Dynamic RWA Based on the Combination of Mobile Agents Technique and Genetic Algorithms in WDM Networks with Sparse Wavelength Conversion

Vinh Trong Le[†], Xiaohong Jiang[††], Son Hong Ngo[†] and Susumu Horiguchi[†,††]

Summary

Genetic Algorithms (GA) provide an attractive approach to solving the challenging problem of dynamic routing and wavelength assignment (RWA) in optical Wavelength Division Multiplexing (WDM) networks, because they usually achieve a significantly low blocking probability. Available GA-based dynamic RWA algorithms were designed mainly for WDM networks with a wavelength continuity constraint, and they cannot be applied directly to WDM networks with wavelength conversion capability. Furthermore, the available GA-based dynamic RWA algorithms suffer from the problem of requiring a very time consuming process to generate the first population of routes for a request, which may results in a significantly large delay in path setup. In this paper, we study the dynamic RWA problem in WDM networks with sparse wavelength conversion and propose a novel hybrid algorithm for it based on the combination of mobile agents technique and GA. By keeping a suitable number of mobile agents in the network to cooperatively explore the network states and continuously update the routing tables, the new hybrid algorithm can promptly determine the first population of routes for a new request based on the routing table of its source node, without requiring the time consuming process associated with current GA-based dynamic RWA algorithms. To achieve a good load balance in WDM networks with sparse wavelength conversion, we adopt in our hybrid algorithm a new reproduction scheme and a new fitness function that simultaneously takes into account the path length, number of free wavelengths, and wavelength conversion capability in route selection. Our new hybrid algorithm achieves a better load balance and results in a significantly lower blocking probability than does the Fixed-Alternate routing algorithm, both for optical networks with sparse and full-range wavelength converters and for optical networks with sparse and limited-range wavelength converters. This was verified by an extensive simulation study on the ns-2 network simulator and two typical network topologies. The ability to guarantee both a low blocking probability and a small setup delay makes the new hybrid dynamic RWA algorithm very attractive for current optical circuit switching networks and also for the next generation optical burst switching networks

Key words: *Mobile agents, genetic algorithm, dynamic routing and wavelength assignment, WDM networks, wavelength conversion*

1. Introduction

Due to their huge bandwidth capacity, all-optical networks based on wavelength-division-multiplexing (WDM) technology hold great promise for serving as the backbone of the next generation Internet. In a wavelength-routed WDM network, data are routed in optical channels called lightpaths. To establish a lightpath without wavelength conversion, the same wavelength is required on all the links along the path; this is referred to the *wavelength continuity constraint*. The wavelength continuity constraint usually results in a high blocking probability in a wavelength-routed WDM network, and a common approach to alleviating this constraint is the adoption of wavelength converters.

A WDM network is referred to as a network with full wavelength conversion if each node of the network has wavelength conversion capability [1]. On the other hand, a WDM network is referred to as a network with sparse wavelength conversion if only a sub-set of the network nodes has wavelength conversion capability. Furthermore, we say that a node is capable of full-range wavelength conversion if a wavelength channel on any input of the node can be converted to any wavelength channel on any output; a node is capable of limited-range wavelength conversion if a wavelength channel on any input of the node can only be converted to particular wavelength channels on any output [2]. Since the wavelength converters are expensive devices, it is practically infeasible to equip each network node with a wavelength converter. Therefore, current research efforts focus mainly on networks with sparse wavelength conversion [1],[2][3],[4],[5],[6].

In WDM networks, the Routing and Wavelength Assignment (RWA) problem concerns determining the routes and wavelengths to be used to establish lightpaths for connection requests. The RWA problem can be generally classified into two types: static and dynamic. In the former type, network topology and connection requests are given in advance, and the problem is to find a solution that minimizes network resources. It has been proved that the static RWA problem is NP-complete [7]. In the latter type, lightpaths are dynamically established upon the arrival of requests. If no route with free wavelength is available for a request, the request will be blocked. Generally, dynamic RWA algorithms aim to minimize the total blocking probability in the entire network.

[†] Graduate School of Information Science, Japan Advanced Institute of Science and Technology, Japan. Email: {vt-le, sonhong}@jaist.ac.jp
[††] Graduate School of Information Sciences, Tohoku University, Sendai, Japan. Email:{jiang,susumu}@ecei.tohoku.ac.jp

The dynamic RWA problem is more challenging, and many approaches have been proposed for solving it [2],[7],[8],[9],[10]. In the fixed routing approach, a single fixed route is predetermined for each source-destination pair. Whenever a request arrives, its fixed route is attempted for wavelength assignment. In fixed alternate routing scheme, a set of routes is pre-computed for each source-destination pair and stored in an ordered list at the source node's routing table. As a connection request arrives, one route is selected from the set of pre-computed routes. Fixed alternate routing always achieves better performance than fixed routing, and this approach actually shows a good trade-off between performance and control overhead [2]. In the adaptive routing approach, the route is computed based on the current network state at the arrival of a request, so it obtains the best performance [8],[9]. However, adaptive routing requires a relatively longer setup delay and a higher control overhead, including special support from control protocols to keep track of global network states.

In a WDM network with sparse wavelength conversion, the dynamic RWA problem needs deliberate study to take full advantage of the gain from wavelength conversion [10],[11]. However, conventional dynamic routing algorithms may not work well in an environment with sparse or/ and full wavelength conversion [10]. GAs represent a promising approach to the RWA problem in WDM networks, and several GA-based RWA algorithms have been proposed [12],[13],[14],[15]. In [12],[13], N. Banerjee and S. Pandey et al. formulated both the static RWA and dynamic RWA problems as a multi-objective optimization problem and solved it using genetic algorithms. However, Banerjee and Pandey's dynamic RWA algorithms require the re-routing of available connections and thus are not suitable for the high capacity WDM networks. Also, these authors did not consider wavelength conversion in their GA-based RWA algorithms. D.Bisbal et al. [14] proposed a novel GA-based algorithm for dynamic RWA problem in WDM networks with wavelength continuity constraint, and V.T.Le et al. [15] improved Bisbal's algorithm by using a more general cost function to achieve a better load balance. Bisbal's algorithm does not require the re-routing of available connections and it can achieve a significantly low blocking probability. However, it can not be applied directly to the dynamic RWA problem for WDM networks with wavelength conversion capability, and it also requires a very time consuming process based on random searching to generate the first population of routes for a new request, which can result in a significantly large setup delay.

In this paper, we focus on the dynamic RWA problem in WDM networks with sparse wavelength conversion, and propose a novel hybrid algorithm for solving it based on a combination of mobile agents technique [16],[17],[18],[19],[20] and genetic algorithm. The main contributions of our work are the following:

- We extend Bisbal's algorithm used for the dynamic RWA problem in WDM networks without wavelength conversion to solve the dynamic RWA problem in WDM networks with sparse wavelength conversion.

- We propose a new reproduction scheme and a more general fitness function that simultaneously takes into account the path length, number of free wavelengths and wavelength conversion capability in route evaluation, such that a good load balance and low blocking probability can be achieved. We also propose some general formulas for determining the key parameters in the new fitness function.

- We adopt a new scheme, based on the mobile agents technique [20], for generating the first population of routes for the hybrid algorithm. In this scheme a suitable number of mobile agents are kept in the network to cooperatively explore the network states and continuously update the routing tables, such that the first population routes for a new request can be determined promptly based on the routing table of its source node without requiring the very time consuming random searching process of old GA-based dynamic RWA algorithms.

