In Proceedings of The Fifth IASTED International Conference on Optical Communications Systems and Networks, pages 58-63, Banff, Alberta, Canada, July 2005.

WAVELENGTH ASSIGNMENT FOR LIGHT-TREE PROTECTION IN WDM OPTICAL NETWORKS

Lan Kong, Lin Li, Jitender S. Deogun and Stephen D. Scott
Dept. of Computer Science and Engineering, University of Nebraska,
Lincoln, NE 68588-0115, U.S.A
email:{kong,lili,deogun,sscott}@cse.unl.edu

Abstract

In this paper, we address the wavelength assignment problem for light-tree protection under different traffic models. For static traffic, we formulate the problem mathematically and propose two new heuristics (tabu search and iterated hill-climber) to solve it. For dynamic traffic, in contrast to previous work, we analyze the traffic characteristics and propose a Low Cost First Serve (LCFS) strategy to maximize the gain and throughput of network systems. Numerical results show our heuristics can achieve good performance in terms of session blocking probability.

Keywords: wavelength assignment, light-tree protection, static traffic, dynamic traffic

1 Introduction

In all-optical networks, data are transmitted through lightpaths from source to destination in the optical domain without undergoing opto-electronic conversion. A *light-tree* supports multicast applications such as HDTV and video conference. Moreover, by incorporating the multicasting capability with light-tree at the routing nodes, we can increase the logical connectivity of the network and may further reduce the average packet hop distance and the number of transceivers [1]. Since high-speed networks carry huge volumes of traffic, even a brief interruption may lead to severe consequences. The protection problem for a light-tree is more challenging than that for a lightpath because a network failure will impact several destinations of the session.

Generally, the light-tree protection problem consists of two separate steps: light-tree routing and wavelength assignment. Previous work, such as Singhal & Mukherjee [2] proposed to build two kinds of trees, *primary tree* and *backup tree*, for light-tree protection. Zhou & Subramaniam [3] gave a overview of the survivability problem and reviewed different protection schemes. Zang et al. [4] compared different light-path routing and wavelength assignment algorithms under static and dynamic traffic. Although there exist many heuristics for light-path routing and wavelength assignment, due to the different characteristics of light-tree and light-path, many of them cannot be applied to light-tree protection directly. In this paper, we focus on the wavelength assignment problem for light-tree protection and analyze our heuristics with both static and dynamic traffic models. Given a time series of primary and backup trees (e.g., generated by algorithms such as Kong [5]), we heuristically optimize their wavelength assignment. Considering the cost of wavelength converters and the state of related technology, throughout the paper, we assume the light-tree based multicast sessions obey the *wavelength-continuity constraint*.

2 Problem Definition and Protection Schemes

In the following, we give a formal definition of wavelength assignment problem for light-tree protection.

Definition 1. *Wavelength Assignment for Light-Tree Protection in WDM Optical Networks (WA-LTP).*
Given a 4-tuple $< G, W, Q, T >$, where $G = (N, L)$ is a directed graph representing the network topology (N is the set of nodes in the network and L is the set of directed links); W is the set of wavelengths on a fiber link; $T = \{(p_i, b_i)\}$ is the sequence of the trees, one pair per session (p_i represents the primary tree and b_i represents its backup tree); Q is a set of quadruples $\{(s_i, D_i, t_i, h_i)\}$ (for the ith session, s_i represents the source node, D_i is the set of destination nodes, t_i represents the starting time and h_i represents its duration). The goal is to maximize the total number of sessions established in the network over all time steps by assigning appropriate wavelengths to the primary and backup trees for each session[1].

The general problem of WA-LTP is NP-hard since a light-tree is the generalization of a lightpath and the wavelength assignment problem for lightpath is NP-hard [6]. we can categorize light-tree protection into two types: *dedicated light-tree protection (DLTP)* and *shared light-tree protection (SLTP)*. In DLTP, we allocate an available wavelength to the backup tree of each established session. The resources are reserved during the life-time of the session, which is called a $1+1$ scheme. The signals are transmitted concurrently in the primary tree and backup tree. In SLTP, two backup trees share the same wavelength channel on a link if the corresponding primary trees are edge-disjoint, which is called shared light-tree protection and it is a $1:N$ scheme. In this case, the signals are transmitted

---

[1] Note that establishing a session means both the primary and backup trees are assigned successfully.

only in the primary trees if no failure occurs. Two primary trees will not fail at the same time since they do not share a common edge. If there is a failure, the end nodes of the affected primary tree will use a special signaling protocol (e.g.,*automatic protection switching*) to inform the source node to switch over to the backup tree for transmission. Thus, SLTP utilizes the capacity more efficiently, while still achieving 100 percent recovery from failures.

