

# Algorithms for Allocating Wavelength Converters in All-Optical Networks

Gaoxi Xiao and Yiu-Wing Leung, *Senior Member, IEEE*

**Abstract**—In an all-optical wide area network, some network nodes may handle heavier volumes of traffic. It is desirable to allocate more full-range wavelength converters (FWC's) to these nodes, so that the FWC's can be fully utilized to resolve wavelength conflict. In this paper, we propose a set of algorithms for allocating FWC's in all-optical networks. We adopt the simulation-based optimization approach, in which we collect utilization statistics of FWC's from computer simulations and then perform optimization to allocate the FWC's. Therefore, our algorithms are widely applicable and they are not restricted to any particular model or assumption. We have conducted extensive computer simulations on regular and irregular networks under both uniform and nonuniform traffic. Compared with the best existing allocation, the results show that our algorithms can significantly reduce: 1) the overall blocking probability (i.e., better mean quality of service) and 2) the maximum of the blocking probabilities experienced at all the source nodes (i.e., better fairness). Equivalently, for a given performance requirement on blocking probability, our algorithms can significantly reduce the number of FWC's required.

**Index Terms**—All-optical WDM networks, simulation-based optimization, wavelength converter.

## I. INTRODUCTION

WAVELENGTH division multiplexing (WDM) divides the bandwidth of an optical fiber into multiple wavelength channels, so that multiple users can transmit at distinct wavelengths through the same fiber concurrently [1]–[3]. In all-optical WDM networks, the information remains in optical form throughout the network, so that the electronic bottleneck can be avoided.

In an all-optical wide area network (WAN), a source-to-destination path usually consists of multiple hops. If a transmission can occupy the same wavelength on every hop, it can remain in optical form within the network. Otherwise, it encounters wavelength conflict and it has to be blocked. To reduce the blocking probability, we can equip the network nodes with *wavelength converters* (WC's) [4] to resolve wavelength conflict. Specifically, when a transmission encounters a wavelength conflict on a hop, we can use a WC to convert its wavelength to another one, so that it can remain in optical form on this hop.

WC's can be distinguished into two types: 1) a *full-range wavelength converter* (FWC) [4]–[10] can convert an incoming wavelength to any outgoing wavelength and 2) a *limited-range wavelength converter* [11]–[13] can convert an incoming wavelength to a subset of the outgoing wavelengths. When the number of FWC's in a node is equal to the total number of outgoing wavelength channels of this node (which is equal to the number of outgoing fibers times the number of wavelength channels per fiber), FWC's are always available when they are needed. We call this scenario a *complete wavelength conversion*. Some previous studies have shown that if every node can provide complete wavelength conversion, the blocking probability can be significantly reduced [5], [6]. Since WC's are being prototyped in research laboratories and are still costly (e.g., see [4]), the cost of complete wavelength conversion is high. It may be more cost-effective to use a fewer number of FWC's; this scenario is called *partial wavelength conversion*. Given a limited number of FWC's, it is necessary to allocate these FWC's to the node. To demonstrate the effects of partial wavelength conversion, two different allocations of FWC's have been studied in the literature [7], [8].

- Subramaniam *et al.* [7] considered the following allocation of FWC's for analytical tractability: some randomly selected nodes are equipped with sufficient number of FWC's to support complete wavelength conversion, and the remaining nodes are not equipped with any FWC. Compared with complete wavelength conversion, this allocation can give nearly the same blocking probability when the number of nodes with complete wavelength conversion is large enough.
- Lee and Li [8] considered the following allocation for performance study: every node is equipped with the same and limited number of FWC's. Compared with complete wavelength conversion, this allocation can give nearly the same blocking probability when the number of FWC's per node is large enough. To the best of our knowledge, this allocation requires the smallest number of FWC's to achieve a given blocking probability.

Although the studies in [7] and [8] were not aimed to optimize the allocation of FWC's, they reached the important conclusion that partial wavelength conversion is more cost-effective than complete wavelength conversion.

An alternative approach to wavelength conversion was recently proposed and investigated in [11]–[13]. In these studies, two cases are considered: 1) the network adopts FWC's and it has sufficient number of FWC's to provide complete wavelength conversion and 2) the same as case

Manuscript received July 29, 1997; revised October 1, 1998 and March 13, 1999; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor K. N. Sivarajan.

G. Xiao is with the Department of Electrical Engineering, Polytechnic University, Brooklyn, NY 11201 USA (e-mail: gxiao@bethel.poly.edu).

Y.-W. Leung is with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong (e-mail: y.leung@ieee.org).

Publisher Item Identifier S 1063-6692(99)06366-9.

1) except that each FWC is replaced by an LWC. It was demonstrated that both cases can result in nearly the same blocking probability.

In the last revision of this paper, we note that there were two recent papers on allocating a limited number of FWC's [9], [10]. In these papers, the blocking probability was derived for some specific cases under the statistical independence assumption. Based on this blocking probability, algorithms were proposed to allocate a limited number of FWC's. However, these algorithms are only applicable for these specific cases and assumption.

We note that in an all-optical WAN, some of the nodes are often required to handle heavier volumes of traffic. It is because the topology of a WAN is usually irregular, and the traffic is often nonuniform. It is desirable to allocate more FWC's to the nodes handling heavier volumes of traffic, so that the FWC's can be fully utilized to resolve wavelength conflicts.

In this paper, we design a set of optimization algorithms for allocating FWC's in all-optical networks. We adopt the simulation-based optimization approach, in which we collect utilization statistics of FWC's from computer simulations and then perform optimization to allocate the FWC's. Therefore, our algorithms are widely applicable and are not restricted to any specific model or assumption. We have conducted extensive computer simulations on regular and irregular networks under both uniform and nonuniform traffic. Compared with the best existing allocation, the results demonstrate that our algorithms can significantly reduce: 1) the overall blocking probability (i.e., better mean quality of service) and 2) the maximum of the blocking probabilities experienced at all the source nodes (i.e., better fairness). Equivalently, for a given performance requirement on blocking probability, our algorithms can significantly reduce the number of FWC's required. In addition, we demonstrate that our simulation-based approach is robust under simulation and estimation uncertainty.

## II. NETWORK MODEL

We consider an all-optical WAN. The network consists of  $N$  nodes and the network topology can be irregular. Every network link consists of a set of distinct wavelength channels. We consider the node configuration proposed in [8] because it requires the least number of FWC's to obtain the same blocking performance. This node configuration is called *share-per-node* [8] and is shown in Fig. 1. After demultiplexing, the wavelength channels are fed to the first switch. If a wavelength channel does not need wavelength conversion, it is switched to an appropriate multiplexer for outgoing transmission. Otherwise, it is switched to the FWC bank in which it is converted to another wavelength by a FWC, and then it is switched to an appropriate multiplexer for outgoing transmission. We use the shortest path routing.

## III. ALLOCATING FULL-RANGE WAVELENGTH CONVERTERS

Using the share-per-node structure, we need to determine the number of FWC's in the FWC bank of every node. Our

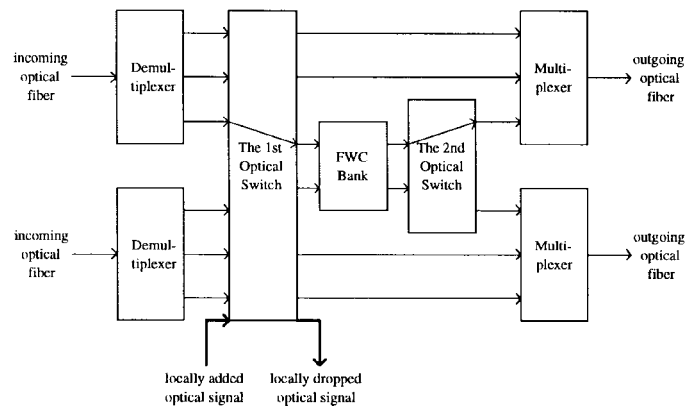


Fig. 1. Node configuration [8].

objectives are to reduce: 1) the overall blocking probability (i.e., better mean quality of service) and 2) the maximum of the blocking probabilities experienced at all the source node (i.e., better fairness). Equivalently, for a given blocking requirement, our objective is to reduce the required number of FWC's.

The blocking probability for all-optical networks is available in analytical form only under some simplifying assumptions, specific traffic models, or specific routing and wavelength assignment methods [14], [15]. Therefore, we adopt simulation-based optimization approach, so that our method is widely applicable and is not restricted to any specific model or assumption.

Our main idea is as follows. First, when there is complete wavelength conversion, we record the utilization statistics of FWC's in every node by computer simulations. Specifically, we let node  $i$  require  $M_i$  FWC's for complete wavelength conversion, where  $M_i$  is equal to the number of outgoing fibers of node  $i$  times the number of channels per fiber, and  $M = \max\{M_1, M_2, \dots, M_N\}$ . We measure the *utilization matrix*  $\mathbf{U} = [U_{i,j}]_{(1 \leq i \leq N) \times (0 \leq j \leq M)}$  in computer simulations, where  $U_{i,j}$  is the percentage of time that  $j$  FWC's are being utilized simultaneously in node  $i$ . Second, when there are a limited number of FWC's, we optimize the allocation of the given number of FWC's based on  $\mathbf{U}$ . This is an approximate approach because when there is partial wavelength conversion, the utilization matrix is changed. Nevertheless, this approximation is good for the following reason. In a well-engineered network, the traffic load handled by each node should not approach or exceed its capacity, so that the blocking probability can be kept at a reasonably low value (say, 0.01). Even if node  $i$  has  $M$  FWC's for complete wavelength conversion, it is likely that only some of them are being used at a time while the others are left idle. Therefore,  $U_{i,M}, U_{i,M-1}, \dots$ , are relatively small. For this reason, when node  $i$  is equipped with a fewer number of FWC's, the utilization matrix is only changed slightly. In Section V, we will present simulation results to demonstrate that when there are a small number of FWC's, our approximate approach can already result in blocking probabilities close to that with complete wavelength conversion.

