# Genetic Algorithm for Finding Minimal Cost Light-Forest of Multicast Routing on WDM Network

Der-Rong Din (丁德榮)

Department of Computer Science and Information Engineering

National Chang-Hua University of Education

E-mail: deron@ms45.hinet.net

## Abstract

*Wavelength-division multiplexing* (*WDM*) is an effective technique to make use of the large amount of bandwidth in optical fibers to meet the bandwidth requirement of applications. In this paper, the multicast routing problem on WDM networks is studied. Specifically, the formulation for the WDM multicast routing problem is given. Since the problem is NP-hard, a genetic algorithm (GA) constructs a source-based light-forest (consisting one or more light-trees) for the multicast request is proposed to solve this problem. Simulation results demonstrate that the proposed GA can run efficiently.

**Keywords**: Light-forest, multicast routing, wavelength division multiplexing (WDM), genetic algorithm, path-oriented encoding.

## 1. Introduction

*Wavelength-division multiplexing* (*WDM*) is an effective technique to make use of the large amount of bandwidth in optical fibers to meet the bandwidth requirement of applications. It is generally believed that the next-generation INTERNET will be largely based on a WDM backbone [1]. All-optical networks consist of connecting optical fibers and nodes. Each node has a dynamically configurable optical switch or router, which supports wavelength-based switching or routing. Since there is no electro-optic (E/O) or optic-electronic (O/E) conversion, all-optical networks greatly increase the throughput capacity [2].

There are two types of architectures of WDM optical networks: *single-hop* systems and *multi-hop* systems [3]. In single-hop systems, a communication channel should use the same wavelength throughout the route of the channel; while in multi-hop systems, a channel can consist of multiple light paths (a light path, for point to point communication, is a path that occupies the same wavelength throughout the path) and wavelength conversion is allowed at the joint nodes of two light-paths in the channel.

Multicasting is the transmission of information from one source to multiple destinations simultaneously, *i.e.*, a one-to-many communication technique. Many broadband services such as video conferences, distance learning, and web casting employ multicasting for data delivery. The support of multicasting in the future WDM networks is essential for these applications. Thus, issues concerning supporting multicasting in INTERNET over WDM networks need to be studied [4].
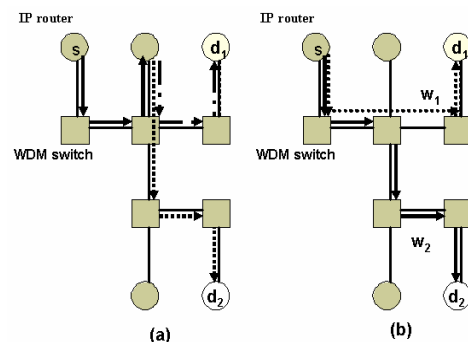


**Fig. 1**. The multicast schemes of WDM networks.

There are several schemes for multicasting data over WDM networks [5]. As shown in Fig. 1(a), a source IP router, as well as each IP router on a multicast tree constructed by the IP layer,

can make copies of a data packet and transmit a copy to each of its child (i.e., immediate downstream router). However, this requires optic-electro-optic (O/E/O) conversions of every data packet at every router on the multicast tree, which may be undesirable and inefficient because routers may be overloaded, IP layer forwarding may introduce a long latency, and data will lose bit-rate and format transparency, and so on. The above-mentioned O/E/O conversions can be avoided by constructing a virtual topology consisting of a set of light-paths (i.e., wavelength-routed paths) from the multicast source to each destination as in Fig. 1(b). However, this is equivalent to have multiple unicasts (i.e., one-to-one connections) in a WDM network. In a previous study [6], if the light-paths were not well designed, the network bandwidth consumed by a large multicast request may become unacceptable. This is because that the problem of efficient multicast routing in a WDM network is already a complicated one due to the fact that one needs to consider wavelength assignment and routing path in a WDM network, simultaneously.

In this paper, the multicast routing in a single-hop WDM networks is studied. Given the network topology, and the multicast request, the goal is to find a light-forest which consists of one or more light-trees (rooted at a multicast source) be used to deliver multicast traffic to all intended destinations efficiently such the total cost is minimized. All nodes are assumed to be *multicast incapable* (MI).

