Dynamic Wavelength Allocation in All-Optical Ring Networks*

Ori Gerstel
Optical Networking Group, Tellabs Operations Inc.†

Shay Kutten
Industrial Eng. Dept., Technion, Haifa, Israel‡

ABSTRACT: We focus on wavelength allocation schemes for all-optical WDM ring networks. For an N node network we characterize the traffic by its load $L_{\max}$ (the maximum number of lightpaths that share a link) and do not assume knowledge of the arrival/departure processes. We prove that shortest path routing produces a routing which has at most twice the load of the optimal solution. We show that at least $0.5L_{\max}\log_2 N + L_{\max}$ wavelengths are required by any algorithm in the worst case, and develop an algorithm which requires up to $3L_{\max}\log_2 N$ wavelengths. For the case when the load is high and blocking is necessary we present an improved algorithm.

Introduction

As WDM systems start emerging from laboratories and being deployed in commercial contexts, the main resource allocation problem associated with WDM networks — of efficiently allocating wavelengths to lightpaths — becomes of crucial importance. The focus of past research on the allocation of wavelengths in WDM networks (e.g, [1, 2] and several works in [3]) has been based on a probabilistic approach, in which the arrival/departure process of lightpaths is known (or assumed). In addition, many of these works are based on unproven heuristics often applicable to arbitrary topologies. The drawback of these approaches is that they are more suitable to high numbers of low-end connections such as telephone calls, than to high-end connections such as lightpaths, for which blocking is not a probable reaction of the service provider.

The current research takes a different approach. Instead of focusing on general topologies but restricted arrival/departure processes of lightpath requests, we assume no knowledge on these processes (and characterize the traffic by a single simple parameter — its load) but restrict the topology to rings only. We believe that this topological restriction still yields results of high practical value since ring networks are the predominant topology for current access/interoffice networks, and are thus expected to be the first physical topology on which WDM networks are deployed outside laboratories and testbeds. We also focus on worst case analysis of the problem as it is the only approach which guarantees robust bounds on the performance of the system. This restricted model enables us to study in greater depth the behavior of the problem and achieve almost tight bounds for different scenarios.

*A preliminary version of this paper was published as an IBM research report no. RC20462, May 1996

†The work was supported in part by DARPA grant MDA 972-95-C-0001 while being with the IBM T.J. Watson Research Center.

‡Part of the work was done while being with the IBM T.J. Watson Research Center.

We separate the *routing* problem, of determining which part of the ring should be used to connect the source and destination of a lightpath request, from the *wavelength allocation* problem, of assigning a wavelength to each route. This technique is justified for the following reasons:

- The network users may choose to have control on the routing to support fault tolerance (namely, two routes may require disjoint paths as they are responsible for backing up each other),
- Additional considerations, such as constraints on propagation delays may require some route to take the short alternative around the ring,
- Computationally efficient solutions to the combined routing and wavelength allocation problem which allocate resources optimally are not plausible [4].

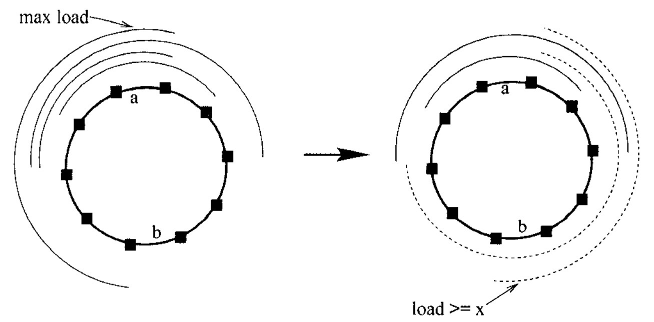
Resource allocation problems can be classified as *static* problems, in which the entire set of requests is known in advance and *dynamic* problems in which the requests arrive at different times and the algorithm has to react without knowledge of the future.

Substantial analytical research has been carried out on the worst-case performance of the static routing and wavelength allocation problems for optical networks, starting with classic problems of wavelength allocation in chain and ring networks [5], through various models of tree topologies [1, 6], and general mesh topologies [2, 6]. On the other hand, numerous heuristic techniques have been proposed for determining the lightpaths in static and dynamic systems using average case analysis [7, 8, 9, 10, 3]. However, to the best of our knowledge, this is the first study of the worst-case behavior in the dynamic model [11]. This work has spawned additional work on this subject [12, 13, 14].