Our proposed hybrid algorithm is applicable to the dynamic RWA problem for both WDM optical networks with sparse and full-range wavelength converters and those with sparse and limited-range wavelength converters. Since the promising fixed-alternate routing algorithm shows a good trade-off between performance and control overhead, and a fixed-alternate algorithm with a small number of alternate routes can asymptotically approach the performance of an adaptive routing algorithm in terms of blocking probability [2], we use the Fixed-Alternate algorithm in this work for comparison with our hybrid algorithm. Extensive simulation results based on two typical network topologies show that our proposed algorithm can adapt well to traffic variations and significantly outperforms the fixed-alternated routing algorithm in terms of blocking probability.

The rest of this paper is organized as follows: In Section 2, we briefly present some background about GA algorithms and mobile-agents technique. Section 3 presents our new hybrid algorithm based on the combination of mobile agents and GA. Section 4 provides the simulation results and discussion. We conclude this paper in Section 5.

3

2. Genetic Algorithms and Mobile Agents Technique

In this section, we briefly introduce the main ideas of genetic algorithms and mobile agent techniques for network routing. In particular, we describe in more detail the mobile agents-based algorithm proposed by S.H. Ngo et al. [20], which will be adopted in our new hybrid algorithm to generate the first population of routes for a request

2.1 Genetic Algorithms

Genetic Algorithms (GA) [21],[22] are a class of probabilistic searching algorithms based on the mechanism of biological evolution. A GA begins with an initial population of individuals, each of them represents a feasible solution to the problem being tackled. Then the GA applies a set of genetic operations to the current population to generate a better one. This process is repeated until a good solution is found or after predefined number of iterations. It is notable that genetic operations are critical for a GA and they must be designed deliberately to achieve a good performance. Readers are referred to [23],[24],[25], and [26] to find some advanced results about genetic operations developed by S. Kobayashi's group [27].

2.2 Mobile Agents Technique for Routing in WDM Networks

Routing in communication networks can be resolved efficiently by means of Ant Colony Optimization (ACO) [16],[17],[19], in which the routing solution can be built using the behavior of ant-based mobile agents in their foraging of network states. These collective agents indirectly communicate through pheromone trailing (stigmergy) in the environment, and an agent can find a "good" route in terms of the shortest, least congested path from the source to the destination by following the pheromone trails of others. Garlick et al. [18] proposed an ACO-based algorithm to solve the dynamic RWA problem, in which at the arrival of a new connection request a number of mobile agents are launched from the source to search for the routes to the destination, and the final best path for the connection request is determined when all mobile agents complete their exploitation tasks. As a new set of mobile agents must be launched after the arrival of every new connection request, Garlick's algorithm may incur a very large setup delay due to the need to wait for all ants to complete their search. To overcome the problem associated with Garlick's algorithm, S.H. Ngo et al. [20] proposed a new mobile agents-based algorithm for solving the dynamic RWA problem in WDM networks without

wavelength conversion. By keeping a suitable number of ants in a network to cooperatively explore the network states and continuously update the pheromone tables, this algorithm enables the route for a connection request to be determined promptly by the current states of routing tables, with a small setup delay. Note, however, that the algorithm proposed in [20] usually results in a higher blocking probability than that of the GA-based algorithm for dynamic RWA in WDM networks without wavelength conversion [14],[15]. Since the main idea of the algorithm in [20] will be adopted in our hybrid algorithm for generating the first population of routes, we describe this algorithm here in more detail.

In Ngo's algorithm, a network node i with k_i neighbors is equipped with a probabilistic pheromone table $R_i = [r^i_{n,d}]_{ki, N-1}$ with N-1 rows (N is number of network nodes) and k_i columns, as illustrated in Fig. 1.

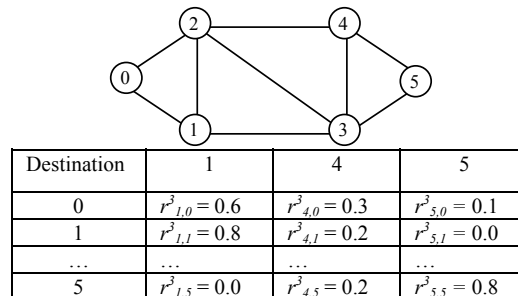| Destination	1	4	5
0	$r^3_{1,0} = 0.6$	$r^3_{4,0} = 0.3$	$r^3_{5,0} = 0.1$
1	$r^3_{1,1} = 0.8$	$r^3_{4,1} = 0.2$	$r^3_{5,1} = 0.0$
…	…	…	…
5	$r^3_{1,5} = 0.0$	$r^3_{4,5} = 0.2$	$r^3_{5,5} = 0.8$

Fig. 1. A simple network and its pheromone table of node 3.

In the pheromone table, each row corresponds to a destination node and each column corresponds to a neighbor node; the value $r^i_{n,d}$ is used as the selection probability of neighbor node n when an ant is moving toward its destination node d.

Ants are launched from each node with a given probability ρ to a randomly selected destination every T time units; here ρ and T are design parameters. Each ant is considered to be a mobile agent: it collects information on its trip, performs pheromone table updating on visited nodes, and continues to move forward as illustrated in Fig. 2.
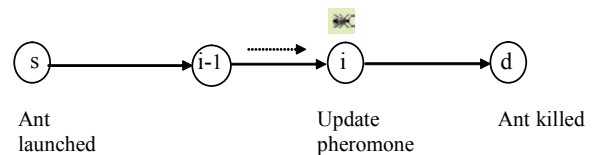
Fig. 2. Ant's moving and updating tasks

Whenever an ant visits a node, it updates the pheromone table element with the information gathered during its trip. Suppose an ant moves from source s to destination d following the path ($s,…, i$-1, $i,…,d$). When the ant arrives at node i, it updates the entry corresponding

to the node s as follows: the probability of neighbor i-1 is increased while the probabilities of other neighbors are decreased. The pseudo-code of the main steps in this algorithm can be summarized as follows:

{Ant generation}

```
Do
      For each node in network
            Select a random destination;
            Launch ants to this destination with a probability ρ
      End for
      Increase time by a time-step for ants' generation
Until (end of simulation)
```

{Ant foraging}

```
For each ant from source s to destination d do (in parallel)
      While current node i <> d
            Update pheromone table elements
            Push trip's state into stack
            If  (found at next hop)
                  Move to next hop
            Else
                  Kill ant
            End if
      End while
End for
```

When a new connection request arrives at its source node, its route is determined promptly from the routing tables: starting from the source node, the next hop will always be the neighbor that has the highest selecting probability, and this principle is applied until the destination node. With this method, the route is already determined upon the arrival of a connection request, so Ngo's algorithm is an attractive tool in GA-based dynamic RWA algorithms for generating the first population of routes, which is usually based on the idea of very time consuming random searching [14]

3. Dynamic RWA by Combining GA and Mobile Agents

Note that the blocking probability of GA-based algorithms for dynamic RWA is usually significantly lower than that of mobile agents-based algorithms [14],[15],[20]. However, GA-based algorithms may incur a very large setup delay because they require a very time consuming random searching process to generate the first population of routes after the arrival of a new request. On the other hand, in the mobile agents-based algorithm [20], the routes can be promptly determined upon the arrival of a connection request, so a small setup delay is guaranteed. Based on the above observations, we are motivated to propose a hybrid algorithm based on the combination of mobile agents and genetic algorithms for dynamic RWA in WDM networks with sparse wavelength conversion.

