PAPER

Ant-Based Alternate Routing in All-Optical WDM Networks

Son-Hong NGO[†a)], **Xiaohong JIANG**[††], ***Nonmembers***, ***and*** **Susumu HORIGUCHI**[††], ***Member***

SUMMARY We propose an ant-based algorithm to improve the alternate routing scheme for dynamic Routing and Wavelength Assignment (RWA) in all-optical wavelength-division- multiplexing (WDM) networks. In our algorithm, we adopt a novel twin routing table structure that comprises both a *P*-route table for connection setup and a pheromone table for ants' foraging. The *P*-route table contains *P* alternate routes between a source-destination pair, which are dynamically updated by ant-based mobile agents based on current network congestion information. Extensive simulation results upon the *ns-2* network simulator indicate that by keeping a suitable number of ants in a network to proactively and continually update the twin routing tables in the network, our new ant-based alternate routing algorithm can result in a small setup time and achieve a significantly lower blocking probability than the promising alternate shortest-path (ASP) algorithm and the fixed-paths least congestion (FPLC) algorithm for dynamic RWA even with a small value of *P*.

key words: *ant-based routing, routing and wavelength assignment, WDM networks, alternate routing*

1. Introduction

All-optical networks using wavelength-division- multiplexing (WDM) technology are now considered very promising to meet the huge bandwidth demand of the next generation Internet. In wavelength-routed WDM networks, data is switched and routed in all-optical domain via lightpaths. The Routing Wavelength and Assignment (RWA) problem concerns in determining a path and a wavelength to establish lightpaths for connection requests, and the RWA problem plays an important role in improving the performance of WDM networks [1], [2]. The RWA problem can be classified into the static RWA and dynamic RWA problems. In the static RWA problem, the connection requests are given in advance. In contrast, the dynamic RWA considers the case where the connection requests arrive randomly. The dynamic RWA is more challenging; therefore, heuristic algorithms are usually employed in resolving this problem [3]. Without wavelength converters, the same wavelength must be assigned on every link of a lightpath, this referred to as the wavelength-continuity constraint.

In this paper, we focus on the dynamic RWA problem under the wavelength-continuity constraint. Static routing approaches are available for the dynamic RWA problem, such as the shortest-path routing (SP) and the alternate shortest-path routing (ASP) [3], [4]. These approaches compute statically a set of shortest paths without acquiring the current network state. The main advantage of alternate shortest-path routing is its simplicity, in the sense it has a small setup time and low control overhead while providing a significantly lower blocking probability than the shortest path routing [5]. Adaptive routing approaches, such as adaptive-unconstraint routing using exhaustive search (AUR-E) [6] or least-loaded routing (LLR) [7], are more efficient than the static routing methods in terms of blocking probability. However, the main problems of these adaptive routing methods are their longer setup delay and higher control overhead, including the requirement of global network's state on each node. To solve the dynamic RWA problem efficiently, Li et al. [8] proposed an advanced alternate dynamic routing algorithm, called fixed-paths least congestion (FPLC). This algorithm routes a connection request on the least congested path out of a set of predetermined paths. It is shown that FPLC outperforms the fixed-alternate routing method. Li et al. also proposed in [8] a FPLC-N(k) method using neighborhood information from only k links on each searched path, and they proved that this FPLC-N(k) method can achieve a good trade-off among between network performance, setup delay and control overhead. It is notable, however, that the current alternate routing algorithms such as ASP or FPLC are all based on a set of fixed and static paths computed in advance, so these algorithms have little adaptability to network traffic variations and may significantly limit the routing performance in terms of blocking probability.

Inspired from the behaviors of natural ant system, a new class of ant-based algorithms for network routing is currently being developed. Previous work has shown the potential success of ant-based routing for both packet switching networks (e.g. AntNet [10]) and circuit switching telephone networks (e.g. ABC [11]). Recently, Garlick et al. [12] proposed an algorithm for dynamic RWA problem via Ant Colony Optimization. This algorithm can achieve a good network performance in terms of blocking probability, but it suffers from a high setup delay because this algorithm uses too many ants to search a path after a connection request arrives.

In this paper, we propose a hybrid ant-based routing algorithm (HABR) for dynamic RWA by combining the best of mobile agent approach's good adaptability [9] and the

Manuscript received April 21, 2005.
Manuscript revised September 15, 2005.