The multicast problem in WDM networks is known to be NP-hard. Since finding an optimal solution to this problem is NP-hard, and that an exact search for optimal solutions is impractical due to exponential growth in execution time. Moreover, traditional heuristic methods and greedy approaches should trap in local optima. *Genetic algorithms* (*GA*) have been trusted as a class of general-purpose search strategies that strike a reasonable balance between exploration and exploitation. Genetic algorithm proposed by John Holland [7] has been constructed as robust stochastic search algorithms for various optimization problems. GA searches by exploiting information sampled from different regions of the solution space. The combination of crossover and mutations helps

GA escape from local optima. These properties of GA provide a good global search methodology for the multicast problem.

In this paper, a genetic algorithm is proposed to construct a *light-forest* (consisting of one or more source-based light-trees) for a given multicast session so that multicast data can be delivered to all the members of the session.

The major contributions of this paper are briefly described as follows: a smart coding method together with the decoding method of GA for the multicast problem on WDM network is constructed. Moreover, heuristic algorithm is used to generate the initial population of the GA and to speed up the convergence of GA.

The rest of this paper is organized as follows. In Section 2, the assumptions and definitions of the problem are described. The details of the proposed genetic algorithm are presented in Section 3. Experimental results are given in Section 4. Final, conclusions are given in Section 5.

## 2. Problem Formulation

### 2.1 Network Model

A multicast connection includes one source port and a few destination ports. According to the wavelength assignment to the ports, there are three different multicast models, namely, the *Multicast with Same Wavelength* (*MSW*) model, the *Multicast with Same Destination Wavelength* (*MSDW*) model and the *Multicast with Any Wavelength* (*MAW*) model (see Fig. 2)[8].
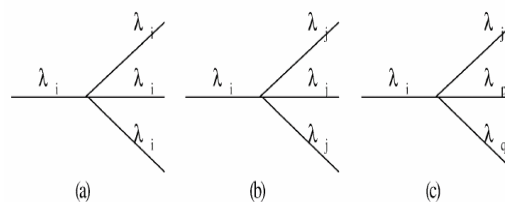


Fig. 2. Three multicast models (a) MSW (b) MSDW (c) MAW.

In this paper, each WDM node is assumed to be *multicast incapable* and has some functions as follows:
- *Drop only*: When the locally attached router is a destination, and there is no need to forward a copy to any downstream node.
- *Continue only*: When the locally attached router is not a destination and there is a

downstream destination.

● *Drop and continue*: When the locally attached router is a destination and there is a downstream destination.

Assume that the source node (the uniquely one) has the function that it is possible to send multiple ``copies'' to the same output using different wavelengths and along the different paths. That is, the source of a multicast request has multiple transmitters (or a tunable one), and hence can transmit to as many children as needed when constructing a multicast tree (rooted at itself). Similarly, a source can transmit to its children on different wavelengths using different transmitters.

## 2.2 Problem Description

A WDM network is modeled by an undirected and connected graph $G$ $(V, E, c, w)$, where $V$ is the vertex-set of $G$ representing the set of nodes in the network with $|V|=n$, and $E$ is the edge-set of $G$ with $|E|=m$ corresponding to fiber links between nodes in the network. Each link carries two opposite-directed fibers in the two directions of the link. Each edge $e$ in $E$ is associated with a cost $c$ $(e)$, which $c$ $(e)$ represents the communication cost of $e$, $c$ $(e)$ can be interpreted as the monetary cost of using link $e$. Let $w$ be the number of wavelengths provided by the WDM network. Function $c$ is additive over the links of a path $P$ $(u, v)$ between two nodes $u$ and $v$, *i.e.*

$$c(P(u,v)) = \sum_{e \in P(u,v)} c(e) \qquad (1)$$

A multicast request is considered in the system, denoted by $r(s, D)$. The request is for setting up a multicast channel from source node $s$ to a group of destination nodes $D= \{d_1, d_2... d_{|D|}\}$ ($s$ not in $D$). The goal is to find the routing trees (or tree) and assign wavelengths to them. Let $T_k$ $(s, D_k)$ be the *light-tree* for the request $r(s, D_k)$ on wavelength $k$, where $k$ is in $\{1,2,... ,w\}$, $D= \cup_{k=1,2,...,w} D_k$ and $D_i \cap D_j = \varnothing$, for $i \neq j$; $i, j \in \{1, 2, ..., |D|\}$, $T= \cup_{k=1,2,...,w} T_k$ and $T$ is called the *light-forest* for the request $r(s, D)$. The cost of a light-tree $T_k(s, D_k)$ on wavelength $k$ is defined to be the sum of the cost of all links in the light-tree $T_k$ $(s, D_k)$. It can be formally defined as

