An efficient algorithm for optimal allocation of wavelength converters in wavelength routing optical network

Wei Li (李 蔚)[1], Jun He (何 军)[1], Yuan Li (李 源)[2], Deming Liu (刘德明)[1], and Dexiu Huang (黄德修)[1]

[1]*Department of Optoelectronic Engineering, Huazhong University of Science and Technology, Wuhan 430074*

[2]*Department of Computer Science, Central China Normal University, Wuhan 430070*

Received April 19, 2004

In a wavelength routing optical network (WRON), the optimal allocation of wavelength converters (WCs) is very important to minimize the number of WCs, enhance the fiber utilization, reduce the blocking probability, etc.. In this paper, a novel simplified network model with shared WCs has been proposed. An effective algorithm for optimal allocation of shared WCs has been presented by using a revised Dijkstra algorithm and genetic algorithm (GA). The effectiveness of the revised algorithm was verified through the simulation on Nature and Science Foundation (NSF) net of USA. This revised algorithm can achieve blocking probability 36% less than the algorithm in previous work, and the calculating time of the minimum blocking probability can be reduced dramatically.

OCIS codes: 060.4510, 060.4250.

Wavelength routing optical network (WRON) is the current practicable solution for intelligent optical network. In WRON, a link should be set up between two nodes on the same wavelength. It is the so-called wavelength continuity constraint. The all-optical wavelength converters (WCs) transform the input wavelength to the desirable output wavelength to improve the utilization efficiency of wavelengths and reduce the wavelength blocking probability, which seems more suitable for the all-optical network. However, they are very expensive and the operating spectrum is very limited. So the optimal allocation is very important for reducing the necessary number of WCs[1,2].

The WCs allocation is a combinatorial optimization problem due to the regional searching of feasible solutions, which usually accompanies with many constraints. Several models and algorithms have been proposed in Ref. [3−8]. Thiagarajan[3] introduced a network model with arbitrary topologies and presented an optimal converter placement algorithm using auxiliary graphs. Siregar[4] followed the model of Ref. [3] and used genetic algorithm (GA) for placing full-range WCs. The authors[5−8] studied the placement of limited range WCs and calculated the approximate blocking probabilities based on different heuristic algorithms. Harai[7] discussed the placement of full-range WCs in shared-per-link structure.

All above researches are proved to be extremely time-consuming and not very efficient. In this paper, in order to get a more efficient algorithm, a novel simplified network model with shared WCs has been proposed, which extends and simplifies from the layered graph model in Ref. [3] for shared limited-range wavelength conversion. An effective algorithm has also been presented to optimize allocation of shared WCs by using a revised Dijkstra algorithm and GA. The effectiveness of the revised algorithms was verified through the simulation on NSF net (a typical network funded by National Natural Science Foundation of USA). It is shown that this revised algorithm can realize blocking probability 36% less than the algorithm in Ref. [3] and the calculating time for reaching the minimum blocking probability can be reduced dramatically.

As an improvement and extension of the model of Ref. [3], our network model can be introduced as a directed graph $G = (V, L)$, where V is the set of vertices and L is the set of directed edges, which represent nodes and the fiber links, respectively. l_{ij} is the directed link from node i to node j, and ρ_{ij} is the link load per wavelength

$$\rho_{ij} = \frac{\sum_{s,d} \lambda_{sd}}{F}, \tag{1}$$

where F is the number of wavelengths on each link, $\{\lambda_{sd}\}$ is the traffic matrix, λ_{sd} $(s \neq d)$ denotes the mean number of calls that arrives at source node s from destination node d per unit time, and $\lambda_{sd} = 0$ when $s = d$. The summation includes all (s, d) pairs whose path from s to d is on l_{ij}. It is assumed that the network λ_{sd} is so small that $\rho_{ij} < 1$ for all $i, j \in V$. The call durations are assumed to be exponentially distributed and the calls arrive at each node is a Poisson process. The load on a given wavelength of a link is assumed statistically independent of the link loads with respect to other wavelengths and links. This "independence assumption" is commonly used to analyze the blocking probability of the optical networks, which is always used in each generation of GA to yield the approximate value for the blocking probability very quickly[4].

