Journal of Network and Computer Applications 33 (2010) 512–522

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

QoS multicast tree construction in IP/DWDM optical internet by bio-inspired algorithms

Hui Cheng [a,b,*], Xingwei Wang [c], Shengxiang Yang [b], Min Huang [c], Jiannong Cao [d]

[a] Department of Computer Science, Sun Yat-Sen University, Guangzhou 510006, China
[b] Department of Computer Science, University of Leicester, University Road, Leicester LE1 7RH, UK
[c] College of Information Science and Engineering, Northeastern University, Shenyang 110004, China
[d] Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

ARTICLE INFO

Article history:
Received 8 May 2008
Received in revised form
29 September 2009
Accepted 3 January 2010

Keywords:
Optical Internet
QoS
Multicast
Ant colony optimization
Artificial immune algorithm

ABSTRACT

In this paper, two bio-inspired Quality of Service (QoS) multicast algorithms are proposed in IP over dense wavelength division multiplexing (DWDM) optical Internet. Given a QoS multicast request and the delay interval required by the application, both algorithms are able to find a flexible QoS-based cost suboptimal routing tree. They first construct the multicast trees based on ant colony optimization and artificial immune algorithm, respectively. Then a dedicated wavelength assignment algorithm is proposed to assign wavelengths to the trees aiming to minimize the delay of the wavelength conversion. In both algorithms, multicast routing and wavelength assignment are integrated into a single process. Therefore, they can find the multicast trees on which the least wavelength conversion delay is achieved. Load balance is also considered in both algorithms. Simulation results show that these two bio-inspired algorithms can construct high performance QoS routing trees for multicast applications in IP/DWDM optical Internet.

1. Introduction

Dense wavelength division multiplexing (DWDM) (Tancevski, 2003; Wei et al., 2008) is a key technology to exploit the tremendous bandwidth provided by optical fibers. By integrating IP network layer and DWDM physical layer, IP/DWDM optical Internet can provide high channel bandwidth and low transmission latency for data communication. Therefore, it has emerged as a promising candidate for next-generation networks (Guo et al., 2008). Nowadays, group communication (Law et al., 2005; Wang et al., 2006) becomes an important network application due to the increasing trend of collaborative work among a group of users such as video conference, content distribution and group disaster rescue. Group communication is normally conducted in a real-time manner which requires that the transmission delay from the source to any destination should not exceed the upper bound specified by the user. End-to-end delay (Chen et al., 2008; Giacomazzi, 2009; Tu et al., 2008) is an important QoS metric which determines the user QoS satisfaction degree (Cheng et al., 2009). To enable the group communication, the IP/DWDM optical Internet must provide the support of QoS multicast (Chen et al., 2008; Yan and Deogun, 2003), one of the essential capabilities for next-generation networks.

Multicast (Li and Li, 2005; Ye et al., 2007) is an important network service, which is the delivery of information from a source to multiple destinations simultaneously using the most efficient strategy to deliver the messages over each link of the network only once, creating copies only when the links to the destinations split. It provides underlying network support for collaborative group communications. In IP/DWDM optical Internet, the problem of QoS multicast is to search a feasible and effective routing tree (Hwang et al., 2007), through which the information can be delivered from the source to all the destinations. This problem is proved to be NP-hard (Znati et al., 2002). Therefore, efficient heuristic multicast algorithms should be developed to find the cost suboptimal tree and assign wavelengths to all the tree links.

Ant colony optimization (ACO) (Dorigo and Stützle, 2004) and artificial immune algorithm (AIA) (Castro and Timmis, 2002) are two bio-inspired intelligent computation techniques which are very powerful in solving combinatorial optimization problems. Especially, ant colony optimization approach has been widely applied to routing in the Internet and wireless mobile networks (Iyengar et al., 2007; Osagie et al., 2008; Sim and Sun, 2003). The multicast problem, due to its intractability, also attracts lots of interests of applying ACO to solve it (Hu et al., 2009; Wang et al., 2009). In optical networks, ACO has also been proved

* Corresponding author.
E-mail address: hc118@le.ac.uk (H. Cheng).

1084-8045/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2010.01.001

successful in solving quite a few problems such as the single-hop wavelength assignment (Chin, 2005), the fault-tolerant dynamic routing and wavelength assignment (RWA) (Pavani and Waldman, 2008), etc. Although AIA is not used as widely as ACO, several papers have shown its successful applications in unicast routing (Masutti and de Castro, 2009), multicast (Vijayalakshmi and Radhakrishnan, 2008), route guidance (Yang et al., 2007), etc. However, few works has been reported on integrated multicast routing and wavelength assignment (iMRWA) by ACO or AIA.

Both ACO and AIA have shown their strong search capabilities in the aforementioned applications. Considering that the iMRWA problem involves the search of the optimal solution in a huge solution space, the deterministic algorithms which construct only one multicast tree for the wavelength assignment cannot deal with it well. Therefore, these two optimization methods are worthy of in-depth investigation. In this paper, we propose two heuristic algorithms which search cost suboptimal routing trees based on ACO and AIA, respectively. For each candidate tree, a dedicated wavelength assignment algorithm is applied to minimize the number of wavelength conversion and thereby reduce the conversion delay. The end-to-end delay resulting from the wavelength assignment is considered in the tree evaluation. Thus, the wavelength assignment is integrated with the search of the low cost multicast trees. By this way, the cost of the multicast tree can approach the optimum whilst the user QoS requirement can be satisfied.

The rest of this paper is organized as follows. Section 2 describes the related work. Section 3 describes the network model and mathematical model. Section 4 presents the preliminary work including the solution expression and the wavelength assignment algorithm. Sections 5 and 6 describe the ACO and AIA-based algorithms, respectively. Simulation results are presented in Section 7. Finally, we conclude the paper in Section 8.

2. Related work

In recent years, some research work has been done in the area of multicast in optical network. They can be mainly classified into two types. The first type reports deterministic algorithms (Chen and Wang, 2002; Hwang et al., 2007; Jia et al., 2001; Jia et al., 2001) and the second type reports intelligent search heuristic algorithms (Chen and Tseng, 2005; Din, 2008). In Chen et al. (2008), an integer linear programming (ILP) method was proposed for the small-scale network and a heuristic algorithm was proposed for the large-scale network and. The algorithms proposed in this paper belong to the latter. In the following, we briefly review several representative algorithms in both types.

In Chen and Wang (2002), the proposed algorithm consists of a heuristic multicast algorithm and a wavelength assignment algorithm. It defines four kinds of costs associated with the WDM multicast. The multicast tree is produced by combining the optimal unicast light paths with the aim of minimizing the total cost of the multicast session. The wavelength assignment algorithm sets up the objective to minimize the wavelength conversion cost of the multicast trees.

In Hwang et al. (2007), this paper has discussed the multipoint-to-multipoint multicast problems in the all-optical networks with the full-capacity converter. It states that the light-tree algorithms can reduce the number of links needed for one-to-multipoint multicast transmission, but require more links to establish connections for multipoint-to-multipoint multicast transmission. In this work, a ring-tree-based routing and wavelength assignment (RTRWA) solution was proposed. The RTRWA algorithm finds an optimal ring path that connects all multicast session members with unidirectional links and connects the remaining nodes to the ring path using the light-tree.

In Jia et al. (2001), two QoS multicast algorithms for routing and wavelength assignment were proposed. Both algorithms utilize Minimum Spanning Tree (MST) to construct low cost multicast trees. In the tree construction procedure, the algorithms deal with the case that the multicast end-to-end delay from the source to a destination exceeds the pre-specified upper bound. The wavelength assignment is based on the greedy strategy, i.e., trying to assign a currently used wavelength to the multicast tree.