$$c(T_k(s,D_k)) = \sum_{e \in T_k(s,D_k)} c(e) \qquad (2)$$

Similarly, the cost of the light-forest $T$ is defined as

$$c(T) = \sum_{k=1}^{w} c(T_k(s,D_k)) \qquad (3)$$

In this paper, an input optical signal can only be forward to an output port leading to its child, node can perform "*drop and continuous*" function which then transmit the signal to its child until all nodes in the $D_k$ receive it. That is, no light splitting function is enabled on node, thus the degree of node (expect one) of light-tree on each wavelength is less or equal to 2.

The objective cost of multicasting a message on a WDM network, has two components. The first one is the sum of the cost of the light-forest $c(T)$; the other is the number of the (distinct) wavelengths used in the network, because the number of wavelengths that could be used in a WDM network is limited. In a single-hop WDM network, two paths must use different wavelengths if their routes share a common link. Let $y_k=1$, if wavelength $k$ is used by the light-forest $T$; $y_k = 0$, otherwise. Thus, the total cost can be formulated as:

$$c(r(s, D)) = c(T) + \alpha \times \sum_{k=1}^{w} y_k, \qquad (4)$$

where α is the ratio of the cost of light-forest to that of the wavelengths used. The problem is formally restated as follows: given a multicast request $r(s, D)$ in a WDM network, find a set of light trees and assign wavelengths to them and construct a light-forest $T$. The objective is to minimize the sum of the cost of the number of distinct wavelengths time ratio α and the cost of the light-forest $T$.

**Example 1:** A key observation is that, due to multicast incapable constraint of the WDM switch, a single light-tree may not sufficient for multicasting data to all the destinations in a multicast session. An example is shown in Fig. 3(a), where in a WDM network with 18 nodes, node $s$ is the source of a multicast request, and set $D=\{1, 2, 3, 4, 5, 6\}$ is the set of destinations. When using the shortest path heuristic to find a shortest tree from $s$ to $D$, it is possible that nodes 7 is used to forward data to nodes 1, 2, 3 and 6

as shown in Fig. 3(c). Since each node is multicast-incapable, the result tree violated the light-splitting constraint. As a result, path $P(s, 3)$ or $P(s, 6)$ should be reassigned to another wavelength. In this case, a second light-tree which overlaps on link $(s, 7)$ with the first one has to be constructed, resulting in a ``light-forest'', as shown in Fig. 3(c). Thus, $D_1 = \{1, 3, 4, 5\}$, $D_2 = \{2, 6\}$. Alternately, disjoint-path can also be found (e.g. $s \rightarrow 9 \rightarrow 1 \rightarrow 13 \rightarrow 3$) as shown in Fig. 3($d$).

Given that source node $s$ needs to send out two ``copies'' on link $(s, 7)$, two wavelengths (or branches) are needed on link $(s, 7)$ in a wavelength-routed network. Note that, for the same multicast request, using different heuristics will likely result in light-forests with different costs.

## 3. Genetic Algorithm

In this section, the details of GA developed to solve the problem of minimal cost multicast routing problem in WDM network are given. The development of GA requires: (1) a chromosomal coding scheme, (2) genetic crossover operators, (3) mutation operators, (4) a fitness function definition, (5) a replacement strategy, and (6) termination rules.
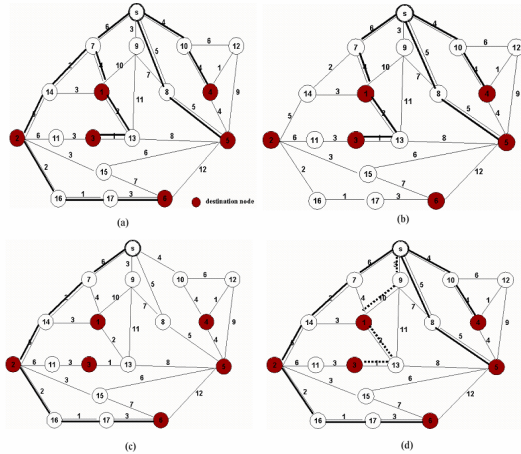


Fig. 3. Graph of Example 1.

