \sum_{e \in E} U_e[t]$.

1) Phase I - Deterministic Lightpath Scheduling: Given a D-SLD $(s^i, d^i, t^i, \tau^i, l^i)$ and a network topology $G(V, E)$ with wavelength availability information, the problem is to provision the lightpath with two possible objectives as follows:

- Objective 1 (MWL): Minimizing the number of used Wavelength-Links between time slot t^i and $t^i + \tau^i - 1$, i.e., minimizing $\sum_{t^i \le t \le t^i + \tau^i - 1} U[t]$.
- Objective 2 (LB): Load-Balancing, i.e., minimizing $\max_{t^i \le t \le t^i + \tau^i - 1, e \in V} U_e[t]$. In other words, LB strives to minimize the maximum number of used wavelengths among all the links in the network.

2) Phase II - Lightpath Re-optimization: Given a D-SLD that is blocked in phase I and a set of D-SLDs that have been scheduled, but have not been in service, the problem is to determine how to re-provision those lightpaths. The objective is to provision the blocked D-SLD without blocking any of the already scheduled D-SLDs and without changing their schedules.

III. Deterministic Lightpath Scheduling Algorithm

We develop a deterministic lightpath scheduling algorithm for solving the problem in phase I. Given a D-SLD, we need to compute a path, a wavelength, and a starting time slot for the request. The algorithm should find a solution quickly so that the requestor will be provided an answer in a short time.

We use fixed-alternate routing as the path selection scheme. A set of pre-computed *k*-shortest paths between the source and destination are used as the candidate routes for the demand. They are denoted by $\{P_1, P_2, \ldots, P_k\}$. These *k*-shortest paths should satisfy the path length constraint on the demand.

We develop a wavelength assignment scheme called *Slotted First-Fit* (*SFF*) in our deterministic lightpath scheduling algorithm. A wavelength is said to be available for a set of contiguous time slots if and only if it is available in each of these time slots. Given the starting time slot, the duration and a pre-computed routing path for a D-SLD, SFF picks up the first common wavelength that is available on every link of the path in all the time slots from its starting time slot through its ending time slot.

The complete design of our deterministic lightpath scheduling algorithm is described in Algorithm 1. The inputs are a topology graph $G(V, E)$ and a D-SLD represented by (s, d, t, τ, l), where the value of t is an integer in a range $[t_{lower}, t_{upper}]$. For a time-fixed D-SLD, $t_{lower} = t_{upper}$. For a time-window D-SLD, $t_{lower} < t_{upper}$. The algorithm loops through each possible starting time slots between t_{lower} and t_{upper} (in step 3) and each candidate path of the *k*-shortest paths (in step 4). Then, it uses *SFF* to find a wavelength (in step 5) and inserts the solution, i.e., the combination of a path, a wavelength and a starting time, into a solution list L (in step 7). If the solution list is empty after looping through all the possible starting time slots and candidate paths, the demand is blocked (in step 12). Otherwise, it schedules the demand by using the solution that has the minimum objective value (in step 14). The objective values are calculated using either the MWL objective or the LB objective, whose expressions are shown below:

MWL: the number of links on the path.
LB: $\max_{t \le x \le t+\tau-1, e \in P} u_e[x]$.

Algorithm 1 Deterministic Lightpath Scheduling Algorithm

Input: A D-SLD request (s, d, t, τ, l) and $G(V, E)$
Output: A schedule of the demand or refusal

1: empty the solution list L
2: compute the *k*-shortest path set $\{P_1, P_2, \ldots, P_k\}$ between s and d with path length no greater than l
3: **for** $t = t_{lower}$ to t_{upper} **do**
4: **for** $P \in \{P_1, P_2, \ldots, P_k\}$ **do**
5: use SFF as wavelength assignment scheme to find an available wavelength w on the path P
6: **if** a wavelength is found **then**
7: insert (P, w, t) as a solution into the solution list L
8: **end if**
9: **end for**
10: **end for**
11: **if** the solution list L is empty **then**
12: the demand is blocked
13: **else**
14: compute the objective value for each solution in the list L and select the one with minimized objective value
15: **end if**

A. Re-optimization at Blocking

We develop a heuristic algorithm for performing the re-optimization at blocking scenario. Upon occurrence of a blocking in phase I, the algorithm re-provisions the scheduled lightpaths in the system with the hope of accommodating the blocked D-SLD. The scheduled lightpaths have been assigned paths and wavelengths, before the blocked lightpath request arrives, which may not be optimal at the time when the blocked

lightpath arrives. By running the re-optimization at blocking, we have a second chance for finding resources to schedule the blocked lightpath.

