Worst-Case Analysis of Dynamic Wavelength Allocation in Optical Networks*†

Ori Gerstel
Optical Networking Group, Tellabs Operations, 15 Saw Mill River Road, Hawthorne, NY 10532,
ori@tellabs.com

Galen Sasaki
Department of Electrical Engineering, University of Hawaii
2540 Dole Street, Honolulu, HI 96822
sasaki@spectra.eng.hawaii.edu

Shay Kutten
Department of Industrial Engineering, Technion, Haifa 32000, ISRAEL,
kutten@ie.technion.ac.il

and

Rajiv Ramaswami
Optical Networking Group, Tellabs Operations, 15 Saw Mill River Road, Hawthorne, NY 10532,
rajiv@tellabs.com

Abstract

This paper proposes algorithms for allocation of wavelengths to connections (lightpaths) in optical wavelength division multiplexed networks, predominantly for ring topologies. The worst-case situation is considered where no blocking is allowed, and there are no assumptions on the traffic arrival and holding times. The traffic is characterized only by its *load* L, which is the maximum number of lightpaths that can be present on any link assuming no blocking.

We start with networks without wavelength conversion, consider a static scenario and prove that the known algorithm which requires $2L-1$ wavelengths is optimal. For a dynamic scenario we show that shortest path routing produces a routing which has at most twice the load of the optimal solution. We also show that at least $0.5L\log_2 N + L$ wavelengths are required by any algorithm for rings of N nodes and present an algorithm that uses at most $L\log_2 N + L$ wavelengths for rings and $2(L-1)\log_2 N$ for trees. For rings, the known *First-Fit* algorithm is shown to require at most $2.53L\log_2 N + 5L$ and at least $0.9L\log_2 N$ wavelengths.

When limited wavelength conversion is allowed, we first show how to use expanders to insure no blocking in arbitrary topologies. Then we present conversion patterns for rings with conversion degree $d = 2$ which require $L\log_2 L + 4L$ or $2L\log_2\log_2 L + 4L$ wavelengths. In a different traffic model where lightpaths are never taken down, the number of wavelengths needed is shown to be only $\max\{0, L-d\} + L$ for a conversion degree of d.

*The work of O. Gerstel, G. Sasaki, and R. Ramaswami was sponsored in part by DARPA grant MDA-972-95-C-0001.

†This paper was submitted for publication to the IEEE/ACM Transactions on Networking journal. This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

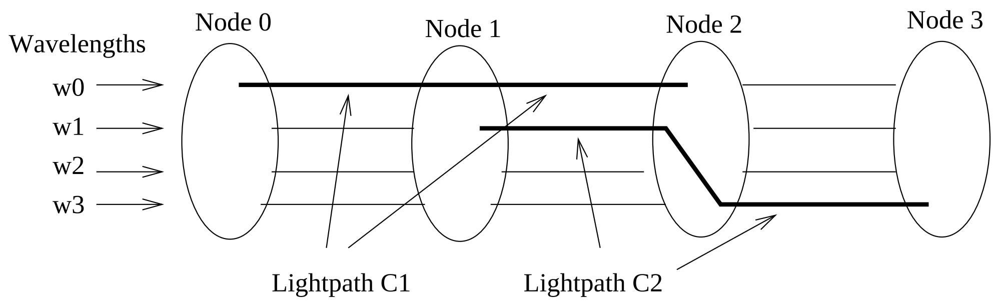


Figure 1: A WDM network with wavelengths $\{\omega_0, \omega_1, \omega_2, \omega_3\}$ and two lightpaths $C1$ and $C2$.

1 Introduction

In this paper we consider *wavelength division multiplexed* (WDM) optical networks. WDM networks use multiple communication channels over a single optical fiber. The channels are at different *wavelengths*. These networks support *lightpaths*, which are end-to-end circuit-switched communication connections that traverse one or more links and use one WDM channel per link.

Figure 1 shows a WDM network that is composed of four nodes with optical fiber links, where each link has four channels at wavelengths $\{\omega_0, \omega_1, \omega_2, \omega_3\}$. Channels at the same wavelength are connected at a node. However, *wavelength conversion* devices are required to connect channels at different wavelengths. For example, lightpath $C2$ needs a wavelength converter at node 2, while lightpath $C1$ requires no wavelength conversion. Wavelength conversion can help improve the utilization of the channels, but at additional cost and complexity.

1.1 Problem description and related works

We study the problem of allocating channels to lightpaths to insure no blocking under different models for the lightpath arrival and termination requests. In the *static* model, all lightpath requests are given in advance. In the *incremental* model, requests arrive as time goes by but are never terminated. In the fully *dynamic* model, requests arrive and depart in time.

Note that we do not allow blocking. We believe that this assumption is more suitable for our case than a statistical model that allows blocking [1, 2, 3, 4, 5]. This is because lightpaths carry data at high bit rates (several gigabits/second) are usually set up on a provisioning basis. As a result, the network operator will try to satisfy the demand by upgrading its network (resulting in a change of the topology) rather than blocking the request. On the other hand, this model may result in over-engineering the network to support pathological sets of requests. In other words, it may be possible to support most sets of lightpath requests using very few wavelengths. However there may be some specific request sets that need a large number of wavelengths to prevent blocking.

The model uses the following assumptions:

Wide-sense non-blocking: Existing lightpaths cannot be disrupted in the process of accommodating new demand, due to their high quality of service requirements. Thus it is impossible to rearrange the configuration of lightpaths.

Load constraint: The traffic is modeled by a single parameter termed its *load* L, which is defined to be the maximum number of lightpaths that can be on any link at any time assuming no blocking. Clearly, it is necessary that $L \leq W$, where W is the number of wavelengths, otherwise blocking will occur. Our goal will be to determine the smallest possible value for W that can support these lightpath requests.

This model assumes little knowledge of the traffic. Statistical models, on the other hand, assume certain arrival statistics (e.g., Poisson) and holding times for lightpath requests, as well as a certain traffic distribution (e.g., uniform traffic) which may not accurately reflect the traffic demand.

Separate routing and wavelength allocation: As in many earlier works, we separate the routing problem from the wavelength allocation problem. The justification for this approach is three-fold: (1) The network users may choose to have control on the routing to support fault tolerance (namely, two lightpaths may require disjoint paths as they are responsible for backing up each other), (2) Additional considerations such as constraints on propagation delays may require some lightpaths to take the shortest path around the ring, and (3) Computationally efficient solutions to the combined routing and wavelength allocation problem which allocate resources optimally are not plausible even for the simpler static case [6].

Our goal is to minimize the number of wavelengths required to support all lightpath requests with a given load. Our approach will be develop algorithms that determine the wavelength allocation using a certain maximum number of wavelengths, called the *upper bound*. We also give *lower bounds*, usually by providing an example of a lightpath request set for which any algorithm needs at least this many wavelengths. An algorithm that achieves the lower bound is said to be an optimal algorithm.

Note that an optimal algorithm according to our definition is not necessarily optimal for every instance of the problem. It is only optimal in the worst-case, i.e., it does the best possible wavelength allocation for the worst-case request set. It may produce rather poor wavelength allocations for other request sets. Much research still needs to be done to find algorithms which are good for every instance of the problem.

Note also that the results for this problem vary depending on our assumptions underlying the directivity of the lightpath requests and the network links. We assume that both the physical link and lightpaths are undirected. However, directed lightpaths and/or links could be considered and the resulting bounds vary depending on these assumptions [7, 8, 9]. We believe that our choice of undirected links and lightpaths is more appropriate for the current telco infrastructure which usually assumes undirected links and requests.

The static wavelength allocation problems in rings is the same as the problem of coloring circular arc graphs and an algorithm that does the allocation using at most $2L-1$ wavelengths is given in [10]. For tree topologies, several different models were discussed and optimal results presented in [7, 8, 9]. For arbitrary topologies, even the wavelength allocation problem itself becomes very hard [11, 8].

The incremental model[1] was discussed for linear topologies and rings and an algorithm presented in [12]. This algorithm was shown to be optimal in [13].

All the work above deals with networks without any wavelength conversion capabilities. The static allocation problem with limited conversion capabilities was studied in [14].

As for the incremental model, we modify the algorithm of [12] to produce a good wavelength allocation for the case of limited conversion. The dynamic model in this context was presented in our preliminary works [15, 16, 17], the results of which comprise the current paper.

One of the main conclusions from the results presented herein is that, at least as far as worst case analysis is concerned, fully dynamic scenarios result in significant degradation of the utilization of wavelengths over static and incremental scenarios, and that the difference between the efficiency of incremental scenarios and fully dynamic ones (i.e., the fact that deletions are allowed) grows logarithmically with the network size. Another important conclusion is that very limited amount of wavelength conversion

[1] Also called the online or semi-dynamic model.

Model	Conv:	No conversion		$d = 1$		$d > 1$		Full
Static	Lower	$2L - 1$	Thm. 1	$L + 1$	[14]	L	[14]	L (trivial)
	Upper	$2L - 1$	[10]	$L + 1$	[14]	L	[14]	L (trivial)
Incre-mental	Lower	$3L$	[13]	?		?		↑ same
	Upper	$3L$	[12]	?		$\max(L, 2L - d)$	Thm. 11	↑ same
Dyna-mic	Lower	$0.5L \log_2 N + L$	Thm. 6	← same		?		↑ same
	(FF)	$0.9L \log_2 N$	Thm. 8	← same		?		↑ same
	Upper	$L \log_2 N + L$	Thm. 4	← same		$\min(L \log_2 L + 4L,$ $2L \log_2 \log_2 L + 4L)$	Thm. 10 ($d = 2$)	↑ same
	(FF)	$2.53L \log_2 L + 5L$	Cor. 1	?		?		↑ same

Figure 2: Summary of worst-case bounds on the number of wavelengths for different lightpath arrival models on ring networks and for different wavelength conversion capabilities. The lower bound indicates that there is a set of lightpath requests with load L for which no algorithm can produce a better assignment. The upper bound indicates that there is an algorithm that can perform the wavelength allocation using that many wavelengths for any set of lightpath requests with load L.

results in substantial improvements in the numbers of required wavelengths. This was realized earlier for the static case [14], but we show that it is the case for the incremental and dynamic models as well. On the other hand, the statistical models of [2, 1] predict lower gains due to wavelength conversion. The difference is probably due to the inherent differences between the traffic models.