3 Wavelength Assignment Under Static Traffic

In the static traffic model, all trees are to be assigned are available at the same time and all trees are persistent, i.e. they will not end. Thus we have $t_i = 1$ and $h_i = \infty$ for each session. Correspondingly, we replace a sequence of the sessions with a set of sessions. We begin our presentation with an ILP formulation and then give two heuristics based on Tabu Search and iterated Hill-Climber approaches, which have been observed to be better than other greedy approaches in avoiding local optima.

3.1 Integer Linear Program Formulation

The WA-LTP problem can be defined as an integer linear programming problem (ILP). Table 1 lists variables and constants used in the formulation.

Table 1. Constants and variables used in the ILP formulation.

Variable	Description
N	Set of nodes in the network
L	Set of links in the network
$G = (N, L)$	Directed graph representing network topology
W	Set of wavelengths.
$\psi = (s, D)$	Multicast session ψ
$XP_{i,\lambda}$	$= 1$ if the primary tree of the session ψ_i is established with the wavelength λ; $= 0$ otherwise.
$XB_{i,\lambda}$	$= 1$ if the backup tree of the session ψ_i is established with the wavelength λ; $= 0$ otherwise.
EP_i	$= 1$ if the primary tree of the session ψ_i is established; $= 0$ otherwise.
EB_i	$= 1$ if the backup tree of the session ψ_i is established; $= 0$ otherwise.
e_i	$= 1$ if the session ψ_i is established; $= 0$ otherwise.
P_l	Set of sessions that have link l in their primary trees.
B_l	Set of sessions that have link l in their backup trees.
PM_{ψ_j,ψ_k}	$= 1$ if the primary tree of the session ψ_j shares an edge with the primary tree of the session ψ_k; $= 0$ otherwise.

I. Objective function: Maximizing the total number of multicast sessions established.

$$Maximize: \ \ Z = \sum_{\psi_i \in Q} e_i$$

subject to the constraints that defined below.

II. Constraints:

Link constraint (DLTP) — Two trees with a common link cannot share a wavelength.

$$\sum_{\psi_i \in P_l} XP_{i,\lambda} + \sum_{\psi_i \in B_l} XB_{i,\lambda} \ \leq \ 1 \quad l \in L, \ \lambda \in W \tag{1}$$

Link constraint (SLTP) — In Constraint 2, if a primary tree shares a common link with another primary tree or a backup tree, the same wavelength will not be assigned to both trees. In Constraint 3, if two backup trees share a common link and their corresponding primary trees share a common edge, then they will not reserve the same wavelength[2].

$$\sum_{\psi_i \in P_l} XP_{i,\lambda} + XB_{j,\lambda} \ \leq \ 1 \tag{2}$$
$$\psi_j \in B_l, \ \ l \in L, \ \ \lambda \in W$$

$$\sum_{\psi_i \in P_l} XP_{i,\lambda} + XB_{j,\lambda} + XB_{k,\lambda} \ \leq \ 1 \tag{3}$$
$$\psi_j, \psi_k \in B_l, \ PM_{\psi_j,\psi_k} = 1, \ \ l \in L, \ \ \lambda \in W$$

Tree constraints — A primary tree or a backup tree is established by using one and only one available wavelength.

$$EP_i = \sum_{\lambda \in W} XP_{i,\lambda} \quad \psi_i \in Q \tag{4}$$

$$EB_i = \sum_{\lambda \in W} XB_{i,\lambda} \quad \psi_i \in Q \tag{5}$$

Session constraint — A session is established if and only if both primary and backup trees for this session are assigned.

$$2 \cdot e_i = EP_i + EB_i \quad \psi_i \in Q \tag{6}$$

III. Bounds: All $XP_{i,\lambda}, XB_{i,\lambda}, EP_i, EB_i, e_i, \in \{0, 1\}$.

We can obtain optimal solutions through ILP formulations. However, they are not practical for large-size networks with tens of nodes. Efficient heuristics must be developed.