The algorithm operates on the time-overlapped lightpath set that overlaps with the blocked D-SLD. Except for the blocked lightpath, all the other lightpaths in the set are in the *scheduled* state. Algorithm 2 presents the algorithm for re-optimization at blocking for time-fixed D-SLD. It is the building block of the algorithm for time-window D-SLD, which will be introduced later in this section. The basic idea of the re-optimization algorithm is to re-provision the scheduled lightpaths according to an order that may help in accommodating more lightpaths.

The input to the algorithm is the blocked lightpath and all the scheduled (not yet in service) lightpaths in the network. In the first step, the algorithm collects the set of scheduled lightpaths that overlap with the blocked lightpath. It then releases the wavelengths reserved for these scheduled lightpaths. Step 3 (see Algorithm 2) sorts the lightpaths according to an order, which we will discuss in detail later in the section. Then, each lightpath of these sorted lightpaths is re-provisioned using the deterministic lightpath scheduling algorithm presented in Algorithm 1. If all the lightpaths are successfully provisioned, the algorithm returns *success*; otherwise, returns *failure*. Note that in step 5 (in Algorithm 2), the starting time of each lightpath remains unchanged when the deterministic lightpath scheduling algorithm is called.

Algorithm 2 Algorithm of re-optimization at blocking for time-fixed D-SLD

Input: the blocked lightpath and all scheduled lightpaths
Output: *success* or *failure*

1: collect the set of time-overlapped scheduled lightpaths that overlaps with the blocked lightpath and represent it as C.
2: release all the scheduled lightpaths in C.
3: sort lightpaths in the set C (based on the ordering schemes defined in this section)
4: **for** each lightpath $c \in C$ according to the order sorted in step 3 **do**
5: re-provision the lightpath c using the deterministic lightpath scheduling algorithm (Algorithm 1) with the load-balancing objective.
6: **if** step 5 fails **then**
7: restore the re-provisioned lightpaths
8: return *failure*
9: **end if**
10: **end for**
11: return *success*

Algorithm 3 presents the algorithm for re-optimization at blocking for a time-window D-SLD. It is based on the re-optimization algorithm for time-fixed D-SLD (Algorithm 2). If the blocked D-SLD is a time-window D-SLD, we simply loop through all the possible starting times for the blocked D-SLD using the re-optimization algorithm for the time-fixed D-SLD. If any iteration in the loop returns success, the algorithm returns success. Otherwise, it returns failure.

The ordering scheme used in step 5 of the re-optimization algorithm in Algorithm 2 is crucial. On the one hand, the scheduled demands in the network may be provisioned out of order on the time line; for example, the demand with an earlier starting time may be scheduled later when it arrives. Some blocking may be eliminated if we re-provision them according to order of their actual starting times. On the other hand, based on the work in [6], the chances of provisioning all lightpaths become higher if the network resources are first assigned to those lightpaths that are most difficult to be provisioned, e.g. the lightpath with the maximum number of hops on the shortest path between its source and destination. However, the introduction of time slots complicates the ordering scheme since we have an extra dimension, the time line.

Algorithm 3 Algorithm for re-optimization at blocking for time-window D-SLD

Input: The blocked lightpath with time-window $[t_{lower}, t_{upper}]$ and all scheduled lightpaths
Output: *success* or *failure*

1: **for** $t = t_{lower}$ to $t = t_{upper}$ **do**
2: set the starting time of the blocked lightpath to t and call the re-optimization at blocking for time-fixed D-SLD (Algorithm 2) using the blocked lightpath as input.
3: **if** step 2 returns *success* **then**
4: return success
5: **end if**
6: **end for**
7: return *failure*

In our approach, we use a combination of multiple keys for ordering. Given a set of scheduled lightpaths, we first sort them according to the non-decreasing order of their starting time. If two lightpaths have the same starting times, we break the tie by sorting them according to the non-decreasing order of the number of links on their minimum-hop paths. If they are still tied, we break the tie by sorting them according to the non-decreasing order of their service durations. Therefore, the lightpaths are sorted according to the rules of the earliest starting time first, maximum hop first, and longest duration first with decreasing priorities.

IV. Numerical Results

In this section, we present numerical results of our proposed approach for on demand lightpath reservation We conduct simulation experiments on a 24-node, 84 link network. The duration of each time slot is set to 15 minutes. In each simulation we simulate 100,000 D-SLDs consisting of a mix ratio of 7:3 of time-fixed and time-window demands. The window size of the time-window demands is uniformly distributed in the range [4, 48] time slots. The duration of D-SLDs is measured in number of time slots and is a weighted non-uniform distribution in the range [1, 50]. The distribution of the inter-arrival time and the starting times of D-SLDs are Poisson processes. We use two metrics for comparison, blocking probability (BP) and service blocking probability

(SBP). The service blocking probability is measured as the ratio of the sum of the durations of blocked D-SLDs to the sum of the durations of all the D-SLDs. Because D-SLDs may have different durations, SBP provides a fair measurement on the network performance. We conduct our experiments in the cases with 8, 16, 32, and 64 wavelengths. The path length constraint is set to 600km, which is considered as a typical reach distance of all-optical signals.