As mentioned earlier, the composite problem of finding a routes for each lightpath request and allocating a wavelength to it is very hard even for ring [4]. It is therefore reasonable to split the problem into two phases: (1) finding a route for each lightpath request and (2) allocating a wavelength to it. As far as the first problem is concerned, an optimal algorithm exists for the static design problem [15]. The dynamic case for this problem is considered in the present work, in which we show that the simple shortest-path heuristics is up to twice away from the optimal solution.

As far as the second problem is concerned, given a sequence of lightpath requests and the physical route for each of them, let the maximum load on any link in the network (denoted $L_{\max}$) be the maximum number of these routes that share a link. This characteristic of a lightpath request system turns out to be the major factor in the worst case analysis of

0-7803-3925-8/97 $10.00 ©1997 IEEE



(a) Shortest path routing

(b) After moving x of the routes to the other option around the ring

Figure 1: Shortest path routes are not much worse than optimal

the system. Clearly, $L_{\max}$ is a lower bound on the number of required wavelengths since each lightpath that shares a link with maximum load must be assigned a different wavelength.

For the static case $2L_{\max}$ wavelengths suffice to support any request pattern [5]. For the dynamic case, the only known results are based on probabilistic models, assuming standard Poisson arrival processes [9]. In the current paper we present a wavelength allocation algorithm that requires at most $W \leq 3L_{\max}\log_2 N$ wavelengths for a ring with N nodes[1] and prove a lower bound of $0.5L_{\max}\log_2 N + L_{\max}$, thereby proving our algorithm to perform at most six times worse than the optimum. We also present an improved algorithm that has better blocking probability for higher loads.

An important conclusion from these results is that, at least as far as worst case analysis is concerned, dynamic scenarios result in significant degradation of the utilization of wavelengths comparing to the static scenarios. Consequently, wavelength conversion is very beneficial in this model.

The paper is organized as follows. in Section 2 we adress the route determination problem and show that the simple shortest path heuristic performs well. In Section 3 we present our algorithm and prove its worst case performance. We also demonstrate that other natural algorithms may have very bad performance. In Section 4 we prove that any algorithm cannot do much better than ours and present and improved algorithm in Section 5, which we prove to be as good in the worst case as the original algorithm as long as no blocking occurs and clearly has better performance if blocking is necessary. In Section 6 we summarize the results and suggest further research.

Route determination

Consider a set of requests for lightpaths for which only source and destination pairs are given for each request. In this section we prove that no algorithm can minimize the maximum load $L_{\max}$ more than a factor of two from the load created by the algorithm which routes each lightpath request on the shortest route between the source and destination.

Given any configuration of lightpaths (possibly after deletions of lightpaths) produced using shortest path routing, let L_{shrt} denote the maximum load ($L_{\max}$) for this case. Consider a link a with maximum load L_{shrt} depicted in Figure 1. Also consider the link b which is diametrically opposite to a on the ring. Since routes of lightpaths that cross a are the shortest possible, none of them crosses b as well (otherwise they would traverse more than half of the ring). Therefore, in any other solution that does not route x of them through a, these x are routed through b, and thus the load on b is at least x. It follows that the maximum load in any such solution cannot be reduced below $\frac{L_{\mathrm{shrt}}}{2}$ of the load L_{shrt} used by the shortest path algorithm, by changing the routes of some of the requests to the other alternative around the ring.

[1] This result has been improved by a factor of three in a subsequent paper [12].