3.2 Heuristic Algorithms

One of the simplest schemes is *Least-Cost Session First (LCSF)* in which we first sort the sessions in increasing order of $|T_i|$, where $|T_i|$ denotes the cost of the primary tree p_i and the backup tree b_i for session ψ_i. The cost of a light-tree is the sum of the costs along all of its fiber links. Then the sessions are established one after another by assigning a pair of appropriate wavelengths for its primary tree and backup tree. The procedure follows a first-fit approach. When searching for an available wavelength, a lower numbered wavelength is considered before a higher

---

[2] Recall that in a directed graph, each edge contains two links in opposite directions.

numbered wavelength. The first available wavelength is then selected. In this way, we always try to pack all of the in-use wavelengths toward the low end of the wavelength space. It should be noted that the primary and backup trees for a specific session don't have to be assigned the same wavelength.

3.2.1 Tabu Search and Iterated Hill-Climber

To further improve the performance of LCSF approach, we propose a Tabu search heuristic algorithm, which is referred to as TS-LTP. Our solution space is defined as the set R of successfully established sessions. We use the solution obtained by the LCSF algorithm as the initial solution r_0 of TS-LTP. Given a particular solution $r \in R$, the neighborhood of such a solution is denoted by $N(r)$. A solution in the neighborhood is obtained in two steps. Let X_j be the set of sessions whose primary trees are assigned with λ_j. Our first step is to unassign all primary and backup trees of X_j, placing them in a queue which is the set of previously unassigned sessions sorted by cost and followed by X_j sorted by cost. Then we use the first-fit approach to assign wavelengths on the sessions in the queue in order. The final solution from first-fit is $N_j(r)$. $N(r)$ is the union[3] of $N_j(r)$ over all j. Among the neighbors, the solution that establishes most sessions is chosen to be the next current solution, and the current move will be forbidden for next t iterations. An exception exists when a tabu move generates a solution which establishes more sessions than the best solution (r^*) found so far. In this circumstance, this solution is taken as the next point. This feature is called *aspiration criterion*. The terminating condition we adopt in TS-LTP is a k ($k = 100$ in our algorithm) iterations in which the best solution found cannot be improved.

The second heuristic we propose is an iterated Hill-Climber, denoted as IHC-LTP, where we first generate a current solution by randomly assigning sessions with free wavelengths until no more sessions can be assigned. Then, a set of neighbors of current solutions are generated by using the same method as in TS-LTP. The new current solution which establishes the most number of sessions is selected from the solutions in this set that are better than the current solution. The procedure stops if there are no better neighbors. After iterating this procedure a k ($k = 20$ in our algorithm) times, the best overall solution is returned. With this approach, we try to escape local optima by starting a new search from a random location.

4 Wavelength Assignment Under Dynamic Traffic

4.1 Low Cost First Serve Heuristic

Under dynamic traffic, the requests come one by one in real time. Correspondingly, each sessions will be established base on its arrival time t_i and will hold its resources until the holding time h_i ends. So we cannot use the preplanned methods (e.g., ILP formulation) for traffic assignment, and the objectives are: (1) minimize the session blocking probability. (2) assign the wavelengths to the sessions as quickly as possible. Fast on-line heuristics have to be developed to meet the objectives. We use *Erlang* to measure the traffic rate which equals the total number of sessions per holding time (e.g., if the average request number is η per unit time and the average duration of each request is τ unit time, the Erlang is $\eta \times \tau$). We evaluate the performance of our heuristics by the blocking probability of sessions which is calculated by the ratio of the number of sessions blocked to the number of total sessions.

In developing our approach, we notice that the objective is to minimize the blocking probability, and establishing a session with small tree cost and short holding time brings the same gain as establishing a session with large tree cost and long holding time. Thus, it may be beneficial if we selectively drop some sessions with high cost even if there are wavelengths available. We define the session cost as $c_i = |L_i| * h_i$, where $|L_i|$ is the cost of the trees of session ψ_i (e.g. the total number of links), and and h_i is the holding time of session ψ_i. Our algorithm uses c_i to decide whether to establish session ψ_i. I.e. if c_i is large, we may not establish session ψ_i, expecting that more upcoming sessions with smaller costs can be assigned later. With some abuse of the terminology, we call our approach *Low Cost First Serve (LCFS)*. This is in contrast to *First Come First Serve (FCFS)*, which establishes sessions as they arrive if sufficient bandwidth is available.