With the above idea, we divide the problem into the following two subproblems.

- 1) Record the utilization matrix via computer simulations.
- 2) Based on the utilization matrix, optimize the allocations of the FWC's.

In the following subsections, we design algorithms to solve these subproblems.

#### A. Recording Utilization Matrix

We record the utilization matrix via simulation experiments. One important issue is that, when there is wavelength conflict, we need to determine where we should perform wavelength conversion. Different methods can lead to different utilization matrices. In our study, we design and adopt one possible method to resolve wavelength conflict that gives good results. However, our simulation-based optimization methodology is also applicable to any other conflict resolution method.

For any given call duration statistics, we can generate the duration for each transmission. Therefore, when a new request arrives, we can determine its finish time. We use this feature when we record the utilization matrix as follows.

##### 1) Recording Algorithm:

- 1) When a transmission request arrives, identify those transmissions that have been finished since the last transmission request arrives, record their duration and the number of FWC's that have been used on each node of their source-to-destination paths, then release all the FWC's used by them.
- 2) If at least one wavelength (the same one) is available on every hop of the source-to-destination path (this wavelength channel is called a *clear channel*), admit this transmission request and assign a clear channel to it on a *first-fit* basis [16]. Otherwise, go to step 3.
- 3) If there is at least one free wavelength channel (at any wavelength) on every hop of the source-to-destination path, execute the following steps:
  - a) Execute *Conflict Resolution Algorithm* (to be explained) to: i) select a wavelength requiring the smallest number of FWC's and ii) select the one that minimizes the maximum number of FWC's being used on every node of the path, when there is more than one choice.
  - b) After selecting the nodes that perform wavelength conversion (called the *tuning nodes*), select a wavelength between any two consecutive tuning nodes on a first-fit basis.

Otherwise, the transmission request is blocked.

- 4) Repeat the above steps for the next and new transmission request.  $\square$

In step 1, we perform recording only when a transmission request arrives. In this manner, it is not necessary to monitor the finish time of all the ongoing transmissions, so that the algorithm can be simpler. In step 3(a), we resolve wavelength conflict to fulfill two objectives: i) the resulting source-to-destination path requires the smallest number of FWC's and ii) when there is more than one choice, we select the one that

minimizes the maximum number of FWC's being used on every node of the path. In [8], Lee and Li proposed a *graph transformation* method to minimize the required number of FWC's (i.e., to fulfill the first objective). To tackle the second objective, we modify and enhance the graph transformation method.

The main idea is to transform the problem of resolving wavelength conflict into an equivalent shortest path problem in a directed graph, where the length of a path in the directed graph is determined by: 1) the total number of FWC's used and 2) the maximum number of FWC's being used on every node of the source-to-destination path. By determining the shortest path in this directed graph, we can fulfill both of our objectives. We construct the directed graph as follows. Along the source-to-destination path in the network, the *intermediate nodes* (excluding the source and destination nodes) are indexed from 1 to  $L$ . Let  $W(l)$  denotes the number of FWC's being used on the  $l$ th intermediate node. Now perform the following steps to construct a directed graph.

- 1) For wavelength  $\lambda$  on the  $j$ th hop of the path, let  $c(\lambda, j)$  be the weight of this wavelength channel. If it is available, then  $c(\lambda, j) = M$ ; otherwise,  $c(\lambda, j) = \infty$ .
- 2) For the  $l$ th intermediate node ( $1 \leq l \leq L$ ), we create a vertex  $v_i(\lambda_i, l)$  for every incoming wavelength channel  $\lambda_i$ , and create a vertex  $v_o(\lambda_o, l)$  for every outgoing wavelength channel  $\lambda_o$ . The weight of the edge connecting vertex  $v_i(\lambda, l)$  to vertex  $v_o(\lambda, l)$  for any  $\lambda$  is zero. The edge connecting  $v_i(\lambda_i, l)$  and  $v_o(\lambda_o, l)$  for any  $\lambda_i \neq \lambda_o$  is called a *converter edge*, and its weight is  $M + W(l)$ .

Fig. 2(a) and (b) illustrate the construction of the directed graph.

We recall that the length of a path in the directed graph must reflect two quantities: i) the total number of FWC's used and ii) the maximum number of FWC's used on the network nodes [e.g., see Fig. 2(c)]. The traditional shortest path algorithms can only tackle the first quantity. To tackle both quantities, we modify the Dijkstra's algorithm to solve our problem as follows.

##### 2) Conflict Resolution Algorithm:

- 1) Initially, label the source vertex  $s$ , and leave all the other vertices unlabeled. Let  $d(s) = 0$  and  $d(y) = \infty$  for all  $y \neq s$ . Let  $x = s$ .
- 2) Let  $a(x, y)$  be the weight for the edge from vertex  $x$  to vertex  $y$ . For each unlabeled vertex  $y$ , if  $a(x, y) \leq M$ , compute  $d(y)$  as follows:

$$d(y) = \min\{d(y), d(x) + a(x, y)\}. \quad (1)$$

If  $M < a(x, y) < \infty$  (i.e., the edge from vertex  $x$  to vertex  $y$  is a converter edge), compute  $d(y)$  as follows:

$$d(y) = \min\{d(y), [d(x) - d'(x)] + [M + \max\{d'(x), a'(x, y)\}]\} \quad (2)$$

where  $d'(x)$  is the maximum of the number of FWC's being used in all the previous nodes up to  $x$

$$\begin{cases} d'(x) = \text{mod}(d(x), M) \\ a'(x, y) = \text{mod}(a(x, y), M). \end{cases} \quad (3)$$

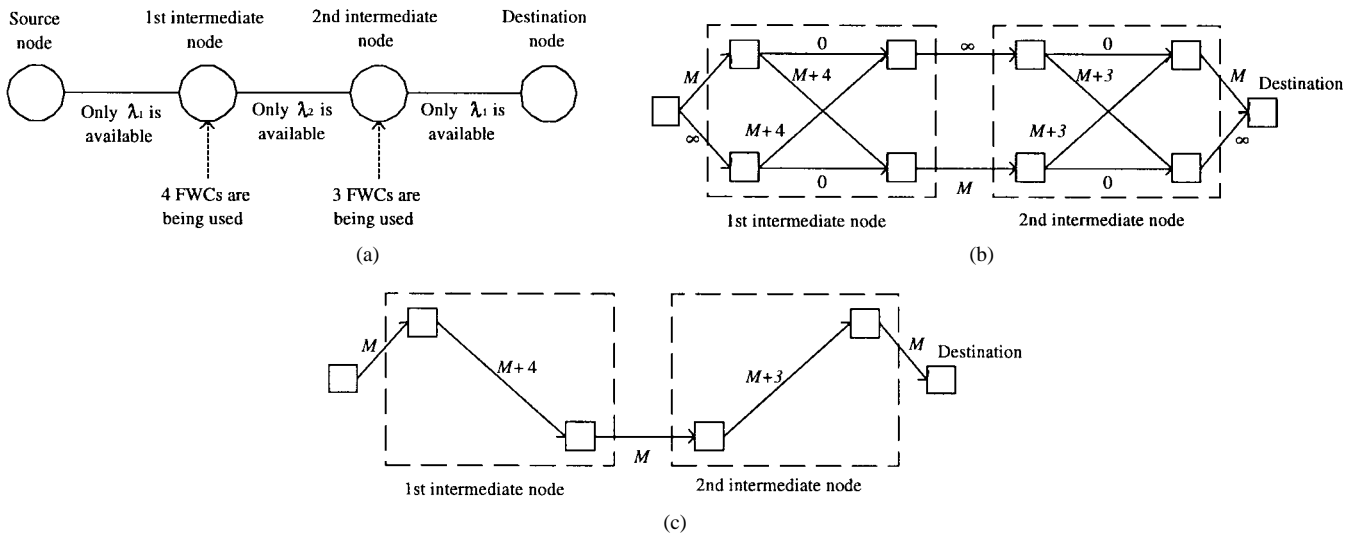


Fig. 2. An example to illustrate the graph transformation method to resolve wavelength conflict. (a) A source-to-destination path with two intermediate nodes and ten wavelength channels/link. (b) A directed graph for the source-to-destination path shown in (a). (c) The shortest path in the directed graph. The weight of the shortest path is  $5M + \max(3, 4) = 5M + 4$ .

In other words,  $d'(x, y)$  is the number of FWC's being used on the intermediate node; and  $d(x)$  is equal to  $M$  times the total number of hops and converter edges up to  $x$ , plus  $M$ , and plus the maximum number of FWC's being used on all the previous nodes up to  $x$ . Label the unlabeled vertex  $y$  with the smallest value of  $d(y)$ .