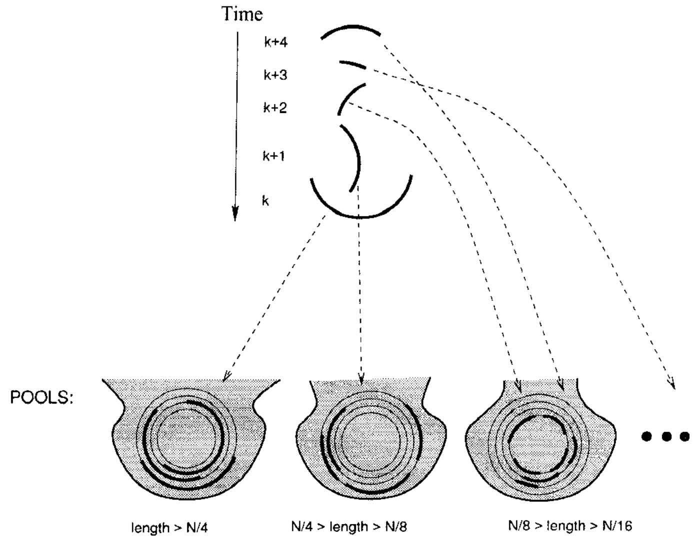


Figure 2: Operation of the dynamic case algorithm

Theorem 1. *At any given time, the load of the current set of undeleted lightpaths, S, produced by shortest path routing does not exceed twice the optimal offline (and certainly online) load of S.*

Wavelength allocation: upper bound

The DWLA-1 algorithm (Dynamic WaveLength Allocation, see Figure 3) allocates wavelengths in a dynamic setting with a performance guarantee of $W \leq 3L_{\max}\log_2 N$ for a ring with N nodes (as long as $L_{\max} \leq L_{\mathrm{alg}}$ for some predetermined value L_{alg}). The key idea behind it is to avoid fragmentation of the wavelength resource by allocating each wavelength to routes of approximately the same length. Refer to Figure 2 for a pictorial demonstration of the algorithm in which requests fall down from the top of the figure and get sorted into the pools based on their length.

In what follows we assume that the ring contains $N = 2^k$ nodes — the presentation of the results is simplified by this assumption, and it is easy to see how other cases are dealt with.

The main claim to be proven for this simple algorithm is that we do not run out of wavelengths in Step 3, as long as the load does not exceed the load for which the algorithm was designed, L_{alg}.

Lemma 1. *Consider lightpaths of length $N \cdot 2^{-i-1} \leq \ell(x) < N \cdot 2^{-i}$. If the maximum load of such lightpaths does not exceed L_{alg} then $3L_{\mathrm{alg}}$ wavelengths suffice for them.*

0. INPUT: receive add/delete lightpath requests, one at a time
   DATA STRUCTURE: Define a set of wavelength pools $\{\text{POOL}(i)\}_{i=0}^{\log_2 N-1}$, each pool containing $3L_{\text{alg}}$ wavelengths (L_{alg} being the anticipated maximum load).
1. If the request is to delete a lightpath, delete it and mark the relevant wavelength segment as free. Otherwise:
2. Let x be the current request to add a lightpath and let $\ell(x)$ be its length (the number of links traversed by the route). Choose i such that $N \cdot 2^{-i-1} \leq \ell(x) < N \cdot 2^{-i}$.
3. Find a free segment of a wavelength in pool $\text{POOL}(i)$ which can accommodate x, and allocate x on this wavelength.
4. Handle next request.

Figure 3: Dynamic allocation of lightpath requests (DWLA-1)

Proof. Consider two sets of links on the ring: one, $\mathcal{A} = \{A_j\}_{j=0}^{2^i-1}$, with ring segments of size $N \cdot 2^{-i}$ between each "consecutive" pair A_j and $A_{(j+1) \bmod 2^i}$. The other, $\mathcal{B} = \{B_j\}_{j=0}^{2^i-1}$, having the same distance between each consecutive pair, and located in the middle of the segments defined by links in $\mathcal{A}$.

Given a lightpath x with length $N \cdot 2^{-i-1} \leq \ell(x) < N \cdot 2^{-i}$, it crosses no more than one A_x in $\mathcal{A}$. However, if it crosses no such link, it has to cross one link of $\mathcal{B}$, say B_x. Thus, if x does not cross any link in $\mathcal{A}$, it shares B_x with up to $L_{\text{alg}} - 1$ other lightpaths that do not cross an $\mathcal{A}$ link. Let $LP(l)$ denote the lightpaths that cross link l. Note that the lightpaths in $LP(B_x)$ do not overlap lightpaths in any other $LP(B_y)$. Thus, L_{alg} wavelengths suffice for them.