In Jia et al. (2001), the objective of the multicast algorithms is to minimize the number of used wavelengths. For a given set of multicast requests with bounded delay, the algorithms can construct trees and assign wavelengths to them. Two basic algorithms *A* and *B* were firstly proposed. Then two optimization algorithms *C* and *D* were proposed to further minimize the number of wavelengths over the results produced by *A* and *B*. Algorithms *C* and *D* integrate routing and wavelength assignment by using rerouting and reassigning techniques.

In Din (2008), the problem of optimal multiple multicast was investigated on WDM ring networks without wavelength conversion. Given a set of multicast requests, it applied several genetic algorithms to select a suitable path(s) and wavelength(s) for each request to minimize the used wavelengths. Since there is no wavelength conversion, a constraint is added that no any paths using the same wavelength pass through the same link. In Chen and Tseng (2005), the multicast routing under delay constraint problem was considered in a WDM network where nodes have different light splitting capabilities. The problem is first reduced to the MST problem and then solved by well-designed genetic algorithms.

In Chen et al. (2008), the problem of multicast routing and wavelength assignment with delay constraint was investigated. The objective is to find an optimal light-forest with minimum cost and assigning wavelengths to the light-trees in the light-forest for routing a request under a given delay bound in a WDM network. The paper proposed two algorithms to deal with the small-scale and the large-scale network, respectively. For the small-scale network, an integer linear programming model was proposed to formulate and solve the problem. For the large-scale network, an efficient heuristic was proposed to produce approximate solutions in polynomial time.

In Chen and Wang (2002), Chen and Tseng (2005), Chen et al. (2008), Jia et al. (2001), Jia et al. (2001), the delay requirement is bounded by a fixed value. In Din (2008), Hwang et al. (2007), the delay constraint is not considered at all. However, the setting of a fixed delay upper bound is not good enough for multicast applications where the users have flexible QoS requirements or they cannot describe the requirements accurately. The algorithms in Chen et al. (2008), Din (2008), Jia et al. (2001), Jia et al. (2001) are only applicable to single-hop WDM networks since no wavelength conversion is allowed. Therefore, they pose a limitation that all the links in a tree can only use the same wavelength. The algorithm in Chen and Wang (2002) separates routing and wavelength assignment. As a result, it is possible that there are no available wavelengths for the multicast tree or the wavelength assignment leads to deteriorated QoS performance.

In our algorithms, we consider the delay constraint since it is a very important QoS parameter in real-time applications. Furthermore, to be more practical, we propose the flexible QoS parameters concept and define it by interval values instead of single value used before. The two bio-inspired algorithms proposed in this paper can handle the optical networks well of any size, i.e., from small- to large-scale. By integrating the wavelength assignment into the multicast tree search, we can get rid of the unexpected case that a multicast tree is found by efforts but the wavelength assignment on it is unavoidably poor. However, one drawback resulted from the integration method is that since lots of stand-by multicast trees needs to be checked, the computation may be extensive.

3. Model description

3.1. Network model

IP/DWDM optical Internet can be modeled as a directed and connected graph $G(V, E)$, where V is the set of nodes representing optical nodes and E is the set of edges representing optical fibers that connect the nodes. Graph $G(V, E)$ contains $|V|$ nodes and $|E|$ edges. Each edge carries two oppositely directed fibers for data transmission in the two directions of the edge. Each directed fiber is called a link. Every node $v_t \in V$ has multicast capability by equipping an optical splitter (Ibrahim et al., 2007). We assume an optical signal can be split into an arbitrary number of optical signals at a splitter. Thus, there is no node degree restriction on the routing tree. Otherwise, the multicast problem turns to be a degree-constrained Steiner (Wang and Jan, 2007) problem.

Since the all-optical wavelength converter is still in its early development stage and the optoelectronic conversion not only is very expensive but also has limited performance, we assume only partial nodes are equipped with full-range wavelength converter (Wong et al., 2003) in the network. The full-range wavelength converter is able to convert optical signal on a wavelength into any other wavelength. The wavelength conversion also introduces additional processing and control delay called wavelength conversion delay. Without loss of generality, we assume the conversion between any two different wavelengths has the same delay at any optical node with the wavelength converter, i.e., $t(v_t) \equiv t$. If there is no wavelength conversion at an intermediate node v_i, we set $t(v_i)=0$.

Each link $e_{ij}=(v_i,v_j) \in E$ is associated with three parameters:

- $\Lambda(e_{ij})$, the set of available wavelengths. $\Lambda(e_{ij}) \subseteq \Lambda=\{\lambda_1,\lambda_2,\ldots\lambda_w\}$, Λ is the set of wavelengths supported by each link in the network.
- $\delta(e_{ij})$, the transmission delay. Here, $\delta(e_{ij})=\delta(e_{ij})$.
- $c(e_{ij})$, the link cost.

3.2. Mathematical model

In graph $G(V,E)$, we consider a request for multicast connection setup, $R(s,D,\Delta)$, where s denotes the source node, D represents the set of destinations. Different from previous algorithms (Chen and Wang, 2002; Chen and Tseng, 2005; Jia et al., 2001; Jia et al., 2001), we define Δ as the delay requirement interval specified by the user. It is more practical to represent the delay requirement by an interval rather than a single value because in practice the network information is inaccurate and the user QoS requirement is often flexible (Bouillard et al., 2008). The lower and the upper bound of the delay interval are determined by the user and the application.

The route of the multicast connection is represented by a tree $T=(X_T, F_T)$, $X_T \subseteq V$, $F_T \subseteq E$. The total cost of T is defined as

$$Cost(T) = \sum_{e_{ij} \in F_T} c(e_{ij}). \tag{1}$$

The communication delay on a path consists of two components, i.e., link transmission delay and wavelength conversion delay. Let $P(s, d_i)$ denote the path from source node s to any destination node d_i in T and let D_{sd_i} denote the path delay. We have

$$D_{sd_i} = [\sum_{v_i \in P(s,d_i)} t(v_i) + \sum_{e_{ij} \in P(s,d_i)} \delta(e_{ij})]. \tag{2}$$

The delay of T is defined as

$$Delay(T) = \max\{D_{sd_i}, \forall d_i \in D\}, \tag{3}$$

which is the maximum delay between the source node and all the destination nodes. We set $\Delta=[\Delta_{low}, \Delta_{high}]$ and then the user QoS satisfaction degree is defined as

$$Degree(QoS) = \begin{cases} 100\% & Delay(T) \le \Delta_{low} \\ \dfrac{\Delta_{high} - Delay(T)}{\Delta_{high} - \Delta_{low}} & \Delta_{low} < Delay(T) < \Delta_{high} \\ 0\% & Delay(T) \ge \Delta_{high} \end{cases}. \tag{4}$$

The algorithm should select the links with more available wavelengths to balance the network load and thereby reduce the call blocking probability. The load on a link is defined as the number of channels over that link. We can adjust it by defining proper link cost functions. For example, by defining heuristic cost functions, for the link with more available wavelengths, the cost takes smaller value. In the proposed algorithm, we define

$$c(e_{ij}) = w - |\Lambda(e_{ij})|. \tag{5}$$

The key optimization objective considered in this paper is to minimize the tree cost whilst the user QoS satisfaction degree is still high. Hence, we solve the problem of QoS multicast in the optical network by finding an optimal multicast tree $T^*(X_{T^*}, F_{T^*})$, $\{s\} \cup D \subseteq X_{T^*}$, $F_{T^*} \subseteq E$, satisfying

$$Cost(T^*) = \min_T\{Cost(T)\}, \tag{6}$$

where T denotes any multicast tree spanning s and D in $G(V, E)$. In addition, the end-to-end delay of tree T^* should not exceed the upper bound of the delay interval. Otherwise the user cannot accept it due to poor QoS performance. So we have

$$Delay(T^*) \le \Delta_{high}. \tag{7}$$

Furthermore, for any link on tree T^*, there should exist at least one available wavelength. Otherwise, the multicast connection cannot be set up. So we have

$$\forall e_{ij} \in F_{T^*}, \quad |\Lambda(e_{ij})| \ge 1. \tag{8}$$

4. Preliminaries

4.1. Expression of the solution

We denote the solution by binary coding. Each bit of the binary string corresponds to a different network node. For solution S, the length of S equals the number of network nodes, i.e., $|V|$. The graph corresponding to S is $G'(V', E')$. Let the function $bit(S, i)$ denote the ith bit of solution S. If $bit(S, k)=1$, $v_k \in V'$. If $bit(S, k)=0$, $v_k \notin V'$. If $v_m \in V'$, $v_n \in V'$, and $(v_m, v_n)=e_{nm} \in E$, $e_{nm} \in E'$. For our problem, every solution S corresponds to tree $T_i'(X_i', F_i')$, which is the minimum cost spanning tree of G'. T_i' spans the given nodes set U, $U=\{s\} \cup D$.