Figure 1 through 4 plot BP and SBP of lightpath scheduling approaches with and without re-optimization in those test cases with 8, 16, 32, 64 wavelengths, respectively. As shown in the figure, the approach with re-optimization reduces both the blocking probability and the service blocking probability significantly. Table I presents the average improvement under different experimental settings. On average, compared to the approach without re-optimization, our approach eliminates 49.8%, 58.9%, 58.8%, and 54.7% blockings in phase II for different cases. The performance gain in terms of service blocking probability remains at the same level.

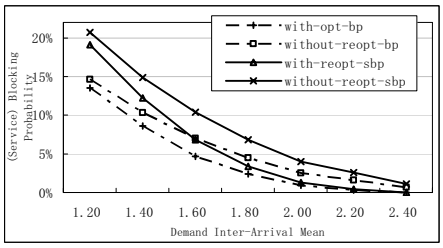

Fig. 1. Results with and without re-optimization at blocking in the network with 8 wavelengths

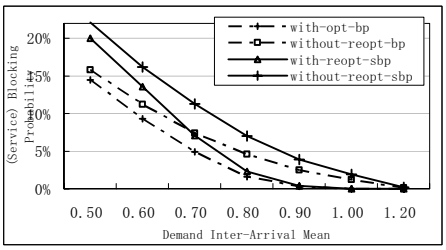

Fig. 2. Results with and without re-optimization at blocking in the network with 16 wavelengths

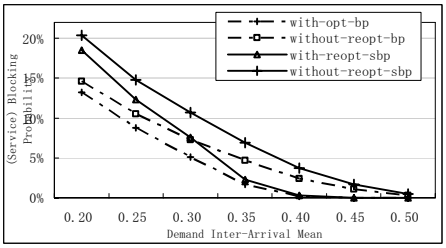

Fig. 3. Results with and without re-optimization at blocking in the network with 32 wavelengths

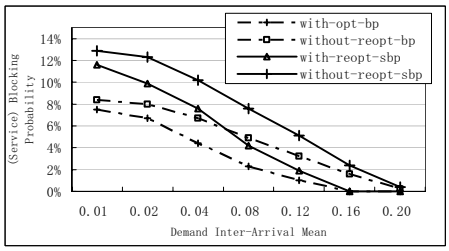

Fig. 4. Results with and without re-optimization at blocking in the network with 64 wavelengths

TABLE I

PERFORMANCE IMPROVEMENT BY RE-OPTIMIZATION.

Number of Wavelengths	Improvement of Re-Optimization on BP	Improvement of Re-Optimization on SBP	Average Number of Lightpaths per Run	Average Running Time (seconds)
8	49.8%	51.8%	263	0.06
16	58.9%	59.9%	640	0.37
32	58.8%	59.1%	1429	1.10
64	54.7%	51.8%	3283	3.89

V. CONCLUSION

In this study we addressed dynamic lightpath scheduling problem. We considered both time-fixed and time-window D-SLDs. Compared to predefined static demands, the unpredictable dynamic demands have a very limited potential for optimization. We proposed an efficient two-phase lightpath scheduling scheme to reduce about half of the blockings in the network with advance reservation. This is a great performance improvement for those network users who require both dynamic and deterministic scheduled lightpath services. In this study, we did not consider the lightpath protection required for D-SLDs. However, such application usually require strict backup for the working lightpaths. Future work should focus on the lightpath scheduling problem under protection requirements.

REFERENCES

[1] J. Kuri et al., "Routing and wavelength assignment of scheduled lightpath demands," *Journal of Selected Areas in Communications (JSAC),* vol. 21, no. 8, pp. 1231-1240, October 2003.

[2] A. Maach and A. Hafid, "Resources reservation in advance in slotted optical networks," *Third International Conference on Optical Communications and Networks (ICOCN 2004),* Hong Kong, China, November 2004.

[3] J. Kuri, N. Puech, M. Gagnaire, and E. Dotaro, "Routing foreseeable lightpath demands using a tabu search metaheuristic," *IEEE GLOBECOM 2002,* Taipei, Taiwan, November 2002.

[4] B. Wang, T. Li, X. Lou, Y. Fan, and C. Xin, "On service provisioning under a scheduled traffic model in reconfigurable WDM optical networks," *IEEE BroadNets '05,* Boston, MA, USA, October 2005.

[5] S. Figueira, N. Kaushik and S. Naiksatam, "Advance reservation of lightpaths in optical-network based Grids," *IEEE BroadNets/GridNets 2004,* San Jose, CA, USA, October 2004.

[6] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: an approach to high bandwidth optical WAN's," *IEEE Transactions on Communications,* vol. 40, pp. 1171-1182, July 1992.