Split $\mathcal{A}$ into two sets: every other A_j will be in a set $\mathcal{A}'$ and the rest in a set $\mathcal{A}''$. It is easy to see that lightpaths in $LP(A_x)$ and lightpaths in $LP(A_y)$ for $A_x \neq A_y \in \mathcal{A}'$ do not overlap and can be allocated the same set of L_{alg} wavelengths. The same argument holds for lightpaths which cross links in $\mathcal{A}''$. □

Theorem 2. *As long as* $L_{\max} \leq \frac{W}{3 \log_2 N}$ *the DWLA-1 algorithm does not block any requests.*

Many natural wavelength allocation algorithms do not have good performance in the worst case dynamic scenario. An example is circular-first-fit (CFF). This algorithm tries to allocate a wavelength to a lightpath request starting at a different starting point each time, in a circular fashion: for the i^{th} request it checks the wavelengths: $i \bmod W, (i+1) \bmod W, (i+2) \bmod W, \ldots$ until it find a wavelength to accommodate the request or finishes a scan of all the wavelengths and fails. The following theorem shows that CFF may need a number of wavelengths that depends linearly on N (instead of logarithmically, as in DWLA-1).

Theorem 3. *Given a ring with N nodes, there exists a set of lightpath requests with load* $L_{\max}$*, for which Circular-First-Fit needs at least* $W = 1 + N(L_{\max} - 1)$ *wavelengths to support all the requests.*

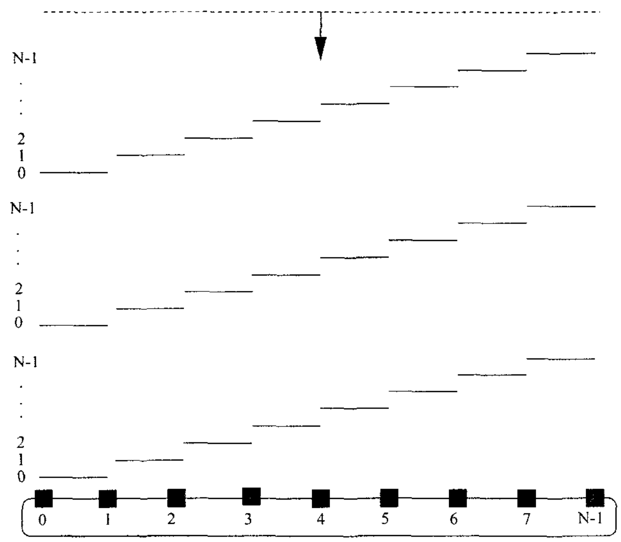


Figure 4: A worst case set of lightpath requests for the Circular-First-Fit algorithm. The requests arrive starting from the bottom left request and proceeding in rows to the top right request. The long dashed request at the top is blocked.

The proof is omitted for the sake of brevity, however, the sketch in Figure 4 corresponds to a configuration of lightpaths upon blocking of the dashed lightpath request.

WaveLength allocation: lower bound

We now prove that in the worst case $W \geq 0.5 L_{\max} \log_2 N + L_{\max}$. We start with $L_{\max} = 2$.

Consider the following scenario, depicted in Figure 5. At each phase i, a request arrives for a lightpath that overlaps all the currently existing $i-1$ lightpaths. Thus any algorithm has to allocate it a new wavelength. Playing an adversary who issues the requests, we manage to manipulate any allocation algorithm (by means of additional add/delete requests) to utilize i wavelengths while the load $L_{\max}$ remains 2 at all times. This process can only be repeated $\log_2 N + 2$ times, since in each phase i, the adversary is forced to issue lightpath requests traversing 2^{i-2} links. More formally, given some allocation algorithm Z, we now describe a worst case scenario specialized for it, in the following phases.

Phases 1 and 2. Two requests arrive to establish lightpaths p_1 and p_2 in the segment $[0, 1]$. Clearly they are allocated different wavelengths by Z.