1.2 Summary of results

A summary of our results on rings appears in Figure 2. In Section 2, we study the static problem and show that the algorithm of [10] is optimal for rings. In Section 3 we consider the dynamic model and prove a lower bound of $0.5L \log_2 N + L$ and upper bound of $L \log_2 N + L$ wavelengths. We extend our algorithm for trees as well, achieving $W \leq 2L \log_2 N$. In Section 3.4, we consider the well-known channel allocation algorithm called *First-Fit*, which has been shown to be efficient in simulation experiments under the statistical model [18, 1]. We show that First-Fit is good even in the worst case. In particular, we show that First-Fit on a ring requires at least $0.9L \log_2 N$ wavelengths to ensure no blocking and can always do the wavelength allocation using at most $2.53L \log_2 N + 5L$ wavelengths.

In Section 4, we investigate how wavelength conversion can improve the utilization of channels. Our wavelength conversion model, based on [14], assumes that certain pairs of channels in adjacent links may be interconnected. We refer to a pair of channels that may be connected as being *compatible* signifying that a lightpath that uses one channel on one link may use any channel that is compatible to the first, on the next link. signal on one may be switched and/or converted to the other. The *conversion degree* of a network is the maximum number of channels which are compatible with any channel. The wavelength conversion capability of a network can be measured by its conversion degree. For example, a network with full conversion capability has conversion degree W, the number of wavelengths. While a network with no wavelength conversion has conversion degree one. We show that if W is sufficiently large then there exists a conversion pattern between adjacent channels which enables networks with arbitrary topology to insure no blocking as long as the traffic load is at most δW, where $\delta > 0$ is some fraction independent of W and N. This result however, does not directly lead to practical solutions since δ is quite small. We also present results for a ring network with conversion degree two, which indicate that $W \leq \min\{L \log_2 L + 4L, 2L \log_2 \log_2 L + 4L\}$ wavelengths suffice to guarantee no blocking. For the incremental model we show that much more efficient utilization of wavelengths is possible and $W \leq \max\{0, L - d\} + L$ wavelengths suffice. Conclusions are given in Section 5. We also present there a different view of Figure 2.

2 Static wavelength allocation, No conversion

We start by considering the simplest case, in which the full set of lightpaths is given in advance. This case is applicable in many networks, in which the required set of lightpaths is determined as part of the network design phase of a higher level network.

Routing the lightpaths so as to minimize the maximum load can be done optimally in this case [19, 20]. However determining the minimum number of wavelengths for a given set of lightpath requests is NP-hard even for rings [6]. [10] proposed an algorithm that does the wavelength allocation using at most $2L-1$ wavelengths. We prove that in the worst case, $W = 2L-1$ wavelengths are required for any algorithm, showing that the algorithm of [10] is optimal according to our definition.

Theorem 1 *Given a ring with $N > 2L$ nodes, there exist lightpath patterns that require $W = 2L-1$ wavelengths.*

Proof. Consider the set of requests depicted in Figure 3. These requests are divided into three groups: $\mathcal{A} = \{a_1, ..., a_{L-1}\}$, $\mathcal{B} = \{b_1, ..., b_{L-1}\}$ and $\{c\}$. All the routes in group $\mathcal{A}$ overlap on link A, and all the routes in group $\mathcal{B}$ overlap on link B. In addition, each $a_i \in \mathcal{A}$ overlaps all the $b_j \in \mathcal{B}$ for $j < i$ in the part of the ring below the line $[A, B]$, and all $b_j \in \mathcal{B}$ for $j \geq i$ above that line. In addition, c overlaps all the other routes. Thus, we have $2(L-1)+1$ routes that overlap each other and need a different wavelength each, a total of $W = 2L-1$ wavelengths. The maximal load is clearly L.

More formally, number the nodes in the ring starting at an arbitrary node 0, and proceeding clockwise up to node $N-1$. Define

$$a_1 = [0, \tfrac{N}{2}], a_2 = [1, \tfrac{N}{2}+1], ..., a_i = [i-1, \tfrac{N}{2}+i-1], ..., a_L = [L-1, \tfrac{N}{2}+L-1],$$

$$b_1 = [\tfrac{N}{2}, 1], b_2 = [\tfrac{N}{2}+1, 2], ..., b_i = [\tfrac{N}{2}+i-1, i], ..., b_L = [\tfrac{N}{2}+L-1, L],$$

$$c = [\tfrac{N}{2}-1, \tfrac{N}{2}+L-1].$$

Clearly the above arguments hold for this general case definition as well. □

3 Dynamic wavelength allocation, No conversion

In this section we present almost tight results for allocating wavelengths to lightpaths when no wavelength conversion is allowed for the dynamic traffic model. We first demonstrate that not every natural algorithm has good performance in this respect. For example, consider the following circular-first-fit (CFF) algorithm. This algorithm is almost identical to the First-Fit algorithm that we will consider later, except that it tries to allocate a wavelength to a lightpath starting at a different starting point each time, in a circular fashion: for the i^{th} request it checks the wavelengths: $i \bmod W, (i+1) \bmod W, (i+2) \bmod W, ...$ 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 the DWLA algorithm presented below).

Theorem 2 *Given a ring with N nodes, there exists a set of lightpaths with load L, for which Circular-First-Fit needs at least $W = 1 + N(L-1)$ wavelengths to support all the requests.*

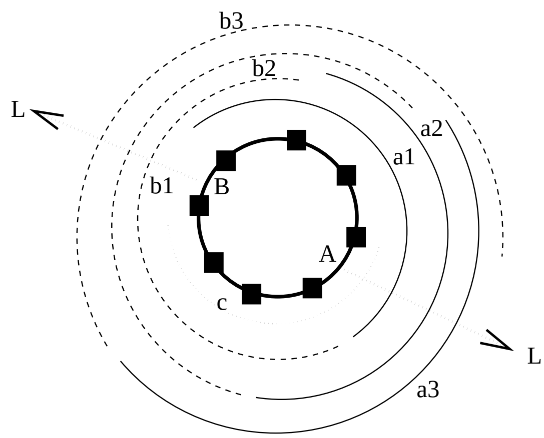


Figure 3: A worst-case set of lightpaths for the static wavelength allocation problem

Proof. Consider the configuration in Figure 4, which is created by single hop lightpath requests which come in rounds. At each round a request comes for a lightpath on link 1, then on link 2, and so on until the N^{th} link. CFF allocated a new wavelength to each request. After a total of W requests, a last request arrives with a route that span the entire ring (depicted as a dashed line in the figure). If $W \leq N(L-1)$ this request is blocked since there is no wavelength that can accommodate it through its entire route. Note that if $W > N(L-1)$ there cannot exist such a last request as the maximum load is violated. $\square$

3.1 Routing on a ring

A necessary step before solving the wavelength allocation problem is to determine the route that each lightpath takes (between the two available choices). 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 shortest path routing yields a maximum load L_{shrt} which is up to twice from the optimal load.

Given any configuration of lightpaths (possibly after deletions of lightpaths) produced using shortest path routing, let L_{shrt} denote the maximum load (L) for this case. Consider a link a with maximum load L_{shrt} depicted in Figure 5. 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 $\frac{L_{\text{shrt}}}{2}$ by changing the routes of some of the requests to the other alternative around the ring.

3.2 Efficient algorithm for rings and trees

Three types of network topologies are considered in this section: the line, ring, and tree. For each of them, the minimum number of wavelengths that suffices to insure no blocking is given when no wavelength conversion is possible.

Definition 1 *Let $W_{line}(N,L)$ (resp., $W_{ring}(N,L)$, and $W_{tree}(N,L)$) denote the number of wavelengths*

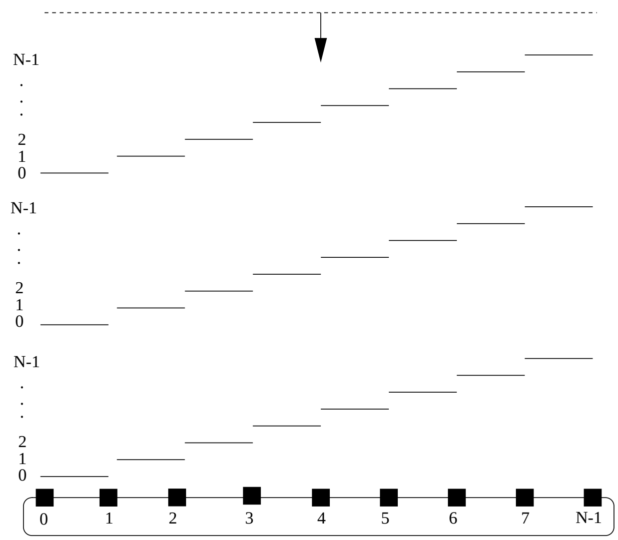


Figure 4: A worst case set of lightpaths for the Circular-First-Fit algorithm. The requests arrive starting from the bottom left request and proceeding in rows to the top right request.