When the minimum cost spanning tree serves as the multicast tree, if there exist leaf nodes which do not belong to U, we prune the tree by deleting these nodes and their adjacent links. We then denote the pruned tree as $T_i(X_i, F_i)$. Another problem is that the graph corresponding to S may be unconnected. Since every subgraph of graph G' has a minimum cost spanning tree, the

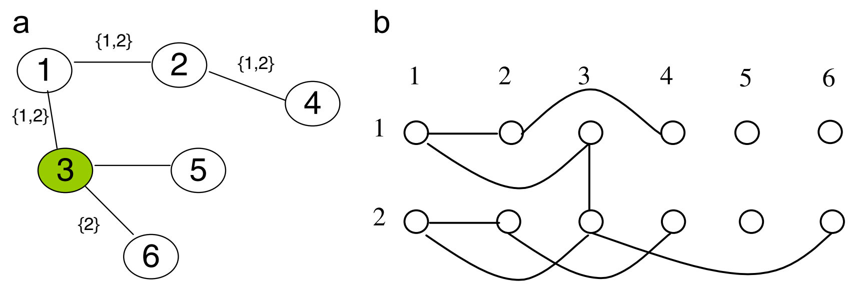

Fig. 1. The illustration of the construction of a wavelength graph: (a) physical network topology, (b) the corresponding wavelength graph.

solution S corresponds to a minimum cost spanning forest, which is also denoted by $T_i'(X_i', F_i')$. Similarly, we prune every tree in the forest and denote the pruned forest as $T_i(X_i, F_i)$.

If S corresponds to an unconnected graph, it represents an unfeasible solution. We can make such a solution feasible by both adding penalty and taking smaller value for the user QoS satisfaction degree. Thus, every solution S corresponds to a graph G', which corresponds to a minimum cost spanning forest T_i' (a forest can have only one tree). After pruning T_i', the forest T_i corresponding to S is obtained.

4.2. The algorithm for wavelength assignment

If T_i is a tree, we assign wavelengths to it. The objective of the proposed wavelength assignment algorithm is to minimize the delay of the tree by minimizing the number of wavelength conversion. Thereby the user can get a high QoS satisfaction degree. The proposed algorithm is based on the ideas of wavelength graph (Guo et al., 2009; Lee et al., 2003). First we construct the wavelength graph WG for the tree $T_i(X_i, F_i)$ by the following method.

(1) $N=|X_i|$, $w=|\cup_{e_{ij}\in F_i}\Lambda(e_{ij})|$. In WG, we create N^*w number of nodes, namely v_{ij}, for $i=1,2,\ldots,w$ and $j=1,2,\ldots,N$. All the nodes are arranged into a matrix with w rows and N columns. Row i represents the corresponding wavelength λ_i' and each column j represents a node v_j' in T_i. A mapping table is created to record the corresponding relationship between i and λ_i', and another one is created to record the relationship between j and v_j'. The two tables will help reversely map the paths in WG back to the paths and wavelengths in T_i.

(2) For $i=1,2,\ldots,w$, in the ith row, we add a horizontal directional link (v_{ij}, v_{ih}) between column j and column h if there exists a link $e_{jh}'=(v_j', v_h')$ in T_i from node v_j' to node v_h' and the wavelength λ_i' is available on this link. We assign the transmission delay $\delta(e_{jh}')$ as its weight.

(3) For $j=1,2,\ldots,N$, in the jth column, for $\forall i_1, i_2, i_1\neq i_2$, we add a vertical bidirectional link $(v_{i_1 j}, v_{i_2 j})$ between row i_1 and row i_2 if node v_j' in T_i has the wavelength conversion capability. We assign the wavelength conversion delay t as its weight.

Using the above steps the wavelength graph WG is constructed. A vertical link in WG represents a wavelength conversion at a node and a horizontal link in WG represents an actual link in T_i. For convenience, we denote the nodes in WG by sequential node number $1\sim N^*w$. The sequential node number for the node in the ith row and jth column in WG is

$$x=(i-1)*N+j. \tag{9}$$

Fig. 1 illustrates an example of constructing the wavelength graph. Fig. 1(a) is the physical network topology G where nodes 1–6 represent the optical nodes. In the bracket near a link, 1 and/or 2 represent that wavelength 1 and/or wavelength 2 are available on that link. Node 3 is an optical node with wavelength conversion capability. Fig. 1(b) is the generated wavelength graph corresponding to the physical network topology.

We treat the wavelength graph WG as an ordinary network topology graph and run the wavelength assignment algorithm. $P(x, y_k)$ is the shortest path from source node s to destination node d_k in WG. We have:

$$i=(x-1)/N+1, \tag{10}$$

$$j=(x-1)\%N+1. \tag{11}$$

Using the above two expressions and the two mapping tables created in step (1), we can reversely map the paths consisted of the sequential node numbers back to the links and wavelengths in T_i conveniently. Thus the wavelength assignment is completed.

However, multiple wavelengths on a link may transmit the same message on the multicast tree, as is illustrated by the input link (0, 3) in Fig. 2. This results in the waste of the scarce wavelength resource. We adopt the following method to deal with it. First, among all the downstream destinations of the input link, we select the one which has the maximum end-to-end delay. Then to the input link, we assign the wavelength, which is used by the output link leading to that destination. Thus, the extra wavelength conversion

Wavelength assignment algorithm

Input: the wavelength graph WG where the source node and all the destination nodes correspond to the column numbers in the matrix, i.e.,

$j_s, j_{d_1}, j_{d_2}\cdots, j_{d_m}$

.

Output: the wavelength assignment result for tree T_i.

begin
for (k=1, $k\leq m$, k+ +)
{
  for (i=1, $i\leq w$, i+ +)
  {
    $x_i=(i-1)*N+j_s$;
    for (j=1, $j\leq w$, j+ +)
    {
      $y_{jk}=(j-1)^*N+j_{d_k}$
      ;
      Apply the Dijkstra's shortest path algorithm to find the shortest path $P(x_i, y_{jk})$ from node x_i to node y_{jk};
    }
    $P(x_i, y_k)=\min\{P(x_i, y_{jk}), 1\leq j\leq w\}$;
  }
  $P(x, y_k)=\min P(x_i, y_k), 1\leq i\leq w\}$;
}
end

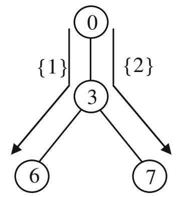

Fig. 2. Illustration of wavelength resource waste.

delay is added to the path with the least end-to-end delay. Therefore, its effect on the delay of the routing tree is compromised. In Fig. 2, assuming that the end-to-end delay to destination 7 is larger than destination 6, wavelength 2 is assigned to link (0, 3) and a wavelength conversion is necessary between link (0, 3) and link (3, 6).