Phase 3. A third request p_3 arrives for a lightpath in the segment $[1, 2]$. If Z allocates to it a wavelength which is different from those allocated to p_1 and p_2, then the phase ends — so far three wavelengths have been allocated. On the other hand, if Z allocates to p_3 the same wavelength that was allocated to either p_1 or p_2 (say p_1), then a request arrives for deleting p_1, and yet another lightpath addition request p_4 arrives for a lightpath in $[0, 2]$. Clearly Z allocates a third wavelength for p_4.

Phase 4. Phases 1–3 are repeated in the segment $[2, 4]$ as well. After which it is easy to see that it is possible

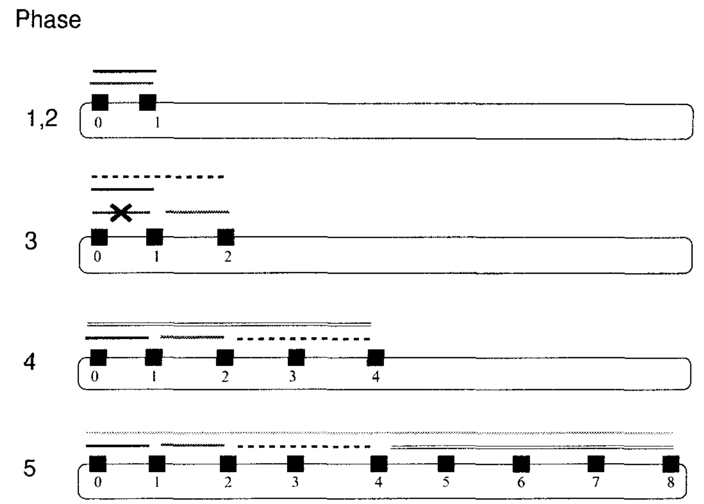


Figure 5: A worst case dynamic scenario of lightpath requests

to choose three non-overlapping lightpaths in segment [0, 4] which have been allocated different wavelengths. For the rest of the lightpaths, delete requests are generated. Now, a new lightpath add request arrives for a lightpath in [0, 4]. Z has to allocate a new wavelength to it, resulting in a total of four different wavelengths. Note that $L_{\max}$ is still at most two.

⋮

Phase i. After repeating Phases 1 to $i-1$ in segments $[0, 2^{i-3}]$ and $[2^{i-3}, 2^{i-2}]$, and deleting superfluous lightpaths to achieve a configuration of $i-1$ non-overlapping lightpaths of different wavelengths, a new request arrives to add a lightpath in the segment $[0, 2^{i-1}]$. Z allocates a new i^{th} wavelength to it, since it overlaps $i-1$ other wavelengths.

⋮

Phase $\log_2 N + 2$. The last lightpath request arrives in the segment $[0, N-1]$. Z allocates wavelength $\log_2 N + 2$ to it.

This process required $W \geq \log_2 N + 2$ wavelengths, with a maximum load of $L_{\max} = 2$. Thus, $W \geq 0.5 L_{\max} \log_2 N + L_{\max}$. To generalize the worst case to any (even) value of $L_{\max}$, we multiply the number of arriving lightpath requests at each phase by $L_{\max}/2$. Since each of these $L_{\max}/2$ requests requires a different wavelength the whole allocation process is inflated by a factor of $L_{\max}/2$ wavelengths per phase, yielding the desired lower bound.

Theorem 4. *For every wavelength allocation algorithm there exists some addition/deletion scenario that requires the algorithm to use $W > 0.5 L_{\max} \log_2 N + L_{\max}$ wavelengths.*

Improving the algorithm

As shown above, the DWLA-1 algorithm guarantees no blocking of requests if the load $L_{\max}$ does not exceed some value L_{alg}. However, it does not necessarily perform well if $L_{\max} > L_{\text{alg}}$. In such cases, some pool $\text{POOL}(i)$ may be overflown and unable to accommodate additional requests of