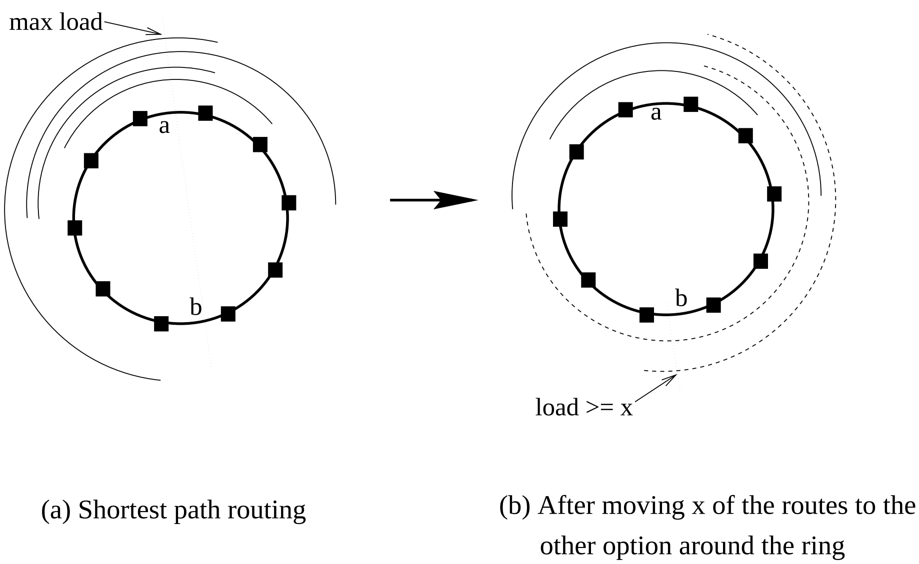


Figure 5: Shortest path routes are not much worse than optimal

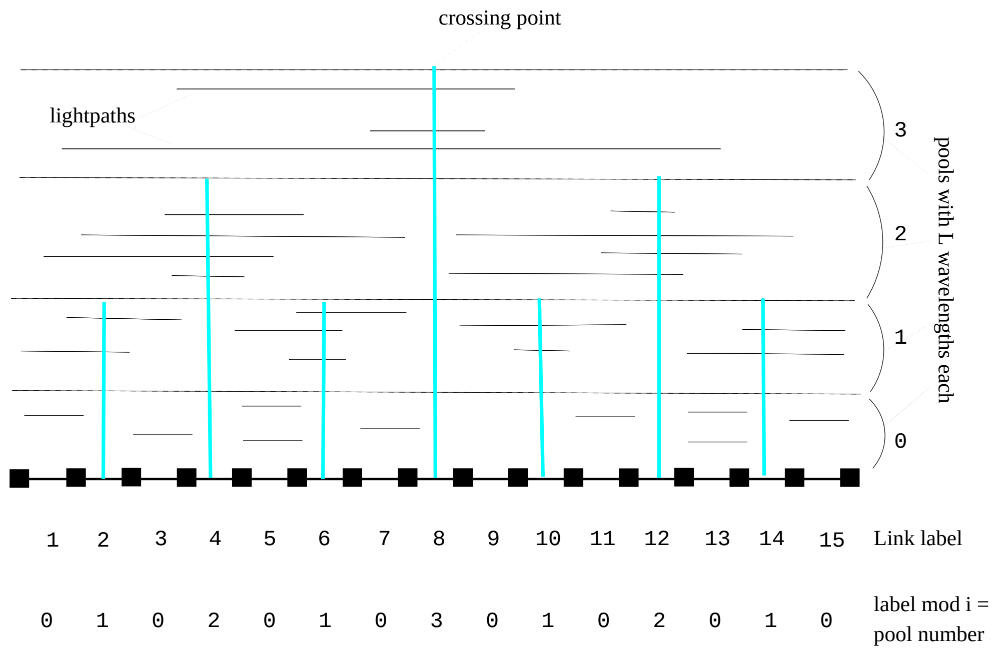


Figure 6: $\log_2 N$ layers of L wavelengths each to accommodate the traffic pattern on a line, as in Figure 7.

required to insure no blocking for any line (resp., ring, and tree) network with at most N nodes and no wavelength conversion, if the load across any link is at most L.

Lemma 1 *If N is even then $W_{line}(N, L) \leq L + W_{line}(N/2, L)$.*

Proof. Note that there is a link e in the network whose removal leaves two line subnetworks X and Y, each with $N/2$ nodes. Let L wavelengths be dedicated to lightpaths that cross e. For these lightpaths, L wavelengths insures no blocking since there can be at most L of them at any time. Dedicate another $W_{line}(N/2, L)$ wavelengths to lightpaths that do not cross e, i.e., those that are entirely in X or Y. This insures no blocking since the subnetworks X and Y each have $N/2$ nodes, and the lightpaths in X can use the same wavelengths as the lightpaths in Y (because they do not intersect at any link). The total number of dedicated wavelengths to insure no blocking for all the lightpaths is $L + W_{line}(N/2, L)$. $\square$

Theorem 3 $W_{line}(N, L) \leq L\lceil \log_2 N \rceil$.

Proof. The theorem follows from Lemma 1 and the fact that $W_{line}(1, L) = 0$. A pictorial example for the classification of lightpaths according to their crossing points is given in Figure 6. $\square$

Theorem 4 $W_{ring}(N, L) \leq L + L\lceil \log_2 N \rceil$.

0. INPUT: receive add/delete lightpaths, one at a time.
   DATA STRUCTURE: Define a set of wavelength pools $\{\textsc{Pool}(i)\}_{i=0}^{\lceil \log_2 N \rceil}$, each pool containing L wavelengths.
1. If the request is to delete a lightpath, delete it and mark the relevant wavelength segment as free. Otherwise:
2. Given a request to add a lightpath x, let the links of the ring be labeled $1, ..., N$. Let i be the maximum value such that x crosses a link labeled y for which $y \bmod 2^i = 0$ (define $y \bmod 1 = 0$ for every y).
3. Find a free segment of a wavelength in pool $\textsc{Pool}(i)$ which can accommodate x, and allocate x on this wavelength.
4. Handle next request.

Figure 7: Dynamic allocation of lightpaths (DWLA)

Proof. Pick a link e in the ring network. Let L wavelengths be dedicated to those lightpaths that cross e. This is enough to insure no blocking for these lightpaths since e can have at most L lightpaths. Dedicate another $L\lceil \log_2 N \rceil$ wavelengths to those lightpaths that do not cross e. These lightpaths can be viewed as ones that are in a line network with N nodes. Theorem 3 implies that these wavelengths are sufficient to insure no blocking. The total number of dedicated wavelengths to insure no blocking for all the lightpaths is $L + L\lceil \log_2 N \rceil$. □

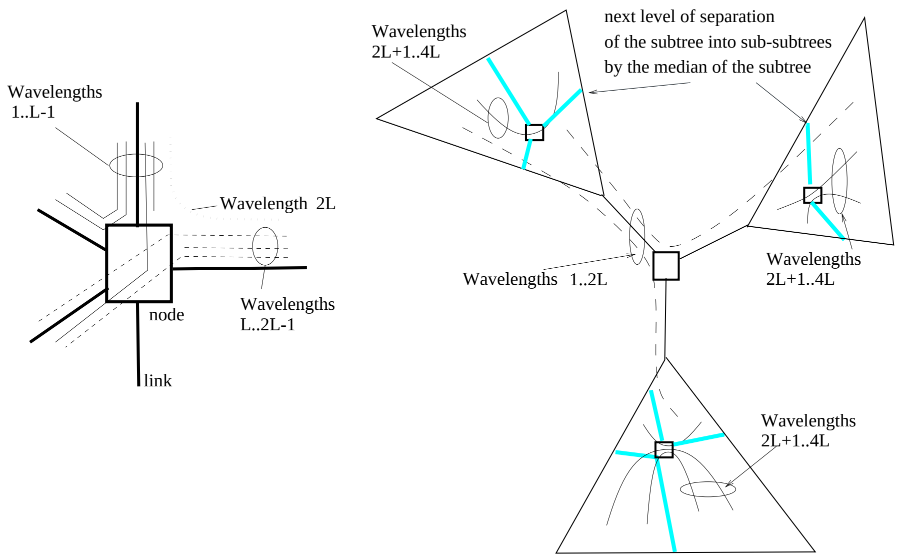
The above theorems imply the Dynamic WaveLength Allocation algorithm (DWLA) described in Figure 7. A similar technique can be used for tree networks as well: find a central node v in the tree, allocate a pool of wavelengths to lightpaths that are routed through v and allocate other pools recursively in the subtrees that do not contain v. However the lightpath configuration at each node may require $2L$ wavelengths as proven in the following lemma (recall that for rings L wavelengths suffice). The recursive application of the scheme may require up to $\log_2 N$ stages (see Figure 8(b)).

Lemma 2 *For* $N \geq 1$, $W_{tree}(N, L) \leq 2L - 1 + W_{tree}(N/2, L)$.

Proof. In every tree there exists a node v, called the *median*, such that its removal leaves a collection of trees $T_1, T_2, ..., T_k$ (for some k) such that each tree has at most $N/2$ nodes [21]. Let $2L-1$ wavelengths be dedicated to those lightpaths that cross v, and let P_v denote those lightpaths that cross v. Note that each lightpath in P_v goes through exactly two links incident to v. Since there can be at most L lightpaths that can traverse any link, a lightpath in P_v can intersect with at most $2L - 2$ other lightpaths in P_v (see Figure 8(a)). Hence, $2L - 1$ wavelengths are enough to insure no blocking of lightpaths in P_v.

Now note that each lightpath that does not cross through v is entirely in one of the trees $T_1, T_2, ..., T_k$ (see Figure 8(b)). Since lightpaths in different trees do not intersect, they may use the same wavelengths. Since each tree has at most $N/2$ nodes, $W_{tree}(N/2, L)$ wavelengths suffices to insure no blocking for lightpaths that do not cross through v.