- 3) If the destination vertex has been labeled (i.e., a path has been determined), identify the tuning nodes, increment  $W(l)$  of these nodes by 1, and then stop. Otherwise, repeat step 2.  $\square$

The above algorithm is modified from Dijkstra's algorithm, and (2) and (3) are tailored for our problem. Specifically, we use these equations to minimize the number of FWC's needed in the path, for once a converter edge is included, the cost of the path has to be increased by at least  $M$  units. When there is a tie, we use (2) and (3) to minimize the maximum number of FWC's being used on every intermediate node, for  $d(x)$  is also determined by the maximum number of FWC's that have been used on all the previous nodes up to  $x$ . The time complexity of the above algorithm can be found to be  $O(n^2)$ , where  $n$  is the number of vertices in the directed graph.

### B. Allocating FWC's

In this subsection, we optimize the allocations of a given number of FWC's based on the utilization matrix  $\mathbf{U}$ .

After allocating a certain number of FWC's to a node, we can get from  $\mathbf{U}$  the percentage of time that this node has sufficient FWC's to serve the transmission. For convenience, we call this quantity the *total utilization*. For example, if node  $i$  has  $J$  WC's, the total utilization is  $\sum_{j=0}^J U_{i,j}$ . To optimize the allocation of a given number of FWC's without any assumption about the traffic pattern, we consider three different objective functions.

- 1) Maximize the sum of total utilizations of all the nodes, so that the overall utilization of FWC's can be improved. As a result, the overall blocking probability can be smaller and, hence, the mean quality of service is better.

- 2) Maximize the product of the total utilizations of all the nodes. In this manner, the overall utilization of FWC's can be improved (i.e., better mean quality of service) and the allocation of FWC's to the nodes can be more fair.
- 3) Maximize the minimum value of total utilization of the  $N$  nodes, so that the allocation of FWC's to the nodes can be more fair.

1) *Maximize the Sum of Total Utilizations*: Let  $T$  denote the total number of available FWC's. For the trivial case that  $T$  is large enough to provide complete wavelength conversion (i.e.,  $T \geq MN$ ), the optimal allocation of FWC's is simple: allocate  $M$  FWC's to every node so that every node can perform complete wavelength conversion. In the following, we consider the case  $T < MN$ .

We define  $x_{i,j}$  as follows:

$$x_{i,j} = \begin{cases} 1, & \text{if } j \text{ or more WC's are allocated} \\ & \text{to node } i \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

The total utilization of node  $i$  is  $\sum_{j=1}^M U_{i,j} x_{i,j}$ . The problem of maximizing the sum of the total utilization of all the nodes can be formulated as follows:

$$\left\{ \begin{array}{l} \text{maximize } x_{i,j} \quad \sum_{i=1}^N \sum_{j=1}^M U_{i,j} \cdot x_{i,j} \\ \text{subject to} \quad 1. \quad \sum_{i=1}^N \sum_{j=1}^M x_{i,j} = T \\ \quad \quad \quad 2. \quad x_{i,j} \geq x_{i,j+1}, \quad i = 1, 2, \dots, N \\ \quad \quad \quad \quad \quad j = 1, 2, \dots, M-1 \\ \quad \quad \quad 3. \quad x_{i,j} \in \{0, 1\}. \end{array} \right. \quad (5)$$

The objective function is the sum of the total utilization of all the nodes. The first constraint ensures that the total number of FWC's allocated to the nodes is  $T$ . The second and the third constraints ensure that  $x_{i,j} \geq x_{i,j+1}$  and  $x_{i,j}$  is binary. The variables to be optimized are  $x_{i,j}$  for all  $1 \leq i \leq N$  and  $1 \leq$

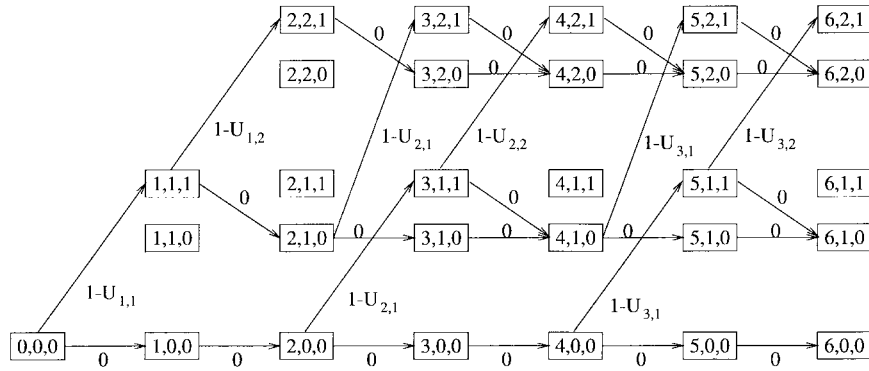


Fig. 3. An example to illustrate the directed graph used in Optimization Algorithm 1;  $T = M = 2$ ,  $N = 3$ .

$j \leq M$ . After optimization, if  $x_{i,1} = x_{i,2} = \dots = x_{i,n} = 1$  and  $x_{i,n+1} = x_{i,n+2} = \dots = x_{i,M} = 0$ , then the optimal number of FWC's allocated to node  $i$  is  $n$ .

We solve the above optimization problem as follows. Observing that  $0 \leq U_{i,j} \leq 1$  and  $T < N \cdot M$ , we can transform the above optimization problem into the following problem which has the same optimal solutions as those of (5)

$$\left\{ \begin{array}{l} \text{minimize}_{x_{i,j}} \quad \sum_{i=1}^N \sum_{j=1}^M (1 - U_{i,j}) \cdot x_{i,j} \\ \text{subject to} \quad 1. \quad \sum_{i=1}^N \sum_{j=1}^M x_{i,j} = T \\ \quad \quad \quad 2. \quad x_{i,j} \geq x_{i,j+1}, i = 1, 2, \dots, N \\ \quad \quad \quad \quad j = 1, 2, \dots, M-1 \\ \quad \quad \quad 3. \quad x_{i,j} \in \{0, 1\}. \end{array} \right. \quad (6)$$

If we remove the second constraint  $x_{i,j} \geq x_{i,j+1}$ ,  $i = 1, 2, \dots, N$ ,  $j = 1, 2, \dots, M-1$ , the above problem reduces to the knapsack problem [17]. Therefore, the above problem can be regarded as a constrained knapsack problem. It is well known that the knapsack problem can be solved by the dynamic programming method [17] which transforms the knapsack problem to an equivalent shortest path problem in directed graphs. To tackle the constraint  $x_{i,j} \geq x_{i,j+1}$ , we modify the dynamic programming method and construct a directed graph for our problem as follows. Every vertex is indexed as  $(i_1, j_1, k_1)$ , where for each  $1 \leq i \leq N$ ,  $1 \leq j \leq M$ ,  $i_1 = (i-1) \cdot M + j$ ,  $j_1 = \sum_{p=1}^i \sum_{q=1}^j x_{p,q}$ , and  $k_1 = x_{i,j}$ . Therefore,  $0 \leq i_1 \leq MN$ ,  $0 \leq j_1 \leq T$ , and  $0 \leq k_1 \leq 1$ . Since  $0 \leq k_1 = x_{i,j} \leq 1$ , the third constraint of (6) has been fulfilled. To fulfill the second constraint, there is an edge connecting vertex  $(i_1, j_1, k_1)$  to vertex  $(i_2, j_2, k_2)$  if and only if: 1)  $i_2 = i_1 + 1$  and  $j_2 = j_1 + k_2$  for  $k_1 \geq k_2$  or 2)  $\text{mod}(i_1, M) = 0$  and  $j_2 = j_1 + k_2$ . The weight of this edge is set to  $1 - U_{i,j}$  (where  $i_2 = (i-1)M + j$ ) if  $j_2 = j_1 + 1$  and is set to 0 if  $j_2 = j_1$ . In this manner, we have incorporated the objective function of (6) into the shortest path problem. To fulfill the first constraint of (6), we find the shortest paths: i) from vertex  $(0, 0, 0)$  to vertex  $(MN, T, 0)$  and ii) from vertex  $(0, 0, 0)$  to vertex  $(MN, T, 1)$ , and then select the shorter

one between them. Fig. 3 shows an illustrative example. The optimization details are given below.

i) *Optimization Algorithm 1:*

1) Construct a directed graph  $G = (V, A)$  where the set of vertices  $V$  is

$$V = \{0, 1, \dots, M \cdot N\} \times \{0, 1, \dots, T\} \times \{0, 1\} \\ - \{0, 1, \dots, M \cdot N\} \times \{0\} \times \{1\}. \quad (7)$$

There is an edge in  $A$  connecting vertex  $(i_1, j_1, k_1)$  to vertex  $(i_2, j_2, k_2)$ , if and only if: i)  $i_2 = i_1 + 1$  and  $j_2 = j_1 + k_2$  for  $k_1 \geq k_2$  or ii)  $\text{mod}(i_1, M) = 0$  and  $j_2 = j_1 + k_2$ . The weight of this edge is set to  $1 - U_{i,j}$  (where  $i_2 = (i-1)M + j$ ) if  $j_2 = j_1 + 1$  and set to 0 if  $j_2 = j_1$ .