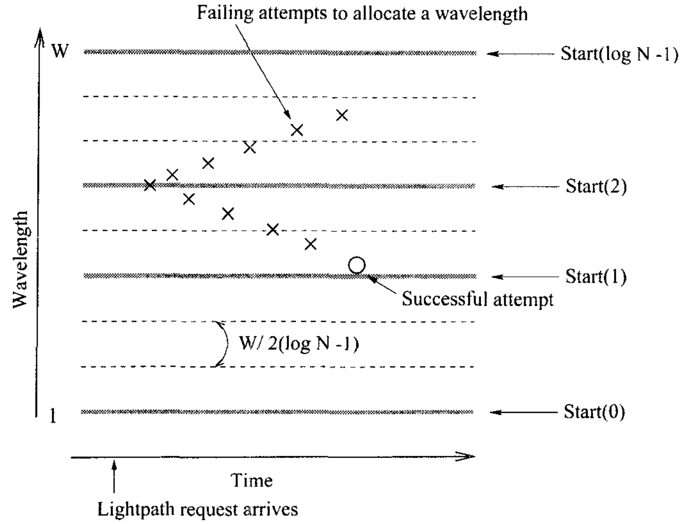


Figure 6: Improved dynamic algorithm (DWLA-2)

length $N \cdot 2^{-i-1} \leq \ell(x) < N \cdot 2^{-i}$, while other pools remain empty. While our lower bound shows that in the worst case there is no way to guarantee no blocking if the load is high, it is still desirable to minimize this blocking. The algorithm DWLA-2 presented in this section has a more flexible pool structure that aims at reducing the "trunking effect" that is caused by the rigid reservation scheme of DWLA-1. We prove that DWLA-2 works as good as DWLA-1 provided that the load is low enough ($L_{\max} < \frac{W}{3\log_2 N}$). On the other hand it works much better for higher loads. This algorithm is a generalization of the Incr/Decr algorithm of [7].

The algorithm starts allocating wavelengths for a request of length $\ell(x)$, at wavelength $\text{START}(i)$. If $\text{START}(i)$ cannot accommodate the request, DWLA-2 searches an available wavelength in the next/previous wavelengths ($\text{START}(i) \pm 1$), and gradually increases the search distance from $\text{START}(i)$ until a free wavelength is found (see Figure 6). For a more formal description refer to Figure 7.

Theorem 5. *The DWLA-2 algorithm never starts blocking requests earlier than DWLA-1.*

Proof. If the load is $L_{\max} \leq \frac{W}{3\log_2 N}$ then no two lightpath requests that use different pools in DWLA-1 will use the same wavelength in DWLA-2, since the number of wavelengths around $\text{START}(i)$ which are scanned first is L_{alg}. As a result DWLA-2 behaves exactly as DWLA-1. If the load exceeds $\frac{W}{3\log_2 N}$ then DWLA-1 will start blocking lightpaths for which the pool is full, while DWLA-2 may continue to allocate wavelengths to these lightpaths, using wavelengths of adjacent pools. □

Corollary 1. *If $L_{\max} \leq \frac{W}{3\log_2 N}$ then DWLA-2 guarantees no blocking.*

Note that DWLA-2 allows any wavelength to be used for any request thereby eliminating the trunking effect. Thus, DWLA-2 typically delays blocking to a much later point in time.

0. INPUT: receive add/delete lightpath requests, one at a time

DEFINE: $\textsc{Start}(i) = \begin{cases} 0 & i = 0, \\ W-1 & i = \log_2 N - 1, \\ (2i+1)\frac{W}{2(3\log_2 N - 1)} + i & \text{otherwise.} \end{cases}$

1. If the request is to delete a lightpath, delete it and mark the relevant wavelength segment as free.
2. Let x be the current request to add a lightpath and let $\ell(x)$ be its length (the number of links traversed by the route). Choose i such that $N \cdot 2^{-i-1} \leq \ell(x) < N \cdot 2^{-i}$.
3. Let $\Delta \leftarrow 0$.
4. If one of the wavelengths $\textsc{Start}(i) \pm \Delta$ can accommodate x, allocate it and finish (goto Step 6).
5. Otherwise, if $\Delta = W$ then block the request else $\Delta \leftarrow \Delta + 1$ and goto Step 4.
6. Handle next request.