The total number of wavelengths for all lightpaths to insure no blocking is at most $2L-1 + W_{tree}(N/2, L)$. □



(a) A single tree node (b) the whole tree

Figure 8: Dynamic allocation of wavelengths that go through a single tree node and the global picture of the tree

Theorem 5 $W_{tree}(N, L) \leq (2L-1)\lceil \log_2 N \rceil$.

Proof. The theorem is implied by Lemma 2 and the fact that $W_{tree}(1, L) = 0$. $\square$

3.3 Lower bound on a ring

We now prove that in the worst case $W \geq 0.5L \log_2 N + L$, thereby proving DWLA to be up to twice away from an optimal solution. We start with $L = 2$.

Consider the following scenario, depicted in Figure 9. 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 remains 2 at all times. This process can only be repeated $\log_2 N$ times, since in each phase i, the adversary is forced to issue lightpaths traversing 2^i 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

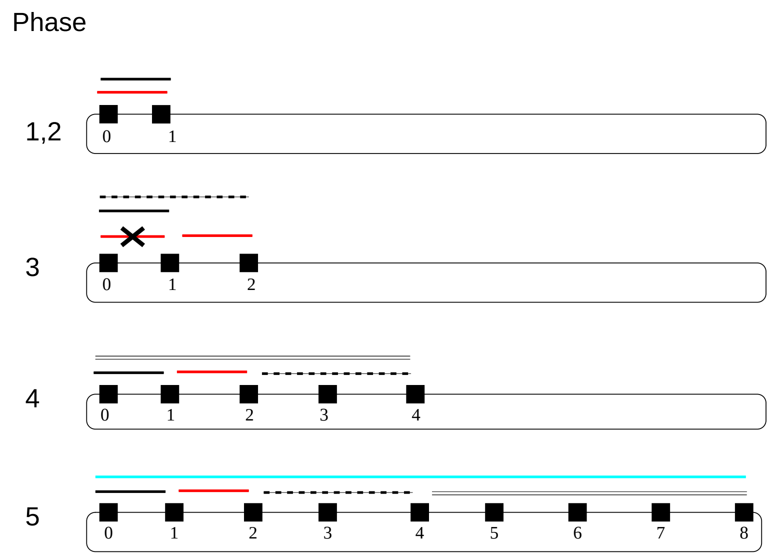


Figure 9: A worst case dynamic scenario of lightpaths

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

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

Theorem 6 *For every wavelength allocation algorithm there exists some addition/deletion scenario that requires the algorithm to use $W > 0.5L\lfloor \log_2 N \rfloor + L$ wavelengths.*

Note that the construction above can be easily modified to work even if there is fixed wavelength conversion at each node, showing that fixed conversion does not help reduce the worst case for dynamic scenarios.

3.4 First-Fit algorithm on a ring

First-Fit is a popular algorithm for assigning a wavelength to a lightpath in a network with no wavelength conversion. It assumes that the wavelengths are labeled $0, 1, ...W-1$, and assigns to a lightpath a wavelength with the lowest label that is available in each link of the lightpath. This algorithm has been studied for the statistical traffic model in [2, 4] and shown to perform well.

Upper and lower bounds on the worst case number of wavelength to insure no blocking for First Fit on a ring are presented in this section.

3.4.1 Upper Bound

Definition 2 *Let $W_{ring}^{FF}(H, L)$ denote the maximum label of a wavelength used by a lightpath of length at most H on any ring network with load at most L and that uses First-Fit.*

Lemma 3 *For $H \geq 1$, $W_{ring}^{FF}(3H, L) \leq 4L - 3 + W_{ring}^{FF}(H, L)$.*

Proof. It will be shown by induction on H that a lightpath p of length $k \leq 3H$ will be assigned a wavelength whose label is at most $4L - 3 + W_{ring}^{FF}(H, L)$.

Consider the case $k \leq H$. Then the lightpath can be assigned to a wavelength at most $W_{ring}^{FF}(H, L)$ by the induction hypothesis. Now consider the case $H < k \leq 3H$. It will be shown that there is a wavelength in the set $W^* = \{W_{ring}^{FF}(H, L) + 1, W_{ring}^{FF}(H, L) + 2, ..., W_{ring}^{FF}(H, L) + 4L - 3\}$ that can be assigned to p. Let $p^* = \{p_1, p_2, ..., p_m\}$ be the set of lightpaths already in the network that intersect p and use wavelengths from the set W^*. Since these lightpaths use wavelengths greater than $W_{ring}^{FF}(H, L)$, their lengths must be greater than H.

Let $e_1, e_2, ..., e_k$ be the sequence of links of the lightpath p. Let $e_1, e_{\lceil k/3 \rceil}, e_{\lceil 2k/3 \rceil}, e_k$ be referred to as the *critical* links for p. Note that each lightpath of p^* has length greater than H, which in turn is at least $k/3$. Thus, each lightpath of p^* intersects at least one of the critical links, see Figure 10. Let $n_1, n_{\lceil k/3 \rceil}, n_{\lceil 2k/3 \rceil}$, and n_k be the number of lightpaths of p^* that cross $e_1, e_{\lceil k/3 \rceil}, e_{\lceil 2k/3 \rceil}$, and e_k, respectively. Since each lightpath of p^* intersects at least one of the critical links, $|p^*| \leq n_1 + n_{\lceil k/3 \rceil} + n_{\lceil 2k/3 \rceil} + n_k \leq 4(L-1)$. Since the number of wavelengths of W^* is $4L - 3$, there must be an available wavelength in W^* for p. Thus, p is assigned a wavelength labeled at most $4L - 3 + W_{ring}^{FF}(H, L)$. □

Theorem 7 $W_{ring}^{FF}(H, L) \leq (4L - 3)\lceil \log_3 H \rceil + L$.

Proof. The theorem follows from Lemma 3, and the fact that $W_{ring}^{FF}(1, L) = L$. □

Corollary 1 *Consider a ring network with N nodes and using the First-Fit algorithm. Then $2.53 \cdot L \log_2 N + 5L$ wavelengths insures no blocking if the load across any link is at most L.*

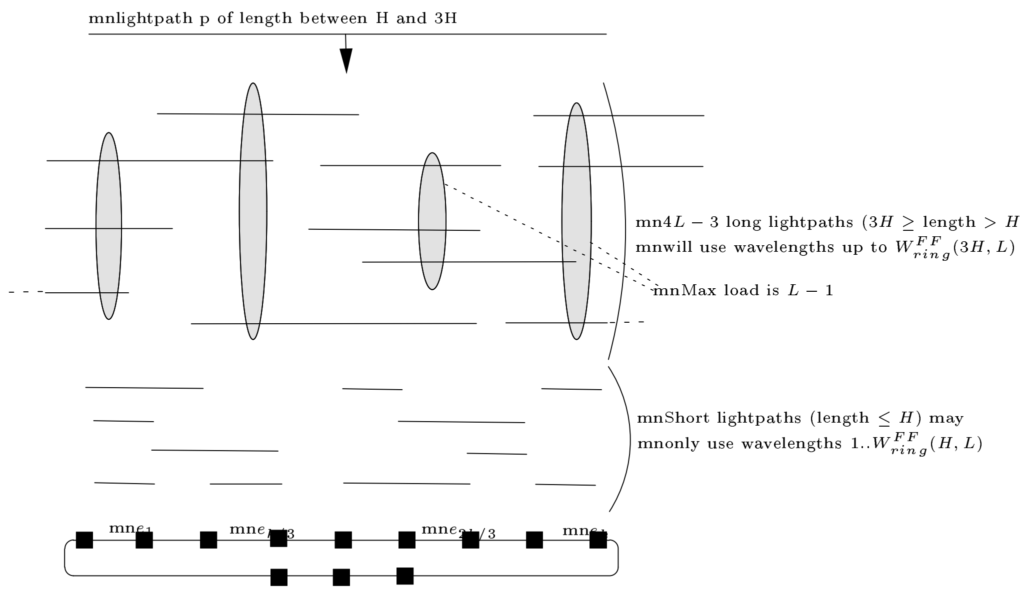


Figure 10: The upper bound argument for First-Fit.

3.4.2 Lower Bound

In Section 3.3 it was proven that any algorithm that assigns wavelengths to lightpaths in an optical ring with N nodes and no wavelength conversion, requires $W \geq 0.5L \log_2 N + L$ wavelengths in the worst case. We now show that if the algorithm is assumed to be First-Fit, a better lower bound on the worst case performance can be proven for reasonable numbers of nodes: $W \geq 0.9L \log_2 N$. The tightening of the bounds is achieved through a denser packing of the overlapping lightpaths which is possible since the chosen wavelengths for lightpaths is known. Consider the following pattern of lightpath add/delete requests, depicted in Figure 11:

Phase 1. A single hop lightpath addition request to connect nodes 0 and 1 arrives and is allocated wavelength 0.

Phase 2. A single hop request to connect nodes 1 and 2 arrives and is allocated wavelength 0 by First-Fit. Another such request arrives and is allocated wavelength 1. Next, the first lightpath between nodes 1 and 2 is deleted. The current configuration is two single hop lightpaths with a load of 1 using two wavelengths.

Phase 3. A two hop lightpath from node 0 to node 2 is requested, and is allocated wavelength 2. Next, the one hop lightpaths are deleted and Phases 1–2 are repeated between nodes 2 and 4.

⋮

Phase i. After phase $i-1$, wavelengths $\{0, 1, ..., i-2\}$ have been used. In order to establish a short[2] lightpath that uses wavelength i it is necessary to rearrange the relative order of lightpaths in the configuration of Phase $i-1$ (the details of the altered schedule of requests is left to the reader). In

[2] As part of the lower bound technique, it is important to have lightpaths that take up as little of the ring as possible since the more phases are applied, the longer the lightpaths become, a fact that limits the number of phases in the lower bound and hence the number of wavelengths that can be proven to be necessary.