### 3.1 Chromosomal Coding

A source-destination pair can be connected by a set of links, which is called a *path*. There are usually many possible paths between any source-destination pair. For example, consider the network shown in Fig. 3(a), the possible paths between $s$ to destination 1 include $s \rightarrow 7 \rightarrow 1$, $s \rightarrow 9 \rightarrow 1$, $s \rightarrow 7 \rightarrow 14 \rightarrow 1$, and so on.

The encoding method (denoted as *path-oriented encoding*) of the GA-based multicast routing algorithm assumes that a path table, consisting of $R_i$ possible paths, has been constructed for each source-destination pair from $s$ to $d_i$. The path-oriented encoding scheme was first proposed in [9] for the point-to-point routing problem and used by Hwang *et al* in [10, 11] to solve the multicast routing problem in general network (not WDM network).

For example, Fig. 4 shows the path table for the source-destination pair $(s, 1)$. The size of the path table, $R_i$, $i=1, 2..., |D|$, is a parameter of algorithm and is initially set to a small number ($R=R_1= R_2= ... =R_{|D|}=16$ or 64). The possible paths in the path table can be sorted according to their cost such that smaller cost paths are assigned smaller path numbers. For a given source node $s$ and destination set D= $\{d_1, d_2, d_3, d_{|D|}\}$, a chromosome can be represented by a string of integers with length $|D|$. A gene $p_i$, $1 < i < |D|$, of the chromosome is an integer in $\{1, 2, ..., R_i\}$, for $i =1, 2, ... ,|D|$, which represents a possible path between $s$ and $d_i$. The relationship among the chromosome, gene, and path table is explained in Fig. 4.
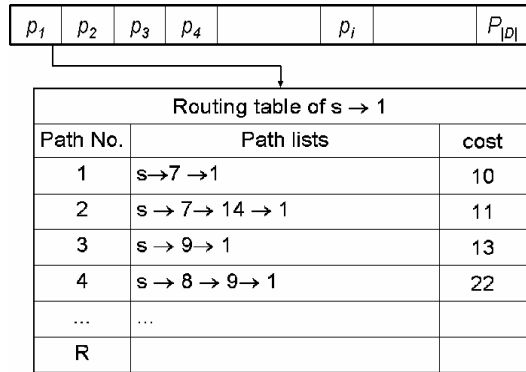


Fig. 4. Chromosome encoding of example 1.

Obviously, a chromosome represents a candidate solution for the multicast routing problem, since it guarantees a path between the source node and any of the destination nodes. However, a chromosome does not necessary represent a tree. Since a chromosome may violates the light-splitting constraint, thus, a decoding method named as *Light-Forest Constructing Algorithm* (*LFCA*), is proposed to transform the chromosome to the corresponding light-forest. The detail of the *Light-Forest Construct Algorithm* (LFCA) is stated as follows:

**Algorithm: Light-Forest Construct Algorithm (LFCA).**

**Step 1**. Extract paths $p_1, p_2, \ldots, p_{|D|}$ from path tables. Sort paths in non-decreasing order according to the cost of each path. Assume that $\{p_{z1}, p_{z2}, \ldots, p_{z|D|}\}$ be the new set of paths with $c(p_{z1}) \leqq c(p_{z2}) \leqq \ldots \leqq c(p_{z|D|})$ and $T_1=T_2=\ldots=T_w=\varnothing$.

**Step 2**. Assign path $p_{z1}$ to wavelength 1, set $w$ to 1, and $T_1=p_{z1}$.

**Step 3**. For each path $p_i$ ($i=2, 3, \ldots, |D|$) performs following Steps.

**Step 4**. Set $j$ to 1. Let *flag* be **FALSE**. While *flag* is **FALSE** and $j$ is less than or equal to $w+1$ perform **Steps 4.1** and **4.2**.