More generally, since the objective of wavelength assignment is to maximize the gain of the system and each session may bring different gains, it is more reasonable to assign each session according to its cost relative to its gain (e.g. financial reward for establishing it). We define this relative cost as

$$RelativeCost = \frac{SessionCost}{Gain}$$

By selectively blocking some sessions with high relative cost, we expect more benefit can be achieved. Correspondingly, when relative cost is considered, we evaluate the performance of our heuristics by the total gain obtained. Another advantage is by computing relative cost, we can balance the utilization of sessions (e.g. giving the broadcast sessions high gain value to keep them from being always blocked).

4.2 Blocking Strategy for LCFS

In this section, we describe the blocking strategy used for our algorithm. Basically, LCFS always attempts to establish a session when its cost is lower than a threshold and drops a session when its cost is higher than another threshold. For the remaining sessions, LCFS establishes them

---

[3] Note that with our definition of $N(r)$, it may be the case that the union over all r of $N(r)$ might not equal the entire solution space. We define $N(r)$ this way to improve our heuristic's efficiency.

probabilistically. Since the threshold needs to vary to accommodate dynamic traffic patterns, we base our thresholds and variance of the session costs. Specifically, we use *Exponentially Weighted Moving Average* (EWMA) to compute the estimated mean value of the tth session (μ_t). When using this procedure, the estimate μ_t is computed as $\mu_t = (1-\alpha) * c_t + \alpha * \mu_{t-1}$, where $0 < \alpha < 1$ is called the gain parameter, and is typically set between 0.1 and 0.2. Correspondingly, the *Mean Square Error* (MSE) is computed as $MSE_t = \frac{1}{t}\Sigma_{i=1}^{t}(c_i - \mu_i)^2$, which is an estimate of the variance of the forecast error. Thus, the estimate of standard deviation (σ_t) is computed as $\sigma_t = \sqrt{\frac{1}{t}\Sigma_{i=1}^{t}(c_i - \mu_i)^2}$. We then define our lower threshold as $\mu - a * \sigma$, and our upper threshold as $\mu + b * \sigma$, where a and b are two real numbers[4], choice of which depends on the application and can be adjusted empirically.

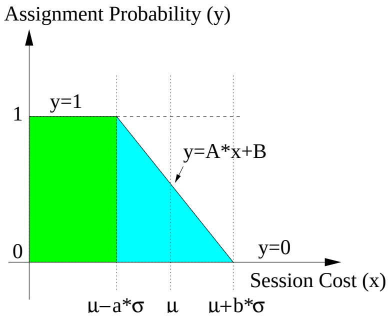

Figure 1. Assignment probabilities of sessions.

We use the slope to represent the selection probability of the sessions with cost between the lower and upper thresholds. As shown in Figure 1, the probability of establishing a session is $y = A * x + B$ with $A = (-1)/((a+b)*\sigma)$ and $B = (\mu + b*\sigma)/((a+b)*\sigma)$ If there are w wavelengths available, where $0 < w \leq W$ and W is the total wavelength number, then we generalize the session selection probability by $y \leftarrow 1 - (1-y)^w$. Other computation method of the probability can also be employed.

5 Numerical Results

We evaluated the performance of our algorithms on 21-node Italian network and 14-node NSFNET. The topologies of NSFNET and Italian networks are shown in Figures 2 and 3, respectively. The Italian network is composed of 21 nodes and 34 edges and NSFNET consists of 14 nodes and 21 edges. Every edge in the network represents two fiber links in opposite directions. We use MCPR algorithm [5] to create the primary and backup trees for each session.

5.1 Results for Static Traffic

In the case of DLTP under static traffic, we randomly generate 20 groups of multicast sessions based on NSFNET and the Italian network topologies, respectively. Each group of sessions includes 20 sessions and each fiber link carries 8 wavelength channels. We attempt to establish the sessions by ILP solver, LCSF, IHC-LTP, and TS-LTP heuristics respectively. The ILP formulation is solved by ILog Cplex Optimizer 6.5. The average numbers of established sessions, denoted by χ, and time obtained by the four approaches are compared in Table 2.

