An Efficient Heuristic for Routing and Wavelength Assignment in Optical WDM Networks

Yongbing Zhang†, Koji Taira‡, Hideaki Takagi†, and Sajal K. Das*

†Institute of Policy and Planning Sciences, University of Tsukuba, Tsukuba, Ibaraki 305–8573 Japan
‡Doctoral Program in Systems and Information Engineering, University of Tsukuba, Tsukuba, Ibaraki 305–8573 Japan
*Department of Computer Science and Engineering, University of Texas at Arlington, Texas 76019–0015 USA

***Abstract*—We propose an efficient heuristic algorithm that sets up and releases lightpaths for connection requests dynamically. We partition the routing and wavelength assignment (commonly known as RWA) problem into two subproblems and solves both of them using a well-known shortest path routing algorithm. For solving the routing subproblem, an auxiliary graph is created whereby the nodes and links in the original network are transformed to the edges and vertices, respectively, and the availability of each wavelength on the input and output links of a node as well as the number of available wavelength converters are taken into account in determining the weights of edges. Furthermore, for solving the wavelength assignment subproblem, an auxiliary graph is also utilized and the cost for wavelength conversion is taken into consideration in the edge weight function. A distinguished feature of our algorithm is that it employs more accurate network information on the availability of both the wavelengths and the wavelength converters than the existing algorithms in deciding the routing and the wavelength assignment. Simulation results show that our algorithm performs much better than previously proposed algorithms with comparable computation time, especially when the number of wavelengths is large while the number of converters at each node is limited.**

I. Introduction

Wavelength-division multiplexing (WDM) is emerging as the dominant technology for the next generation optical networks [1]. Using WDM, multiple signals, distinguished by their wavelengths, can be transmitted on a single fiber and each wavelength operates at its peak speed. In all-optical networks, all nodes or a limited number of nodes may have wavelength conversion function that can convert one input wavelength into another different output wavelength in order to increase the wavelength utilization. The device that performs the wavelength-conversion is called the *wavelength converter*, and is usually expensive due to the economic and spatial factors [2].

A route (a set of links) traversed by data between a source-destination (*s-d*) pair forms an all-optical path with a wavelength assigned on each link and it is called a *lightpath*. In this paper, we consider that a lightpath is assigned to a connection corresponding to each user connection request for its entire duration. Given a set of connection requests, how to set up lightpaths for them is called the *routing and wavelength assignment* (RWA) problem [3]. The objective of an RWA algorithm is to set up lightpaths and assign wavelengths in a manner which minimizes the amount of the request blocking.

In this paper, we propose a new heuristic algorithm, called wavelength- and converter-aware (WCA) algorithm, that solves the dynamic RWA problem efficiently. The WCA algorithm first solves the routing subproblem and then the wavelength assignment subproblem. Both subproblems are formulated as routing problems and solved using a well-known shortest path routing algorithm. The key part of the heuristic algorithm is how to determine the weight function for the edges of the auxiliary graphs for the two subproblems. The advantage of our algorithm is that in routing decisions the availability of both the wavelengths and the wavelength converters and in wavelength assignment the cost of wavelength converters are taken into account. The algorithm yields significant improvements in terms of the request blocking probability over traditional techniques, especially when the number of wavelengths is large while the number of wavelength converters is limited.

II. System Model

A WDM-routed network can be modeled by a directed graph $G(N, L)$, where N and L denote the sets of nodes and communication links, respectively. For simplicity, N and L are also used to denote the numbers of nodes and links, respectively. The bandwidth of each optical fiber link is divided into a set of W wavelengths as communication channels. A connection request of an *s-d* pair is served by setting up a lightpath that is a series of channels belonging to the immediate nodes along the path from the source s to the destination d. The connection occupies the channels until it terminates. It is assumed that connection requests arrive at each node independently and follow the Poisson process. The occupation time of a lightpath by a connection is assumed to be exponentially distributed.