3.1. Hybrid Algorithm for Dynamic RWA

Our hybrid algorithm differs from the algorithm proposed by Bisbal [14] in that we have adopted a more general fitness function and also a new reproduction scheme to account for the effects of wavelength conversion capability on route selection, such that the GA-based approach can be used to solve the dynamic RWA problem in WDM networks with sparse wavelength conversion. Also, the gain of wavelength conversion can be fully explored to improve the blocking behavior of the new algorithm. The hybrid algorithm is executed at the arrival of a connection request between a source-destination node pair. It works with an initial population in which each individual is a possible route between the source-destination node pair. In this work, we use representation, crossover and mutation operators similar to those in [14]. The coding of a route is a vector of integers where each number identifies a node of the route. For the network in Fig.3, the coding of the two routes from node 0 to node 5 are vector (0, 1, 2, 5) and (0, 2, 4, 5).
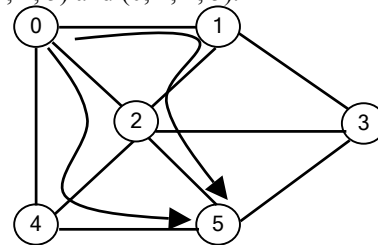

Fig. 3. Two routes from node 0 to node 5 are encoded as (0 1 2 5) and (0 2 4 5).

The main steps of the hybrid algorithm are as follows:

(1) Initialization

Available schemes for generating the first population of routes are usually based on random searching [14], which is very time consuming for large networks and may incur a large setup delay. Here we propose a new approach to generating the first population of routes based on the mobile agents-based algorithm proposed by S.H.Ngo et al. [20].

Since the algorithm in [20] was designed to find only one suitable route between a source-destination node pair, we need to modify it such that it can be used to generate the first population of P routes (where P is a design parameter) for the hybrid algorithm.

To make sure that the first population of P routes is available for a node-pair upon the arrival of a request between this node-pair, we equip each node with a new P-route routing table in addition to the previous pheromone table. The P-route routing table contains N-1 entries, each of which corresponds to a list of P routes to a destination node; these P routes will serve as the first population for the future request between the current node and the

destination node. The role of mobile agents is now two-fold, they need to continuously update both the pheromone table and the *P*-route routing table on each node.

When an agent moves from its source to its destination, it updates the pheromone tables of intermediate nodes in a same way as with Ngo's approach, but it updates the *P*-route routing table of its destination node as follows[1]: when the agent reaches its destination node, it updates the list of *P* routes corresponding to its source node based on the new route it has just found. If the *P*-route table contains less than *P* routes in the route list corresponding to its source node, the new route is directly inserted into that list. Otherwise, if the list has already contained *P* routes, the new one will replace an old route in the list based on the *First-In-First-Out* policy. This replacement is necessary to ensure randomness in the first population of routes. The pseudo code of this process can be described as follows:

{**Updating the *P*-route routing table of destination node**}

If the new route is different from any available routes in the route list of source node Then

        If number of routes in the route list less than *P* Then
        Insert agent's new route into the list
        Else
                Replace an existing route with the agent's new route based on the First-In-First-Out policy
        End if
End if

Since we always keep a suitable number of mobile agents in the network, to cooperatively explore the network states and to continuously update the pheromone tables and *P*-route routing tables, each *P*-route routing table will contain *P* routes for each destination after an initialization period; those routes are always updated based on the current network state and they can serve as the first population of *P* routes upon the arrival of a request.

(2) Determination of Fitness Function

The definition of the fitness function is critical for a GA, because it determines which individual should be chosen in the evolution process. Bisbal et al. [14] define the fitness function of a route as the inverse of the cost of the route, where the cost of the route is defined as the number of hops if there exists at least one common free wavelength on all the links of the route; otherwise, infinite.

Since this fitness function considers only the number of hops in route selection, it tends to take the shortest available path for a connection request. Previous work [10],[28] indicates clearly that this shortest-path based

approach usually leads to unbalanced link utilization and thus significantly degrades network performance.

To guarantee a good load balance while accounting for the gain of using wavelength conversion, we propose here a new and more general fitness function for the hybrid algorithm. The new fitness function considers not only the number of hops on a route, but also the number of free wavelengths and the number of wavelength converters along the route. For a route having t wavelength converters along it, we use the scheme proposed in [29] to divide this route into $t+1$ segments $s_1, s_2 \ldots s_{t+Z}$ as illustrated in Fig. 4.
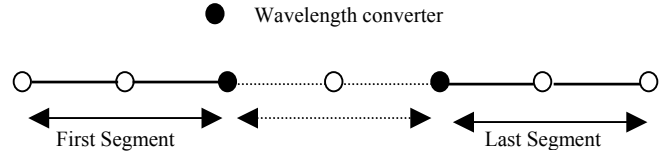

Fig. 4. A route and its segments

For a source-destination node pair, let l_i be the length of route i between the node pair, l_{min} and l_{max} be the length of the shortest route and the longest route between the node pair, respectively, and fw_i be the number of free wavelengths on route i. If we use fw_i^j to denote the number of free wavelengths on the jth segment of route i that has t_i wavelength converters, the fw_i can be determined as:

$$fw_i = \min_{1 \le j \le t_i + 1} \left(fw_i^{\,j} \right) \qquad (1)$$

If $fw_i > 0$, we introduce the following general fitness function f_i for route i:

$$f_i = \alpha \cdot \frac{1}{l_i - l_{min} + 1} + (1-\alpha) \cdot \frac{fw_i}{W} - t_i \cdot C_{wc} \qquad (2)$$

where $\alpha \in [0, 1]$ is a design parameter, W is total number of wavelengths on a link, t_i is the number of wavelength converters along the route, and C_{wc} is the cost of each wavelength converter. If $fw_i = 0$, we just set f_i as zero. In this case, route i cannot be used by the connection request because we cannot find any free lightpath along this route.

The parameter α should be chosen such that the shorter route has a higher fitness value. Let $d = l_i - l_{min} + 1$, then α should meet the following requirement:

$$\alpha \cdot \frac{1}{d} + (1-\alpha) \cdot \frac{1}{W} > \alpha \cdot \frac{1}{d+1} + (1-\alpha) \cdot \frac{W}{W}$$

which is equivalent to:

$$\alpha > \frac{(W-1)(d+1)d}{W + (W-1)(d+1)d} \qquad (3)$$

For a given value of W, the right-hand side of (3) increases as d increases. From the definition of d we can see easily that $1 \le d \le N-2$, where the N is number of network nodes. Thus, we have:

---

[1] More efficient but complex updating for *P*-route routing table is possible, such as using the smart agents technique [30]. However, this paper focuses mainly on how to combine GA and mobile agents technique to solve the dynamic RWA problem, so we just adopt a simple updating scheme here.