The time complexity of the above algorithm is $O(mN^2w^4)$, where m is the number of destination nodes, N is the number of nodes in T_i, w is the number of wavelengths which are available on at least one link on T_i. We can see that they all take small integer values. In addition, as a type of pre-assigning strategy for getting the end-to-end delay of the path, all the wavelength assignments for solutions except the final solution will not be used as the final wavelength assignment result. Hence, the algorithm need not store lots of data and has a low space complexity.

4.3. Fitness function

The fitness of solution S is obtained by computing fitness function f: $S' \rightarrow R^+$, where S' is the set of solutions. f is determined by $Cost(T_i)$ and the user QoS satisfaction degree. After assigning wavelengths to T_i, the delay of T_i is determined and thereby $Degree(QoS)$ is determined. In addition, when the forest corresponding to the solution has more than one tree, the solution is unfeasible. We solve this problem by adding penalty value to the fitness of solutions and taking smaller value of $Degree(QoS)$. The higher the user QoS satisfaction degree, the smaller the fitness value. This is because the objective of our algorithm is to find low cost multicast tree, which also satisfies the user QoS requirement. The fitness function is defined as follows.

$$f(S) = \frac{Cost(T_i) + [count(T_i) - 1] * \rho}{Degree(QoS)} = \frac{\sum_{e_{ij} \in F_i} c(e_{ij}) + [count(T_i) - 1] * \rho}{Degree(QoS)}, \tag{12}$$

where $count(T_i)$ is the number of trees in the forest T_i, ρ is a constant, $\rho > 0$. For every solution, the smaller the fitness value computed by f, the better the solution.

5. Design of the ACO-based QoS multicast algorithm

Ant colony optimization (Dorigo and Stützle, 2004) simulates the ants' behavior of searching paths and it has shown great advantage in solving complex optimization problems. It has popular applications in many combinatorial optimization problems such as routing and job scheduling. ACO is still a random search algorithm. Similar to other bio-inspired techniques, it tries to find the optimal solution through the evolutionary process of the swarm composed of candidate solutions. The process consists of two basic phases, i.e., adaptation phase and cooperation phase. During the adaptation phase, each candidate solution adjusts its structure according to accumulated information. During the cooperation phase, candidate solutions exchange information to produce better solutions.

5.1. ACO-based model

In our problem, a multicast tree is also a Steiner tree (Wang and Jan, 2007). To construct it, ACO is used to select Steiner nodes. For each node $n \in V$, we define $pheromone_n(t)$ as the amount of pheromone. The ant will decide if node n is selected as a Steiner node according to the amount of pheromone remained on n. We define $Probability_n$ as the probability of selecting n as a Steiner node. We have

$$Probability_n^k = \frac{pheromone_n^{\alpha}(t)}{\sum_{s \in allowed_k} pheromone_s^{\alpha}(t),} \tag{13}$$

where $allowed_k$ represents the set of nodes which can be selected by the k-th ant in the next step, and a is set to 1. With the time passing away, a fraction of the pheromone evaporates. We then define $RemainRate$ as the pheromone remaining parameter, which represents the ratio of the pheromone remained on nodes.

We denote the size of ant colony as $AntNum$. When one iteration is completed by the ant colony, the amount of pheromone on the corresponding paths needs to be adjusted according to the following expression.

$$\begin{aligned} &pheromone_n = RemainRate \\ &\quad * pheromone_n + \Delta pheromone_n \Big(RemainRate \in (0,1), \\ &\Delta pheromone_n = \sum_{k=1}^{AntNum} \Delta pheromone_n^k \Big). \end{aligned} \tag{14}$$

where $\Delta pheromone_n^k$ represents the amount of pheromone that the k-th ant remained on the path during the iteration, and $\Delta pheromone_n$ represents the increment of pheromone on node n after this iteration.

Three different models have been proposed in Dorigo and Stützle (2004) i.e., ant cycle system (ACS), ant quantity system (AQS), and ant density system (ADS). Different model has different $\Delta pheromone_n$ In ACS model,

$$\Delta pheromone_n = \begin{cases} \sum \frac{\beta}{fitness_k} & \text{if the } k\text{-th ant uses } n \text{ in its route} \\ 0 & \text{otherwise} \end{cases}. \tag{15}$$

In both AQS and ADS, the local information is used. In ACS, the global information is used. Since ACS has better performance when it is applied to the QoS multicast problem, it is adopted as the basic model in our algorithm. The optimal combination of the parameters in each formula can be determined by simulation experiments. When the algorithm reaches the maximum iteration number or the continuous iterations cannot improve the current optimal solution, it should terminate. In the proposed algorithm, the maximum iteration number is employed as the termination rule.

5.2. Improvements over ACO

Since ACO may converge at the local optimum or at a very slow speed, to enhance its global search capability and improve its search speed, we make the following two improvements.

(a) Record the optimal solution. The current optimal solution is recorded after each iteration.
(b) Change the value of *RemainRate* adaptively. First, the value of *RemainRate* is set to 1. When the current optimal solution cannot be improved after t iterations, the value of *RemainRate* is decreased according to the following rule:

$$RemainRate = \begin{cases} 0.95 * RemainRate_{t-1} & if \;\; 0.95 * RemainRate_{t-1} \geq RemainRate_{\min} \\ RemainRate_{\min} & otherwise \end{cases}, \tag{16}$$

where $RemainRate_{min}$ is the minimum value allowed by *RemainRate*. The reason is explained below.

When the problem size is relatively large, the amount of pheromone on the unsearched solutions will approach 0 due to the evaporation. Therefore, the global search capability of the algorithm will be weakened. When the value of *RemainRate* is relatively high, with the increase of the pheromone on the solutions, the difference between the probabilities of selecting the searched solutions and unsearched solutions is not significant. Thus, the convergence speed of the algorithm is decreased. However, if we reduce the value of *RemainRate*, both the convergence speed and the global search capability are increased. The algorithm mainly relies on random search which makes ACO lose its own advantage. Our strategy of adaptive change can solve this problem well.

The path traversed by the ant colony always follows the one with the strongest pheromone. The research on biological ant colony and ACO has demonstrated that the path with the strongest pheromone normally represents the expected optimal path for both the biological and artificial ant colonies. However, it may not be always true. In artificial ant colony, the violation of the rule is more severe than biological ant colony. Therefore, effective measures should be taken to prevent it. In the proposed algorithm, the following two methods are adopted.

(a) Ant Mutation. Since ACO is still a random optimization algorithm, mutation operation in genetic algorithm (GA) (Yang, 2008) is introduced into it to improve the performance. The ant mutation is to mutate the optimal solution at each iteration, i.e., taking the reverse value of some bits. For example, before mutation, ant[0]=1100101, after mutation, ant[0]=1010101.
(b) Initiation of pheromone. In basic ACO, pheromone is initialized as a fixed value. In the propose algorithm, the amount of pheromone on different nodes are initialized as different values. This setting can make ACO converge to the global optimal solution faster. The amount of pheromone assigned on each node considers its degree.

5.3. Description of the ACO-based QoS multicast algorithm

In IP/DWDM optical Internet, the ACO-based QoS multicast algorithm is described as follows.

ACO-based QoS multicast algorithm

Initialization: set the size of ant colony *AntNum* and the maximum iteration number *NCNum*. Set the ant number counter k=0, and the iteration number counter NC=0. *BestAnt* represents the optimal ant. F represents the fitness of *BestAnt* and is set to be infinity initially. Set the maximum number of iterations through which the optimal ant remains unchanged to C. Initialize the amount of pheromone on different nodes as different values.