Besides transmitting and receiving signals, each node provides the optical switching functions such as switching a

0-7803-7400-2/02/$17.00 (C) 2002 IEEE

wavelength of a connection from an input end to an output end and converting an input wavelength to a different output wavelength. In this paper, it is assumed that each node have a limited number of converters that are shared in the node, and the wavelength conversion is full range of the waveband; *i.e.*, a converter can convert one input wavelength to any other output wavelength.

It is assumed that the routing and the wavelength assignment (RWA) algorithm is decentralized; *i.e.*, the routing and the wavelength assignment decisions are made at each node autonomously. Each node dynamically broadcasts its state on both the wavelength and the converter availability to all other nodes in the network and receives the state information from other nodes. It then determines independently the best route for each arriving connection request. It is assumed that there is one or more dedicated wavelengths for information exchange among the nodes. A routing or wavelength assignment decision for a new connection request is not allowed to affect the existing connections. The objective of our algorithm is therefore to find the *best route* from the source to the destination and to assign the *best wavelength* on each link along the best route for each connection request so that the *blocking probability* of connection requests is minimized.

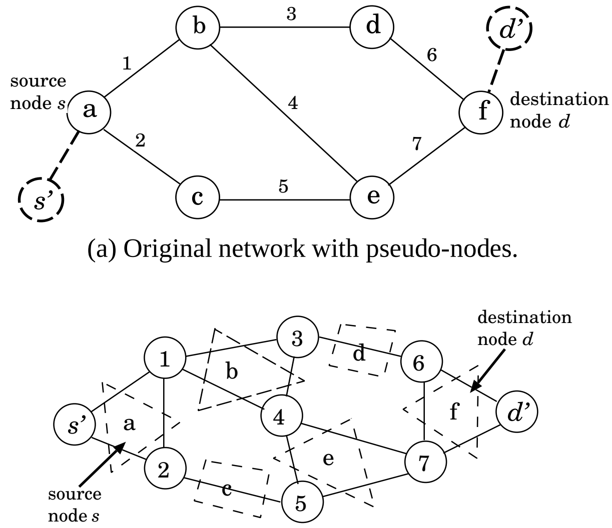
III. Proposed Solution

The WCA algorithm proposed in this paper consists of two components: the routing algorithm and the wavelength assignment algorithm. The overall approach includes the following four steps.

1) *Graph transformation for routing*: Transform the original network to its corresponding auxiliary graph.
2) *Determination of the best route*: Solve the routing algorithm using Dijkstra's algorithm to find the best route between the *s*-*d* pair. If no route with finite length is found, reject the connection request; otherwise, go to Step 3.
3) *Graph transformation for wavelength assignment*: Transform the route in the original network determined in Step 2 to its corresponding auxiliary graph.
4) *Wavelength Assignment*: Solve the wavelength assignment problem using Dijkstra's algorithm to find the best wavelength on each link in order to set up a lightpath. If no lightpath can be set up, reject the connection request; otherwise, accept the connection request.

A. Source-Destination Routing Algorithm

An auxiliary graph of the original network is created by considering the specific characteristics of optical networks in order to determine the best route for an *s*-*d* pair. Before constructing the auxiliary graph, two pseudo-nodes are added to the *s*-*d* pair in the original network. The pseudo-nodes s' and d', called the *pseudo traffic input* and *output points*, are connected to the source s and the destination d with zero cost *pseudo-links*, respectively, as shown in Figure 1(a). The nodes and links in the original network are respectively transformed to the edges and vertices in the auxiliary graph as shown in Figure 1(b). A possible path through an edge with or without wavelength conversion is called an *edge-path* of the edge.



(a) Original network with pseudo-nodes.

(b) Auxiliary graph.

Fig. 1. Illustration of graph transformation.

1) Graph transformation: For a given network $G(N, L)$ and an *s*-*d* pair, an auxiliary graph $G(V, E)$ is created, where V and E denote the sets of vertices and edges, respectively. The procedures of the graph transformation can be listed as follows.