    **Step 4**.1 Test whether $p_{zi} \cup T_j$ is satisfying the light-splitting constraint and not form a cycle? If yes, set *flag* to be **TRUE**, $T_j = p_{zi} \cup T_j$, and assign $p_{zi}$ to wavelength $j$. Otherwise, increase $j$ by 1.

    **Step 4.2** If $j$ is equal to $w+1$ then increase $w$ by 1.

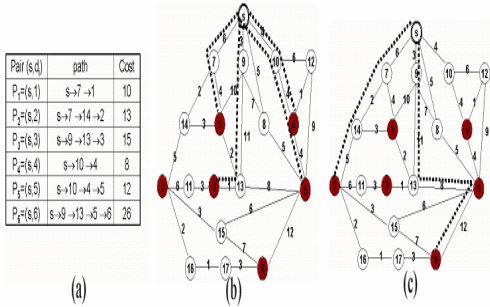**Step 5**. If $w > W$ then set *conflict*=**TRUE** otherwise set *conflict*=**FALSE.**



Fig. 5. Result by applying LFCA to Example 1 on Figure 3.

**Example 2**. Assume the paths extracted from path table of each destination are shown in Fig. 5 (a). After running algorithm LFCA, the result is shown in Fig. 5(b) and (c). Two wavelengths are required to transmit the multicast request. The cost of the light-forest is equal to $38+39+2\times\alpha$.

It is worth to note that, the test of ``Is $T_j \cup p_{zi}$ violated the light-splitting constraint ?'' and ``Does adding $p_{zi}$ to $T_j$ form a cycle?'' can be known by adding path $p_{zi}$ into $T_j$ edge by edge and test ``whether the degree of the end-vertices of edge is greater than 2?'' If no, $p_{zi}$ is added into to light-tree $T_j$ to form a new light-forest, i.e., $T_j=T_j \cup p_{zi}$. Otherwise, $p_{zi}$ is conflicted with light tree $T_j$ and $p_{zi}$ is tried to reassign to next

light-forest $T_{j+1}$. This process is repeated until a wavelength is found for the path.

In [10], the table size $R_i$, for all destinations are the same (i.e., $R=R_1=R_2=\ldots=R_{|D|}$ (say $R=64$) and $R_i$ is fixed during the evaluation of GA. In the scheme proposed in this paper, the path table size is varied. Initially, $R_i$ is set to be a small integer during the evaluating processes, the new source-destination path may insert into the path table. The detail of the data structure and the operation of the path table will be stated later.

**3.2 Genetic crossover operator**

Two types of crossover operators are used in development of the GA: (1) *single-point crossover* (*SPC*); (2) *two-point crossover* (*TPC*). The single-point crossover is the traditional one. In SPC, by randomly selecting two chromosomes (say $P_1$ and $P_2$) for crossover from previous generations and then using a random number generator, an integer $i$ is generated in the range $(1, |D|)$. This number is used as the crossover site. The TPC by randomly selecting two chromosomes (say $P_1$ and $P_2$) for crossover from previous generations and then using a random number generator, two integers $i_1$, $i_2$ are generated in the range $(1, |D|)$. These numbers are used as the crossover sites. It is worth noting that after performing SPC or TPC crossover, LFCA should be used to construct the new light-forest and evaluate the cost of the new chromosomes.

**3.3 Mutation**

Two types of mutations were used to develop this algorithm, they are (1) *single routing path mutation* (*SRPM*), and (2) *multiple routing paths mutation* (*MRPM*).

**(1)** *Single routing path mutation* (*SRPM*)

By randomly selected a destination $d_i$ from a chromosome, the SRPM changes the value of the destination node $d_i$ to a random integer, which is randomly generated between 1 to $R_i$. After single routing path mutation, the light-forest may be changed. Thus, LFCA should be used to construct the new light-forest and evaluate the cost of the new chromosomes.

**(2)** *Multiple routing paths mutation* (*MRPM*)

First, select a number $z$ between 1 and $|D|$. A set $Z$ of $z$ destinations are randomly generated

from a chromosome, the MRPM changes the value of the destination node $d_i$ in $Z$ to a random integer, which is randomly generated between 1 to $R_i$. After multiple routing paths mutation, the light-forest may be changed. Thus, LFCA should be used to construct the new light-forest and evaluate the cost of the new chromosomes.