[†]The author is with the Graduate School of Information Science, Japan Advanced Institute of Science and Technology, 923-1292, Japan.
[††]The authors are with the School of Information Sciences, Tohoku University, Sendai-shi, 980-8579, Japan.
a) E-mail: sonhong@jaist.ac.jp

DOI: 10.1093/ietcom/e89–b.3.748

alternate routing method's simplicity while avoiding their shortcomings. To enable the new HABR algorithm, we adopt a novel twin routing table structure on each network node that comprises a *P*-route table for connection setup and a pheromone table for ants' foraging. The *P*-route table contains a set of *P* feasible paths between a source-destination pair, which are dynamically updated by ant-based mobile agents based on current network congestion information. By keeping a suitable number of ants in a network to continually update the twin routing tables in the network, the candidate alternate routes are available at the arrival of a connection request, thus a small setup delay is guaranteed in our new algorithm. Moreover, our ant-based approach does not require the global information on network states for the route selection as in other adaptive RWA routing algorithms. Extensive simulation results indicate that our algorithm can significantly outperform the ASP when the same number of alternate paths is adopted in both algorithms. Further simulation indicates that by adopting a new method to evaluate the goodness of a route based on both its length and the number of its idle wavelengths, our algorithm can also outperform the FPLC algorithm in terms of blocking probability with a suitable number of ants and a small value of *P*.

The rest of this paper is organized as follows: Sect. 2 presents some related works in more details. Section 3 describes our RWA algorithm that is a combination of mobile agent technique and the alternate routing scheme. Section 4 presents a comparison among our new algorithms, the alternate shortest path (ASP) algorithm and the fixed-paths least congestion (FPLC) algorithm based on an extensive simulation study. Section 5 concludes this paper.

2. Related Works

In this section, we discuss in more details some ant-based approaches for dynamic RWA that has been proposed recently.

In [12], Garlick et al. proposed an adaptive algorithm for dynamic RWA problem via Ant Colony Optimization. Garlick's algorithm works as follows. When a connection request arrives at a node, a number of ants are launched from source to destination to search for paths. Once an ant reaches the destination, it reports a path between the source and destination. Each reported path is scored based on the length and congested information. After all the ants complete their searching, the best path is selected from all the reported paths to establish the connection request. This approach suffers from high setup delay due to the waiting for all ants to complete their search. Basically, ants launching from one node do not cooperate with ants from others by mean of pheromone communication [10], [11].

Recently, Ngo et al. [13] proposed an adaptive RWA algorithm using ant-based approach, called ant-based routing (ABR). In this algorithm, a network node is equipped with a probabilistic pheromone table that contains the selection probability of neighbor node when an ant is moving toward its destination node. Ants are launched from each node with a given probability to a randomly selected destination every time unit. This algorithm ensures that the information about network congestion is well reflected in the pheromone tables. The route for a connection request is selected directly based on the highest selection probability or the second highest probability, thus can reduced the setup time. The results show that ABR algorithm outperforms the alternate shortest path algorithm in terms of blocking probability. Because the ant-based mobile agents in ABR algorithm can well explore the network state, we will inherit this property to improve the alternate routing methods such as ASP or FPLC that are based on a set of fixed and static paths between a source-destination pair.

3. Hybrid Ant-Based Routing (HABR) Algorithm

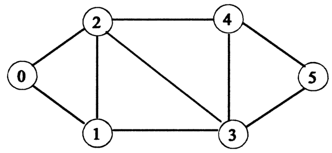
In this section, we present our HABR algorithm for dynamic RWA problem under the wavelength-continuity constraint, that means every network nodes is not equipped with wavelength converters. Each node now has a twin routing table that comprises a *P*-route table for connection setup and a pheromone table for ants' foraging. The *P*-route table contains for each destination a set of *P* dynamic routes between current node and that destination. Upon arrival of a connection request on a node, the k ($k \leq P$) best routes among the *P* alternate routes to the corresponding destination node in the *P*-route table will serve as the candidates for lightpath setup. The main difference between the HABR algorithm and other alternate routing algorithms is that the *P*-route table in a node is now proactively and continually updated by ant-based mobile agents based on the network state [13], thus we have more chances to find a feasible route for lightpath setup based on the dynamic routes in the new *P*-route table than based on simply the static alternate routes as in the available alternate routing schemes.

We suppose that ant-based agents and control packets run in a separated control plane that is carried out in a packet switching network with the same topology as optical network, or in optical domain where control data is transported on a dedicated wavelength [1], [15].