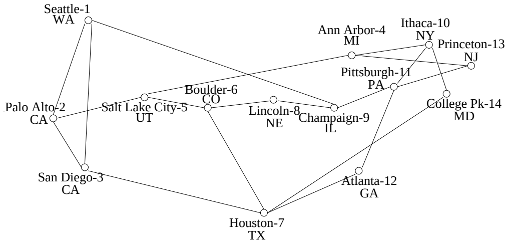

Figure 2. NSFNET

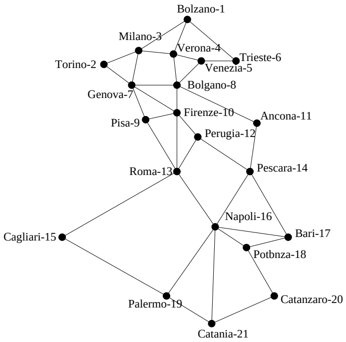

Figure 3. The high-speed Italian network.

Table 2. Results comparison of the ILP, LCSF, IHC-LTP, and TS-LTP approaches for DLTP. Time is in seconds.

Network	NSFNET	Italian
χ^{CPLEX}	16	15.6
CPLEX Time	14904.62	19688.1
χ^{LCSF}	13	12.4
LCSF Time	0.06	0.07
χ^{TS-LTP}	15.2	14.6
TS-LTP Time	0.5	0.6
$\chi^{IHC-LTP}$	14.8	15
IHC-LTP Time	2.9	3.2

All heuristic algorithms solve the problem in short time. The LCSF approach is the most efficient one, but the results are fair. The TS-LTP and IHC-LTP algorithms take longer time than the LCSF algorithm (e.g., 8 to 10 times or longer). However, they can assign about 15% to 20% more sessions than LCSF. TS-LTP and IHC-LTP yield similar results, and IHC-LTP usually takes longer time than TS-LTP.

In the case of SLTP under static traffic, we also randomly generate 20 groups of multicast sessions based on the two topologies, respectively. However, each group includes 25 sessions instead of 20[5]. The average numbers

---

[4] $0 \leq (\mu - a*\sigma) \leq (\mu + b*\sigma)$

[5] With SLTP scheme and 8 wavelength per fiber link, TS-LTP and IHC-

of established sessions and time obtained by the four approaches are compared in Table 3.

Table 3. Results comparison of the ILP, LCSF, TS-LTP, and IHC-LTP approaches for SLTP. Time is in seconds.

Network	NSFNET	Italian
χ^{CPLEX}	20	15.4
CPLEX Time	86400	86400
χ^{LCSF}	16.6	14.8
LCSF Time	0.07	0.1
χ^{TS-LTP}	19	17.4
TS-LTP Time	0.68	0.92
$\chi^{IHC-LTP}$	19	17.2
IHC-LTP Time	4.2	5.4

From the results, we can see that SLTP establishes more sessions than DLTP due to its efficient utilization of wavelength channels. Since the ILP formulation for SLTP contains more constraints and more sessions are considered, the CPLEX optimizer is unable to find the optimal solution for any instance within 24 hours. Thus, in table 3, we only give the best values that CPLEX reports. Sometimes, the results reported by CPLEX solver are worse than those of our heuristics. For example, the average number of sessions established in Italian network is 15.4 by using CPLEX, which is not as good as the ones reported by TS-LTP (17.4) and IHC-LTP (17.2). Similarly, TS-LTP and IHC-LTP algorithms take longer time than the LCSF algorithm, but they can assign about 14% to 18% more sessions than LCSF.

5.2 Results for Dynamic Traffic

To simulate the dynamic traffic, we randomly generate 5000 session requests on both NSFNET and Italian network. The session requests arrive according to a Poisson process with exponentially distributed holding time. Combining different blocking schemes and wavelength assignment algorithms, we evaluate two different heuristics: FCFS and LCFS with and without relative cost.