2) Find the shortest path: i) from vertex  $(0, 0, 0)$  to vertex  $(M \cdot N, T, 0)$  and ii) from vertex  $(0, 0, 0)$  to vertex  $(M \cdot N, T, 1)$ , and select the shorter one between them. If the shortest path passes through the edge with weight  $1 - U_{i,j}$ , the optimal value of  $x_{i,j}$  is 1; otherwise, the optimal value is 0.  $\square$

The time complexity of above algorithm can be analyzed as follows. From (7), we see that the directed graph has  $O(MNT)$  vertices and hence we can use the Dijkstra's algorithm to find the shortest path in this directed graph in  $O(M^2N^2T^2)$  time.

2) *Maximize the Product of the Total Utilizations:* In this subsection, we formulate and solve the problem of maximizing the product of the total utilizations of the nodes. Let  $x_{i,j}$  be as defined in (4). The problem can be formulated as follows:

$$\left\{ \begin{array}{l} \text{maximize}_{x_{i,j}} \quad \prod_{i=1}^N \left\{ \sum_{j=0}^M (U_{i,j} \cdot x_{i,j}) \right\} \\ \text{subject to} \quad 1. \quad \sum_{i=1}^N \sum_{j=1}^M x_{i,j} = T \\ \quad \quad \quad 2. \quad x_{i,0} = 1, i = 1, 2, \dots, N \\ \quad \quad \quad 3. \quad x_{i,j} \geq x_{i,j+1}, i = 1, 2, \dots, N \\ \quad \quad \quad \quad j = 1, 2, \dots, M-1 \\ \quad \quad \quad 4. \quad x_{i,j} \in \{0, 1\}. \end{array} \right. \quad (8)$$

The variables to be optimized are  $x_{i,j}$  for all  $1 \leq i \leq N$  and  $1 \leq j \leq M$ . The objective function is the product of the total utilizations of all the nodes. The first constraint ensures that the total number of FWC's allocated to the nodes is  $T$ . The second constraint ensures that the  $U_{i,0}$  is always included in the objective function so that the objective function will not be zero. The third and fourth constraints ensure that  $x_{i,j} \geq x_{i,j+1}$  and  $x_{i,j}$  is binary.

We solve the above optimization problem as follows. Since

$$0 < \sum_{j=0}^M U_{i,j} x_{i,j} \leq 1, \quad 1 \leq i \leq N$$

maximizing  $\prod_{i=1}^N \sum_{j=0}^M U_{i,j} x_{i,j}$  is equivalent to maximizing the following function:

$$\log \left\{ \prod_{i=1}^N \sum_{j=0}^M (U_{i,j} x_{i,j}) \right\} = \sum_{i=1}^N \log \left\{ \sum_{j=0}^M (U_{i,j} \cdot x_{i,j}) \right\}. \quad (9)$$

We need the following lemma.

*Lemma 1:* For any  $1 \leq i \leq N$ , if

$$\begin{cases} x_{i,0} = 1 \\ x_{i,j} \geq x_{i,j+1} & j = 1, 2, \dots, M-1 \\ x_{i,j} \in \{0, 1\} & j = 1, 2, \dots, M \end{cases} \quad (10)$$

then

$$\log \left[ \sum_{j=0}^M (U_{i,j} \cdot x_{i,j}) \right] = \log U_{i,0} + \sum_{j=1}^M \left\{ x_{i,j} \left[ \log \left( \sum_{s=0}^j U_{i,s} \right) - \log \left( \sum_{s=0}^{j-1} U_{i,s} \right) \right] \right\}. \quad (11)$$

*Proof:*

- 1) If  $x_{i,1} = 0$ , then for any  $1 \leq j \leq M$ ,  $x_{i,j} = 0$ . The result is obviously correct.
- 2) If there exists  $1 \leq j_0 \leq M$  such that  $x_{i,j_0} = 1$  and  $x_{i,j_0+1} = 0$ , then from (10), we have  $x_{i,1} = x_{i,2} = \dots = x_{i,j_0} = 1$ , whereas  $x_{i,j_0+1} = x_{i,j_0+2} = \dots = x_{i,M} = 0$ . Thus

$$\begin{aligned} & \sum_{j=1}^M \left\{ x_{i,j} \left[ \log \left( \sum_{s=0}^j U_{i,s} \right) - \log \left( \sum_{s=0}^{j-1} U_{i,s} \right) \right] \right\} + \log(U_{i,0}) \\ &= \sum_{j=1}^{j_0} \left[ \log \left( \sum_{s=0}^j U_{i,s} \right) - \log \left( \sum_{s=0}^{j-1} U_{i,s} \right) \right] + \log(U_{i,0}) \\ &= \left[ \log \left( \sum_{s=0}^{j_0} U_{i,s} \right) - \log(U_{i,0}) \right] + \log(U_{i,0}) \\ &= \log \sum_{s=0}^{j_0} U_{i,s} \\ &= \log \sum_{s=0}^{j_0} (U_{i,s} \cdot x_{i,s}) \\ &= \log \sum_{j=0}^M (U_{i,j} \cdot x_{i,j}). \end{aligned} \quad (12)$$

- 3) If  $x_{i,M} = 1$ , then  $x_{i,j} = 1$ ,  $j = 1, 2, \dots, M$ . Similar to case 2), formula (11) can be proved to be correct. This completes the proof.  $\square$

Using the above lemma, we can transform optimization problem (8) into an equivalent form of (5), so that we can apply Optimization Algorithm 1 to find the optimal solutions. The details are given in the following algorithm.

*ii) Optimization Algorithm 2:*

- 1) Using the following transformation

$$U'_{i,j} = \log \left( \sum_{s=0}^j U_{i,s} \right) - \log \left( \sum_{s=0}^{j-1} U_{i,s} \right), \quad 1 \leq i \leq N, 1 \leq j \leq M \quad (13)$$

we transform problem (8) to the following form:

$$\begin{cases} \text{maximize}_{x_{i,j}} & \sum_{i=1}^N \sum_{j=1}^M (U'_{i,j} \cdot x_{i,j}) \\ \text{subject to} & 1. \sum_{i=1}^N \sum_{j=1}^M x_{i,j} = T \\ & 2. x_{i,j} \geq x_{i,j+1}, i = 1, 2, \dots, N; \\ & \quad j = 1, 2, \dots, M-1 \\ & 3. x_{i,j} \in \{0, 1\} \end{cases} \quad (14)$$

- 2) Apply Optimization Algorithm 1 to solve (14).  $\square$

3) *Maximize the Minimum Total Utilization:* In this subsection, we formulate and solve the problem of maximizing the minimum total utilization of all the  $N$  nodes. Let  $x_{i,j}$  be as defined in (4). This problem can be formulated as follows:

$$\begin{cases} \text{maximize}_{x_{i,j}} & \left( \min_{1 \leq i \leq N} \left\{ \sum_{j=0}^M U_{i,j} x_{i,j} \right\} \right) \\ \text{subject to} & 1. \sum_{i=1}^N \sum_{j=1}^M x_{i,j} = T \\ & 2. x_{i,j} \geq x_{i,j+1}, 1 \leq i \leq N; \\ & \quad 1 \leq j \leq M-1 \\ & 3. x_{i,j} \in \{0, 1\}. \end{cases} \quad (15)$$

The variables to be optimized are  $x_{i,j}$  for all  $1 \leq i \leq N$  and  $1 \leq j \leq M$ . The objective function is the minimum total utilization of all the  $N$  nodes, and the first constraint ensures that the total number of FWC's allocated to the  $N$  nodes is  $T$ .

The above optimization problem can be solved by the following greedy algorithm.

*iii) Optimization Algorithm 3:*

- 1) Initially, all the  $N$  nodes have no FWC.
- 2) Among the  $N$  nodes, find the node having the smallest total utilization, then allocate one more FWC to this node, and then update the total utilization for this node based on  $\mathbf{U}$ .
- 3) Repeat step 2 until all the  $T$  FWC's have been allocated to the nodes.  $\square$

*Theorem 1:* Optimization Algorithm 3 can find optimal solutions to optimization problem (15).