(1) Generate the initial ant colony randomly.
(2) If $NC < NCNum$, go to step (3); otherwise, the algorithm terminates, go to step (8).
(3) If k=*AntNum*, then k=0; otherwise, the following operation is conducted for the k-th ant:
   (a) Search Steiner node: determine $allowed_k$, the set of nodes which can be selected by the ant. For each node in $allowed_k$, calculate its selection probability, then determine which nodes to be selected according to the probability. In the binary solutions, the bits corresponding to selected nodes are set to 1;
   (b) Wavelengths are assigned to the solution corresponding to the ant using the proposed wavelength assignment algorithm. Then calculate *Degree(QoS)* of the solution. Calculate $f(k)$, the fitness of the ant;
   (c) Update the amount of pheromone on each node according to formulas (13) and (14);
   (d) k=k+1. Go to step (3).
(4) $f(h)=\min\{f(k)|0 \leq k < AntNum\}$, i.e., the h-th ant is the current optimal one in the ant colony. If $f(h) < F$, then update *BestAnt*, i.e., *BestAnt* is replaced by the h-th ant.
(5) If C is reached, then change *RemainRate* adaptively according to formula (16).
(6) Mutate and update ants.
(7) NC=NC+1. Go to step (2).
(8) Use the proposed wavelength assignment algorithm to assign wavelengths to *BestAnt* and record the assignment result. Then the final multicast tree is obtained.

6. Design of the AIA-based QoS multicast algorithm

In life sciences, the natural phenomena such as heredity and immunity have been investigated intensively. Genetic algorithm (Yang, 2008) was proposed by Holland in the middle of 1980s. However, the practice shows that it is far away from simulating the human ability of processing things intelligently. The human intelligence resources could be exploited more deeply. As a result, the concept of immunity was introduced into the engineering optimization area (Castro and Timmis, 2002). Relevant immune knowledge and theory is combined with some existing intelligent algorithms to create new evolutionary theory and algorithms. Intuitively, the overall performance of the new algorithms is improved.

Artificial immune algorithm is developed by adding immune concept and theory into GA. It is feasible and effective. The degradation, which appears in the optimization process of GA, is inhibited by some characteristic information or knowledge derived from problems to be solved. Theoretical analysis and simulation results on some combinatorial optimization problems show that AIA successfully relieves the degradation in GA.

6.1. AIA-based model

Basically, AIA can be regarded as an improvement over GA. Based on the reasonable extraction of vaccine, the immune idea is implemented by two steps, i.e., inoculating vaccine and immunity selection. The inoculating vaccine is to improve the fitness and the immunity selection is to avoid the degradation of the population. The basic procedure of AIA is described as follows.

Basic procedure of AIA

(1) Generate the initial father population A_1 randomly.
(2) Extract vaccine according to the prior knowledge.
(3) If the optimal individual is in the current population, the algorithm terminates and outputs the result; otherwise, the algorithm continues.
(4) Perform the crossover operation over the current *k*th generation A_k, and then get the population B_k.
(5) Perform the mutation operation over B_k, and then get the population C_k.
(6) Perform the operation of inoculating vaccine for C_k, and then get the population D_k.
(7) Perform the immunity selection for D_k, and then get the new generation father population A_{k+1}.

6.2. Vaccine and inoculation

Vaccine, which is used for inoculation, is the most important element and elite of AIA. To select the appropriate vaccine is a critical step for speeding up the searching of the optimal solution. It is very creative and requires skills to select the vaccine. To design the vaccine for a specific problem, the problem itself should be analyzed. More importantly, the characteristic information should be collected because the vaccine has to be made according to the characteristic information.

The objective of the proposed algorithm is to find a multicast tree with low network cost and high user QoS satisfaction degree. The tree should cover all the multicast nodes. All these information inspires us that the optimal adjacent link should be found for each multicast node. The other end node of the optimal adjacent link is named as the optimal adjacent node. In the binary coding solution, we set the bits corresponding to optimal adjacent nodes as 1 and the other bits as 0. Thus, the obtained binary coding solution can be used as the vaccine. To sum up, the idea of making vaccine is that the multicast tree should include the optimal adjacent node of each multicast node.

The optimal adjacent node is defined as follows. We assume that *A* is a multicast node having three adjacent nodes *B*, *C* and *D*. Therefore, *A* has three adjacent links l_1=(*A*,*B*), l_2=(*A*,*C*), l_3=(*A*,*D*). The cost and delay of the link e_{ij} is $c(e_{ij})$ and $\delta(e_{ij})$, respectively. Calculate the value of $c(e_{ij})^*\delta(e_{ij})$. The smaller the value, the better the link. The example is shown in Fig. 3. Since $c(l_1)$=4, $c(l_2)$=5, $c(l_3)$=3, $\delta(l_1)$=4, $\delta(l_2)$=3, $\delta(l_3)$=5, we get $c(l_2)^*\delta(l_2)$= $c(l_3)^*\delta(l_3)$*lt;$c(l_1)^*\delta(l_1)$. Both l_2 and l_3 are the optimal adjacent links. As a result, *C* and *D* are the optimal adjacent nodes. Assume that *D* is selected randomly. In binary coding solution, the bit corresponding to *D* is set to 1. When all the multicast nodes have been processed like this, a binary coding solution is obtained. This solution is just the vaccine that can be used in the proposed algorithm. The process of extracting vaccine is described as follows.

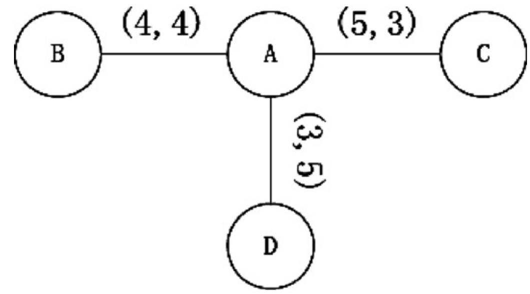

Fig. 3. Example of making vaccine.

Process of extracting vaccine

Initialization: define *immune* as a binary coding solution with all the bits set to 0. Set *temp* as a real number big enough.

(1) If all the multicast nodes have been processed, go to step (5); otherwise, begin to deal with the first link marked as *l* connecting the current multicast node.
(2) If link *l* is null, go to step (4); otherwise, calculate $c(l)^*\delta(l)$. If $temp \le c(l)^*\delta(l)$, go to step (3); otherwise, $temp \le c(l)^*\delta(l)$, and mark the other end node of *l* as *k*.
(3) Deal with the next link and mark it as *l*, go to step (2).
(4) The bit corresponding to *k* is set to 1 in *immune*.
(5) The process terminates.

We also design a simple yet effective process of inoculating vaccine. First, by contrasting the vaccine and the solutions obtained by GA, we find all the bits which correspond to 0 in the obtained solution and correspond to 1 in the vaccine simultaneously. Then set them to 1. Each node corresponding to the bit with the value of 1 is the optimal node of the adjacent multicast node in the vaccine. The joining of these optimal nodes helps select the optimal links for the multicast tree. Therefore, the multicast tree transmits information through links with low cost or delay. The beneficial information kept in the vaccine can be inoculated into the solutions through this process.

6.3. Annealing mechanism

In Section 6.1, it is possible that the offspring individual does not replace the father individual although the fitness of one individual in offspring population is better than its father individual in the father population. To deal with it, annealing operation is implemented to accept the offspring individual in a certain probability. The objective is to encourage some individuals with local good characteristics enter the next iteration, thereby avoiding the local optimum.

The annealing function in the algorithm is as follows.

$$T_N = \ln((T_0/N)+1), \quad T_0 = 100, \tag{17}$$

where *N* is the number of evolutionary generation in GA.

6.4. Description of the AIA-based QoS multicast algorithm

The AIA-based QoS multicast algorithm is described as follows.

AIA-based QoS Multicast Algorithm

Initialization: Population size *P*, generation number *N*, crossover probability *pc*, mutation probability *pm*, counter *i=0*.