$$\alpha > \frac{(W-1)(N-1)N}{W+(W-1)(N-1)N} \qquad (4)$$

The value of C_{wc} should be chosen to make sure that a route with a larger number of wavelength converters will always get a smaller fitness value. For a node pair, suppose that one route has l_1 hops and $t \geq 1$ converters and another route has $l_2 \leq l_1$ hops but $(t+1)$ converters, then the C_{wc} should meet the following requirement:

$$\alpha \cdot \frac{1}{l_1 - l_{min} + 1} + (1-\alpha) \cdot \frac{1}{W} - t \cdot C_{wc} >$$
$$\alpha \cdot \frac{1}{l_2 - l_{min} + 1} + (1-\alpha) \cdot \frac{W}{W} - (t+1) \cdot C_{wc}$$

which is equivalent to:

$$C_{wc} > \alpha \cdot \left(\frac{1}{l_2 - l_{min} + 1} - \frac{1}{l_1 - l_{min} + 1} \right) + (1-\alpha) \cdot \frac{W-1}{W} \qquad (5)$$

Obviously, the right side of inequality (5) reaches its maximum value when $l_2 = l_{min}$ and $l_1 = l_{max}$. The route of $l_2 = l_{min}$ has to accommodate at least 2 converters ($t+1 \geq 2$), thus its number of hops $l_2 = l_{min}$ is not less than 3. On the other hand, the route of l_{max} has $t \geq 1$ converter(s), so we have $l_{max} < N - N_c + 1$, where N_c is the total number of wavelength converters on the network. Thus, we have

$$l_{max} - l_{min} < (N - N_c + 1) - 3 = N - N_c - 2$$

Finally, we can get the following general inequality for C_{wc}:

$$C_{wc} > \alpha \cdot \left(1 - \frac{1}{N - N_c - 2} \right) + (1-\alpha) \cdot \frac{W-1}{W} \qquad (6)$$

(3) Crossover Operator
The operator can only be applied to a pair of routes that have at least one common node, apart from the source and destination nodes. This common node is called the crossover point. If there are many common nodes, one of them is chosen randomly. The offspring is generated by interchanging the second halves of its parent, as illustrated in Fig. 5.

In the crossover stage, the hybrid algorithm examines all possible pairs of routes, beginning with the pairs that include the individual with a higher fitness value, until either all combinations have been considered or the population size becomes twice of the original size.
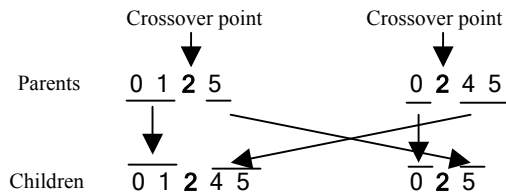
Crossover point   Crossover point

Parents   0 1 **2** 5   0 **2** 4 5

Children   0 1 2 4 5   0 2 5

Fig. 5. Example of crossover operation

(4) Mutation Operator

To do mutation, a node is randomly selected from the route and the selected node is called the mutation point (node). Then, the path from the mutation node to the destination node is randomly selected from the routing table of the mutation node. The path from the source node to the mutation node remains untouched. In the mutation stage, the mutation operator is applied to all individuals whose fitness value is below a given threshold, which is chosen from the mean fitness value of the current generation.

(5) Reproduction and Stopping Conditions
After applying the genetic operators above, the reproduction stage selects the P fittest individuals that have a higher fitness value from both parents and children for the next generation. This process is repeated until the stopping condition is fulfilled and the best individual is selected. Existing results [2] have indicated that to achieve a good performance, a route with wavelength conversion should be chosen only if no routes without wavelength conversion are available for a request. In the reproduction stage of Bisbal's algorithm [14], only the P fittest individuals are selected in a generation without considering the wavelength conversion capability in the selection. To take the advantage of the wavelength conversion capability in evolution, while avoiding the high cost that may be introduced by adopting too many wavelength changes, we propose here a new reproduction scheme. In this new scheme, we first select the P fittest individuals under the constraint of 'without wavelength conversion' as the population for next generation, then we select the l best routes among all these routes equipped with wavelength converter(s) as the backup route candidates. These backup route candidates will be attempted only if none of these P individuals can be used for the connection request under the 'without wavelength conversion' constraint. Our simulation results in Section 4 indicate that by keeping only the one best route ($l=1$) among all the routes equipped with wavelength converter(s) as the backup route candidate in each generation, we can achieve a satisfactory performance in terms of blocking probability.

Let G denote the maximum number of generations and S denote the satisfactory cost value of a route between a node-pair with its initial value being the cost value of the shortest route between the node-pair, then the pseudo code of the GA part in our hybrid algorithm can be summarized as follows

{Genetic algorithm}
t = 0;
Evaluate fitness values of the first population of P routes;
Select the best route with wavelength conversion (BR).
S = shortest distance between (s, d) nodes;
While (t < G AND doesn't exist a route that have length lower or equal S) do
         Do crossover & evaluate fitness value for children;
         Do mutation & evaluate fitness value for children;

```
            Select P fittest individuals for next generation;
            Select the best route with wavelength conversion (BR).
            S = S + 1;
            t = t + 1;
End while
If  exist routes without wavelength conversion then
            select the best route from P;
Else
            select the BR;
End If
```

3.2 Complexity Analysis

Since the time complexity of our hybrid algorithm is dominated by its GA part, so our complexity analysis focuses on the genetic algorithm. Let N be the number of network nodes and N_c be the total number of converters in a network, then the complexity of our hybrid algorithm can be estimated as follows.

The complexity of examining all possible pairs in the crossover stage is $P(P-1)/2$ times the complexity of examining a pair of routes. This operation includes the following steps: search common nodes, create children, check if the children are valid routes and check if the children are different from their parents. All these steps require $O(N)$ time. Moreover, the complexity of evaluating the fitness values for the cases with and without wavelength conversion is $O(PWNN_c)$. Therefore, the complexity of the crossover stage is $O((P(P-1)/2)N+ PWNN_c) = O(P^2N+PWNN_c)$. In the mutation stage, a maximum of P-1 routes are mutated. The fitness values for the cases with and without wavelength conversion must be evaluated. Thus, the complexity of the mutation stage is $O(PWN_cN)$. The reproduction stage only involves sorting the population by decreasing order of fitness value, so complexity of this operation is $O(PlogP)$. In summary, the complexity of the hybrid algorithm is:

$$O(G(P^2N+PW\,N_cN+ PW\,N_cN+PlogP))$$
$$=O(GPN(P+WN_c)).$$

3.3 Wavelength Assignment

The First-Fit wavelength assignment algorithm [9], in which the available wavelength with the smallest index is chosen, can achieve almost the same performance as other complex algorithms and is very simple to implement, so we use this algorithm in our work for wavelength assignment. We consider the following two scenarios:

- If there are no wavelength converters along the selected route, apply the First-Fit wavelength assignment algorithm directly to the selected route.

- If there are $t \geq 1$ wavelength converters along the selected route, we divide the route into $t+1$ segments as illustrated in Fig.4. In the case of full-range wavelength conversion, we apply the first-fit algorithm for each segment. As for the case of limited-range

wavelength conversion, we will check all the free wavelengths on each segment and try to find a lightpath that satisfies the constraint of limited-range wavelength conversion and also has the smallest index.