1) For an *s*-*d* pair, add the pseudo-nodes, s' and d', as the pseudo traffic input and output points and connect them respectively to the source and destination nodes with zero cost pseudo-links as shown in Figure 1(a).
2) Create nodes in the auxiliary graph to denote the links in the original network. Note that the pseudo-nodes are treated as links in the original network.
3) Create edges in the auxiliary graph that denote nodes connecting two links in the original network.

2) Determination of the edge weight: In a wavelength-routed network, a lightpath may not be able to pass through a node for lack of wavelength conversion capacity, even though there are free wavelengths both on the input and output links of the node. It is therefore necessary to consider the states of both the input and output links of a node, *i.e.*, the states of edge-paths of each edge in the auxiliary graph, in routing decisions. In this paper, the states of available wavelengths on the input and output links at each node, the conversion capacity, and the number of available converters are taken into consideration. In order to introduce the edge weight function, the following notation is used.

C total number of converters at a node

W total number of wavelengths a link or a node handles

c number of converters available at a node

w_{in} number of wavelengths available at the input ends of a node

w_{out} number of wavelengths available at the output ends of a node

w number of available edge-paths of an edge on the route of an *s-d* pair that a lightpath can pass the edge without wavelength conversion

w' number of available edge-paths of an edge on the route of an *s-d* pair that a lightpath can pass the edge with wavelength conversion, *i.e.*, $w' = \min\{w_{in} - w, w_{out} - w, c\}$.

The weight of each edge of the auxiliary graph is determined based on the probability of the available edge-paths of an edge. By supposing that the probability of a free edge-path on an edge neither from the source nor to the destination at some time in future is $\dfrac{w + w'}{W}$, the probability that an edge-path will be occupied is given by $1 - \dfrac{w + w'}{W}$. Therefore, the probability that all the edge-paths will be used is given by $\left(1 - \dfrac{w + w'}{W}\right)^{w+w'}$. Then, the probability that at least one edge-path will be free on an edge i in future is given by

$$p_i = 1 - \left(1 - \frac{w + w'}{W}\right)^{w+w'}.$$

The weights of edges originated from the source and targeted to the destination are determined differently from others. By assuming that the probability of a free edge-path on an edge originated from the source in future is w_{out}/W, then the probability that at least one edge-path will be available is given by

$$p_i = 1 - \left(1 - \frac{w_{out}}{W}\right)^{w_{out}}.$$

On the other hand, since data can always arrive at the destination if its input link is free, the probability that at least one edge-path on an edge to the destination will be available in future is always 1; *i.e.*, $p_i = 1$. For the above three types, the weight of an edge i, ρ_i, in the auxiliary graph is defined by the following function as in [4],

$$\rho_i = -\log p_i.$$

The above equation implies that an edge with a higher probability of free edge-paths has a lower value of the edge weight and that the edge weight becomes infinity if there is no free edge-path. Note that ρ_i is determined by the current status of both a node and a link, and is constantly changing.

3) Routing algorithm: Given an auxiliary graph $G(V, E)$ and a connection request between the pseudo traffic input and output points, s'-d' pair, Dijkstra's algorithm is used to search for the min-cost path corresponding to the route in the original network. In the auxiliary graph, however, a route may pass through consecutively the same node twice while it is impossible in the original network. In order not to let this situation to occur, the following constraint is introduced to the routing algorithm.

Constraint: It is prohibited in the routing algorithm that the chosen route goes through two edges belonging to the same node successively in the original network. For example, a route is not allowed to pass from vertex 1 to 3 and then from vertex 3 to 4 in Figure 1(b).

B. Wavelength Assignment Algorithm

In this paper, three wavelength assignment schemes are newly proposed.

1) First Fit Wavelength First (FFW): The FFW algorithm is an extension of the first-fit (FF) algorithm that is commonly used for performance comparison in the literature [5]. The source node attempts to find and assign a free wavelength that is found first along the route determined by the routing algorithm. This algorithm searches for a free wavelength on a link in a predefined order and attempts to use a wavelength converter whenever a wavelength conversion is needed. The connection request will be forwarded to the next node along the route when the trial for finding a free wavelength succeeds. If the request fails, the source node gets the feedback and a different wavelength will be chosen. This process is repeated until there is one free wavelength available or the lightpath cannot be set up.