1) Generate the initial father population *A* randomly. Assign wavelengths to the solutions in *A* by the proposed wavelength assignment algorithm. Then calculate *Degree(QoS)* of each solution. Thus the fitness value of each chromosome is determined.
2) Extract the vaccine by the rules in Section 6.2.
3) $i=i+1$, if $i \leq N$, generate the roulette wheel, go to step 4); otherwise, go to step 9).
4) Generate the gene pool by selecting chromosomes from *A* according to the roulette wheel. The number of chromosomes in gene pool is the population size *P*.
5) Perform the crossover operation over the chromosomes in the gene pool according to the crossover probability *pc*. Reproduce the chromosomes which do not execute the crossover operation. Thus population *B* is obtained. Perform the mutation operation for *B* according to the mutation probability *pm*. Then population *C* is obtained.
6) Inoculate the vaccine for the chromosomes in *C* according to the methods described in Section 6.2. Then the population *D* is obtained.
7) Perform the following immunity selection operation for the chromosomes in *D*, then the offspring generation population *E* is obtained:
   a) Immunity detection: the chromosome whose fitness increases after inoculation is replaced by the counterpart in *A*;
   b) Annealing selection: the chromosome whose fitness decreases after inoculation conducts the annealing selection according to the annealing mechanism in Section 6.3.
8) Calculate the fitness of each chromosome in the offspring population *E*, and replace the father population *A*, go to step 3).
9) Calculate the fitness of each chromosome in population *A*. The chromosome with the least fitness is just the final solution. Assign wavelengths to the solution by the proposed wavelength assignment algorithm. Then the final multicast tree is obtained.

7. Performance evaluation

To implement and evaluate the proposed algorithms, we have developed a simulator in C+ +. The simulation software can provide two options for establishing the network topologies, i.e., random generation and manual generation. The random generation method can produce an arbitrary connected topology under the given number of nodes, number of links and maximum node degree. The manual generation method can let the user draw a specific network topology and specify the parameters for both nodes and links. A snapshot of the simulator is shown in Fig. 4.

Through simulation experiments, we evaluate the performance of both ACO-based and AIA-based multicast algorithms on a real network topology CERNET (Zhao et al., 2002). The network topology consists of 23 nodes and 24 optical links. Because the optimization objective of the proposed algorithms is to minimize the tree cost whilst the user QoS satisfaction degree is still high, there is a tradeoff between the tree cost and delay.

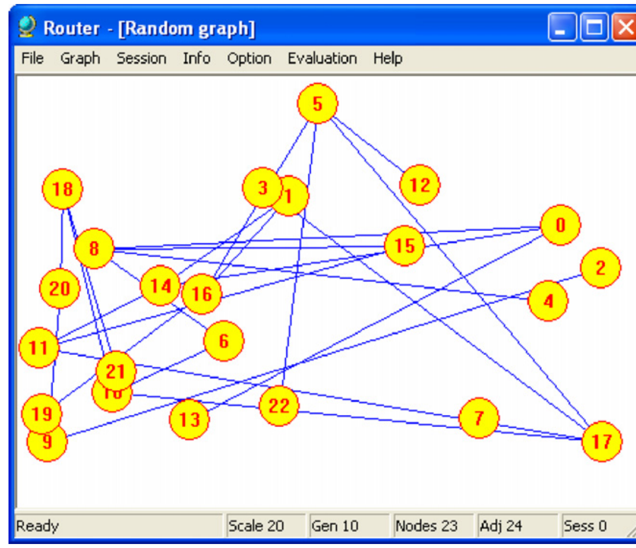

Fig. 4. The snapshot of the simulator.

Therefore, we evaluate the algorithms in two aspects, i.e., the cost and the delay of the final multicast trees. Since GAs have been widely used to solve the QoS multicast problem in optical network (Chen and Tseng, 2005; Din, 2008), the proposed two algorithms are compared with it to show the performance improvements.

In the experiments, the population size was set to 20, the generation number was set to 10, the crossover probability was set to 80%, the mutation probability was set to 5%, and the ratio of multicast nodes to the total network nodes was set to 30%. Referring to the simulation model established in Jia et al. (2001), we set the transmission delay on each link to be a small integer between 1 and 10, which is also in direct proportion to the length of the link. The wavelength conversion delay was set to be a constant integer between 1 and 10. We choose the 50% of all the nodes which have higher degrees to be equipped with wavelength converters. $|\Lambda|=20$ and $10 \leq |\Lambda(e_{ij})| \leq 15$.

If the fitness values of some chromosomes are too large, the difference between other chromosomes will be shielded leading to lower diversity. To avoid this, when *Degree*(*QoS*) is less than a small value *val*, *Degree*(*QoS*) is set to *val* in the fitness calculation. If the solution corresponding to the chromosome is unfeasible, *Degree*(*QoS*) is also set to *val*. By running extensive simulation experiments, the appropriate values for the other parameters of ACO- and AIA-based multicast algorithms are also determined.

In Section 4.2, we have stated that "the objective of the proposed wavelength assignment algorithm is to minimize the delay of the tree by minimizing the number of wavelength conversion". Therefore, in each wavelength assignment, the algorithms have tried to minimize the number of wavelength conversion which will result in the minimization of delay. Through evaluating the delay metric, we evaluate the performance of the algorithms in minimizing the number of wavelength conversions.

7.1. The evaluation of ACO-based algorithm

In the following experiments, the number of ants is equal to the number of nodes in the network, the maximum iteration number is set to 25, *RemainRate* is initialized to 1, and β is set to 200. The maximum number of wavelengths supported by each link is 20 and the number of available wavelengths falls into between 10 and 15. There are 11 different multicast sessions.

These multicast sessions represent multicast groups with various sizes.

7.1.1. The evaluation on the tree cost

By comparing solutions obtained by the ACO-based multicast algorithm with the solution obtained by the GA-based multicast algorithm, we have made quantitative analysis on the tree cost. The results are shown in Table 1.

From Table 1, we can see that the cost of ACO multicast trees is lower than GA multicast trees in most of the cases. It is well known that GA is a widely used and successful optimization algorithm. Hence, as a new random optimization algorithm, the ACO-based multicast algorithm shows very good performance in terms of the tree cost.

7.1.2. The evaluation on the delay

In this paper, we define a new concept, i.e., the user QoS satisfaction degree. The QoS performance of each solution is considered when its fitness is calculated. Therefore, we select the solutions by considering both the tree cost and the maximum end-to-end delay. The use of the user QoS satisfaction degree helps to make an ideal tradeoff between the cost and the delay of the multicast trees.

To evaluate the performance improvement made by using the user QoS satisfaction degree, we also run ACO-based multicast algorithm under the scenario that QoS (i.e., the user QoS satisfaction degree) is not considered. Then we compare the delay of the multicast trees obtained by the algorithms considering QoS and without considering QoS. The results are shown in Fig. 5. From Fig. 5, it shows that the delay of the multicast trees obtained by the algorithm considering QoS is less than the one without considering QoS. It proves that with the use of the user QoS satisfaction degree, the multicast trees with better QoS performance are obtained.

Table 1
The cost comparison results between the final solutions obtained by the ACO-based multicast algorithm and the GA-based multicast algorithm.

Multicast session no.	The ratio of multicast session nodes in the network (%)	The cost of multicast tree	
		ACO	GA
1	13.89	18	18
2	22.22	25	25
3	27.78	36	38
4	36.11	41	40
5	41.67	48	50
6	50.00	50	53
7	58.33	56	55
8	63.89	56	58
9	72.22	63	62
10	77.78	70	72
11	86.11	82	83

7.2. The evaluation of AIA-based algorithm

In the following experiments, we first evaluate the probability of inoculating vaccine. Then we evaluate the performance regarding the tree cost and QoS. There are seven different multicast sessions. These multicast sessions represent multicast groups with various sizes.