4. Numerical Results & Discussion

In this section, we examine the performance of our new hybrid algorithm with an extensive simulation study based upon the ns-2 network simulator [31] and two typical network topologies, as illustrated in Fig. 6. In our simulation, we take W=8 and consider the configurations of adopting 2 and 5 wavelength converters in each network topology. For each converter, we simulate the case of full-range wavelength conversion and also the cases of using 1 or 2 wavelength conversion units; here a node with r conversion units can convert a wavelength w into any wavelength belongs to set $[w-r, w+r]$.
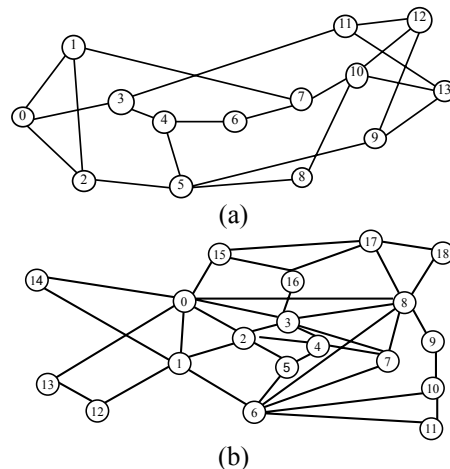

(a)

(b)

Fig. 6 Network topologies adopted in simulation. (a) NSF network with 14 nodes and 21 links. (b) EON network with 19 nodes and 35 links.

To place the wavelength converters properly in a network, we use the placement scheme proposed in [29] to achieve best benefit from the use of converters. When adopting 2 converters, the converters are placed in nodes 3 and 5 for the NSF network, and nodes 0 and 6 for the EON network. When using 5 converters, the converters are placed in nodes 1, 3, 5, 10 and 12 for the NFS network, and nodes 0, 1, 3, 6 and 8 for the EON network.

In our experiments, we used a dynamic traffic model in which the connection requests arrive at the network according to a Poisson process with an arrival rate λ (call/seconds). The session holding time is exponentially distributed with mean holding time μ (seconds). The connection requests are distributed randomly on all the network nodes. If there are N sessions over the network, then the total workload is measured by $N*\lambda*\mu$ (Erlangs). Thus, we can modify N, λ, μ parameters to control

workloads. For performance comparison, we run each simulation based on two routing algorithms: the Fixed-Alternate routing algorithm (FA) with two alternative routes and our algorithm. The GA parameters used in our experiment are set as P=16 and G=8 for both the NSF and EON networks. For the mobile agents part of our hybrid algorithm, we adopt the same set of parameters as that in paper [20] for ants generation and pheromone table updating.

4.1 Determination of α and C_{wc}

Table 1 and Table 2 illustrate the sensitiveness of our hybrid algorithm to the variation of parameters α and C_{wc} for NSF and EON networks. Both tables clearly indicate that by choosing the parameter α properly for both the cases of with and without wavelength conversion, we can reduce the blocking probability considerably. It is interesting to note in Table 1 and Table 2 that by choosing α =0.9 according to formula (4), we can get the best results in terms of blocking probability for both the cases, with and without wavelength conversions.

The results in Tables 1 and 2 also show clearly that when wavelength converters are adopted for both the NSF and EON networks, we can get the best results in terms of blocking probability by first choosing α =0.9 according to formula (4) and then choosing C_{wc} =0.4 according to formula (6). Hereafter, we use the values of α =0.9 and C_{wc} =0.4 in our simulation.

4.2 Effectiveness of Ant-based Initialization and the New Fitness Function

To show clearly the effectiveness of ant-based agent scheme for generating the first generation of GA, we illustrate in Fig. 7 the blocking probability of the proposed algorithm for the cases in which the ant-based scheme and the random searching scheme are adopted for generating the first generation of GA.

The comparison in Fig. 7 shows clearly that the ant-based scheme and the random searching scheme for initialization have a very similar performance in terms of blocking probability under the proposed algorithm. It notable, however, the random searching scheme is usually very time consuming and may result in a significantly large setup delay than that of ant-based scheme (please refer to Table 3 and Table 4).

To further show the effectiveness of the proposed fitness function, we illustrate in Fig. 8 the blocking probability of the proposed algorithm under the scenarios in which the new fitness function proposed in this paper and old fitness function proposed in [14] are adopted.

Table 1. Blocking probability vs. workload for different values of α and C_{wc} on NSF network. F: Full-range conversion.

GA parameters				Workload (Erlangs)				
α	C_{wc}	N_c	r	45	54	63	72	81
0.9		0		0.29	0.63	1.39	2.68	4.29
	0.4	2	1	0.23	0.53	1.21	2.34	3.76
			2	0.22	0.50	1.08	2.22	3.58
			F	0.19	0.42	0.95	2.12	3.43
		5	1	0.23	0.49	1.16	2.22	3.51
			2	0.20	0.43	1.04	2.03	3.37
			F	0.19	0.39	0.95	1.86	3.09
	0.1	2	1	0.27	0.60	1.37	2.54	4.19
			2	0.24	0.57	1.27	2.55	3.95
			F	0.22	0.52	1.21	2.43	3.86
		5	1	0.26	0.61	1.33	2.57	4.06
			2	0.23	0.55	1.23	2.50	3.91
			F	0.22	0.51	1.20	2.40	3.67
0.5		0		0.32	0.84	1.72	3.18	4.87
	0.4	2	1	0.28	0.67	1.65	2.76	4.14
			2	0.27	0.64	1.46	2.62	4.03
			F	0.22	0.54	1.28	2.14	3.94
		5	1	0.25	0.57	1.39	2.37	4.13
			2	0.24	0.55	1.31	2.30	3.82
			F	0.20	0.44	1.09	2.14	3.38
	0.1	2	1	0.30	0.81	1.70	3.13	4.73
			2	0.29	0.75	1.67	2.95	4.55
			F	0.27	0.71	1.64	2.86	4.48
		5	1	0.28	0.73	1.67	3.00	4.69
			2	0.27	0.69	1.56	2.77	4.42
			F	0.26	0.66	1.48	2.65	4.17
0.1		0		0.30	0.68	1.53	2.86	4.62
	0.4	2	1	0.26	0.58	1.36	2.50	3.89
			2	0.25	0.55	1.24	2.36	3.75
			F	0.21	0.47	1.12	2.13	3.61
		5	1	0.24	0.53	1.27	2.30	3.74
			2	0.22	0.47	1.17	2.12	3.52
			F	0.19	0.42	1.02	1.95	3.23
	0.1	2	1	0.28	0.64	1.50	2.82	4.39
			2	0.26	0.62	1.42	2.69	4.19
			F	0.24	0.58	1.37	2.60	4.03
		5	1	0.26	0.67	1.50	2.78	4.33
			2	0.25	0.62	1.41	2.63	4.14
			F	0.24	0.57	1.32	2.52	3.99

Table 2. Blocking probability vs. workload for different values of α and C_{wc} on EON network. F: Full-range conversion.