2) LEast Converter First (LEC): Since the number of wavelength converters at a node may be much less than the number of wavelengths on a link, it is natural to take this factor into account in assigning wavelengths. The LEC algorithm treats the wavelength assignment problem as a routing problem and employs Dijkstra's algorithm to find the solution. An auxiliary graph is created by using an approach similar to that in [6], but in LEC only the best route determined by the routing algorithm needs to be considered.

The weight of a channel edge is determined similarly to that in [6]. That is, an idle channel edge has weight f where f is a positive constant while an occupied channel edge has weight of infinity. The weight of a converter edge is g and is larger than the sum of any free path without wavelength conversion, *i.e.*, $g > nf$ where n is the path length from the source to the destination. Furthermore the weight of an edge corresponding to a switching operation without wavelength conversion at a node is set to zero. The LEC algorithm therefore attempts to set up a lightpath for a connection request with the lowest cost, *i.e.*, with the least number of converters. If no lightpath can be set up, then the connection request will be rejected.

3) Least Conversion Cost First (LCC): This algorithm is implemented similarly to LEC. However, the weight function of LCC is different from that of LEC in the sense that LCC employs a nonlinear cost function for using converters. It is assumed that using a free converter at a node where the

TABLE I

AUXILIARY GRAPH COMPLEXITY OF THE ALGORITHMS*

Algorithm	Complexity of the algorithms					
	$W = 8$		$W = 16$		$W = 32$	
	# vertices	# edges	# vertices	# edges	# vertices	# edges
HW	(14, 37)	(21, 101)	(14,72)	(21, 435)	(14, 142)	(21,1337)
TAW	(14, 51)	(21, 172)	(14,98)	(21,592)	(14, 188)	(21,2103)
NEW	(21, 49)	(49–53,161)	(21,92)	(49–53,538)	(21,175)	(49–53,1893)
TRWA	364	3720	700	13200	1372	49440

converter utilization is higher should pay higher cost (higher penalty). The cost function, $c_i(u_i)$, for using a converter at node i is defined as follows.

$$c_i(u_i) = \frac{1}{C - u_i},$$

where u_i denotes the number of wavelength converters in use at node i. The weight, $\rho_i(u_i)$, of an edge i corresponding to a switching operation with wavelength conversion at a node is defined to be the *differential function* of the above cost function; *i.e.*,

$$\rho_i(u_i) = \frac{dc_i(u_i)}{du_i} = \frac{1}{(C - u_i)^2},$$

IV. PERFORMANCE EVALUATION

Simulation experiments are used to evaluate our heuristic algorithm, WCA, and compare it with other algorithms, Hop-based (HW) and Total wavelengths and Available Wavelength (TAW) in [4] and Total Routing and Wavelength Assignment (TRWA) in [6]. The routing algorithm in HW attempts to set up a lightpath from the source s to the destination d using the smallest number of hops. On the other hand, the routing algorithm in TAW attempts to set up a lightpath with the smallest edge weight that are determined by the available wavelengths and the total wavelengths on each link. TRWA combines the routing and the wavelength assignment algorithms and attempts to set up a lightpath for each request in one operation. The network model used in the simulation is the NSFNET model (shown in Figure 2), *i.e.*, there are 14 nodes and 21 duplex links. It is assumed that each link has the same number of wavelengths W and each node has the same number of wavelength converters C. The performance metrics used for comparison are (i) the blocking probability of the connection requests, and (ii) the computational complexity.