In Figure 4, we compare the simulation results of FCFS and LCFS approaches without considering relative cost, which is equivalent to that *all sessions have the same gain value*. Thus, we can obtain a high gain value if the session blocking performance is good. We observe that LCFS yields significantly lower blocking probability than that of FCFS. In Figure $4(a)$, the average improvement of blocking performance is 10% for DLTP and 7% for SLTP in NSFNET. In Figure $4(b)$, the average improvement of blocking performance is 12% for DLTP and 10% for SLTP in the Italian network. These indicate that dropping high cost sessions can effectively improve the blocking probability. The improvement is more obvious when traffic load is high. For example, in Figure $4(a)$, under DLTP scheme, the blocking probabilities of FCFS and LCFS are 0.15 and 0.13 respectively when $Erlang = 10$, and therefore the improvement is 2%. However, the improvement increases to 10% at 25 Erlang load where the blocking probabilities of FCFS and LCFS are 0.46 and 0.36 respectively. This is because when the traffic load is high, more conflicts will occur between low cost and high cost sessions. By blocking high cost sessions using LCFS, more low cost sessions can be established.

In Figure 5, we compare the total gain obtained based on different approaches when relative cost is considered. The gain value of each session is decided by the the number of its destinations. To further justify the effectiveness of our strategy, we also evaluate the performance of LCFS with and without relative cost consideration (selected or blocked by session cost). As expected, LCFS can dramatically increase the total gain realized. In Figure $5(a)$, where we compare the heuristics under DLTP protection scheme, the experimental results show that the average improvement of LCFS over FCFS is 14.2%. We can also see that LCFS with relative cost consideration outperforms LCFS without relative cost consideration. An average of 3% improvement can be achieved. Similarly, in Figure $5(b)$, under SLTP protection scheme, the average improvements of LCFS with relative cost considered over FCFS and LCFS without relative cost consideration are 11.6% and 3%, respectively. Like the results in Figure 4, the improvement becomes more obvious with high traffic load.

6 Conclusions

We investigated the wavelength assignment problem for light-tree protection in all-optical networks. Since the problem is NP-hard, based on different traffic models, we proposed different heuristics and blocking strategies to improve the system performance. For static traffic, we described the problem as integer linear program formulations and presented two heuristics (TS-LTP and IHC-LTP) to achieve efficient results. Our heuristics are better than other greedy algorithms in avoiding local optima. For dynamic traffic, we developed the Low Cost First Serve (LCFS) strategy to maximize the gain and throughput of network systems and compared it with the First Come First Serve (FCFS) strategy. All mathematical formulations and heuristics are evaluated under both DLTP and SLTP protection schemes. The experimental results show our algorithms can achieve good performance in terms of the blocking probability of the sessions. For example, both TS-LTP and IHC-LTP algorithms outperform LCSF algorithm by assigning 15% to 20% more sessions under DLTP protection scheme, and 14% to 18% under SLTP protection scheme. TS-LTP has similar performance as IHC-LTP but taking less time. Under dynamic traffic, we can achieves much better performance by using LCFS strategy than using FCFS strategy. When there are 8 wavelengths on each fiber link, the average improvement is 10% for NSFNET and 7% for Italian networks under DLTP protection scheme

---

LTP can often assign all 20 sessions successfully.