In order to describe the WCs' placements, $C = (c(1), c(2), \cdots, c(K))$ is used to denote the converter placement vector. Let $1 \leq c(i) < c(i+1) \leq N$, $1 \leq i < K$. The entries of C denote the serial number of nodes in which converters are placed among the nodes sequence. N is the total node numbers and K is the node number which is equipped with converters.

For the nodes without WC, suppose the path p of an end-to-end call from s to d consists of successive links l_{ii_1}, $l_{i_1i_2}$, $\cdots$, l_{i_nj}, where i_1, i_2, $\cdots$, i_n are the nodes between i and j along path p. Considering the wavelength continuity constraint, $\bar{\rho}_{xy}$ is defined as the probability of a given wavelength for link l_{xy}. Then $\bar{\rho}_{ii_1}$, $\bar{\rho}_{i_1i_2}$, $\cdots$, $\bar{\rho}_{i_nj}$ is the probability that the given wavelength is available on all links over the path from i to j. Hence the probability that a call successfully finds a link from i to j is

$$f(i,j) = 1 - (1 - \bar{\rho}_{ii_1}\bar{\rho}_{i_1 i_2}\cdots\bar{\rho}_{i_n j})^F. \quad (2)$$

Let k be the number of converters which placed on the nodes of the path p, where $0 \le k \le K$. Following Ref. [4], the probability of the successfully establishing lightpath on the path p is

$$S_{sd}(C) = \prod_{i=0}^{k} f[c(i), c(i+1)], \quad (3)$$

where $c(0) = s$, $c(k+1) = d$, $C_{sd} = (c(1), c(2), \cdots, c(K)) \in C$ is the converter placement vector for the path p. Thus the blocking probability for the path p is

$$P_{sd}(C) = 1 - S_{sd}(C). \quad (4)$$

Finally, the blocking performance $\Gamma(C)$ of the network for the WCs placement vector C can be written as

$$\Gamma(C) = \frac{\sum_{s,d\in V}\lambda_{sd}P_{sd}(C)}{\sum_{s,d\in V}\lambda_{sd}}. \quad (5)$$

Equation (5) is the blocking probability of the dedicated switch structures network with full-range WCs.

Considering a shared limited-range conversion, the degree of translation D is

$$D = \frac{T}{F-1}, \quad (6)$$

where T is the number of output wavelengths, F is the number of usable wavelengths. In this paper, the degree of translation D is assumed to be same to each WCs.

Therefore, we simply define a shared factor δ_i to express the probability of a call which needs to be converted using converters in the node i

$$\delta_i = \frac{W_i}{F}, \quad (7)$$

where F is the number of layers and is equal to the number of the translated wavelengths, W_i is the number of the converter banks in the node i. $\delta_i = 1$ means F converters stored in ith node, which is a dedicated node structure, $\delta_i = 0.5$ means a half-clear wavelength-convertible switch, and $\delta_i = 0$ means there is no converters in the node i.

Unlike Ref. [6], in this paper, the factor δ_i is used to simplify analysis of the complex node structures as we assume that the wavelengths are assigned randomly with respect to links.

Based on our assuming, $\delta_{c(1)}$, $\delta_{c(2)}$, $\cdots$, $\delta_{c(k)}$ are the shared factors of nodes $c(1)$, $c(2)$, $\cdots$, $c(k)$, respectively. Equation (3) is revised as follows to calculate the probability of the successfully established lightpath with shared limited-range conversion

$$S_{sd}(C_{1,2,\cdots,k}) = (1 - D\times\delta_{c(1)})S_{sd}(C_{2,\cdots,k}) + D\times\delta_{c(1)}S_{c(1)d}(C_{2,\cdots,k}), \quad (8)$$

where $S_{sd}(C_{2,\cdots,k})$ is the probability of successfully establishing the link from s to d with converters $c(2)$, $c(3)$, $\cdots$, $c(k)$ and converters in $c(1)$'s are removed from the node, and $S_{c(1)d}$ $(C_{2,\cdots,k})$ is the probability of successfully establishing lightpath from $c(1)$ to d with $c(2)$, $c(3)$, $\cdots$, $c(k)$ and s subtracted from the path.