7.2.1. The evaluation on the probability of inoculating vaccine

As an important parameter, the probability of inoculating vaccine is tested to determine its optimal value, which can guarantee the algorithm efficiency and optimize the results. The performance of the AIA-based multicast algorithm is compared with the GA-based multicast algorithm since AIA is regarded as an improvement over GA. Ten multicast

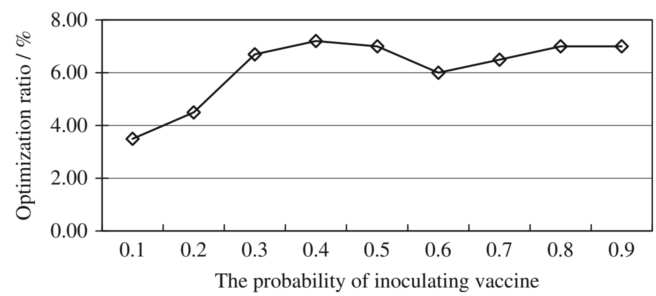

Fig. 6. The effect of the probability of inoculating vaccine on the algorithm performance.

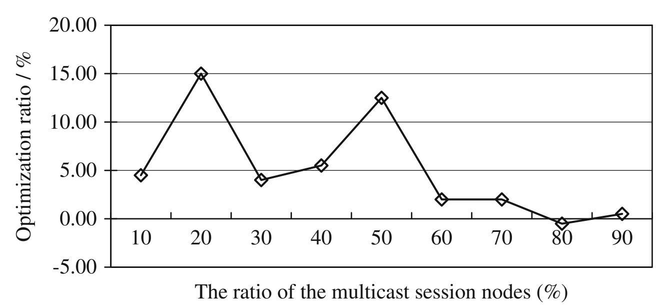

Fig. 7. The effect of the multicast session nodes ratio on the algorithm performance.

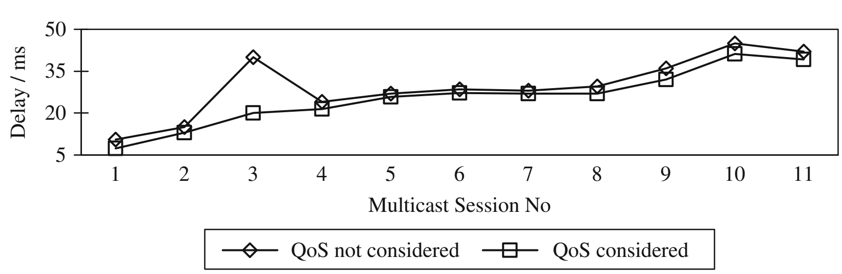

Fig. 5. The delay comparison results between the ACO multicast algorithm considering QoS and without considering QoS.

groups are randomly selected and the value of *VaccinationRate* is tested in 10% incremental interval. Fig. 6 is plotted to show the effect of the probability of inoculating vaccine on the algorithm performance.

In Fig. 6, the *X*-coordinate represents the probability of inoculating vaccine for the solutions, and the *Y*-coordinate represents the decreasing ratio of the fitness obtained by the AIA-based multicast algorithm to the fitness obtained by the GA-based multicast algorithm. The decreasing ratio can also be regarded as the optimization ratio of AIA. Fig. 6 shows that the fitness of AIA is optimized step by step with the increase in the probability of inoculating vaccine. When the probability falls into the interval between 0.3 and 0.4, the algorithm achieves the best performance. After 0.4, the increase of the probability reduces the algorithm efficiency without improving the performance.

7.2.2. *The effect of the multicast session nodes ratio*

Since different multicast sessions have different sizes, we have investigated the effect of the session nodes ratio on the performance of the AIA-based algorithm. The genetic parameters are kept unchanged. *VaccinationRate*=30%. The ratio of the number of session nodes to the total number of the network nodes varies from 10% to 90% in the incremental interval 10%. Each time 10 multicast sessions are tested and the average values are recorded. Then we calculate how much improvement the AIA-based algorithm can achieve compared to the GA-based algorithm in terms of the fitness values. The results are shown in Fig. 7.

From Fig. 7, the *X*-coordinate represents the ratio of the number of session nodes to the total number of the network nodes, and the *Y*-coordinate represents the decreasing ratio of the fitness obtained by the AIA-based multicast algorithm to the fitness obtained by the GA-based multicast algorithm. Similar as in Section 7.2.1, the decreasing ratio can also be regarded as the optimization ratio of AIA. Fig. 7 shows that with the increase of the ratio, the performance improvements of the AIA-based algorithm over the GA-based algorithm become less significant. When the ratio reaches 70%, both of the algorithms show similar performance.

Table 2
The cost comparison results between the final solutions obtained by AIA-based multicast algorithm and the GA-based multicast algorithm.

Multicast session no.	The ratio of multicast session nodes in the network (%)	The cost of multicast tree	
		AIA	GA
1	10.81	52.3	54.4
2	18.92	98.7	99.8
3	27.78	121.9	124.2
4	37.84	175.5	173.9
5	48.64	212.6	213.3
6	59.46	240.5	243.5
7	72.97	286.2	290.3

7.2.3. *The evaluation on the tree cost and delay*

Different from the ACO-based multicast algorithm, the number of available wavelengths on each link falls into the interval between 3 and 17 in the following experiments. The AIA-based multicast algorithm is also compared with the GA-based multicast algorithm in terms of the tree cost and delay. The parameters of GA are fixed and the probability of inoculating vaccine is set to 60%. Seven multicast groups are randomly generated with different ratios of multicast session nodes to the total network nodes. Under each ratio, the average cost and delay of the AIA multicast trees are compared with the GA multicast trees. The results are shown in Table 2 and Fig. 8, respectively.

From Table 2, we can see that the cost of the AIA multicast trees is lower than the GA multicast trees in most of the cases. From Fig. 8, with the increase in the number of multicast session nodes, the results obtained by the AIA-based multicast algorithm gradually approach the results obtained by the GA-based multicast algorithm. The reason is that the number of the paths which can be selected by the algorithms decreases with the increase in the number of the multicast session nodes. Thus the space for further optimization is relatively small. In addition, one AIA multicast result is worse than GA. This is due to the random characteristics of the algorithms since both of them are random search algorithms. The AIA-based multicast algorithm is better than the GA-based multicast algorithm especially when large groups are tested.

8. Conclusions

Driven by the requirements of supporting QoS group communications in IP/DWDM optical Internet, efficient integrated multicast routing and wavelength assignment algorithms are investigated in this paper. Since bio-inspired algorithms are powerful tools for solving the combinatorial optimization problems, both ant colony optimization and artificial immune algorithm are applied to search the optimal multicast tree which has the lowest cost and produces the least wavelength conversion delay. Based on the idea of wavelength graph, a dedicated wavelength assignment algorithm is proposed to minimize the number of wavelength conversion. The wavelength pre-assigning is integrated into the search of the routing trees to make a reasonable tradeoff between the tree cost and the end-to-end delay. Simulation results show that the proposed two bio-inspired QoS multicast algorithms show better performance than the popular GA-based algorithm in terms of both the tree cost and delay. However, our model and work can still be improved in the future, For example, to be more practical, we should allow any

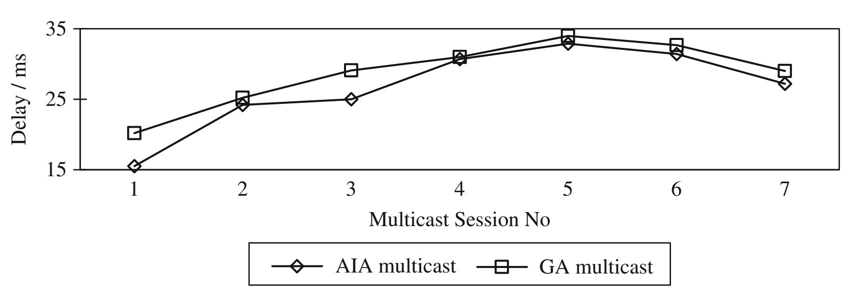

Fig. 8. The delay comparison results between the AIA-based multicast algorithm and the GA-based multicast algorithm.

ratio of nodes equipped with wavelength converters and allow the wavelength conversion delay to be different.