It is assumed that connection requests from a node to each of the others nodes are generated with the equal probability and the request arrivals follow the Poisson process with rate

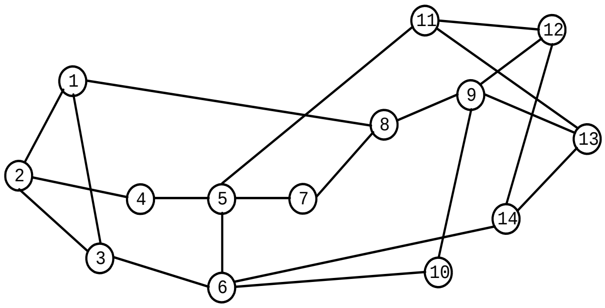


Fig. 2. A wavelength-routed network.

λ. The connection duration time is assumed to be exponentially distributed with mean of 1 time unit. The number of wavelengths W is varies as 8, 16, and 32. The results for 8 and 16 wavelengths are not shown in the figures due to the space limitation. The wavelength conversion factor, denoted by f, is defined to be the ratio of the number of converters at a node to the number of wavelengths; *i.e.*, $f = C/W$, and is changed as 0, 10%, 20%, 50%, and 100%, where 0 means there are no converters whereas 100% means there are full converters. Simulation experiments are run on an 8-CPU Sparc workstation. In each experiment, 200,000 connection requests are generated and the performance metrics are shown as an average in the figures.

TABLE II

COMPUTATION TIME BOUNDS OF THE ALGORITHMS.

Algorithm	Computation time
HW	$O(N^2 + \bar{N}W \log(\bar{N}W) + \bar{N}W^2)$
TAW	$O(N^2 + \bar{N}W \log(\bar{N}W) + \bar{N}W^2)$
NEW	$O(N^3 + \bar{N}W \log(\bar{N}W) + \bar{N}W^2)$
TRWA	$O(N^4W^2)$

Table I shows the numbers of vertices and edges in the auxiliary graphs that the algorithms have to handle. Since the computation time of a routing algorithm depends on its graph complexity, *i.e.*, the numbers of vertices and edges, the com-

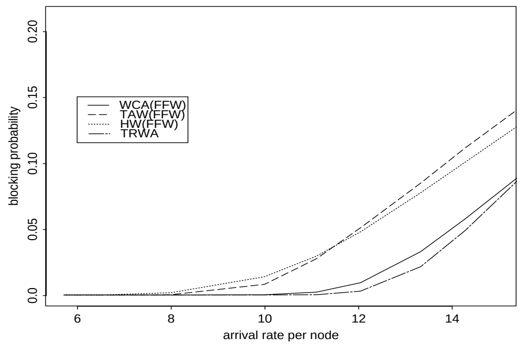
*The two-tuples (x, y) in HW, TAW, and NEW denote the number of vertices or edges in the first and the second subproblems, respectively. The numbers of vertices and edges of the second subproblem in HW, TAW, or NEW are calculated using $\bar{N}$.

putation time of an algorithm can be estimated using its computational complexity. Given a network $G(N, L)$, the computation time of Dijkstra's algorithm depends on how to determine the next node of the minimum shortest-path weight from the source and is $O(N^2)$ by using a simple scheme while $O(N \log N + L)$ by using a Fibonacci heap [7]. In this paper, the latter approach is used.

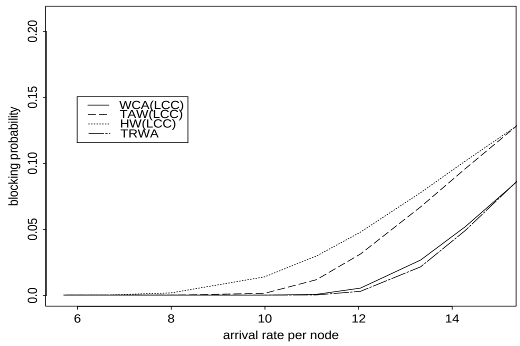
It can be easily shown that the computation times of HW and TAW are bound by $O(N^2)$ while the computation time of WCA is bound by $O(N^3)$. Note that the complexity of the auxiliary graph in WCA depends on the sparsity of the original network, and therefore the number of edges in WCA is much smaller than $O(N^3)$ when the network is sparse. Note also that the computation time of WCA can be reduced much more in practice because of the constraint described in Section 3. Since the auxiliary graph for wavelength assignment in HW, TAW, or WCA is created using the same method and forms a chain of nodes from the source to the destination, the numbers of vertices and edges of their auxiliary graphs become $O(\bar{N}W)$ and $O(\bar{N}W^2)$, respectively, where $\bar{N}$ denotes the average length (number of hops) of the best route from the source to the destination. In most cases, $\bar{N} \ll N$ while $\bar{N} = O(N)$ in the worst case. Therefore, the computation time for wavelength assignment is bound by $O(\bar{N}W \log(\bar{N}W) + \bar{N}W^2)$. On the other hand, since the numbers of vertices and edges in TRWA are $O(N^2W)$ and $O(N^4W^2)$, respectively, its computation time is therefore bound by $O(N^4W^2)$ [6]. The worst case computation times of the algorithms are summarized in Table II.