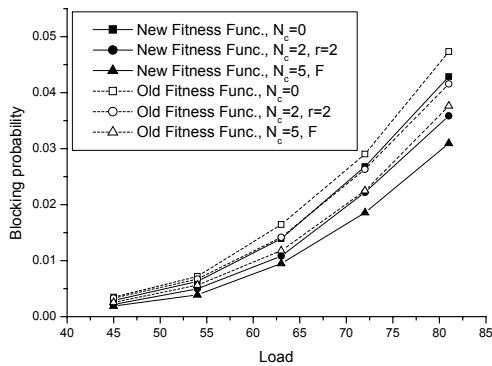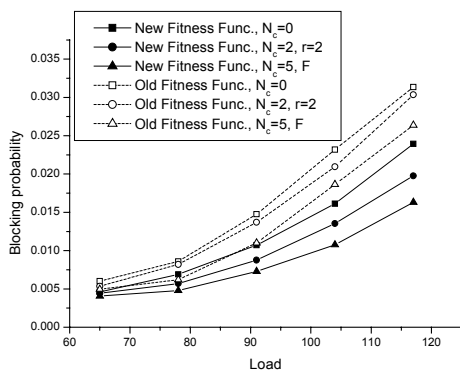| GA parameters				Workload (Erlangs)				
				65	78	91	104	117
α	C_{wc}	N_c	r	Blocking probability (%)				
0.9		0		0.46	0.72	1.07	1.61	2.39
	0.4	2	1	0.45	0.62	0.90	1.45	2.01
			2	0.44	0.57	0.88	1.35	1.98
			F	0.42	0.51	0.84	1.28	1.92
		5	1	0.44	0.58	0.87	1.34	1.97
			2	0.43	0.49	0.77	1.22	1.81
			F	0.41	0.48	0.73	1.08	1.63
	0.1	2	1	0.45	0.65	1.00	1.54	2.31
			2	0.44	0.60	0.96	1.54	2.20
			F	0.42	0.53	0.84	1.37	2.04
		5	1	0.44	0.64	0.94	1.53	2.20
			2	0.43	0.56	0.90	1.47	2.18
			F	0.41	0.53	0.77	1.36	1.98
0.5		0		0.51	0.76	1.34	1.91	2.73
	0.4	2	1	0.50	0.71	0.99	1.78	2.56
			2	0.49	0.62	0.97	1.65	2.42
			F	0.44	0.55	0.95	1.51	2.18
		5	1	0.48	0.60	0.93	1.50	2.19
			2	0.46	0.56	0.85	1.40	2.07
			F	0.43	0.49	0.77	1.20	1.97
	0.1	2	1	0.51	0.70	1.16	1.85	2.69
			2	0.49	0.73	1.15	1.80	2.59
			F	0.46	0.69	1.11	1.78	2.51
		5	1	0.50	0.68	1.13	1.82	2.66
			2	0.46	0.63	1.08	1.74	2.52
			F	0.44	0.60	0.98	1.62	2.41
0.1		0		0.49	0.69	1.22	1.72	2.56
	0.4	2	1	0.47	0.66	0.94	1.57	2.22
			2	0.45	0.59	0.92	1.47	2.13
			F	0.43	0.53	0.89	1.38	2.02
		5	1	0.46	0.60	0.90	1.41	2.06
			2	0.44	0.53	0.81	1.30	1.91
			F	0.42	0.48	0.75	1.13	1.75
	0.1	2	1	0.48	0.65	1.08	1.67	2.43
			2	0.46	0.65	1.04	1.62	2.32
			F	0.43	0.59	0.94	1.53	2.21
		5	1	0.47	0.64	1.02	1.64	2.39
			2	0.45	0.61	0.97	1.59	2.31
			F	0.43	0.57	0.90	1.47	2.14
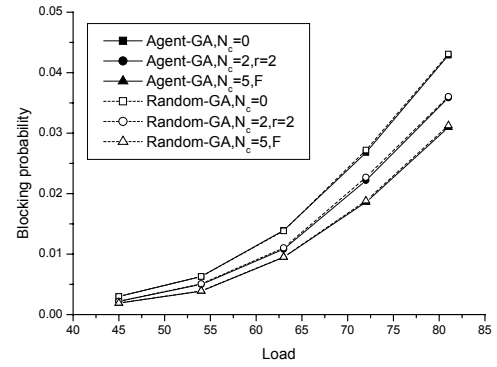
(a)
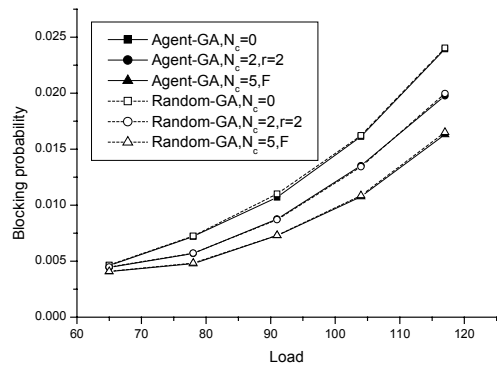
(b)

Fig. 7 Performance of the algorithm with Ant-based agents search in compare with random search for initial population (a) NSF network. (b) EON network.

The results in Fig.8 indicate that the blocking probability corresponding to the new fitness function is significantly lower than that of the old fitness function for all the cases we studied. For the NSF network with two wavelength converters and r=2, the blocking probability corresponding to the new fitness function is about 0.022 while the blocking probability corresponding to the old fitness function is 0.026 under the workload of 72. For the NSF network with a same workload but five wavelength converters, the blocking probability is 0.018 and 0.023 for the new and old fitness functions, respectively. Similar behaviors are also observed in the EON network. For the EON network containing two wavelength converters and having a workload 104, the blocking probability of the new fitness function is 0.013 while the blocking probability of the old fitness function is 0.021. For the same workload of 104, Fig. 8 (b) shows that the blocking probability of the new fitness function is 0.010 while the blocking probability of the old fitness function is 0.016 when the EON network contains five converters.

two wavelength converters, the blocking probability of the FA algorithm is about 0.065 when *r*=2 and the workload is 104, about three times higher than the blocking probability of our algorithm, which is 0.013. For the same values of *r* and workload, Fig. 10 (b) shows that the blocking probability of our algorithm is 0.012 when the EON network has five converters, and this blocking probability is about two times lower than the 0.04 blocking probability of the FA algorithm.
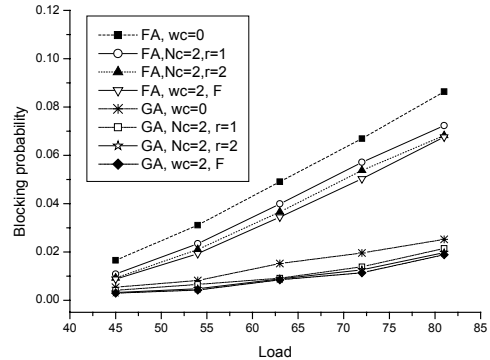
(a)
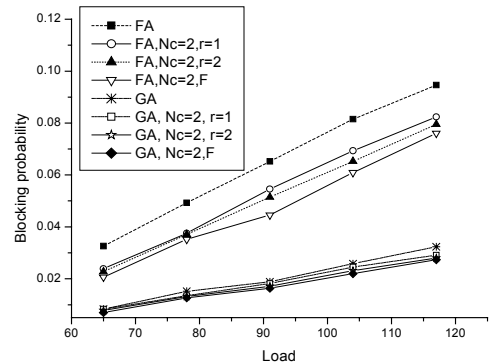
(b)

Fig.9 Blocking probability vs. workload with Nc=2, α =0.9 and C_{wc} =0.4. (a) NSF network. (b) EON network. F: Full-range conversion.