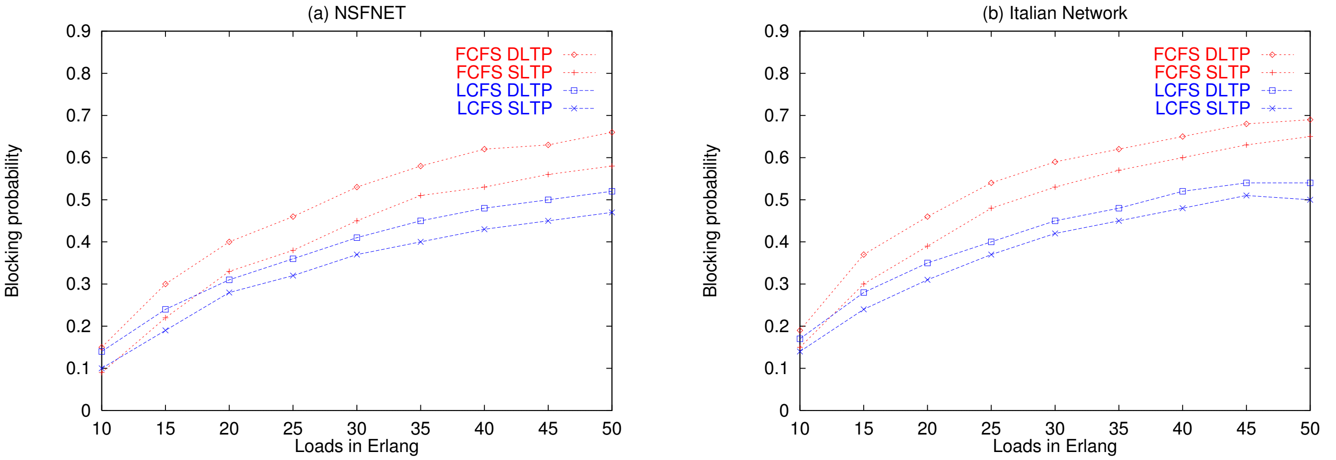

Figure 4. Blocking performance of LCFS and FCFS approaches with dynamic traffic in NSF and Italian networks (8 wavelengths per fiber link).

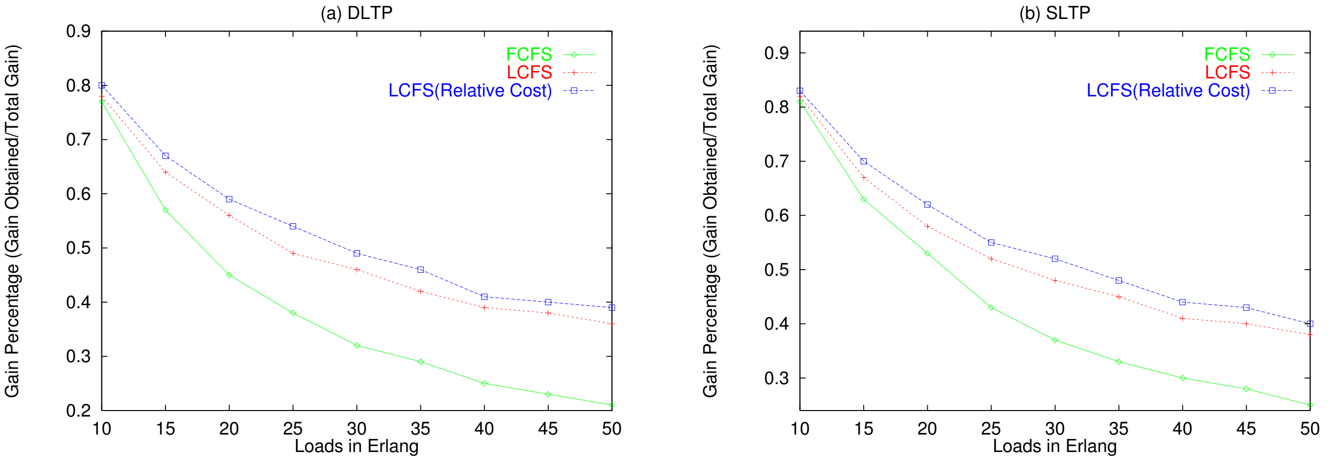

Figure 5. Gain performance of FCFS and LCFS approaches with dynamic traffic in Italian networks.

and 12% for NSFNET and 10% for Italian networks under SLTP protection scheme. When each session has a different gain value and relative cost is considered, LCFS can also dramatically improve the system performance.

Plans for future work include theoretical analysis of the blocking thresholds and evaluation of other blocking strategies under dynamic traffic.

References

[1] L. H. Sahasrabuddhe and B. Mukherjee, "Light-Trees: optical multicasting for improved performance in wavelength-routed networks", *IEEE Communications Magazine*, vol. 37, no. 2, pp. 67-72, Feb. 1999.

[2] N. K. Singhal and B. Mukherjee, "Protecting multicast sessions in WDM optical mesh networks", *IEEE Journal of Lightwave Technology*, vol. 21, no. 4, pp. 884-892, 2003.

[3] D. Zhou and S. Subramaniam, "Survivability in optical networks", *IEEE Network*, vol. 14, no. 6, pp. 24-32, 2000.

[4] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks", *Optical Networks Magazine*, vol.1, no.1, pp. 47-60, Jan. 2000.

[5] L. Kong, J. S. Deogun, and M. Ali, "Preplanned Recovery with Redundant Multicast Trees in Optical Networks", *Proc. of SPIE OptiComm*, pp. 419-423, Oct. 2003.

[6] B. Mukherjee, "Optical communication networks", *McGraw-Hill*, 1997.