### 3.4 Fitness function definition

Generally, genetic algorithms use fitness functions to map objectives to costs to achieve the goal of a minimal cost light-forest. An objective function value is associated with each chromosome and the can be represented the following objective function:

$$\text{minimize} \sum_{j=1}^{w} c(T_j) + \alpha \times \sum_{j=1}^{w} y_j \qquad (5)$$

The light-forest construct algorithm may construct infeasible solution of the WDM network (when the number of used wavelengths $K$ is greater than $W$, i.e., $conflict=\mathbf{TRUE}$). Thus, a penalty is attached to the fitness function in the event the solution is infeasible. Let

$$conflict = \begin{cases} K-W & \text{if } K \text{ is greater than } W \\ 0 & \text{otherwise.} \end{cases} \qquad (6)$$

Then, the formulation above can be rewritten in an unconstrained form:

$$\text{minimize} \sum_{j=1}^{w} c(T_j) + \alpha \times \sum_{j=1}^{w} y_j + \beta \times conflict \qquad (7)$$

where *conflict* is the penalty measure associated with a chromosome, and β is the penalty weight. Since the best-fit chromosomes should have a probability of being selected as parents that is proportional to their fitness, they need to be expressed in a maximization form. This is done by subtracting the objective from a large number $C_{max}$. Hence, the fitness function becomes:

$$\text{maximize } C_{max} - \sum_{j=1}^{w} c(T_j) + \alpha \times \sum_{j=1}^{w} y_j + \beta \times conflict \qquad (8)$$

where $C_{max}$ denotes the maximum value observed, so far, of the cost function in the population. Let cost is the value of the cost function for the chromosome; $C_{max}$ can be calculated by the following iterative equation:

$$C_{max} = max\{C_{max}, cost\}, \qquad (9)$$

where $C_{max}$ is initialized to zero.

### 3.5 Heuristic Initialization

Since this problem is NP-hard, a heuristic algorithm is proposed to generate one of the initial populations of the GA and speed up the convergence of the GA. The proposed heuristic algorithm is *shortest path based*, that is, the shortest path algorithm is used to find the minimal cost path from source to destinations. Given the WDM network and the multicast request $r(s, D)$, for each destination node $d_i$ in $D$ find the minimal cost path $P(s, d_i)$ from source node $s$ to $d_i$ in $G$. Then, paths $P(s, d_i)$, $i=1, 2, ..., D$ are combined to form $P$, that is,

$$P = \cup_{i=1}^{|D|} P(s, d_i) \qquad (10)$$

It is worth noting that $P$ should be a tree. If $P$ satisfies the light-splitting constraint, then $P$ may be the *minimal cost light-tree* of the multicast request. Otherwise, another heuristic is used to find the light-tree or light-forest with light-splitting constraint.

Before introducing the proposed heuristic initial algorithm, some notations are introduced and stated as follows: Let $Edge(P(s, d_i))$ be the set of edges that in path $P(s, d_i)$ or edges that at least one of its endpoints (not $s$) on $P(s, d_i)$. $G' = G - Edge(P(s, d_i))$ be the remaining graph by removing those edges and nodes in $Edge(P(s, d_i))$. For example, assume $G$ be the network shown in Fig. 3(a). If $P(s, 2) = s \rightarrow 7 \rightarrow 14 \rightarrow 2$, then $Edge(P(s, 2)) =\{(s, 7), (7, 14), (14,2), (1, 7), (1, 14), (3, 14), (2, 11), (2,15), (2, 16)\}$.

Let $degree(s)$ be the degree of the source node $s$ on tree $P$, then tree $P$ can be divided into $degree(s)$ sub-trees. Let $v_1, v_2, ..., v_{degree(s)}$ be the nodes in $P$ and incident to the source node $s$, then the sub-tree rooted at $v_i$ is represented by $PT(v_i)$, for $i=1, 2, ..., v_{degree(s)}$. Since the routing tree in a wavelength network must not contain cycle(s) and routing tree should satisfy the light-splitting constraint. Thus, it is easy to find that if there is more than one destination on leaves in this sub-tree, only one of these destinations can be chosen to route by this sub-tree and the other destinations should be rerouted. To determine the rerouting paths of the destinations in the leaves, the farthest-first heuristic is introduced.