Figs. 9 and 10 also show that for both the FA and our algorithms, while we can always decrease blocking probability by adopting wavelength converters, this decrease is inversely proportional to the number of wavelength converters and wavelength conversion capability of each converter. Actually, by adopting only limited-range wavelength converters, our algorithm can achieve performance similar to that of using full-range wavelength converters. For example, for the NSF network with two converters and a workload of 63, the blocking probability of our algorithm is 0.0108 when *r*=2, and this blocking probability is slightly reduced to 0.0095 when we adopt full-range wavelength converters. For the EON

(a)

(b)

Fig. 8 Performance of the new fitness function in comparison with the old fitness function. (a) NSF network. (b) EON network

4.3 Overall Performance Comparisons

To illustrate the overall performance of the our hybrid algorithm, we show in Fig. 9 and Fig. 10 the comparisons between our hybrid algorithm and the Fixed-Alternate (FA) algorithm in terms of blocking probability for the cases of adopting two and five converters, respectively.

The comparisons in Fig. 9 and Fig. 10 show clearly that blocking probability with the new algorithm is significantly lower than with the Fixed-Alternate routing algorithm for all the cases we studied, but the improvement from using our algorithm decreases as the number of wavelength converters increases. For example, for NSF network with two wavelength converters and *r*=2, the blocking probability of FA algorithm is about 0.053 while that of our algorithm is only 0.022 when the workload is 72. For the same set of *r* and workload in the NSF network with five wavelength converters (Fig. 10 (a)), the blocking probability is about 0.041 for the FA algorithm and is 0.020 for our algorithm. Similar behaviors can also be observed in the EON network (Fig. 9 (b) and Fig. 10 (b)). When the EON network contains

network with five converters and a workload of 91, the blocking probability of our algorithm is 0.0077 when $r=2$, and we can only reduce this blocking probability slightly to 0.0073 when we use full-range wavelength converters in the network.

The results in both Fig. 9 and Fig. 10 indicate that our new algorithm significantly outperforms the Fixed-Alternate (FA) in terms of blocking probability. This is partially due the reason that the FA algorithm used in our comparison has only two alternative routes determined statically in advance, so it has very limited adaptability to the state variation of a dynamic network. On the other hand, each generation of our hybrid algorithm has 16 feasible alternate routes (P=16) determined dynamically based on the current network state, so our algorithm has a much better adaptability to the state variation of a dynamic network and thus has more chances to find a free route for a connection request than that of the FA algorithm. Another reason why the proposed algorithm works well in the numerical simulation is due to the powerful evolution capability of GA combined with the new fitness function, such that a good load balance can be achieved by using the hybrid algorithm, which in turn further reduces the blocking probability as indicated in Fig. 8.
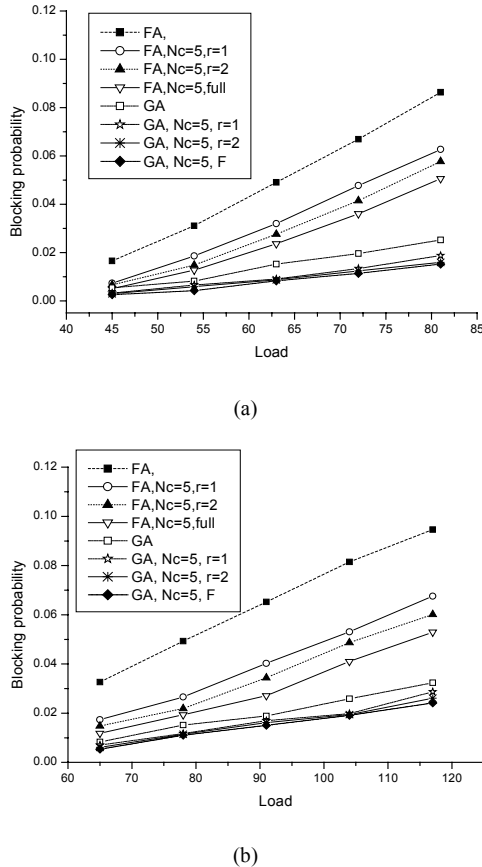

(a)

(b)

Fig.10 Blocking probability vs. workload for Nc=5, α=0.9 and C_{wc}=0.4.   (a) NSF network. (b) EON network. F: Full-range conversion.

To show that our new algorithm can result in a significantly lower setup delay than the old GA-based dynamic RWA algorithm [14], we summarize in Table 3 and Table 4 the average execution time of a request for the NSF network and the EON network based on the simulation in a computer with Pentium III 600 Mhz processor, HDD 10 GB IDE and 512 MB RAM. For comparison, we also include in Table 3 and Table 4 the average execution time of the FA algorithm.

Table 3. Average execution time in the NSF network (*ms*). N_c: number of wavelength converters

Algorithm/ Workload	45	54	63	72	81
old GA, N_c=0	9.72	10.08	10.38	11.39	11.77
old GA, N_c=2	10.91	11.35	11.73	12.48	12.94
old GA, N_c=5	11.54	11.89	12.32	12.75	13.31
new GA, N_c=0	4.72	5.06	5.30	5.82	6.00
new GA, N_c=2	5.46	5.78	6.05	6.84	7.09
new GA, N_c=5	6.01	6.43	6.56	7.14	7.68
FA, N_c=0	0.062	0.076	0.092	0.114	0.144
FA, N_c=2	0.067	0.081	0.098	0.122	0.153
FA, N_c=5	0.076	0.095	0.111	0.130	0.169

Table 4. Average execution time in the EON network (*ms*) N_c: number of wavelength converters

Algorithm/ Workload	65	78	91	104	117
old GA, N_c=0	12.70	13.02	13.26	14.43	14.54
old GA, N_c=2	12.91	13.63	14.08	15.01	15.27
old GA, N_c=5	13.17	14.02	14.45	15.23	15.54
new GA, N_c=0	4.57	4.76	4.94	5.13	5.28
new GA, N_c=2	5.04	5.27	5.62	5.89	6.07
new GA, N_c=5	5.22	5.51	5.92	6.16	6.38
FA, N_c=0	0.072	0.086	0.140	0.177	0.206
FA, N_c=2	0.083	0.099	0.156	0.194	0.221
FA, N_c=5	0.098	0.120	0.167	0.206	0.240

The results in Table 3 and Table 4 indicate that for both the NSF and EON networks, the average execution time of our new algorithm is significantly lower than that of the old GA-based dynamic RWA algorithm. For the NSF network with a workload of 72 Erlangs, the average execution time of the old GA algorithm is 11.39 *m*s, which is almost two times higher than the 5.82 *m*s average execution time of the new algorithm. When the EON network works under a workload of 78 Erlangs, the average execution time of the old GA algorithm is 13.02 *m*s, which is almost three times higher than the average execution time of the new algorithm (4.76 *m*s). It is notable in above two tables that FA algorithm requires a much smaller execution time than that of the conventional

GA algorithm and our hybrid algorithm, because the FA algorithm used in our comparison has only two alternative routes determined statically in advance and it does not involve any evolution process to find a route for a request. However, the smaller execution time of FA comes at the price of a very limited capability to keep a good load balance and thus a significantly higher blocking probability than that of the new hybrid algorithm, as indicated in Figs. 9 and 10.