3.1 Twin Routing Table Structure

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

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. In the *P*-route table, each row corresponds to a destination and contains a list of *P* routes to the destination. Each route is assigned a value that represents the goodness of this route based on its length and the number of idle wavelengths along it. The bigger the goodness of a



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	$r^3_{1,5} = 0.0$	$r^3_{4,5} = 0.2$	$r^3_{5,5} = 0.8$

a) Pheromon table

Destination	List of P routes
0	3.1.0; 3.2.0; ...
1	3.1; 3.2.1; 3.4.2.1; ...
...	...
5	...

b) P-route table

Fig. 1 A simple network and its routing tables of node 3.

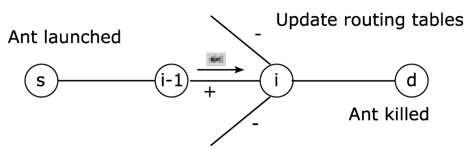


Fig. 2 Ant's moving and updating task.

route, the better path is considered for connection setup.

3.2 Ant's Foraging and Routing Tables Updating

On each network node and for every t time units (second), an ant is launched with a given probability ρ to a randomly selected destination; here ρ and t are design parameters. Each ant is considered to be a mobile agent: it collects information on its trip, performs routing tables updating on visited nodes, and continues to move forward as illustrated in Fig. 2.

3.2.1 Pheromone Table Updating

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 $(s, \dots i-1, i, \dots d)$. 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.

If an ant visits node i at time t, so the next values for routing entry in next time $t+1$ are determined as follows:

$$r^i_{i-1,s}(t+1) = \frac{r^i_{i-1,s}(t) + \delta r}{1 + \delta r} \tag{1}$$

$$r^i_{n,s}(t+1) = \frac{r^i_{n,s}(t) + \delta r}{1 + \delta r}, \forall n \neq i-1 \tag{2}$$

Here, dr is the reinforcement parameter or the amount of trailing pheromone, and dr decreases with the length of the route and increases with the number of available wavelengths:

$$\delta r = \frac{\alpha}{\delta l} + (1-\alpha) * \delta w \tag{3}$$

Here, dl corresponds to the length of the route (the length of route refers to the hop number of the route), dw corresponds to the percent of free wavelengths of this route, α is a scalar parameter that can be used in adjusting the emphasis of path length versus free wavelength percentage. These two factors are computed as follows:

$$\delta l = \beta * (e^{\frac{-1.0}{dl}}),\ \delta w = e^{\gamma * dw} - e^0 \tag{4}$$

Where dl is the difference between the length of current route and the length of the shortest route, dw is the percentage of free wavelengths of the route. Here α, β and γ are designed parameters and they are adjusted to get a good system performance [13].

3.2.2 P-Route Table Updating

Besides updating the pheromone table, ant also updates the P-route table. As mentioned above, each route is associated with a goodness value that is calculated by the following formula:

$$dr = \varphi * \frac{1}{dl+1} + (1-\varphi) * dw \tag{5}$$

Here, φ is a scalar parameter used to adjust the emphasis between the route length and the percentage of free wavelengths. The bigger value of φ, the larger goodness value the shorter route has. The smaller value of φ, the larger goodness value the route with more idle wavelengths has. The parameter φ should be chosen such that the shorter route has a larger goodness value, while for two routes with the same length, the route with a larger number of idle wavelengths gets a larger goodness value. It is formally proved in [14] that the value of φ should be selected based on the following inequality:

$$1 > \varphi > \frac{(W-1)(N-1)N}{W + (W-1)(N-1)N} \tag{6}$$

where N is the number of network nodes and W is the number of wavelengths per fiber.

When an ant reports its trip to a P-route table, it updates only the goodness value if the reported route already exists in the table. Otherwise, ant inserts the new route in the table or replace the route that has the smallest goodness value by the new one.

3.3 Connection Setup

Upon the arrival of a connection request, the k best routes that have the biggest goodness values in the P-route table are selected as the routes candidates for lightpath setup. The lightpath setup is similar to other alternate routing methods: the source node sends in parallel some needle packets [15] along these k routes toward destination to request a path setup. At the destination node, the route with the highest goodness value is selected for connection setup. If no free wavelength is available for all these k routes, the connection request will be blocked.

We use in our algorithm two alternate routes ($k = 2$) as suggested in previous works [5], [8] that alternate routing methods using too many alternate routes do not significantly improve network performance. To select a route for the connection setup in HABR algorithm, not only the number of idle wavelengths but also the network congestion is considered. However, the network congestion information is continuously updated by ant-based mobile agents. Thus, the setup delay depends only on the time that the needle packets get the results along these k alternate routes. It is noticed from Eq. (5) that only the number of idle wavelengths are searched by needle packets. This is also similar to the FPLC algorithm where the maximum number of idle wavelengths is considered. Moreover, it is noticed that the time complexity to select the k best routes from the P-route table is only $O(k * P)$. This is a local search on the source node based on the goodness value of each route. For the above reasons, the setup delay of HABR is similar to FPLC with the same number of alternate routes.