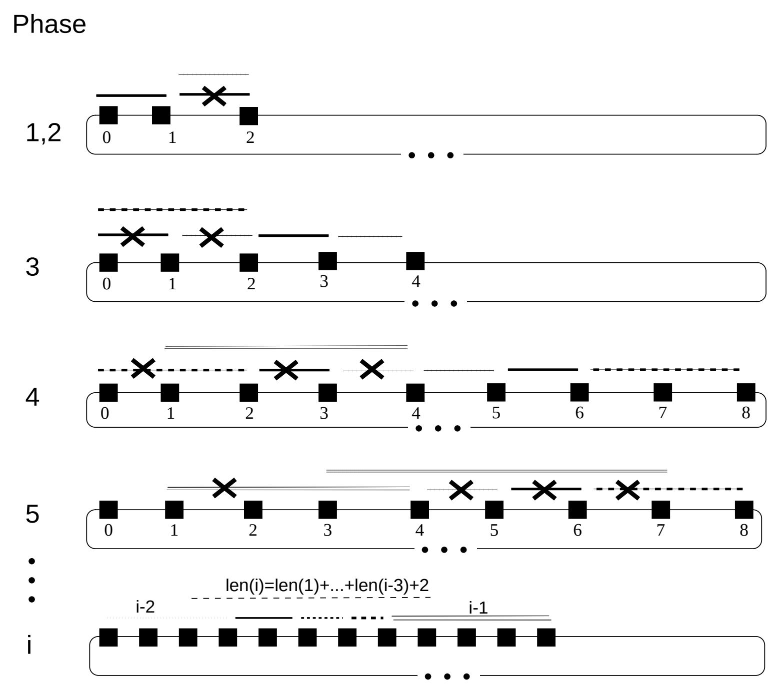


Figure 11: Worst case scenario for First-Fit

the modified configuration, the lightpath which uses wavelength $i-2$ is the leftmost one and the lightpath that uses $i-1$ is the rightmost one. The new request overlaps one link of the leftmost and rightmost lightpaths (and all the intermediate lightpaths). Thus First-Fit allocates wavelength $i-1$ to it.

Theorem 8 *On a ring with N nodes, there exists a sequence of lightpath additions and deletions which require at least $W \geq 0.9L \log_2 N - 1.5L$ wavelengths.*

The proof of the theorem can be found in Appendix A.1.

4 Limited wavelength conversion

In this section we determine the performance of networks with limited wavelength conversion, i.e., the conversion degree of the network is small. We will first consider networks with arbitrary topology, and then networks with a ring topology.

We use the following terminology. Consider a network with wavelengths $\{0, 1, ..., W-1\}$, and channels are numbered according to their wavelengths. For an ordered pair of adjacent links x and y in the network, a bipartite graph $G_{x,y} = (V_x, V_y, E)$ is called its *conversion graph* if $V_x = V_y = \{0, 1, ..., W-1\}$ and for each $(i,j) \in E$, channel i on link x is compatible with channel j on link y. (Note that for a network with full wavelength conversion, the conversion graphs are complete bipartite graphs, while for a network with no wavelength conversion, the conversion graphs have edges $E = \{(i,i) | 0 \leq i < W\}$.) A conversion graph is said to be *symmetric* if $(i,j) \in E$ implies $(j,i) \in E$.

Our performance result (Theorem 9) depends on conversion graphs with particular *expansion* properties, stated next. See also Figure 12(a).

Definition 3 *Consider a bipartite graph (V_1, V_2, E) with each node having at most d incident edges. For each subset of nodes $S \subseteq V_1$, let $\Gamma(S)$ denote the subset of nodes in V_2 that are adjacent to a node in S (i.e., $\Gamma(S) = \{j \in V_2 : \exists i \in S, (i,j) \in E\}$). The graph is called an (α, β, d)-expander, for some $0 < \alpha < \frac{1}{2}$ and $\beta > 1$, if for each subset of nodes $S \subseteq V_1$ such that $|S| \leq \alpha|V_1|$, $|\Gamma(S)| \geq \beta|S|$.*

Lemma 4 *[22] There is a triple (α, β, d), where $0 < \alpha < \frac{1}{2}$ and $\beta > 1$, such that for each n that is sufficiently large, there is a symmetric (α, β, d)-expander with n nodes.*

By having an expander with proper expansion properties as the conversion graph at each node, it can be guaranteed that a lightpath will not be blocked. The wavelength allocation process for a given lightpath starts at one of its end-points at which there is a a set of free wavelengths for the lightpath, and attempts to extend the wavelength allocation for the lightpath one hop at a time. The expanders insure that the set of free wavelengths which may be used by the lightpath, does not decrease below some minimum — see Figure 12(b).

Lemma 5 *Consider a network with W wavelengths per link, such that the conversion graph for every ordered pair of adjacent links is an (α, β, d)-expander, where $0 < \alpha < \frac{1}{2}$ and $\beta > 1$. The network does not block any lightpath as long as the load is at most δW, where $\delta = \min\{\alpha(\beta - 1), 1 - \alpha\}$.*

Proof. Consider setting up a lightpath p in the network, and suppose the lightpath traverses the following sequence of links $(e_1, e_2, ..., e_k)$ for some k. Note that the WDM channels assigned to the lightpath must be compatible from link to link.

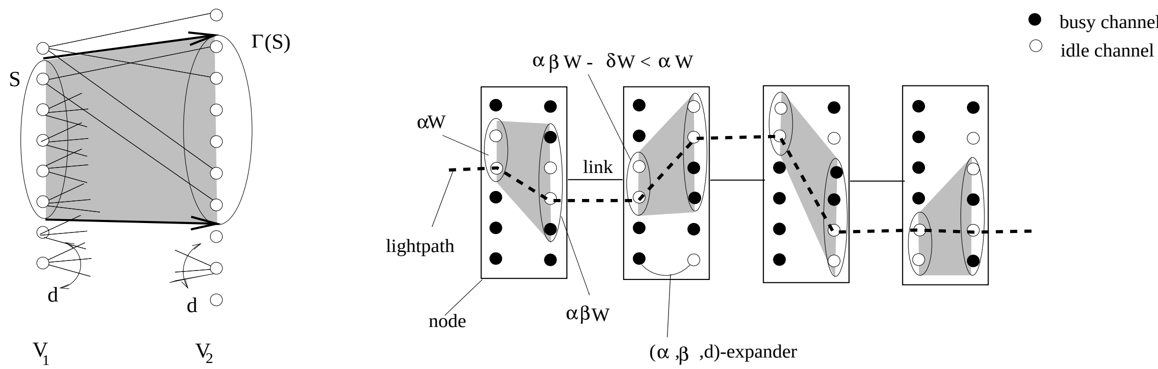
For $i = 1, 2, ..., k$, a WDM channel on link e_i is referred to as being *busy* if it is being used by a lightpath. For $i = 2, 3, ..., k$, a channel on link e_i is also referred to as being *busy* if all the channels on link e_{i-1} it is compatible with are busy. Note that if there is an *idle* (i.e., not busy) channel on link e_k then the lightpath p may be set up.

Next, it will be shown by induction that for $i = 1, 2, ..., k$, there are at least αW idle channels on link e_i. For $i = 1$ this is true since there are at most $\delta W \leq (1 - \alpha)W$ lightpaths through e_1. For $i > 1$, suppose there is a set of αW idle channels on link e_{i-1}. Since the conversion graph for the ordered pair (e_{i-1}, e_i) is an (α, β, d)-expander, there is a set of $\beta\alpha W$ channels on link e_i such that each is compatible with at least one of the αW idle channels on link e_{i-1}. Note that each of the $\beta\alpha W$ channels on link e_i may be busy only if there is a lightpath using it. Since there can be at most $\delta W \leq (\beta - 1)\alpha W$ lightpaths through e_i, there must be at least αW idle channels in link e_i. Thus, there are at least αW idle channels on link e_k, so lightpath p may be set up. □

Theorem 9 *There is a fraction $\delta > 0$ and integer $d > 0$ such that for any network with sufficiently large number of wavelengths W, the network can have conversion degree d and insure no blocking of lightpaths if the traffic load is at most δW.*

Proof. The theorem follows directly from Lemma 4 and Lemma 5. □

For the rest of this section, we focus our attention on ring networks. The next Theorem 10 is for a ring network with conversion degree 2. For this case we present a construction which deploys two different mechanisms. For short lightpaths (lightpaths that traverse a small number of hops) we use the technique from Section 3.2 and do not convert wavelengths. For long lightpaths we split the ring into shorter sections and gradually convert each lightpath at the beginning of each section so that it will use a free wavelength at the end of the section. This result shows that the number of wavelengths can



(a) A Single expander

(b) A lightpath

Figure 12: Using an expander for non-blocking networks.

be decoupled from the size of the network (while in the no-conversion case the number of wavelengths depended on the network size).

Theorem 10 *Let L and N be arbitrary positive integers.*

1. *Let $W = L\lceil \log_2 L \rceil + 4L$. There is a ring network with N nodes, W wavelengths, and conversion degree 2 that that does not block any lightpaths as long as the load is at most L.*

2. *Similarly, there exists such a ring with $W = L\lceil \log_2 \log_2 L \rceil + 4L$.*

The proof of the theorem can be found in Appendix A.2. The subsequent Theorem 11 is for a ring network with conversion degree $d > 1$ and where lightpaths are set up but never taken down (termed the *incremental* model). This model is suitable for networks with growing demands, and with almost no requirements for removing lightpaths which are already in use.