5. Conclusion

In this paper, we proposed a hybrid algorithm for dynamic RWA in optical WDM networks with sparse wavelength conversion. By combining the mobile agents technique with an appropriate genetic algorithm, we have produced a hybrid algorithm that is able to reduce the time consuming process of generating the first population of routes while retaining the GA's attractive ability to achieve a significantly low blocking probability. To take full advantage of wavelength conversion, we also propose a new reproduction scheme and a more general fitness function that simultaneously takes into account the path length, number of free wavelengths, and wavelength conversion capability in route evaluation. Extensive simulation results show that the new hybrid algorithm can achieve a good load balance and always has a lower blocking probability than the promising Fixed-Alternate routing algorithm for both networks, with full-range wavelength conversion and with limited-range conversion. Our simulation results also indicate that in a WDM network with sparse wavelength conversion, limited-range wavelength conversion can actually achieve a performance level similar to that of full-range wavelength conversion. Note that this paper focuses mainly on how to combine GA and mobile agents technique to solve the dynamic RWA problem, so we simply adopt some very basic genetic operations in our hybrid algorithm. We expect that the performance of our hybrid algorithm can be further enhanced by adopting some improved genetic operations, e.g. the advanced genetic operations (specially the crossover operation) developed by S. Kobayashi's group [27]. This remains to be our future research

Acknowledgments

References

[1] S. Subramaniam, M. Azizoglu, and A.K. Somani, "All-optical networks with sparse wavelength conversion", *IEEE Transactions on networking*, vol. 4, no. 4, 1996, pp. 544-557

[2] S. Ramamurthy and B. Mukherjee, "Fixed-Alternate routing and wavelength conversion in wavelength-routed optical networks", *IEEE/ACM Transactions on networking*, vol. 10, no. 2, 2002, pp. 351-367.

[3] J. Kleimberg and E. Kumar, "Wavelength conversion in optical networks", *Proceedings of the tenth annual ACM-SIAM symposium on discrete algorithms*, 1999, pp.566-575

[4] B. Ramamurthy and B. Mukherjee, "Wavelength conversion in WDM networking", *IEEE Journal on selected areas in communication*, vol.16, no.7, 1998

[5] K. C. Lee and V.O.K. Li, "A wavelength-convertible optical network", *Journal of lightwave technology*, vo.11, no. 5/6. 1993

[6] M. Kovacevic and A. Acampora, "Benefits of wavelength translation in all-optical clear-channel networks", *IEEE Journal on Selected Areas in Communications*, vol.14, no.5, 1996, pp. 868-880

[7] R. Ramaswami and K.N. Sivarajan, "Routing and wavelength assignment in all-optical networks", *IEEE/ACM Transactions on Networking*, vol. 3, 1995, pp.489-500.

[8] M. Ahmed and M. Azizoglu, "Adaptive wavelength routing in all-optical networks", *IEEE/ACM Transactions on networking*, vol.6, no.2, 1998, pp.197-206.

[9] H. Zang and et al., "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks", *Optical Networks Magazine*, vol. 1, no.1, 2000, pp. 47-60.

[10] X. Chu, B. Li and Z. Zhang, "A dynamic RWA algorithm in a wavelength-routed all-optical network with wavelength converters", *IEEE INFOCOM 2003*.

[11] E. Kaeasan and E. Ayanoglu, "Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks", *IEEE/ACM Trans. Networking*, vol.6, no.2, 1998, pp.186-196

[12] N. Banerjee, V. Mehta and S. Pandey, "A genetic algorithm approach for solving the routing and wavelength assignment problem in WDM networks", *International Conference on Networks (ICN'04)*, French Caribbean (2004).

[13] S. Pandey, N. Banerjee and V. Mehta, "A new genetic algorithm approach for solving the dynamic routing and wavelength assignment problem in WDM networks", *7th IEEE International conference on High speed Networks and Multimedia Communications (HSNMC'04)*, Toulouse, France (2004).

[14] D. Bisbal and et al, "Dynamic Routing and Wavelength Assignment in Optical Networks by

Means of Genetic Algorithms", *Photonic Network Communications,* vol. 7, no.1, 2004, pp. 43-58.

[15] V.T. Le, S.H. Ngo, X. Jiang, S. Horiguchi and M. Guo, "A Genetic Algorithm for Dynamic Routing and Wavelength Assignment in WDM Networks", Proc. Inter. Symp. Parallel and Distributed Processing and Applications. HongKong, Dec.2004.

[16] G. Di Caro and M. Dorigo, "AntNet: A mobile agents approach to adaptive routing", *Technical Report* 97-12, IRIDIA, Universite Libre de Bruxelles, 1997.

[17] E. Bonabeau, M. Dorigo, G. Theraulaz, "Swarm intelligence: from natural to artificial systems", *Oxford University Press, Inc*., NewYork, 1999.

[18] R. M. Garlick and R. Barr, "Dynamic wavelength routing in WDM networks via Ant Colony Optimization", *Ant Algorithms,* Springer-Verlag Publishing, September 2002.

[19] T.White, B. Pagurek, and F. Oppacher, "ASGA: Improving the Ant System by Integration with Genetic Algorithm", *Proc. Third Conf. Genetic Programming (GP/SGA'98)*, 1998, pp. 610-617.

[20] S.H. Ngo, X. Jiang, S. Horiguchi and M. Guo, "Ant-Based Dynamic Routing and Wavelength Assignment in WDM Networks", *Proc. International Conference on Embedded and Ubiquitous Computing (EUC2004),* LNCS 3207, pp. 829-838, Aizu-Wakamatsu City, Japan, August 2004.

[21] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc., 1997

[22] Z. Michalewicz, *Genetic Algorithms + Data Structres = Evolution programs*, Springer-Verlag, 1992.

[23] O.Takahashi and S.Kobayashi, "An Adaptive Neighboring Search Using Crossover-Like Mutation For Multimodal Function Optimization," IEEE International Conference on Systems, Man and Cybernetics, pp.261-267 (2001).

[24] Y.Nagata and S.Kobayashi, "Analyses of Genetic Algorithms for Traveling Salesman Problems for Effective Searches," The 32ND ISCIE International Symposium on Stochastic Systems Theory and its Applications, pp.44-45 (2000).

[25] Y.Nagata and S.Kobayashi, "An analysis of Edge Assembly Crossover for the Traveling Salesman Problem," Proc. 1999 Int. Conf. on Systems, Man and Cybernetics, III-628-633 (1999).

[26] Y. Nagata and S. Kobayashi, "Edge Assembly Crossover: High-power Genetic Algorithm for the Traveling Salesman Problem," Proc. of 7th Int. Conf. on Genetic Algorithms, pp.450-457 (1997).

[27] http://www.fe.dis.titech.ac.jp/

[28] C.F. Hsu, T.L. Liu, and N.F. Huang, "Performance of adaptive routing in wavelength-routed networks with wavelength conversion capability", *Photonic Network communications*, 2003, pp. 41-57

[29] X. Chu, B. Li and I. Chlamtac, "Wavelength converter placement under different RWA algorithms in wavelength-routed all-optical networks", *IEEE/ACM Transactions on Communications*, vol. 51, no.4, 2003, pp. 607-617

[30] E. Bonabeau and et al, "Routing in telecommunication networks with smart ant-like agents," *Proc. IATA*, Lectures Notes in AI, vol. 1437, Springer Verlag (1998).

[31] The network simulator, ns-2. http://www.isi.edu/nsnam/ns/index.html, 2003.