For the wavelength assignment, any method in [3] can be applied. We adopt the First Fit heuristic because it is simple but still can result in a good performance. In the First Fit heuristic, all wavelengths are considered and the first available one is selected.

The pseudo-code of the main steps in our 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;
   Endfor
   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;
         Update P-route table;
         Push trip's state into stack;
         If (found a next hop)
            Move ant to next hop ;
         Else
            Kill ant;
         End if
   End while
End for

{Connection setup}
Select k best route from the P-route table;
If (no wavelength available)
   Consider a blocking case;
Else
   Setup lightpath on highest goodness route;
```

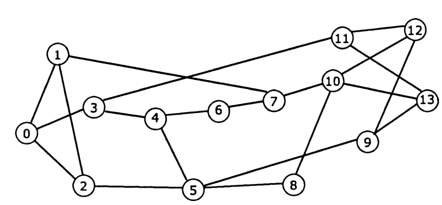
4. Simulation and Results Analysis

In this section, we verify the performance of our HABR algorithm based on simulation in a circuit-switched routing module for WDM network in the *ns-2* Network Simulator [16]. For comparison, we also conduct simulation for the ASP and FPLC algorithms that adopt two alternate routes. Two network topologies used in our simulation are the NSF network and the EON network, as illustrated in Fig. 3

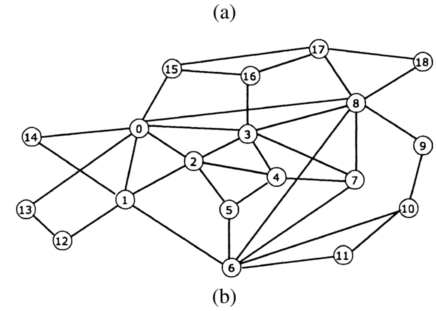
We adopt the general traffic model widely used in performance analysis of data communication networks [17]. There are totally T arriving traffic sessions; Arriving sessions are distributed randomly over the network. Each traffic session has many connection requests: for each traffic session, connection requests arrive according to a Poisson process with an arrival rate λ (call/s). Connection holding time is exponentially distributed with mean μ (s). The total network load is measured by $T * \lambda * \mu$ (Erlang).

The average holding time is set as $\mu = 10$ s, and two values of the number of wavelengths, $W = 8$ and $W = 16$, are considered here. For each case, the number of traffic sessions, the arrival rate are selected to have a reasonable range of traffic load such that the total blocking probability is about 5% - a practical value for WDM networks. The time step for ant's generation is set as $t = 1$ ms, and the delay for each link is assumed as 10 ms. To get a stable result, each experiment is run in 2000 s and it is repeated five times to get the average value of blocking probability.

As explain in the previous section, we keep the number of alternate routes $k = 2$. ASP and FPLC also use two alternate routes for two reasons. Firstly, alternate routing meth-



(a)



(b)

Fig. 3 Network topologies: (a) NSF network with 14 nodes and 21 links. (b) EON network with 19 nodes and 35 links.

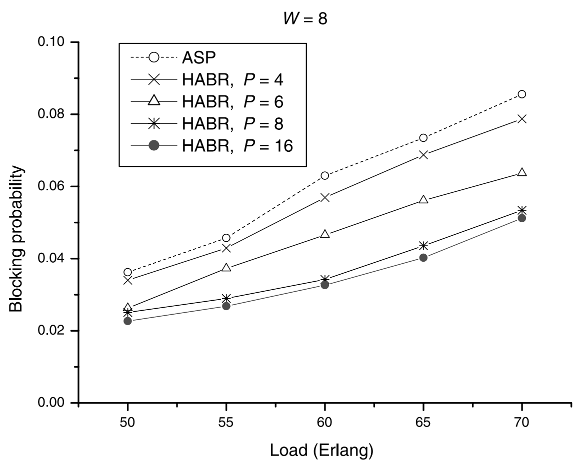


Fig. 4 Performance of HABR on difference value of P when $k = 2$.

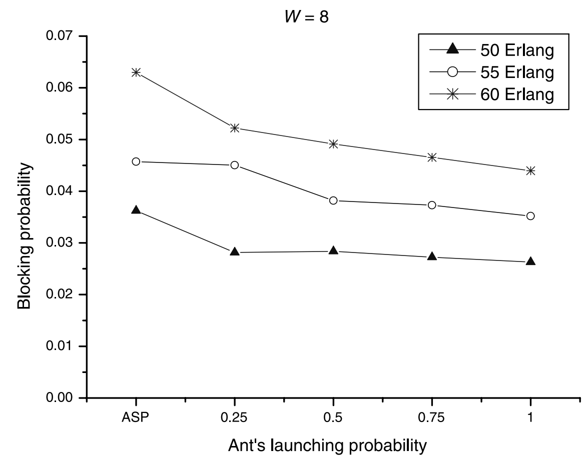


Fig. 5 Performance gain of HABR with difference value of ant's launching probability in compare with ASP on the NSF network.

ods like ASP and FPLC using more than two routes do not significantly improve performance [5], [8], [18]. Secondly, the more alternate routes are searched by needle packets over the network, the much higher cost for the connection setup process is; Thus we should apply the same number of alternate routes in all ASP, FPLC and HABR algorithms for a fair performance comparison.