Theorem 11 *Consider an incremental traffic model where lightpaths may be set up but never taken down. Let L and d be integers that satisfy $L \geq 1$ and $d > 1$. Then there is a ring network with conversion degree d and $\max\{0, L-d\} + L$ wavelengths that does not block any lightpaths as long as the load is at most L.*

The theorem can be proven by modifying the results for the case of no wavelength conversion in [12]. We provide an outline of the modifications in the Appendix A.3.

5 Conclusions

In this paper we analyzed the worst-case performance of wavelength allocation schemes for linear and ring topologies. The worst-case model we used determines the maximum traffic load that can be supported without any blocking, given the number of wavelengths available. We showed that the common first-fit algorithm, which does well in simulations that allow blocking, is also quite good in the worst-case, requiring at most $2.53L\log_2 N + 5L$ wavelengths in a ring network without wavelength conversion. A better algorithm for this case uses only $L\log_2 N$ wavelengths. The latter scheme was proven to be up to twice away from the best possible solution.

We also demonstrated that limited wavelength conversion can increase the utilization of WDM channels. Our results show that the number of wavelengths needed to insure no blocking is independent of the number of nodes. This presents an improvement over the case of no wavelength conversion, in which the number of wavelengths grows logarithmically with the network size.

For the incremental case, the number of required wavelengths is much lower than that for the fully dynamic case. Very limited conversion of conversion degree 2 enables to achieve 33% decrease in the number of wavelengths that are needed to support a given load, and the number of wavelengths linearly decreases (at least) with increased conversion degree.

Figure 13 plots the maximum supported load under different scenarios for $W = 32, N = 16$. Loads achieved by our algorithms, as well as known upper bounds on the loads are plotted. Note that, in contrast to the perspective taken by the rest of the paper and summarized in Figure 2, here the system (namely, N and W) is fixed and the maximum supported load is plotted against the amount of wavelength conversion.

The following are our main conclusions:

- If worst-case guarantees are required, the system has to be significantly over-designed. Wavelength conversion helps to reduce this phenomenon.
- The lightpath arrival process plays a crucial role in determining the number of wavelengths required: If the lightpaths are known in advance, the system need not be significantly over-designed; If lightpaths arrive but are not deleted some more wavelengths need to be allocated. However, if fully dynamic scenarios need to be taken into account, the number of wavelengths to guarantee no blocking needs to be very large.
- In the same context, it has been noted in [14] that very limited wavelength conversion helps a lot for the static case. In other words, rearranging existing lightpaths to accommodate new ones will enable the system to support much higher loads, with limited conversion. Our current algorithms for more dynamic cases require more conversion capabilities, but there is still much room for designing better wavelength assignment algorithms, particularly with limited wavelength conversion.

A Appendix

A.1 Proof Of Theorem 8

To prove the theorem, we first determine the length of a lightpath added in Phase i.

Lemma 6 *Let $len(i)$ denote the length of the lightpath added at Phase i (the one that uses the i^{th} wavelength). Then $len(1) = 1, len(2) = 1, len(3) = 2$, and for $i > 3$,*

$$len(i) = 2 + \sum_{j=1}^{i-3} len(j) \quad .$$

Proof. For $i = 1, 2$, and 3, the lemma is easy to check. For $i > 3$, the lightpath of the lemma spans all the lightpaths from phases 1 to $i - 3$, which are adjacent and non-overlapping. In addition it shares one link with the lightpath from phase $i - 2$ and one link from the lightpath of phase $i - 1$. □

It is interesting to note the following similarity between $len(i)$ and Fibonacci series:

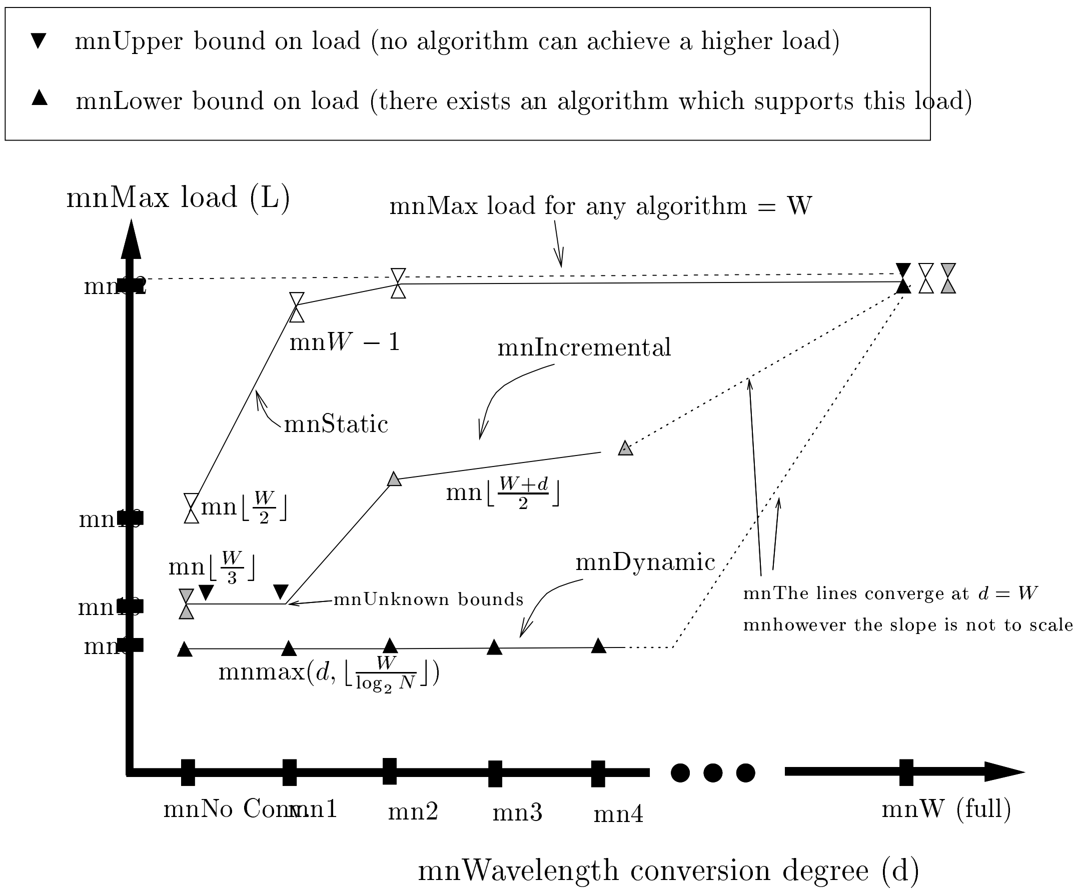


Figure 13: Maximum supported load for a ring network with $W = 32$ wavelengths and $N = 16$ nodes. White triangles indicate results for the static model, gray for the incremental model, and black for the dynamic model.

Lemma 7

$$len(i) = \begin{cases} 1, & \textit{if } i = 1, 2, \\ 2, & \textit{if } i = 3, \\ len(i-1) + len(i-3), & \textit{otherwise.} \end{cases}$$

Proof. It follows from Lemma 6 that for $i > 3$,

$$len(i) - len(i-1) = (2 + \sum_{j=1}^{i-3} len(j)) - (2 + \sum_{j=1}^{i-4} len(j)) = len(i-3) \quad .$$

□

Lemma 8 *An upper bound for $len(i)$ is: $len(i) \leq 1.465575^i$.*

Proof. The proof is by induction. For $i = 1, 2$, and 3, by numerical calculation it can be shown that $len(i) \leq (1.465575)^i$. Now consider the case when $i > 3$, and suppose that for all $j < i$, $len(j) \leq (1.465575)^j$. From Lemma 7, $len(i) = len(i-1) + len(i-3)$. Thus, $len(i) \leq (1.465575)^{i-1} + (1.465575)^{i-3}$ $= (1.465575)^{i-3}((1.465575)^2 + 1) \leq (1.465575)^{i-3}(1.465575)^3 = (1.465575)^i$. Thus, the lemma is true. □

Lemma 9 *Let $N(i)$ denote the number of links in the segment of the ring used by i phases of the above described lightpath pattern. Then*

$$N(i) = \sum_{j=1}^{i} len(j) = len(i+3) - 2 \quad .$$

Proof. The proof follows from the lightpath pattern and Lemma 6. □

To finish the proof of the theorem, by Lemma 8 and Lemma 9 we get $N(i) \leq 1.465575^{i+3}$. Since it is possible to apply Phase i only if $N \geq N(i)$ we get $\log_{1.465575} N \geq i + 3$ or $i \leq \log_{1.465575} N - 3 = \frac{\log_2 N}{\log_2 1.465575} - 3 < 1.81335 \log_2 N - 3$. Since the above scenario has a maximum load of 2, it is possible to duplicate it $L/2$ times and require $iL/2$ wavelengths. Thus the number of necessary wavelengths is $W \geq \frac{1.81335}{2} L \log_2 N - 1.5L$.

A.2 Proof Of Theorem 10

To prove the first part of the theorem we define a ring network with N nodes, $L\lceil\log_2 L\rceil + 4L$ wavelengths, and conversion degree 2 that insures no blocking as long as the maximum load is at most L. We may assume that $N \geq L$, otherwise, Theorem 4 implies Theorem 10.

The ring has nodes numbered $0, 1, ..., N-1$ going clockwise, and for $i = 0, 1, ..., N-1$, the link between nodes i and $(i+1) \bmod N$ is numbered i. The ring is partitioned into segments, each having at least L links but less than $2L$ links. (Note that such a partitioning is always possible.) Now, $L\lceil\log_2 L\rceil + L$ wavelengths are dedicated to lightpaths that do not cross segments (termed *local* lightpaths), and these wavelengths do not have wavelength conversion. Since the lightpaths are confined to segments with less $2L$ links, Theorem 3 implies that there will be no blocking of these lightpaths.