*Proof:* By contradiction. Assume that Optimization Algorithm 3 cannot find the optimal solution to optimization problem (15). We let  $W^*(i)$  be the optimal number of FWC's allocated to node  $i$ , and  $W_a(i)$  be the number of FWC's allocated to node  $i$  by Optimization Algorithm 3. In addition, let  $S^*(i)$  and  $S_a(i)$  be the total utilization of node  $i$  corresponding to  $W^*(i)$  and  $W_a(i)$ , respectively. In the optimal allocation of FWC's, suppose node  $i_0^*$  has the smallest total utilization; in the allocation by Optimization Algorithm 3, node  $i_0$  has the smallest total utilization. In other words, we have

$$\begin{cases} S^*(i_0^*) = \min\{S^*(1), S^*(2), \dots, S^*(N)\} \\ S_a(i_0) = \min\{S_a(1), S_a(2), \dots, S_a(N)\} \end{cases} \quad (16)$$

since

$$S^*(i_0) \geq S^*(i_0^*). \quad (17)$$

We have three possible cases

$$\begin{cases} \text{case (a)} & W^*(i_0) > W_a(i_0) \\ \text{case (b)} & W^*(i_0) = W_a(i_0) \\ \text{case (c)} & W^*(i_0) < W_a(i_0). \end{cases} \quad (18)$$

Since the total utilization of a node is monotonically increasing with the number of FWC's allocated to it, both case (b) and case (c) will lead to a contradiction to the assumption that the solution from Optimization Algorithm 3 is not optimal. For case (a), since the total number of available FWC's is fixed, there must exist at least one node (called node  $k_0$ ), such that

$$W^*(k_0) < W_a(k_0). \quad (19)$$

If  $S^*(k_0) \leq S_a(i_0)$ , then

$$S^*(i_0^*) \leq S^*(k_0) \leq S_a(i_0) \quad (20)$$

which leads to a contradiction. If  $S^*(k_0) > S_a(i_0)$ , then we have

$$S_a(k_0) > S^*(k_0) > S_a(i_0). \quad (21)$$

Combined with (19), we see that Optimization Algorithm 3 has allocated more FWC's to the node with larger total utilization (node  $k_0$ ), but not the node with the smallest total utilization (node  $i_0$ ), which leads to contradiction again. This completes the proof.  $\square$

#### IV. ROUTING AND WAVELENGTH ASSIGNMENT ALGORITHM

Routing and wavelength assignment (RWA) for all-optical network is a hot research topic and several RWA algorithms have been proposed (e.g., [5], [16], [18], [19]). These algorithms were designed for the networks having either complete wavelength conversion or no wavelength conversion. In this section, we design a new RWA algorithm for the networks in which different nodes may be allocated different number of FWC's.

The critical problem is that when a certain number of FWC's have been allocated to each node, how should we select the tuning nodes for every transmission request in order to get

good blocking performance? Our main ideas for this problem are as follows.

- 1) Once a transmission request arrives, select the set of tuning nodes such that the required number of FWC's is minimized.
- 2) When there is more than one choice, select the one that maximizes the minimum number of free FWC's in each tuning node of the source-to-destination path. For simplicity, we call the tuning node with minimum number of free FWC's as the *critical node*.
- 3) When there is more than one choice, select the one that has the maximum number of FWC's installed on the critical node. When there is still more than one choice (though this rarely happens), randomly select one choice.

Based on the above ideas, we design the following algorithm for routing and wavelength assignment. For the  $i$ th intermediate node ( $1 \leq i \leq L$ ), let  $N_t(i)$  and  $N_a(i)$  be the total number of FWC's and the number of FWC's being used in this node, respectively. Therefore, the number of free FWC's in this node is  $N_t(i) - N_a(i)$ . The details of our RWA algorithm are as follows.

*i) RWA Algorithm:*

- 1) Check if there is at least one clear channel on the source-to-destination path. If one exists, assign this clear channel to the transmission request; if there is more than one channel, select one of them on a first-fit basis; if there is none, go to step 2.
- 2) If there is at least one free wavelength channel (at any wavelength) on every hop of the source-to-destination path, execute the following steps:
  - a) Construct a directed graph in a manner similar to that in Conflict Resolution Algorithm. For each free wavelength channel on every hop, the weight of the corresponding edge is  $M$ . On every intermediate node  $l$ , the weight of the edge between the node  $v_i(\lambda_i, l)$  and node  $v_o(\lambda_o, l)$  is

$$c(\lambda_i, \lambda_o, l) = \begin{cases} M + S, & \text{if } \lambda_i \neq \lambda_o \\ 0, & \text{if } \lambda_i = \lambda_o \end{cases} \quad (22a)$$

where

$$S = \begin{cases} \frac{M}{N_t(l) - N_a(l)} + \left(1 - \frac{N_a(l)}{N_t(l)}\right), & \text{if } N_t(l) > N_a(l) \\ \infty, & \text{if } N_t(l) = N_a(l). \end{cases} \quad (22b)$$

Apply (1)–(3) in the Conflict Resolution Algorithm to find the shortest path from the source to the destination.

- b) Determine the set of tuning nodes and increment  $N_a(l)$  of each tuning node by 1.

Otherwise, the transmission request is blocked.  $\square$

The RWA algorithm can minimize the number of FWC's required by each transmission because the weight of each converter edge is at least  $M$ . When there is more than

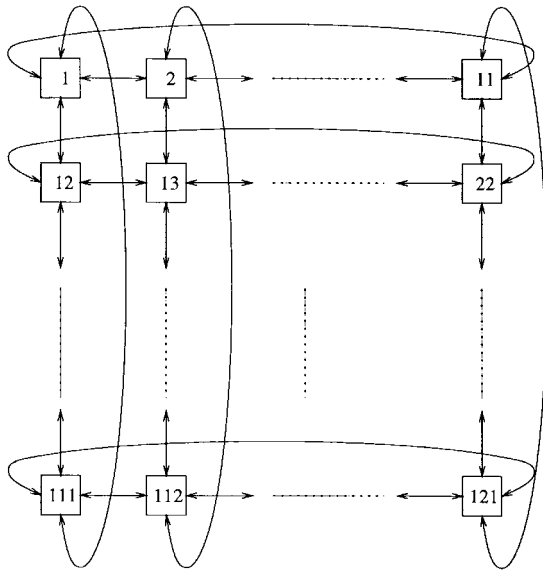


Fig. 4. The  $11 \times 11$  torus-mesh network, where every edge represents a link which is composed of two separate fibers going in opposite directions and each fiber has ten channels.

one choice requiring the same number of tuning nodes, (22) ensures that we can select the choice that maximizes the minimum number of free FWC's in each tuning node. It is because a smaller value of  $M/[N_t(l) - N_a(l)]$  implies a larger value of  $N_t(l) - N_a(l)$  (i.e., a larger number of free FWC's in node  $l$ ). When there is still more than one choice, (22) can also ensure that we can select the choice that has the maximum number of FWC's installed on the critical node. It is because, for the same value of  $N_t(l) - N_a(l)$ , a larger  $N_t(l)$  can lead to a smaller value of  $(1 - N_a(l)/N_t(l))$ .

V. NUMERICAL RESULTS AND DISCUSSIONS

We use computer simulations to evaluate the performance of the proposed allocation method. The main steps are as follows.

- 1) Conduct a computer simulation for any given network with complete wavelength conversion and any given traffic load and pattern. During simulation, execute the Recording Algorithm to record the utilization matrix.
- 2) Based on the recorded utilization matrix, execute Optimization Algorithm 1 (or 2 or 3) to optimize the allocation of FWC's.
- 3) Conduct another computer simulation for the same network with the allocation of FWC's determined in step 2. During simulation, execute the RWA Algorithm to perform routing and wavelength assignment for each new request and record the blocking probability.

We have conducted extensive computer simulations to study the effectiveness of our algorithms. We consider a regular network (an  $11 \times 11$  torus-mesh network with 121 nodes [5], see Fig. 4) and an irregular network with 100 nodes (see Appendix A), where every network link is composed of two separate fibers going in opposite directions and each fiber has ten channels. The torus-mesh network has been adopted by many researchers for performance evaluation of all-optical network (e.g., see [5], [7]). We consider both uniform and

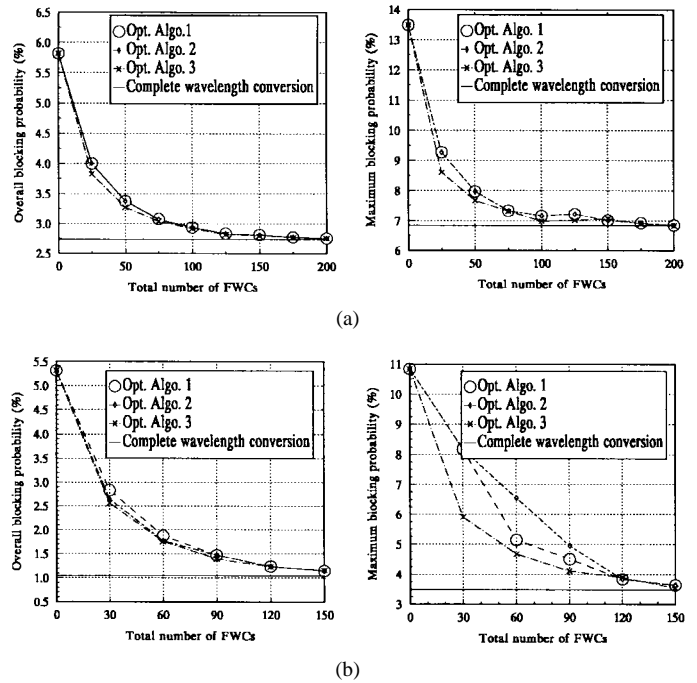


Fig. 5. Performance of Optimization Algorithms 1-3. (a) Uniform traffic on the irregular network with traffic load 130 Erlangs. The number of FWC's required for complete wavelength conversion is 3800. (b) Nonuniform traffic on regular network with traffic load 160 Erlangs. The number of FWC's required for complete wavelength conversion is 4840.