Each experiment is conducted in an initial time without the traffic load to get an initial list of P routes. We found that ants can report a list of P shortest paths because ants only consider the path length at initial period. This confirms the results in previous work [9] that ant-based mobile agents tend to find the shortest paths if only the path length is considered in path selection process.

To find a suitable value of P for simulation, we compare the performance of HABR with ASP by keeping ant's launching probability at $\rho = 1$. Figure 4 shows the network blocking probability versus the total traffic load with $P = 4, 6, 8$ and 16 on the NSF network topology. When P is small, HABR slightly outperforms ASP, the difference is more significant when $P = 6$ or 8 but there is no much improvement when $P = 16$. This can be explained intuitively that when P is small, the P-route table is not large enough to contain many feasible routes reported by ants, which will reduce the chance to find a good route when k best routes are selected. Moreover, a small value of P may cause routes to be replaced too frequently so a good route may be replaced by another route that could become a bad route later. When P increases, ants can report a larger number of feasible routes into P-route table, so the blocking probability will be decreased. Too large value of P does not much increase performance significantly, because the number of good routes between two nodes is not large in reality. In fact, a too high value of P will increase the complexity of HABR, so take $P = 6$ or 8 are good enough for HABR to significantly outperform ASP in terms of blocking probability.

Figure 5 illustrates the performance of the HABR algorithm on the NSF network topology when $P = 8$ with different values of ant's launching probability ρ. We observe that the blocking probability decrease significantly as ρ increases. When the value of ρ is big enough to have a suitable numbers of ants exploring the network, the blocking probability does not decrease significantly anymore. In fact, too many ants in the networks ($\rho \approx 1$) will cause a high control overhead. However, our simulation results (Fig. 5) show that a good performance can be obtained within a large range value of ρ, as we can see in the figure when $\rho \in [0.5, 1]$. We will select $\rho = 0.75$ for our experiments because with this value, the blocking probability is significantly decreased in compare with ASP algorithm.

The blocking probability versus the traffic load of the routing algorithms ASP, FPLC and HABR with different numbers of wavelengths are shown in Fig. 6 and Fig. 7 for the NSF network and the EON network, respectively. All of the routing algorithms use $k = 2$ alternate routes. From previous experiments, we select $P = 6$ and $\rho = 0.75$ because HABR can perform a good network performance with these values. These figures show clearly that our HABR always outperform much better than both ASP and FPLC algorithms in terms of blocking probability under different traffic load conditions.

The above results can be explained as follows. Our HABR algorithm can outperform the ASP algorithm because the k alternate routes used for connection setup in HABR are continually updated by ant-based mobile agents, thus these k routes are much better than the fixed routes used by ASP. Moreover, each route in HABR is assigned with a goodness value as in Eq. (1); this goodness value allows ants to introduce into P-route table the routes with smaller path length and bigger number of idle wavelengths. As a result, HABR will select the least congested route in a more adaptive manner to the traffic variation than FPLC does. By consequence, our HABR algorithm can outperform the FPLC algorithm in terms of blocking probability.