Figure 3 shows the probability versus traffic load with 32 wavelengths when the conversion factor, f, is 20%. As expected, the TRWA algorithm performs the best among the algorithms, but its computational complexity is exhaustive as shown in Table I. It shows that TRWA may not be practical for a real system, especially in a large network with a large number of wavelengths. On the other hand, WCA yields a blocking probability close to that of TRWA while the computation time is close to that of the other algorithms. It can be observed that WCA outperforms significantly both the HW and TAW algorithms over a wide range of traffic load and that the improvement gain becomes obvious as the number of wavelengths increases (*e.g.*, over 40% when f = 20%). Note that HW may behave better than TAW when the traffic load is high. This is because when the traffic load becomes saturated, a route with the least hops yields less waste of system resources and therefore provides better performance.

In addition to Figure 3, it can be observed from Figure 4 that LCC can improve the performance further over FFW even though the performance gain depends on the routing algorithms. An efficient routing algorithm provides performance improvement large enough and therefore leaves little room to a wavelength assignment algorithm to acquire further performance improvement. Another observation is that a larger performance gain can be obtained for a longer route from the source to the destination. In the NSFNET model under consideration, the average lengths of a route of HW, TAW, and WCA are 2.2, 2.9, and 2.7, respectively, and TAW provides the largest performance gain.



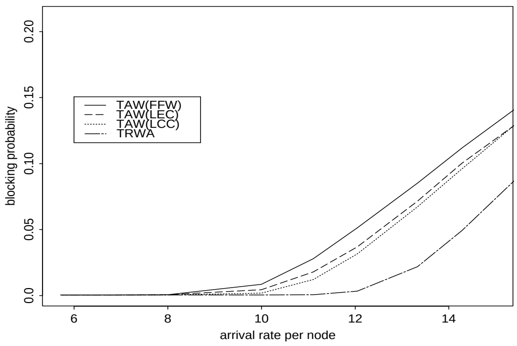
(a) FFW



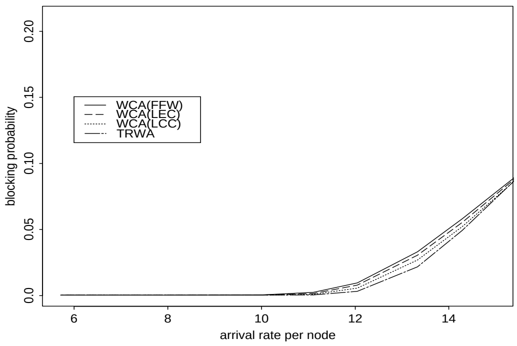
(b) LCC

Fig. 3. Blocking probability vs. traffi c load with 32 wavelengths and 20% converters.

Figure 5 shows the comparison of WCA and TAW with 32 wavelengths and various degrees of the conversion factor, f. It can be observed that the performance (blocking probability) is sensitive to f when it is small (*e.g.*, $f = 10$ or 20%); but if f becomes larger (*e.g.*, $f = 50\%$) there is little room for any further improvement over the performance. This trend can be observed clearly when the number of wavelengths becomes large. This coincides with the fact stemmed from the effectiveness of WCA; *i.e.*, only a small number of converters is enough to provide good performance close to the case with the full number of converters. On the other hand, the TAW algorithm is much more sensitive to the number of converters as shown in Figure 5(b), and its performance becomes comparable to WCA only when the number of converters is close to the number of wavelengths.