Therefore, the blocking probabilities can be calculated for all the paths from Eqs. (4), (5), and (8), and then the blocking performance $\Gamma(C)$ of the network for the converter placement vector C can be obtained.

Dijkstra algorithm can determine the shortest path to each pair of the source and destination nodes when the network topology is given[3,7]. In an optical network, the number of hops is used to measure the lengths of the links. Dijkstra algorithm does not consider the congestion control of the network. So in this paper, a revised Dijkstra algorithm is proposed by revising the hops numbers of the links considering the congestion.

The revised algorithm is described as follows:

Step 1) Choose a node n from the sequence V, which has not been selected yet;

Step 2) Dijkstra (n, $\{\lambda_{sd}\}$) procedure is used to obtain the shortest path node n;

Step 3) Revise the traffic matrix $\{\lambda_{sd}\}$ by the rule: If a shortest path goes through a set of links, each link in the set will be added to α;

Step 4) If all the nodes of the set V have been selected, finish the revised Dijkstra algorithm and built the shortest paths for every node of the set V, otherwise goes to Step 1).

Here α denotes a revised factor, which is used to increase the hops of a link. When the algorithm has decided the shortest path of a node, all chosen links' distance measurements are adjusted by the revised factors. If a link has been chosen m times, its distance measurement is added to $m \times \alpha$, where α must satisfy $\alpha < [1/(N(N-1))]$, because the hop is adjusted to control the congestion but cannot increase one real hop to a practical fiber at the same time, and then the shortest path of the network is changed. Let $\alpha = 1/(N \times N)$, therefore when the algorithm finds several links having equal hops, it will choose the link of the least chosen by their revised factors.

The revised Dijkstra algorithm is used to optimize the blocking probability performance in GA, which is shown in the simulation of NSF net.

GA is a very efficient heuristic algorithm for solving the nondeterministic polynomial-complete (NP-C) problem, especially in optics[9]. Generally, a suitable encoding of possible solution in a vector representation is needed. In this paper, the converter placement is represented by an array of values 0 and 1[9].

Using this special kind of coding, we first produce a so-called initial population by randomly generating a suitable number of vectors in the described manner. Furthermore it is necessary to evaluate the quality of these solution vectors. Therefore we apply Eq. (5) to the fitness function of GA. Now many iterations of the GA are executed in order to improve the quality of the given solution vectors (individuals), as shown in Fig. 1.

At the beginning of each iteration, some vectors of high quality are selected to produce better solution. There are two different methods used to produce new vectors, crossover, and mutation. After producing new vectors,

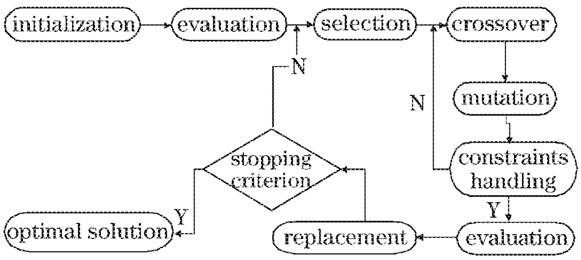

Fig. 1. The process of GA.