We can also observe from Fig. 6 and Fig. 7 that when the traffic load is small, the HABR algorithm just slightly

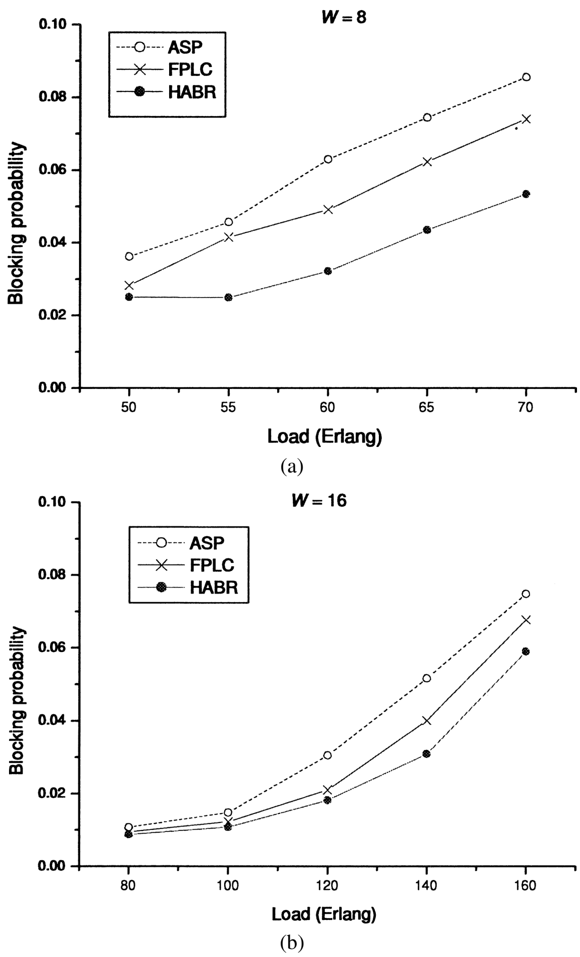


Fig. 6 Comparisons between HABR and others algorithms on NSF network with $P = 6,\ k = 2,\ \rho = 0.75$.

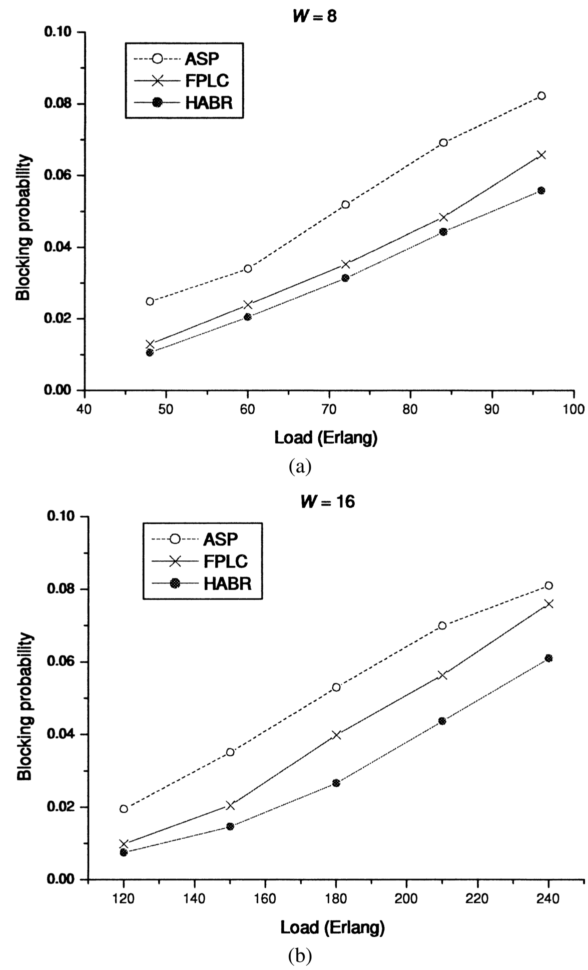


Fig. 7 Comparisons between HABR and others algorithms on EON network with $P = 6,\ k = 2,\ \rho = 0.75$.

outperforms the FPLC and ASP algorithms. In contrast, the difference is much more significant when the traffic load is high. This is because the ASP algorithm is a fixed routing algorithm, while the FPLC algorithm is a dynamic routing but it is not flexible enough to adapt to the traffic variation. The FPLC algorithm is dynamic in terms of using only the information of idle wavelengths on a set of pre-computed routes. In contrast, the HABR algorithm is much more dynamic in terms of using information about networks congestion that is explored by mobile agent. That is why our algorithm can much more outperform the ASP and FPLC algorithm under different value of traffic load, even when the traffic load is very high.

To show the control overload caused by the ant-based mobile agents, we compute by simulation the number of update on each routing table P-route when setting the time step for ant's generation $t = 1$ ms. It is observed from Fig. 8 that the number of update is proportional to the ant's launching probability ρ. When ρ increases, the number of update increases too, so the blocking probability is decreased. The number of update is highest when $\rho = 1$, which may causes a high control overhead. However, as shown is Fig. 5, a significant reduce of blocking probability could be obtained when

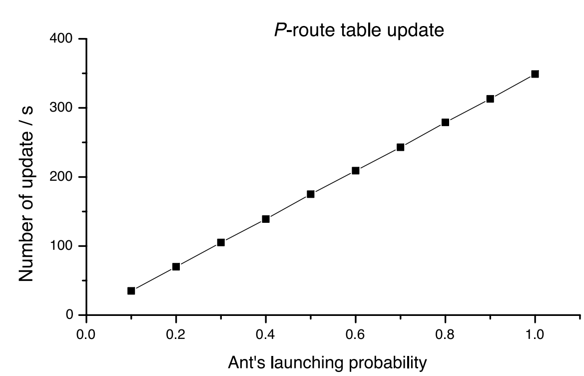


Fig. 8 Number of updates for a P-route table versus ant's launching probability when $t = 1$ ms, EON network.

$\rho \approx 0.5 \div 0.75$, it means that the number of update is about 150 to 250 times per second.

In summary, the HABR algorithm can always outperform significantly the ASP and the promising FPLC algorithms in term of blocking probability while using the same alternate routes. The setup process of HABR is similar to other alternate methods as it takes only k alternate routes for connection setup. Moreover, the ant-based mobile agents always explore and update the *P*-route tables before connection requests arrive, so our HABR algorithm does not increase the setup time in compare with other alternate routing methods.

It is noticed that we can apply more advanced techniques of alternate routing methods to improve furthermore the performance of our HABR algorithm such as routing using neighborhood information [8], or another techniques to evaluate the goodness of a route to introduce the load balancing as in [12]–[14]. However, these problems are out of scope of this paper. In this work, we use the simple function to evaluate the goodness of a route as in Eq. (1) in order to emphasize the advantages when using ant-based approach in combine with alternate routing method to improve the network performance. The problem of how to find the ant's launching probability and the number of routes P to reduce the control overhead still remain open for furthermore research.