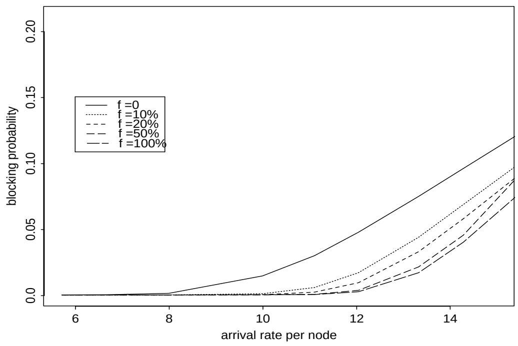


(a) TAW

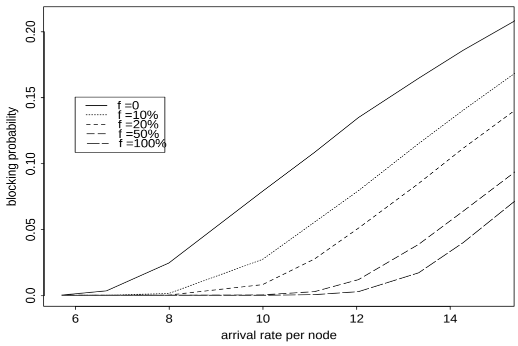


(b) NEW

Fig. 4. Comparison of various wavelength assignment algorithms with 32 wavelengths and 20% conversion ratio.



(a) WCA



(b) TAW

Fig. 5. Comparison of WCA and TAW with 32 wavelengths and various conversion factors.

V. Conclusions

In this paper, a new efficient heuristic algorithm, called WCA, for WDM-routed optical networks is proposed. The algorithm is implemented by partitioning the routing and the wavelength assignment into two subproblems so that the computation time is reduced largely. In determining the best route for an *s*-*d* pair, the states of the available wavelengths on the input and output links of a node along with the number of converters are taken into account in the edge weight function. Furthermore, in determining the best wavelengths to set up a lightpath along the best route the cost for using converters is introduced in the weight function of the auxiliary graph so that the selected wavelengths yields the least conversion cost.

The simulation results show that WCA outperforms significantly other algorithms with comparable computation time in an NSFNET model. Furthermore, the results also show that WCA is specially effective when the number of wavelengths is large while the number of wavelength converters is limited. For example, using WCA the performance can be improved over 30% on TAW or HW when the number of wavelengths is equal to or larger than 16 and the conversion factor is 20%.

References

[1] R. Ramaswami and K.N. Sivarajan. *Optical Networks–A Practical Perspective, 2nd Ed.* Morgan Kaufmann Publishers, Inc., San Francisco, 2002.

[2] L. Li and A. Somani. Dynamic wavelength routing using congestion and neighborhood information. *IEEE/ACM Trans. Networking*, 7(5):779–786, October 1999.

[3] Z. Zhang and A.S. Acampora. A heuristic wavelength assignment algorithm for multihop WDM networks with wavelength routing and wavelength re-use. *IEEE/ACM Trans. Networking*, 3(3):281–288, June 1995.

[4] T. Fabry-Asztalos, N.M. Bhide, and K. M. Sivalingam. Adaptive weight functions for shortest path routing algorithms for multi-wavelength optical WDM networks. In *Proc. IEEE ICC'00*, pages 1330–1334, June 2000.

[5] A. Mokhtar and M. Azizoglu. Adaptive wavelength routing in all-optical networks. *IEEE/ACM Trans. Networking*, 6(2):197–206, April 1998.

[6] K.C. Lee and V.O.K. Li. A wavelength-convertible optical network. *J. Lightwave Tech.*, 11(5/6):962–970, May/June 1993.

[7] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press, Cambridge, Massachusetts, 2001.