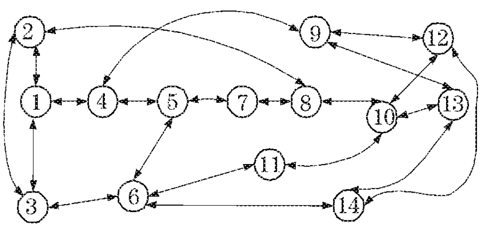

Fig. 2. NSF net with 14 nodes and 21 links.

constraints handling must run to remove infeasible children, whose "1" bits do not equal to their parents' and rebuild these children from their parents to satisfy constraints. When the numbers of children arrive at the population of the generation in GA, the next population will be constructed by substituting the worse solution vectors of parent generation.

In the present study, a simple stopping criterion is used based on the numbers of generations. Our algorithm step stops when the generation counter exceeds the preset maximum number of generations.

The effectiveness of the revised algorithms was verified through the simulation on NSF net of USA, shown as Fig. 2, which consists of 14 nodes. The optimal objective is the blocking probability performances of the network.

We assume $F = 5$ for the number of wavelengths on each link and $\lambda_{sd} = 0.1$ for the rate of calls generated for every pair of source and destination nodes, respectively, same as the condition of Ref. [3].

In our experiment, we assume $D = 100\%$ as the degree of translation. The GA parameters are defined as: population size = 10; probability of crossover = 0.6; probability of mutation = 0.005; maximum number of generations = 20.

We place $C_n = 10$ converter banks on NSF net. Using our optimal algorithm, let $r = 2$ and placement vector have 14 bits. The GA performances using the revised Dijkstra algorithm and Dijkstra algorithm are presented in Tables 1 and 2, respectively.

The results show that the performance of the blocking probability has been improved dramatically. The blocking probability can be reduced to 36% less than that in Ref. [3], when the number of genetic generations is 20 and the number of wavelengths is 7.

The results of pure GA and the optimal algorithm are shown in Table 3. The placement vector has 70 bits in pure GA and 14 bits in the optimal algorithm. Assuming that the node traffic is uniform, all the node loads are equal to 0.1, and $D = 100\%$, C_n =10.

The results of the optimal algorithm have been presented with different degrees of translation, shown in Fig. 4.

We can see that the optimal algorithm only spends 16 generations for reaching the best individual, while the pure GA spends 725 generations arriving at it. The results show that the best fitness value related to the blocking probability in the optimal algorithm in Fig. 4 is the same as the value in Fig. 3, where $\Gamma(C) = 0.0074254$.

Table 1. Optimal Convert Placement Based on the Dijkstra Algorithm for GA

R	Optimal Placement	Blocking Probability
1	(4)	0.014554
2	(4, 6)	0.010329
3	(4, 6, 8)	0.0074356
4	(4, 6, 8, 10)	0.0057422
5	(4, 5, 6, 8, 10)	0.0046977
6	(4, 5, 6, 8, 9, 10)	0.0037218
7	(2, 4, 5, 6, 7, 8, 9,10)	0.0027774

Table 2. Optimal Convert Placement Based on the Resived Algorithm for GA

R	Optimal Placement	Blocking Probability
1	(6)	0.011204
2	(6, 10)	0.0074254
3	(6, 8, 10)	0.005649
4	(4, 6, 8, 10)	0.0041197
5	(4, 6, 8, 10, 13)	0.0035537
6	(4, 6, 8, 9, 10, 13)	0.0030138
7	(4, 5, 6, 8, 9, 10, 13)	0.0024768

Table 3. The Comparing Results of Pure GA and Our Optimal Algorithm

	Pure GA	Optimal Algorithm
Population Size	40	10
Probability of Crossover	0.6	0.6
Probability of Mutation	0.005	0.005
Maximum Number of Generations	1000	20

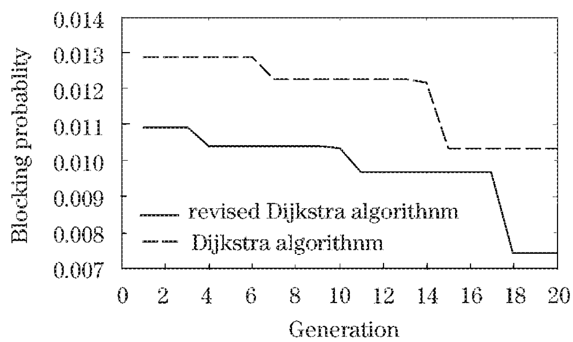

Fig. 3. Blocking probability versus generation using the revised and non-revised Dijkstra algorithms.