5. Conclusion

In this paper, we proposed a hybrid algorithm for dynamic RWA in WDM networks based on the combination of ant-based mobile agents technique and alternate routing method. To enable the new hybrid algorithm, we proposed to use a novel twin routing table structure that consists of a *P*-route table for routing and a pheromone table for mobile agents' foraging. Our new algorithm is highly adaptive in the sense that it always keeps a suitable number of ants to continuously exploring the network state and proactively updating the *P*-route routing table. Our simulation results indicate that with a small number of routes in the *P*-route table and a suitable number of ants, our algorithm can always outperform the promising alternate shortest-path routing algorithm and the fixed-path least congestion routing algorithm in terms of blocking probability while guaranteeing a small setup time in compare with other adaptive routing methods. In the further, we will more evaluate the HABR algorithm in various network situations and its applicability to real optical fiber networks.

Acknowledgements

This research is partly supported by the Grand-In-Aid of scientific research (B) 14380138 and 16700056, Japan Science Promotion Society.

References

[1] R. Ramaswami and K. Sivarajan, Optical Networks: A Practical Perspective, Morgan Kaufman Publishers, San Francisco, CA, 2002.

[2] B. Mukherjee, Optical Communication Networks, McGraw-Hill, New York, 1997.

[3] H. Zang, J.P. Jue, and B. Mukhesjee, "A review of routing and wavelength assignment approaches for Wavelength-Routed Optical WDM Networks," Optical Networks Magazine, vol.1, pp.47–63, Jan. 2000.

[4] A. Birman, "Computing approximate blocking probabilities for a class of all-optical networks," IEEE J. Sel. Areas Commun., vol.14, no.5, pp.852–857, June 1996.

[5] S. Ramamurthy and B. Mukherjee, "Fixed-alternate routing and wavelength conversion in wavelength-routed optical networks," Proc. IEEE GLOBECOM, pp.2295–2302, Nov. 1998.

[6] H.T. Mokhtar and M. Azizoglu, "Adaptive wavelength routing in all-optical networks," IEEE/ACM Trans. Netw., vol.6, no.2, pp.197–206, April 1998.

[7] B. Zhou and H.T. Mouftah, "Adaptive least loaded routing for multi-fiber WDM networks using approximate congestion information," Proc. IEEE ICC, vol.5, pp.2745–2749, 2002.

[8] L. Li and A.K. Somani, "Dynamic wavelength routing using congestion and neighborhood information," IEEE/ACM Trans. Netw., vol.7, no.5, pp.779–786, Oct. 1999.

[9] M. Dorigo and V. Maniezzo, "Ant system: Optimization by a colony of cooperating agents," IEEE Trans. Sys. Man Cybern. B, Cybern., vol.26, no.1, pp.29–41, 1996.

[10] M. Di Caro and M. Dorigo, "AntNet: Distributed stigmergetic control for communications networks," J. Artificial Intelligence Research, pp.317–365, Dec. 1998.

[11] R. Schoonderwoerd, O. Holland, and J. Bruten, "Ant-like agents for load balancing in telecommunications networks," Proc. 1st Inter. Conf. Autonomous Agents, pp.209–216. ACM Press, Feb. 1997.

[12] R.M. Garlick and R.S. Barr, "Dynamic wavelength routing in WDM networks via Ant Colony Optimization," in Ant Algorithms, pp.250–255, Springer-Verlag Publishing, 2002.

[13] S.H. Ngo, X. Jiang, and S. Horiguchi, "Adaptive routing and wavelength assignment using ant-based algorithm," Proc. 12th IEEE ICON, vol.2, pp.482–486, Singapore, Nov. 2004.