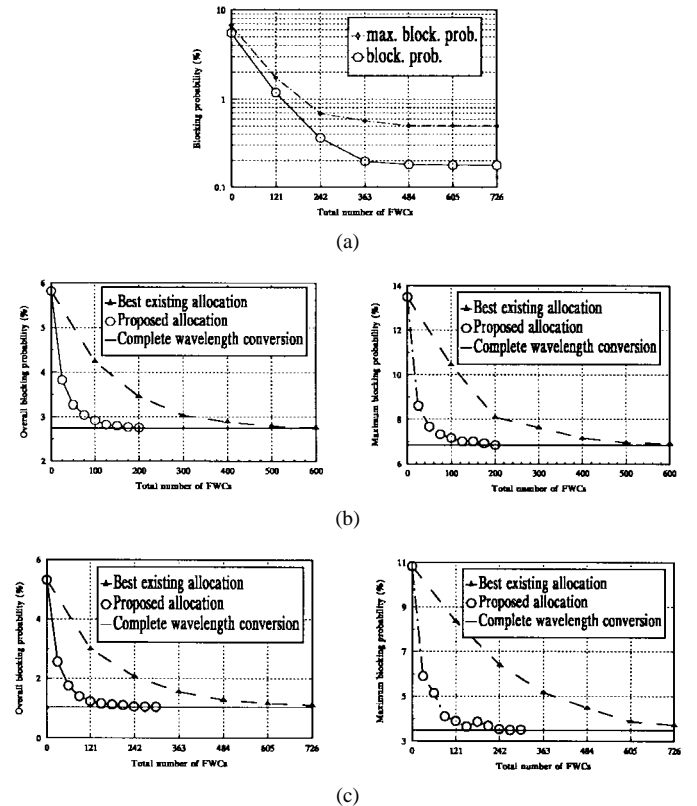
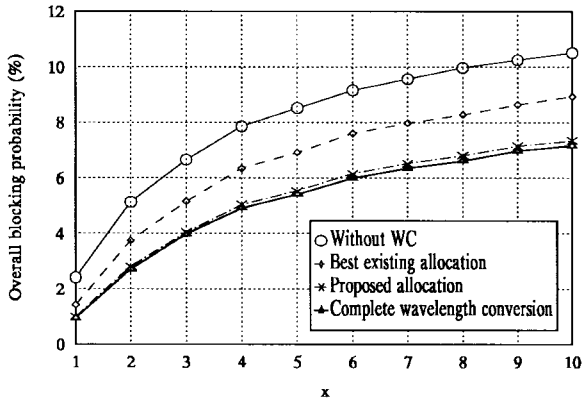
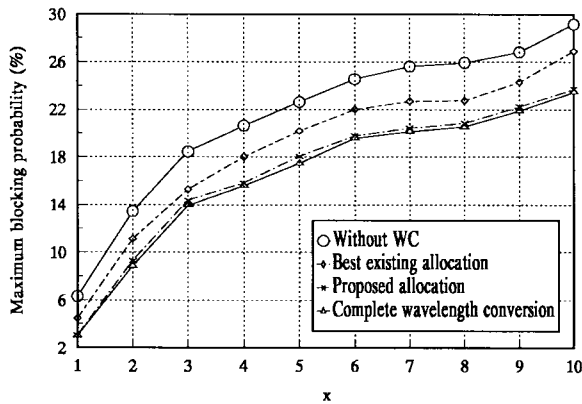


Fig. 6. Performance of the proposed allocation and the best existing allocation. (a) Regular network and uniform traffic with traffic load 180 Erlangs. Our allocation is the same as the best existing allocation. (b) Irregular network and uniform traffic load 130 Erlangs. The number of FWC's required for complete wavelength conversion is 3800. (c) Regular network and nonuniform traffic with traffic load 160 Erlangs. The number of FWC's required for complete wavelength conversion is 4840.





(a)



(b)

Fig. 7. Performance of the proposed method and the best existing allocation versus the traffic parameter  $x$ , where a larger  $x$  specifies a more nonuniform traffic. Network topology is irregular, traffic load is 100 Erlangs, and there are 100 FWC's. The number of FWC's required for complete wavelength conversion is 3800. (a) Overall blocking probability. (b) Maximum blocking probability.

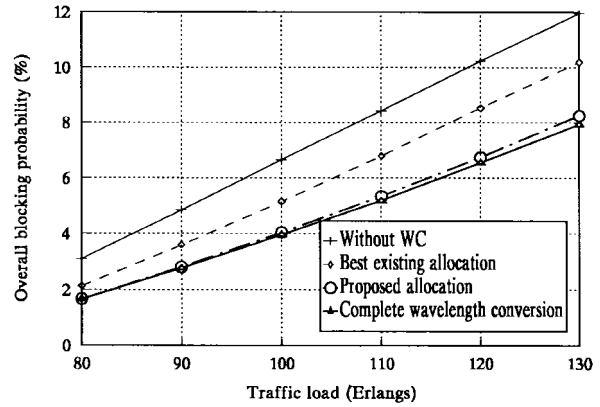
nonuniform traffic and they are described in Appendix B. For the nonuniform traffic mode, there is a traffic parameter  $x$  where a larger  $x$  specifies a more nonuniform traffic.  $x$  is defined in (B.2) in Appendix B.

We consider two performance measures: 1) overall blocking probability (i.e., the average of the blocking probabilities experienced at all the source nodes) and 2) maximum blocking probability (i.e., the maximum of the blocking probabilities experienced at all the source nodes). The first performance measure can measure the mean quality of service, while the second one can measure the fairness. To make comparisons, we apply the blocking probability with complete wavelength conversion to evaluate the performance of the proposed algorithms under partial wavelength conversion.

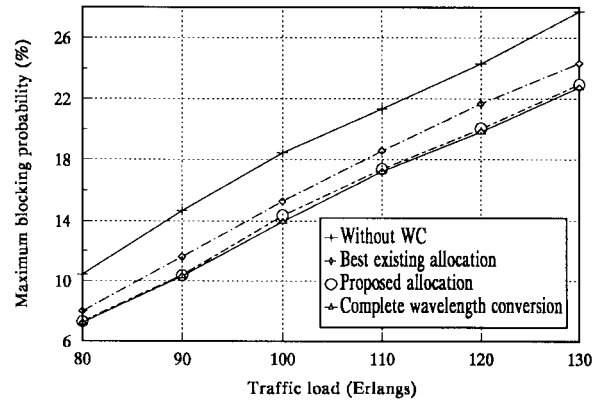
In subsection A, we compare the performance of our algorithms with the best existing allocation and present the performance improvement. In subsection B, we demonstrate that our algorithms are robust under simulation and estimation uncertainty.

A. Performance

Fig. 5 shows the performance of Optimization Algorithms 1–3. When each node has one or more FWC's on average,



(a)



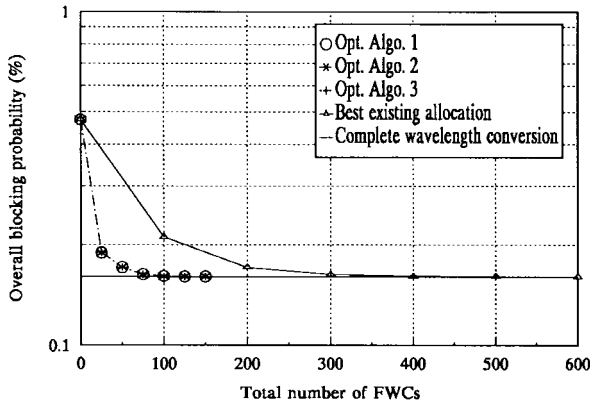
(b)

Fig. 8. Performance of the proposed method and the best existing allocation under different traffic load. Network is irregular, traffic is nonuniform with  $x = 3$ , and there are 100 FWC's. The number of FWC's required for complete wavelength conversion is 3800. (a) Overall blocking probability. (b) Maximum blocking probability.

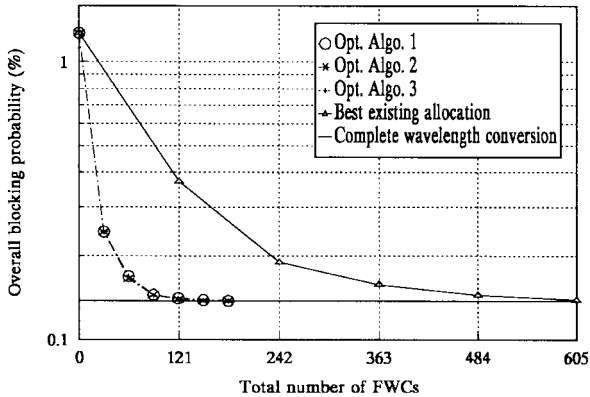
we see that all the three algorithms can result in blocking probabilities close to those with complete wavelength conversion. In addition, we see that these algorithms have similar performance for our regular and irregular networks. Since Optimization Algorithm 3 has the smallest time complexity, it can be regarded as the most efficient one for these two networks. Therefore, unless otherwise specified, we consider Optimization Algorithm 3 in the remaining part of this section.

Fig. 6 shows the performance of our allocation and the best existing allocation [8]. From this figure, we observe the following points.

- When each node has one or more FWC's on average, our method can already result in blocking probabilities close to those with complete wavelength conversion. This demonstrates that our approximate approach is very good.
- When the network topology is regular and the traffic is uniform, Fig. 6(a) shows that our method and the best existing allocation have the same performance. It is because every node handles the same amount of traffic in this special case, and hence, the optimal allocation is to allocate the same number of FWC's to every node. Therefore, the best existing allocation can be regarded as a special case of our allocation.