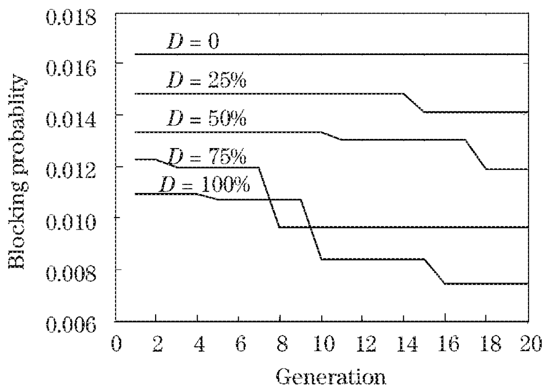

Fig. 4. GA performance for the NSF net with different degrees of translation D.

In Fig. 3, because the pure GA must search bigger region than the optimal algorithm, it needs more generations and individuals. It is obvious that the optimal algorithm is more efficient and less time-consuming than pure GA.

Figure 4 shows that full-range translation is very effective to reduce the blocking probability of networks. It also shows the significant improvements in optimal blocking probability can be achieved by using limited-range translation as the same conclusion in Ref. [5].

In this paper, an efficient algorithm for optimally using number of shared limited-range converters in WRON is presented. Our purpose is to provide an optimal solution for a certain population of individuals in a limited number of generations of GA, which can be used to solve other NP-C problems in the optical networks, such as routing and wavelength assignment (RWA). The effectiveness of this method has been verified through the simulation of NSF net. We have compared the performance of the optimal method with pure GA method. The revised algorithm can realize blocking probability 36% less than the algorithm of Ref. [3], and the calculating time for reaching the minimum blocking probability can be reduced dramatically.

A major difference between the pure GA and the optimal algorithm is the size of the search space for the optimal solution. Pure GA searches all combinations of converter placements in every layer of the network. For the large networks, when the number of layers grows, it needs explosive computational time for searching the optimal solution. On the other hand, the optimal algorithm only searches the combination of the nodes in the networks and does not need to compare all layer blocking probabilities.

This work was supported by Wuhan Science & Technology Great Foundation under Grant No. 2002100513004.
W. Li's e-mail address is weilee@hust.edu.cn.

References

1. M. Zhang, P. D. Ye, F. Zhang, Y. P. Zhao, and J. Wang, Chin. Opt. Lett. **1,** 3 (2003).
2. Y. B. Ye, X. P. Zheng, H. Y. Zhang, X. Teng, W. W. Shi, Y. H. Li, and Y. L. Guo, Chin. J. Lasers (in Chinese) **29,** 722 (2002).
3. S. Thiagarajan and A. K. Somani, in *Proceedings of INFOCOM 1999* 916 (1999).
4. T. H. Siregur, H. Takagi, and Y. B. Zhang, IEICE Trans. Commun. **E85-B,** 1075 (2002).
5. D. Y. Li and X. H. Jia, in *Proceedings of International Conference: Parallel Processing Workshops 2001* 277 (2001).
6. T. Tripathi and K. N. Sivarajan, IEEE J. Sel. Areas Commun. **18,** 2123 (2000).
7. H. Harai, M. Murata, and H. Miyahara, IEEE J. Sel. Areas Commun. **16,** 1051 (1998).
8. I. Alfouzan and A. P. Jayasumana, Optical Networks Magazine **4,** 46 (2003).
9. H. Zhang, S. L. He, X. H. Chen, W. Sun, and Z. D. Liu, Acta Opt. Sin. (in Chinese) **20,** 1027 (2000).