Acknowledgment

This work was supported in part by the Program for New Century Excellent Talents in University, the Engineering and Physical Sciences Research Council (EPSRC) of UK under Grant EP/E060722/1, the National Natural Science Foundation of China under Grant no. 60673159 and 70671020, the National High-Tech Research and Development Plan of China under Grant no. 2007AA041201, Specialized Research Fund for the Doctoral Program of Higher Education under Grant no. 20070145017.

Reference

Bouillard A, Gaujal B, Lagrange S, Thierry E. Optimal routing for end-to-end guarantees using Network Calculus. Performance Evaluation 2008;65:883–906.

Castro LN, Timmis J. Artificial immune systems: a new computational intelligence approach. Secaucus, NJ: Springer-Verlag; 2002.

Chen B, Wang JP. Efficient routing and wavelength assignment for multicast in WDM Networks. IEEE Journal on Selected Areas in Communications 2002;20(1):97–109.

Chen M-T, Tseng S-S. A genetic algorithm for multicast routing under delay constraint in WDM network with different light splitting. Journal of Information Science and Engineering 2005;21:85–108.

Chen M-T, Lin BMT, Tseng S-S. Multicast routing and wavelength assignment with delay constraints in WDM networks with heterogeneous capabilities. Journal of Network and Computer Applications 2008;31:47–65.

Cheng H, Wang X, Yang S, Huang M. A multipopulation parallel genetic simulated annealing-based QoS routing and wavelength assignment integration algorithm for multicast in optical networks. Applied Soft Computing 2009;9:677–84.

Chin TS, An ant algorithm for single-hop wavelength assignment in WDM mesh network, In: Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence, IEEE Press, New York, 2005, pp. 111-117.

Din D-R, Genetic algorithm for finding minimal cost light-forest of multicast routing on WDM network, Journal of Artificial Evolution and Applications vol. 2008 (2008), Article ID 536913, 20 p.

Dorigo M, Stützle T. Ant colony optimization. Cambridge, MA: MIT Press; 2004.

Giacomazzi P. Closed-form analysis of end-to-end network delay with Markov-modulated Poisson and fluid traffic. Computer Communications 2009;32:640–8.

Guo L, Wang X, Song Q, Wei X, Hou W, Yang T, Yang F. New insights on survivability in multi-domain optical networks. Information Sciences 2008;178:3635–44.

Guo L, Wang X, Cao J, Zhang Y, Zheng X, Wang X, Hou W. A survivable routing algorithm with differentiated domain protection based on a virtual topology graph in multi-domain optical networks. Information Sciences 2009;179:2453–9.

Hwang I-S, Lee S-N, Chen C-C. A novel ring-tree-based routing and wavelength assignment algorithm for multicasting in DWDM optical mesh networks. Information Sciences 2007;177:5558–72.

Hu X, Zhang J, Zhang L, Swarm intelligence inspired multicast routing: an ant colony optimization approach. In: M. Giacobini (Ed.), Proceedings of the 6th European Workshop on the Application of Nature-inspired Techniques for Telecommunication Networks and other Parallel and Distributed Systems, Tubingen, Germany, 2009, pp. 51-60.

Ibrahim MH, Lee S-Y, Chin M-K, Kassim NM, Mohammad AB. Multimode interference optical splitter based on photodefinable benzocyclobutene (BCB 4024-40) polymer. Optical Engineering 2007;46(1):013401.

Iyengar SS, Wu H-C, Balakrishnan N, Chang SY. Biologically inspired cooperative routing for wireless mobile sensor networks. IEEE Systems Journal 2007;1(1):29–37.

Jia X, Du D, Hu X. Integrated algorithms for delay bounded multicast routing and wavelength assignment in all optical networks. Computer Communications 2001;24:1390–9.

Jia X, Du D, Hu X, Lee M, Gu J. Optimization of wavelength assignment for QoS multicast in WDM networks. IEEE Transactions on communications 2001;49(2):341–50.

Law LK, Krishnamurthy SV, Faloutsos M, Fireworks: an adaptive group communications protocol for mobile ad hoc networks, In: R. Boutaba (Ed.), Proceedings of the 4th International IFIP-TC6 Networking Conference, Waterloo, Canada, 2005, pp. 853-868.

Lee H, Choi H, Subramaniam S, Choi H-A. Survivable embedding of logical topologies in WDM ring networks. Information Sciences 2003;149:151–60.

Li L, Li C. A QoS multicast routing protocol for dynamic group topology. Information Sciences 2005;169:113–30.

Masutti TAS, de Castro LN. Neuro-immune approach to solve routing problems. Neurocomputing 2009;72:2189–97.

Osagie E, Thulasiraman P, Thulasiram RK, PACONET: imProved ant colony optimization routing algorithm for mobile ad hoc networks, In: Proceedings of the 22nd International Conference on Advanced Information Networking and Applications, IEEE Press, New York, 2008, pp. 204-211.

Pavani GS, Waldman H. Restoration in wavelength-routed optical networks by means of ant colony optimization. Photonic Network Communications 2008;16:83–91.

Sim KM, Sun WH. Ant colony optimization for routing and load-balancing: survey and new directions, IEEE Transactions on Systems. Man and Cybernetics-Part A: Systems and Humans 2003;33(5):560–72.

Tancevski L. Intelligent next generation WDM optical networks. Information Sciences 2003;149:211–7.

Tu Y-C, Chen M-C, Sun Y-S, Shih W-K. Enhanced bulk scheduling for supporting delay sensitive streaming applications. Computer Networks 2008;52:971–87.

Vijayalakshmi K, Radhakrishnan S. Artificial immune based hybrid GA for QoS based multicast routing in large scale networks (AISMR). Computer Communications 2008;31:3984–94.

Wang X, Cao J, Cheng H, Huang M. QoS multicast routing for multimedia group communications using intelligent computational methods. Computer Communications 2006;29:2217–29.

Wang C-F, Jan R-H. A factoring approach for the Steiner tree problem in undirected networks. Information Sciences 2007;177:2418–35.

Wang H, Shi Z, Li S. Multicast routing for delay variation bound using a modified ant colony algorithm. Journal of Network and Computer Applications 2009;32:258–72.

Wei X, Guo L, Wang X, Song Q, Li L. Availability guarantee in survivable WDM mesh networks: a time perspective. Information Sciences 2008;178:2406–15.

Wong KKY, Marhic ME, Uesaka K, Kazovsky LG. Wavelength exchange: a novel function for optical networks. Information Sciences 2003;149:161–9.

Yan S, Deogun J. Multi-drop path model for multicast routing and wavelength assignment. Information Sciences 2003;149:113–34.

Yang L, Lin J, Wang D, Jia L. Dynamic route guidance algorithm based on artificial immune system. Journal of Control Theory and Applications 2007;5(4):385–90.

Yang S. Genetic algorithms with memory- and elitism-based immigrants in dynamic environments. Evolutionary Computation 2008;16(3):385–416.

Ye B, Guo M, Chen D, Lu S. A degree-constrained QoS-aware routing algorithm for application layer multicast. Information Sciences 2007;177:3613–26.

Zhao YX, Wu JP, Yin X. From active to passive- progress in testing of Internet routing protocols. Journal of Computer Science & Technology 2002;17(3):264–83.

Znati TF, Alrabiah T, Melhem R. Low-cost, delay-bounded point-to-multipoint communication to support multicasting over WDM networks. Computer Networks 2002;38:423–45.