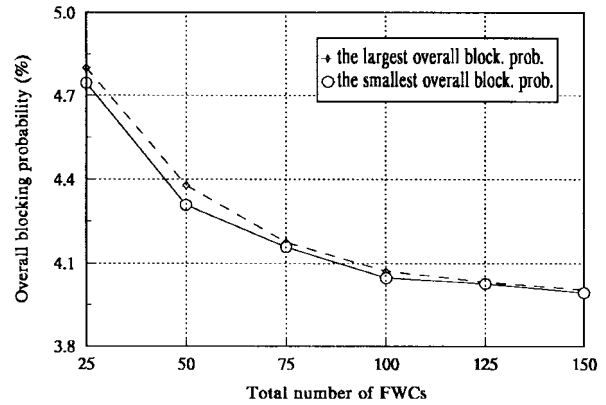
(a)



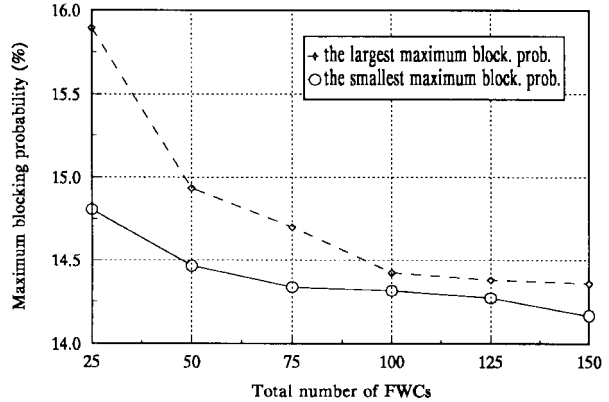
(b)

Fig. 9. Performance of the proposed method and the best existing allocation for low blocking probability. (a) Irregular network and uniform traffic with traffic load 70 Erlangs. The number of FWC's required for complete wavelength conversion is 3800. (b) Nonuniform traffic on regular network with traffic load 110 Erlangs. The number of FWC's required for complete wavelength conversion is 4840.

- When the network topology is irregular and the traffic is uniform, Fig. 6(b) shows that our method can give significantly better performance than the best existing allocation, especially when the number of available FWC's is small. For example, when the number of available FWC's is 100, the overall blocking probabilities of our method and the best existing allocation are 2.916% and 4.244%, respectively (i.e., our method can reduce the overall blocking probability by 31.3%). In addition, the maximum blocking probabilities of our method and the best existing allocation are 7.158% and 10.460%, respectively (i.e., our method can reduce the maximum blocking probability by 31.6%). From another point of view, when we want to achieve a given blocking performance, our method requires significantly fewer WC's than those required by the best existing allocation. For example, if we want to ensure that the overall blocking probability is about 3%, the number of FWC's required by our method and the best existing allocation are 100 and 300, respectively.
- When the network topology is regular and the traffic is nonuniform, Fig. 6(c) shows that our method can also give significantly better performance than the best existing allocation. For example, when the number of



(a)



(b)

Fig. 10. Robustness of the proposed method under simulation uncertainty on an irregular network with nonuniform traffic with  $x = 3$  and traffic load 100 Erlangs. There are ten simulation runs. (a) Overall blocking probability. (b) Maximum blocking probability.

available FWC's is 121, our method can reduce the overall and maximum blocking probability of the best existing allocation by 59.0% and 53.5%, respectively.

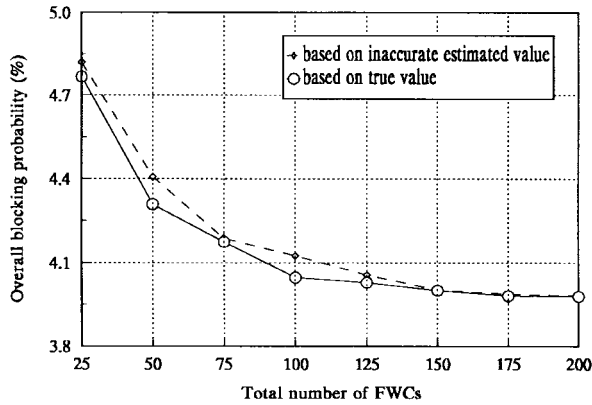
Figs. 7 and 8 compare the performance of the proposed method with the best existing allocation when the traffic becomes more nonuniform and the traffic load becomes heavier. We see that our method is significantly better than the best existing allocation, and its performance is quite close to that with complete wavelength conversion.

Fig. 9 shows the performance of the proposed allocation and the best existing allocation for low blocking probability. We observe similar results: 1) the proposed allocation can result in blocking probabilities close to that with complete wavelength conversion and 2) the proposed allocation is significantly better than the best existing allocation.

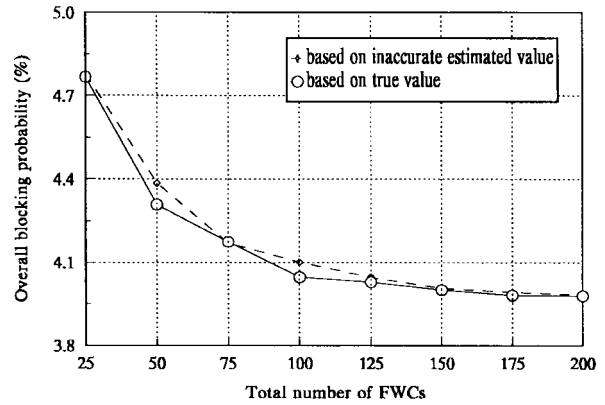
### B. Robustness

In computer simulations, uncertainty is unavoidable. In this subsection, we demonstrate that our simulation-based optimization method is robust under: 1) simulation uncertainty and 2) estimation uncertainty of *traffic pattern* and *traffic load* [20].

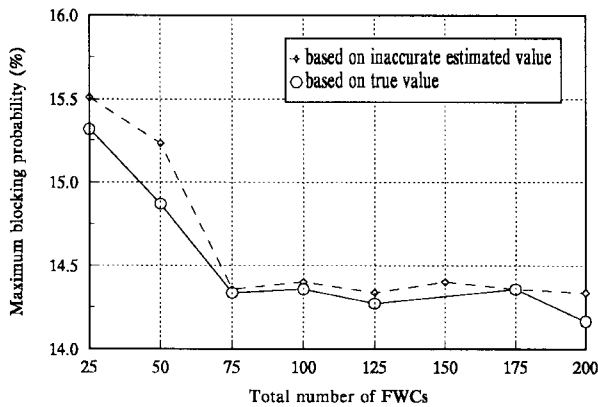
To study the stability of Optimization Algorithms 1–3 under simulation uncertainty, we conducted ten independent simula-



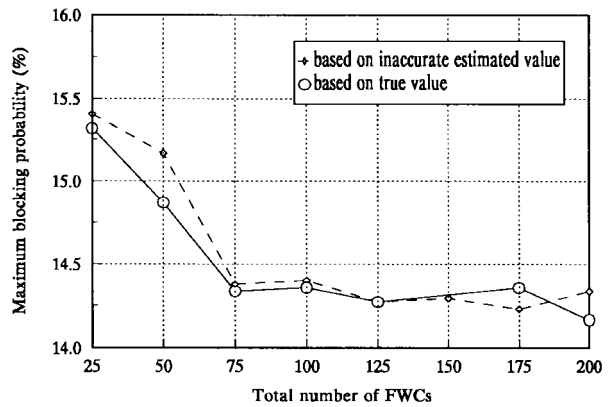
(a)



(a)



(b)



(b)

Fig. 11. Robustness of the proposed method under estimation uncertainty of traffic pattern. The network is irregular with traffic load 100 Erlangs. The traffic is nonuniform and the true value and the inaccurate estimated value of  $x$  are 3 and 2, respectively. (a) Overall blocking probability. (b) Maximum blocking probability.

Fig. 12. Robustness of the proposed method under estimation uncertainty of traffic load. Network is irregular and traffic is nonuniform with  $x = 3$ . The true and the inaccurate estimated traffic load are 100 and 90 Erlangs, respectively. (a) Overall blocking probability. (b) Maximum blocking probability.

tion experiments on the irregular network using different kinds of random number generators and different seeds. Fig. 10 shows the results. We see that our method is robust, and the blocking probability is relatively less sensitive to the uncertainty than the maximum blocking probability.

Figs. 11 and 12 show that our method is robust under estimation uncertainty of traffic pattern and traffic load respectively. In particular, the blocking probability is relatively less sensitive to the uncertainty than the maximum blocking probability.

## VI. CONCLUSION

In this paper, we proposed a set of algorithms for allocating FWC's in all-optical networks. We adopted the simulation-based optimization approach, in which we collect utilization statistics of FWC's from computer simulations and then perform optimization to allocate the FWC's. Therefore, our algorithms are widely applicable and are not restricted to any particular network model or assumption. After optimization, different nodes may be allocated different number of FWC's. To utilize these FWC's efficiently, we proposed a routing and wavelength assignment algorithm.

We have conducted extensive computer simulations on regular and irregular networks under both uniform and nonuniform

traffic. Compared with the best existing allocation, our method can significantly reduce the overall blocking probability (i.e., better mean quality of service) and the maximum blocking probability (i.e., better fairness). Equivalently, for a given performance requirement on blocking probability, our method can significantly reduce the number of FWC's required. In addition, we demonstrated that our simulation-based optimization approach is robust under simulation and estimation uncertainty.