Figure 7: Improved dynamic allocation of lightpath requests (DWLA-2)

Summary and further research

In this paper we have studied the problem of allocating wavelengths to lightpaths in a WDM ring system in which wavelength conversion is not possible. We have first shown that shortest path routing is a good heuristics for wavelength routing. Next, we have suggested an algorithm which uses up to $3L_{\max} \log_2 N$ wavelengths. Another way to look at this result is a guarantee of no blocking as long as the load does not exceed $\frac{W}{3\log_2 N}$. We have also shown that in the worst case this algorithm performance is up to six times the minimum possible number of wavelengths that any algorithm needs. We have suggested an improved algorithm, DWLA-2, which has the same guarantees if the load is low enough, and at the same time has delayed and better blocking probability if the load is higher. While we have indicated why this algorithm should perform better than DWLA-1, it is still necessary to demonstrate this empirically.

An interesting remaining issue is to find an algorithm with better worst case performance. However, for deployment in real implementations, the average performance of the algorithm must be studied as well and should prove competitive with other allocation algorithms.

Acknowledgment. We would like to thank Rajiv Ramaswami for the fruitful discussions.

References

[1] I. Chlamtac, A. Ganz, and G. Karmi, "Lightnets: Topologies for high-speed optical networks," *IEEE/OSA Journal on Lightwave Technology*, vol. 11, pp. 951–961, May/June 1993.

[2] A. Aggarwal, A. Bar-Noy, D. Coppersmith, R. Ramaswami, B. Schieber, and M. Sudan, "Efficient routing and scheduling algorithms for optical networks," in *Proc. 5th Annual ACM-SIAM Symp. on Discrete Algorithms*, pp. 412–423, Jan. 1994.

[3] R. L. Cruz, G. R. Hill, A. L. Kellner, R. Ramaswami, and G. H. Sasaki, eds., *IEEE JSAC / IEEE/OSA JLT: Special Issue on Optical Networks*, June 1996.

[4] M. Garey, D. Johnson, G. Miller, and C. Papadimitriou, "The complexity of coloring circular arcs and chords," *SIAM Journal on Discrete Math*, vol. 1, no. 2, pp. 216–227, 1980.

[5] A. Tucker, "Coloring a family of circular arcs," *SIAM Journal on Applied Math*, vol. 29, no. 3, pp. 493–502, 1975.

[6] P. Raghavan and E. Upfal, "Efficient routing in all-optical networks," in *Proc. 26th ACM Symp. Theory of Computing*, pp. 134–143, May 1994.

[7] A. Birman and A. Kershenbaum, "Routing and wavelength assignment methods in single-hop all-optical networks with blocking," in *IEEE Infocom'95*, pp. 431–438, 1995.

[8] 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.

[9] M. Kovačević and A. Acampora, "Benefits of wavelength translation in all-optical clear-channel networks," *IEEE JSAC / IEEE/OSA JLT: Special Issue on Optical Networks*, vol. 14, no. 5, pp. 868–880, 1996.

[10] R. Ramaswami and K. Sivarajan, "Routing and wavelength assignment in all-optical networks," *ACM/IEEE Transactions on Networking*, pp. 489–500, Oct. 1995. An earlier version appeared in Infocom'94.

[11] O. Gerstel and S. Kutten, "Dynamic wavelength allocation in WDM ring networks," Research Report RC 20462, IBM Research Division, Watson Research Center, May 1996. To appear in ICC'97, Montréal, Canada.

[12] O. Gerstel, G. Sasaki, and R. Ramaswami, "Dynamic channel assignment for wdm optical networks with little or no wavelength conversion," in *The 34th Allerton conf. on Communication, Control, and Computing*, 1996.

[13] O. Gerstel, R. Ramaswami, and G. Sasaki, "Benefits of limited wavelength conversion in wdm ring networks," in *OFC'97 Tech. Digest*, 1997.

[14] O. Gerstel, G. Sasaki, S. Kutten, and R. Ramaswami, "Worst case analysis of dynamic wavelength allocation in optical networks." Submitted to IEEE/ACM ToN, 1997. Also published as IBM research report RC 20717, 2/1997.

[15] A. Frank, T. Nishizeki, N. Saito, H. Suzuki, and E. Tardos, "Algorithms for routing around a rectangle," *Discrete Applied Math*, vol. 40, pp. 363–378, 1992.