The other $3L$ wavelengths are dedicated to supporting lightpaths that cross segments, i.e., *inter-segment* lightpaths. These pools deploy wavelength conversion. The idea behind the pools is to provide the equivalent of a non-blocking switching network in each segment. When an inter-segment lightpath is considered, it is allocated wavelengths in each segment separately, starting from the first clockwise segment in its path and ending at the last one. Focusing on some intermediate segment through which the lightpath is routed, the lightpath comes into the segment using whatever wavelength x was allocated to it in the previous segment. Since the load does not exceed L, there is a free wavelength y among the $3L$ wavelengths[3] of the pool at the other end of the segment. In order to get from x to y it is necessary to switch the lightpath in a non-blocking manner.

More formally, the following graph G^* is used to describe how the WDM channels are compatible. In G^*, WDM channels are compatible if they are incident to a common vertex[4]. It is straightforward to check that this ring network has conversion degree 2.

- **The vertices of G^*:** There are N stages of vertices where each stage has $2L$ vertices, and stage i represents node i in the ring network. In each stage i, there are two types of vertices called *u-vertices* and *v-vertices* and are labeled $\{u_0(i), u_1(i), ..., u_{L-1}(i)\}$ and $\{v_0(i), v_1(i), ..., v_{L-1}(i)\}$ respectively.
- **The edges of G^*:** For $i = 0, 1, ..., N-1$, there are $3L$ edges between stage i and stage $(i+1) \bmod N$ vertices corresponding to the $3L$ WDM channels on link i. The $3L$ channels are of three types: *shift channels*, *u-channels*, and *v-channels*, where there are L of each type. The enlarged part of Figure 14 shows a subgraph of G^* corresponding to a single segment of the ring network between nodes m and n. For $i = m, m+1, ..., n-1$, the channels between the stage i and stage $i+1$ vertices (i.e., the channels of link i) are as follows:
  - Each u-vertex $u_j(i)$ (for $0 \le j < L$) has a u-channel between it and $u_j(i+1)$. It also has a shift channel between it and
  $$\begin{cases} u_{(j+1) \bmod L}(i+1), & \text{if } i < n-1 \\ v_j(i+1), & \text{if } i = n-1 \end{cases}$$
  - Each v-vertex $v_j(i)$ (for $0 \le j < L$) has a v-channel between it and
  $$\begin{cases} v_j(i+1), & \text{if } i < n-1 \\ u_j(i+1), & \text{if } i = n-1 \end{cases}$$

Next, we describe how inter-segment lightpaths are assigned channels, and begin by introducing some terminology. Consider a segment of the ring network and its corresponding subgraph of G^* as shown in Figure 14. First, the u-vertices in stages m and n will be referred as *joining vertices* or *J-vertices* because they "join" segments together. The J-vertices in stages m and n are called the *left* and *right* J-vertices, respectively. A J-vertex is *busy* if it has a lightpath going through it, and is *idle* otherwise. Second, note that u-channels form L paths of channels crossing the segment, and these will be called *u-paths*. Similarly, the v-channels form L paths, and these paths will be referred to as *v-paths*. A u-path or v-path is *busy* if there is a lightpath on it, and *idle* otherwise. An idle u-path or v-path is called *available* if the right J-vertex that it is connected to is also idle.

Inter-segment lightpaths are assigned to channels as follows. The allocation for a lightpath is done segment by segment, starting from one end of the lightpath and going clockwise around the ring network.

[3] In fact, only L of the $3L$ wavelengths are needed for this purpose. The other $2L$ wavelengths are needed for some technical reason explained later.

[4] Recall the definition of a conversion graph in the beginning of Section 4. The current graph G^* is the result of a concatenation of the conversion graphs of all the nodes in the ring. This concatenation is done by uniting each vertex j in the right set of vertices (V_y) at node i with vertex j in the left set of vertices (V_x) at node $i+1$.

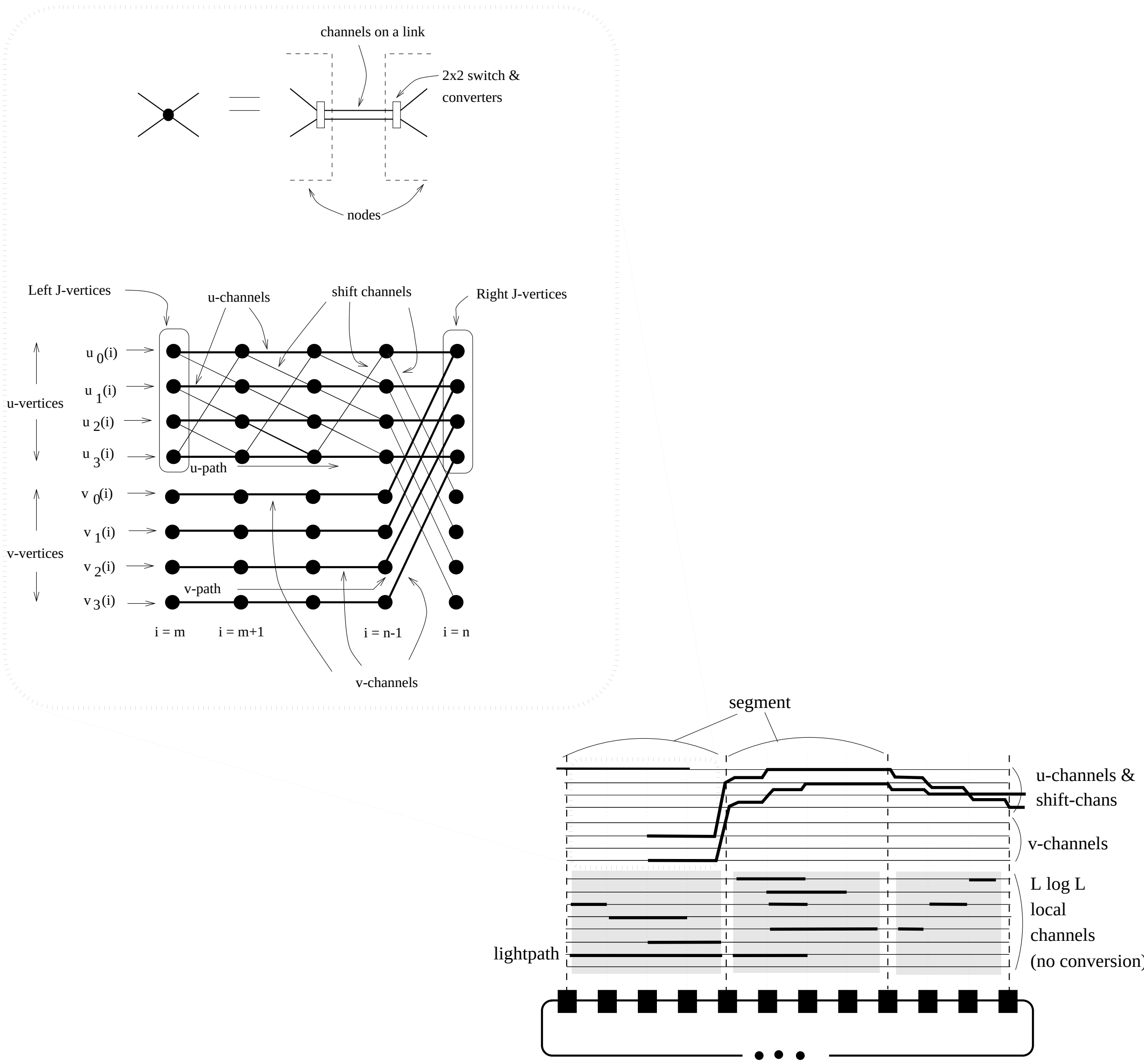


Figure 14: lightpath configuration in the proposed network and an enlarged subgraph of G^* corresponding to the segment of the ring with links $\{m, m+1, ..., n-1\}$, and $L = 4$. The u-channels and v-channels are shown as thick lines, while the shift channels are shown as thin lines.

In the first segment, the lightpath follows an available v-path to the corresponding idle (right) J-vertex. In an intermediate segment, the lightpath starts from an idle (left) J-vertex, follows shift channels until it reaches an available u-path, and then follows the u-path to an idle (right) J-vertex. In the final segment, the lightpath starts from an idle (left) J-vertex and follows shift channels.

We now argue that the allocation works, i.e., a lightpath will not be assigned channels already used by existing lightpaths. First note that in a segment, if a lightpath follows a u-path or v-path then it goes through a right J-vertex. Thus, the existence of an idle (right) J-vertex implies there is an available u-path and v-path. Since the load is L and there are L right J-vertices, if a lightpath is to be set up through a right J-vertex then there is at least one idle right J-vertex. Therefore, there is at least one available u-path and v-path.

Now consider a lightpath p that is about to be assigned channels, and consider the segments it traverses. In its first segment, an available v-path can be found for p. In an intermediate segment, p starts from an idle left J-vertex, and can proceed along shift channels without overlapping an existing lightpath. This is possible since all the existing lightpaths that start from the left J-vertices, first follow the shift channels before going along a u-path. Since the segment has at least $L-1$ links, the lightpath p can reach all u-paths by just following shift channels. In particular, it will be able to reach an available u-path. In the final segment, p starts from an idle left J-vertex, and so it can follow a sequence of shift channels without overlapping an existing lightpath. Therefore, the lightpath will not overlap with an existing one.