## APPENDIX A

### GENERATION OF AN IRREGULAR NETWORK

The irregular network is randomly generated. To ensure that the resulting network is not far from the reality, we adopt the following generation method.

- 1) Start from the  $10 \times 10$  mesh network with 100 nodes and 180 bidirectional links.
- 2) Randomly delete 20 links from the network while ensuring that the resulting network is not disconnected.
- 3) Randomly add 30 links to the network as follows. For the  $j_1$ th node on the  $i_1$ th row and the  $j_2$ th node on the  $i_2$ th row, we define the *distance* between them as follows:

$$d_{(i_1, j_1), (i_2, j_2)} = \sqrt{(i_1 - i_2)^2 + (j_1 - j_2)^2}. \quad (\text{A.1})$$

To ensure that a node is not directly connected to a very far-away node, we randomly select two nodes, and add a link between them if and only if: 1) there is no existing link between them and 2) their distance is not larger than  $3\sqrt{2}$ . This step is repeated until 30 links have been added.

We execute the above steps to get a sample network for our simulation experiments. This network is irregular with 100 nodes and 190 bidirectional links. The path length between any two nodes varies from 1 to 11 and the average is 5.1628 hops. The number of links connected to a node varies from 2–6 and the average is 3.8.

## APPENDIX B

### DEFINITION OF THE TRAFFIC PATTERNS

The arrivals of transmission requests follow a Poisson process and the total arrival rate is  $\lambda_T$ . The duration of each transmission is exponentially distributed. The *traffic matrix* is  $\mathbf{I} = [I_{i,j}]_{N \times N}$  where  $\mathbf{I}_{i,i} = 0$  and  $I_{i,j}$  ( $i \neq j$ ) denotes the probability that there is a transmission request from node  $i$  to node  $j$  [20]. Therefore, the arrival rate from node  $i$  to node  $j$  is  $\lambda_T I_{i,j}$ . When all the nondiagonal entries of  $\mathbf{I}$  are equal to each other, the traffic is uniform; otherwise, the traffic is nonuniform.

The nonuniform traffic on the regular network is as follows. Based on the network topology shown in Fig. 4, we divide the network into nine parts, as follows:

<b>1</b> (4 × 4)	<b>2</b> (4 × 3)	<b>3</b> (4 × 4)
<b>4</b> (3 × 4)	<b>5</b> (3 × 3)	<b>6</b> (3 × 4)
<b>7</b> (4 × 4)	<b>8</b> (4 × 3)	<b>9</b> (4 × 4)

Let  $I'_{i,j}$  (where  $\mathbf{i}$  could be equal to  $\mathbf{j}$ ) denote the probability that there is a transmission request between two different nodes in the  $\mathbf{i}$ th part and the  $\mathbf{j}$ th part, respectively. The nonuniform traffic is defined as follows:

$$\begin{cases} I'_{1,3} = \frac{5}{25784} \\ I'_{1,7} = I'_{3,9} = \frac{3}{25784} \\ I'_{1,9} = I'_{3,7} = \frac{8}{25784} \\ I'_{\mathbf{i},\mathbf{j}} = \frac{1}{25784}, & \text{otherwise} \end{cases} \quad (\text{B.1})$$

where 25784 is a normalization constant and it ensures that probabilities sum to one.

The nonuniform traffic on the irregular network is as follows. We divide the network into the two parts where the nodes in the upper five rows belong to part **1** and those in the lower five rows belong to part **2**. The nonuniform traffic

is defined as follows:

$$\begin{cases} I'_{1,2} = \frac{x}{4900 + 5000 \cdot x} \\ I'_{1,1} = I'_{2,2} = \frac{1}{4900 + 5000 \cdot x} \end{cases} \quad (\text{B.2})$$

where  $x$  is a parameter such that a larger  $x$  specifies a more nonuniform traffic, and  $(4900 + 5000 \cdot x)$  is a normalization constant.

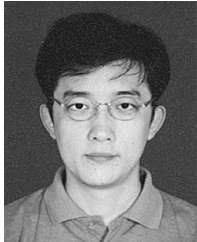
### ACKNOWLEDGMENT

The authors would like to gratefully acknowledge Dr. M. Kovacevic for giving his programs to them and helpful discussions, and Dr. S. Subramaniam for his patient explanation of his results reported in [7]. In addition, they would like to thank the anonymous reviewers. They gave many helpful and insightful comments that improved the quality of the paper.

### REFERENCES

- [1] B. Mukherjee, "WDM-based local lightwave networks—Part II: Multihop systems," *IEEE Network*, pp. 20–32, July 1992.
- [2] C. A. Brackett, "Dense wavelength division multiplexing networks: principles and applications," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 948–964, Aug. 1990.
- [3] P. E. Green Jr., "Optical Networking update," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 764–779, June 1996.
- [4] B. S. Glance, J. M. Wiesenfeld, U. Koren, and R. W. Wilson, "New advances in optical components needed for FDM optical networks," *J. Lightwave Technol.*, vol. 11, pp. 882–890, May/June 1993.
- [5] M. Kovacevic and A. Acampora, "Benefits of wavelength translation in all-optical clear-channel networks," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 868–880, June 1996.
- [6] R. Ramaswami and K. N. Sivarajan, "Routing and wavelength assignment in all-optical networks," *IEEE/ACM Trans. Networking*, vol. 3, pp. 489–500, Oct. 1995.
- [7] S. Subramaniam, M. Azizoglu, and A. K. Somani, "All-optical networks with sparse wavelength conversion," *IEEE/ACM Trans. Networking*, vol. 4, pp. 544–557, Aug. 1996.
- [8] K.-C. Lee and V. O. K. Li, "A wavelength-convertible optical network," *J. Lightwave Technol.*, vol. 11, pp. 962–970, May/June 1993.
- [9] H. Harai, M. Murata, and H. Miyahara, "Allocation of wavelength convertible nodes and routing in all-optical networks," in *Proc. SPIE, All-Optical Communication Systems: Architecture, Control and Network Issues III*, Oct. 1997, vol. 30, pp. 277–287.
- [10] S. Subramaniam, M. Azizoglu, and A. K. Somani, "On the optimal placement of wavelength converters in wavelength-routed networks," in *Proc. INFOCOM'98*, vol. 2, pp. 902–909.
- [11] J. Yates, J. Lacey, D. Everitt, and M. Summerfield, "Limited-range wavelength translation in all-optical networks," in *Proc. IEEE INFOCOM'96*, vol. 3, pp. 954–961.
- [12] V. Sharma and E. A. Varvarigos, "Limited wavelength translation in all-optical WDM mesh networks," in *Proc. IEEE INFOCOM'98*, vol. 2, pp. 893–901.
- [13] T. Tripathi and K. Sivarajan, "Computing approximate blocking probabilities in wavelength routed all-optical networks with limited-range wavelength conversion," in *Proc. INFOCOM'99*, vol. 1, pp. 329–336.
- [14] R. A. Barry and P. A. Humblet, "Models of blocking probability in all-optical networks with and without wavelength changers," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 858–867, June 1996.
- [15] A. Birman, "Computing approximate blocking probabilities for a class of all-optical networks," *IEEE J. Select. Areas Commun.*, vol. 14, pp. 852–857, June 1996.
- [16] I. Chlamtac, A. Ganz, and G. Karmi, "Lightpath communications: An approach to high bandwidth optical WAN's," *IEEE Trans. Commun.*, vol. 40, pp. 1171–1182, July 1992.
- [17] A. Schrijver, *Theory of Linear and Integer Programming*. New York: Wiley, 1986.
- [18] K. Bala, T. E. Stern, D. Simchi-Levi, and K. Bala, "Routing in a linear lightwave network," *IEEE/ACM Trans. Networking*, vol. 3, pp. 459–460, Aug. 1995.

- [19] Z. Zhang, D. Guo, and A. Acampora, "Logarithmically scalable routing algorithms in large optical networks," in *Proc. IEEE INFOCOM'95*, vol. 3, pp. 1290–1299.
- [20] H. Yoon, M. T. Liu, K. Y. Lee, and Y. M. Kim, "The knockout switch under nonuniform traffic," *IEEE Trans. Commun.*, vol. 43, pp. 2149–2156, June 1995.
- [21] A. Ganz and Y. Gao, "Time-wavelength assignment algorithm for high performance WDM star based system," *IEEE Trans. Commun.*, vol. 42, pp. 1827–1836, Feb./Mar./Apr. 1994.



**Gaoxi Xiao** was born in Xi'an, China, in 1970. He received the Ph.D degree from Hong Kong Polytechnic University in 1999.

He is currently a postdoctoral Research Fellow at the Department of Electrical Engineering, Polytechnic University, Brooklyn, NY. His main research interests include optical networks, TCP/IP, and analysis and design of optimization algorithms.



**Yiu-Wing Leung** (M'92–SM'96) received the B.Sc. and Ph.D. degrees from the Chinese University of Hong Kong in 1989 and 1992, respectively.

Since 1997, he has been an Associate Professor in the Department of Computer Science, Hong Kong Baptist University. His research interests include computer and communication networks, internet computing, multimedia systems, and algorithmic and heuristic techniques. He has published more than 40 journal papers in these areas, most of which have appeared in various IEEE journals.