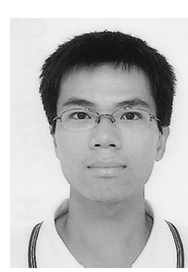
[14] V.T. Le, X. Jiang, S.H. Ngo, and S. Horiguchi, "Dynamic RWA based on the combination of mobile agents technique and genetic algorithm in WDM networks with sparse wavelength conversion," Proc. 19th IEEE IPDPS, p.61a, Colorado, USA, April 2005.

[15] R. Ramaswami and A. Segall, "Distributed network control for wavelength routed optical networks," Proc. IEEE INFOCOM, pp.138–147, San Francisco, CA, March 1996.

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

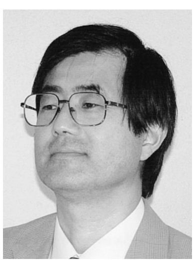
[17] A. Girard, Routing and Dimensioning in Circuit-Switched Networks, Addison-Wesley, 1990.

[18] E. Karasan and E. Ayanoglu, "Effects of wavelength routing and selection algorithms on wavelength conversion gain in WDM optical networks," IEEE/ACM Trans. Netw., vol.6, no.2, pp.186–196, April 1998.



Son-Hong Ngo received the B.S. from Faculty of Information Technology, Hanoi University of Technology and M.S. degrees from Francophone Institute for Computer Science (IFI) in 2000 and 2002 respectively. During 2001–2002, he stayed in Software, Systems and Networks Laboratory (LSR) in National Polytechnical Institute of Grenoble (INPG), France to study about active networking in wire/wireless networks. He is now a doctoral student in the Graduate School of Information Science, JAIST (Japan Advanced Institute of Science and Technology). His current research interest is about mobile agents approach for optical networks routing.

Xiaohong Jiang received his B.S., M.S. and Ph.D. degrees in 1989, 1992, and 1999 respectively, all from Xidian University, Xi'an, China. He is currently an associate professor in the Department of Computer Science, Graduate School of Information Science, Tohoku University, Japan. Before joining Tohoku University, Dr. Jiang was an assistant professor in the Graduate School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), from Oct. 2001 to Jan. 2005. Dr. Jiang was a JSPS (Japan Society for the Promotion of Science) postdoctoral research fellow at JAIST from Oct. 1999–Oct. 2001. He was a research associate in the Department of Electronics and Electrical Engineering, the University of Edinburgh from Mar. 1999–Oct. 1999. Dr. Jiang's research interests include optical switching networks, WDM networks, interconnection networks, IC yield modeling, timing analysis of digital circuits, clock distribution and fault-tolerant technologies for VLSI/WSI. He has published over 60 referred technical papers in these areas. He is member of IEEE.



Susumu Horiguchi received the B.Eng. the M.Eng. and Ph.D. degrees from Tohoku University in 1976, 1978 and 1981 respectively. He is currently a professor and Chair of Department of Computer Science, the Graduate School of Information Science, Chair of Department of Information Engineering, Faculty of Engineering, Tohoku University. He was a visiting scientist at the IBM Thomas J. Watson Research Center from 1986 to 1987. He was also a professor in the Graduate School of Information Science, JAIST (Japan Advanced Institute of Science and Technology). He has been involved in organizing international workshops, symposia and conferences sponsored by the IEEE, IEICE, IASTED and IPS. He has published over 150 papers technical papers on optical networks, interconnection networks, parallel algorithms, high performance computer architectures and VLSI/WSI architectures. Prof. Horiguchi is a senior member of IEEE and member of IPS and IASTED.