Note that the shift- and u-channels in each segment form a wide-sense non-blocking cross-connect function that enables to convert the wavelength of a lightpath coming into the segment (on one of the left J-vertices), to any other wavelength at the output of the segment (right J-vertices). This is done by a simple "matrix" cross-connect and requires L stages. As a result the size of each segment is L hops. If instead a different wide-sense non-blocking network is used, say [23], in which only $\lceil log_2 L \rceil^2$ stages are necessary[5], it is possible to reduce the segment size to $\lceil log_2 L \rceil^2$. It follows that $W \leq L\lceil log_2(2\lceil log_2 L\rceil^2)\rceil + 3L \leq 2L\lceil \log_2 \log_2 L\rceil + 4L$ which is significantly lower for large values of L. This completes the proof of the second half of the theorem.

A.3 Proof Of Theorem 11

We will discuss how the results of [12] can be modified to prove Theorem 11. The ring network in the theorem has W WDM channels per link, where $W = \max\{0, L-d\} + L$ and L is the maximum expected load of the lightpaths. The channels are numbered $\{0, 1, ..., W-1\}$.

The Incremental WaveLength Allocation algorithm (IWLA) is a modification of the algorithm $COLOR$ in [12] — which solves the problem for the no-conversion case. The algorithm assigns incoming lightpaths to sets called *shelves*. If $d \geq L$ then there is only one shelf and it is numbered 0. If $d < L$ then there are $L-d+1$ shelves and they are numbered $0, 1, ..., L-d+1$. In what follows $\textsc{Shelf}(i)$ denotes the collection of lightpaths that have been assigned to shelf i. Also, for a lightpath p, $L(p/S)$ denotes the maximum load experienced on links along p by lightpaths in some set S. In other words, it is the value $\max_{e \in p} L(e/S)$, where $L(e/S)$ denotes the number of lightpaths in S that traverse e. The pseudo-code for IWLA is shown in Figure 15. The crux of the algorithm is in Step 4, in which IWLA chooses in which shelf i to place a given lightpath based on the load the lightpath experiences ignoring the lightpaths in shelves above i.

Using arguments similar to [12] we shall show that IWLA will assign each lightpath to some shelf. We shall also show that IWLA insures that the load of the lightpaths in $\textsc{Shelf}(0)$ is at most d and the load in every other shelf is at most 2. Thus, d channels of $\textsc{Pool}(0)$ and 2 channels in the other pools

[5]The switching network in [23] requires $8\lceil log_2 L\rceil^2$ stages, but it is claimed that the factor of 8 can be eliminated.

0. INPUT: A sequence of lightpaths $p_0, p_1, ...$ such that they have load at most L.
   DATA STRUCTURE: A collection $\{\textsc{Shelf}(i)\}_{i=0}^{L-d+1}$, where for $i \geq 0$, $\textsc{Shelf}(i)$ is a set of lightpaths.
   A collection of pools $\{\textsc{Pool}(i)\}_{i=0}^{L-d+1}$ where $\textsc{Pool}(0)$ contains wavelengths $0, ..., d$ and each other pool contains a disjoint pair of consecutive wavelengths.
   (The lightpaths in $\textsc{Shelf}(i)$ will be accommodated by $\textsc{Pool}(i)$.)
1. For each $i \geq 0$, set $\textsc{Shelf}(i) = \emptyset$
2. Upon arrival of the next lightpath request whose route is p do:
3. Set $i = 0$
4. While $L(p/\textsc{Shelf}(0) \cup ... \cup \textsc{Shelf}(i)) > i + d$, Set $i = i + 1$
5. Set $\textsc{Shelf}(i) = \textsc{Shelf}(i) \cup \{p\}$
6. Accommodate the request using wavelengths in $\textsc{Pool}(i)$.

Figure 15: Incremental allocation of lightpath requests (IWLA)

will support the lightpaths, provided that the channels in each pool are compatible. In what follows we provide the necessary modifications to prove these points.

Definition 4 *Given a configuration of requests up to some given time T, let the lightpath requests be numbered according to the order of their arrival $p_1, p_2, ..., p_k$. Let $F_i = \{p_1, ..., p_{i-1}\}$ denote the lightpaths which arrived before p_i and let $T_i = \cup_{j=0}^{i} \textsc{Shelf}(j)$ denote the set of lightpaths in selves 0 to i at the time T.*

The next lemma states that when a lightpath is put in a shelf i, the maximum load it experiences in shelves $0, ..., i$ does not occur in segments where shelf i populates another lightpath.

Lemma 10 *If for some $x < y$ and $i > 0$, $p_x \cap p_y \neq \emptyset$ and $p_x, p_y \in \textsc{Shelf}(i)$ then $L(p_x \cap p_y / F_y \cap T_{i-1}) \leq i + d - 1$.*

Proof. Assume by contradiction that $L(p_x \cap p_y / F_y \cap T_{i-1}) > i + d - 1$. Then, since $p_x \in \textsc{Shelf}(i)$, when p_y arrives $L(p_x \cap p_y / F_y \cap T_i) > i + d$ and p_y will be placed in a higher shelf — contradiction. □

We now show that a pair of lightpaths in the same shelf $i > 0$ cannot fully contain each other.

Lemma 11 *[12] For each $i > 0$, if $p_x, p_y \in \textsc{Shelf}(i)$ and $x < y$ then $p_x \not\subseteq p_y$ and $p_y \not\subseteq p_x$.*

Proof. If, by contradiction, $p_y \subseteq p_x$ then $L(p_y / F_y \cap T_{i-1}) \leq i + d - 1$ and IWLA would have placed p_y in a shelf below i. If $p_x \subseteq p_y$ then when p_y arrives it experiences a load of $i + d + 1$ along p_x and is placed by the algorithm in shelf above i. In either case p_y would not have been placed in shelf i — a contradiction. □

Lemma 12 *The maximum number of lightpaths that belong to $\textsc{Shelf}(0)$ and overlap is d. The maximum number of lightpaths that belong to $\textsc{Shelf}(i)$, $i > 0$ and overlap is 2.*

Proof. The first part is obvious: In Step 4 IWLA assigns lightpaths to $\textsc{Shelf}(0)$ as long as the load a new lightpath experiences does not exceed d. As to the second part: if the load at some shelf is above 2, there exist at least three lightpaths p_x, p_y and p_z which overlap at this point. Sort these lightpaths by their starting point: $p_x = (s_1, ..., e_1), p_y = (s_2, ..., e_2)$, and $p_z = (s_3, ..., e_3)$ where $s_1 \le s_2 \le s_3$. Since none of them is contained in the other by Lemma 11, $e_1 \le e_2 \le e_3$.

By Lemma 10, $L(p_x \cap p_y / F_{\max(x,y)} \cap T_{i-1}) \le i+d-1$ and $L(p_z \cap p_y / F_{\max(z,y)} \cap T_{i-1}) \le i+d-1$. Since $p_y = (p_x \cap p_y) \cup (p_z \cap p_y)$ and since the load only grows as more lightpaths are added $L(p_y/F_y \cap T_{i-1}) \le i + d - 1$ and p_y would have been placed in a lower shelf than $\textsc{Shelf}(i)$. □

Lemma 13 *The maximum shelf used by IWLA is* $\textsc{Shelf}(L - d)$.

Proof. In order for an lightpath p_x to be placed in a higher shelf, $\textsc{Shelf}(L - d + 1)$, the load it experienced should be $L(p_x/F_x \cap T_{(L-d+1)-1}) > ((L-d+1)-1)+d = L$, contradicting the maximality of L. □

By Lemma 12 it follows that the pools of wavelengths can support the corresponding shelves. By Lemma 13 it follows that the algorithm will always find a shelf for a new lightpath. Thus the algorithm is correct. The number of wavelengths consumed by IWLA is clearly $d + 2(L - d) = 2L - d$ if $L > d$ and a single shelf if $L \le d$. This completes the proof of Theorem 11.

References

[1] R. Barry and P. Humblet, "Models of blocking probability in all-optical networks with and without wavelength changers," *IEEE JSAC / IEEE/OSA JLT: Special Issue on Optical Networks*, vol. 14, no. 5, pp. 858–867, 1996.

[2] 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.

[3] J. Yates, J. Lacey, D. Everitt, and M. Summerfield, "Limited-range wavelength translation in all-optical networks," in *IEEE Infocom'96*, pp. 954–961, 1996.

[4] A. Birman, "Computing approximate blocking probabilities for a class of optical networks," *IEEE JSAC / IEEE/OSA JLT: Special Issue on Optical Networks*, vol. 14, no. 5, pp. 852–857, 1996.

[5] 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.

[6] 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.

[7] 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.

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

[9] M. Mihail, C. Kaklamanis, and S. Rao, "Efficient access to optical bandwidth," in *36th Symp. on Foundations of Computer Science*, pp. 548–557, 1995.

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

[11] 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.

[12] M. Ślusarek, "Optimal online coloring of circular arc graphs," *Informatique Theoretique et Applications*, vol. 29, no. 5, pp. 423–429, 1995.

[13] H. Kierstead and W. Trotter, "An extremal problem in recursive combinatorics," *Congressus Numerantium*, vol. 33, pp. 143–153, 1981.

[14] R. Ramaswami and G. Sasaki, "Multiwavelength optical networks with limited wavelength conversion," in *IEEE Infocom'97*, 1997.

[15] 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.

[16] 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.

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

[18] 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.

[19] 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.

[20] R. Vachani, A. Shulman, P. Kubat, and J. Ward, "Multicommodity flows in ring networks," *INFORMS Journal of Computing*, vol. 8, no. 3, pp. 235–242, 1996.

[21] B. Zelinka, "Medians and peripherians on trees," *Arch. Math. (Brno)*, pp. 87–95, 1969.

[22] S. Arora, T. Leighton, and B. Maggs, "On-line algorithms for path selection in a non-blocking network," in *ACM Symp. on Theory of Computing*, pp. 149–158, 1990.

[23] D. Cantor, "On non-blocking switching networks," *Networks*, vol. 1, pp. 367–377, 1972.