The basic idea of the farthest-first is that for each sub-tree $PT(v_i)$, $i=1, 2, ..., v_{degree(s)}$, only the farthest destination

$$c(P(s,\ v)) \geq \max_{u \in PT(v_i)}\ c(P(s,\ u))\quad (11)$$

is routed by the path $P(s,\ d_i)$ on $PT(v_i)$, the others should be re-routed. Consider the network shown in Fig. 3(a), there are four destinations in the sub-tree $PT(7)$, they are nodes 1, 2, 3 and 6. The farthest (maximal cost) destination is the node 6, thus the routing paths to destinations 1 and 3 should be rerouted (Note, destination 2 needed not be re-routed since 2 is passed through by the path $P(s, 6)$). Similar, the farthest destinations in the sub-trees $PT(8)$ and $PT(10)$ is node 5 and 4, respectively. Since the routed paths and nodes used by farthest path in each sub-tree can not be used twice. Moreover, the rerouting paths together with exist paths can not form a cycle (or cycles) or violate the light-splitting constraint. Thus, the path $P(s,v)$ to the farthest destinations and $EDGE(P(s,v))$ in each sub-trees should be removed. Let *UNREACH* be the set of all destinations not been routed which are sorted in nonincreasing order according to the minimal cost path from source *s* to destinations in *G*. The Farthest-First Algorithm is described as follows:

**Algorithm: Farthest-First**

**Step 1.** Let $w=1$, $G^1 = G^2 = ... = G^W = G$. For each destination $d_i$ in *D* find the minimal cost path $P(s,d_i)$. Combine paths to form a routing tree *P*. If *P* satisfied the light-splitting constraint, then **DONE**. Otherwise perform **Step 2**.

**Step 2.** For each sub-tree $PT(v_i)$, rooted at $v_i$ for $i=1, 2, ..., degree(s)$, find the farthest destination node $v_i'$, the routing path $P(s, v_i')$ and $EDGE(P(s, v_i'))$. Construct $G'$ by removing edges in $EDGE(P(s, v_i'))$, $i=1, 2, ..., degree(s)$ from *G* and construct *UNREACH*. While *UNREACH* is nonempty perform **Step 3**.

**Step 3.** Take the farthest destination in *UNREACH*, say *v*, find the minimal cost path and the minimal value *w* where there is a path $P(s,v)$ in $G^l$, $l=1, 2, ..., w+1$. Assign path $P(s, v)$ to wavelength *l*, subtract $EDGE(P(s, v))$ from $G^l$, remove the destination *v* and other destination passed by path $P(s, v)$ from UNREACH.

The process is illustrated as follows: First, find the shortest-path tree *P* (as shown in Fig. 6(a)) and remove those edges in $EDGE(P(s, 6))$,

$EDGE(P(s, 5))$, and $EDGE(P(s, 4))$ from *G*, that is, find $G'=G- EDGE(P(s, 6)) - EDGE(P(s, 5)) - EDGE(P(s,4))$. The remaining graph is shown in Fig. 6(b). for example, $UNREACH= \{3, 1\}$. Since the node 3 is the farthest one, the minimal cost path $P'(s, 3)$ from *s* to 3 in *G'* is $s \rightarrow 9 \rightarrow 11 \rightarrow 3$. Further, after subtracting $EDGE(P'(s, 3))$ from *G'*, since there is no path from *s* can be route to destination 1, another wavelength network is needed to reroute the path. Thus, consider the wavelength graph $G^2=G$, the minimal cost path from *s* to 1 is $s \rightarrow 7 \rightarrow 1$. Final, the light-forest is shown in Fig. 6(c).

### 3.6 Replacement strategy

Initially, assume $N_{population}$ be the number chromosomes to be generated and $N_{parent}$ connection chromosomes are randomly constructed. In the process of selection, $N_{population}/2$ pairs of connections are randomly selected for crossover to generate the new generation of chromosomes. After crossover, chromosomes are sorted according to the fitness function in increasing order, $N_{parent}$ chromosomes with smaller fitness is selected to construct the new generation. The values of $N_{population}$ and $N_{parent}$ will be determined through experiments.

### 3.7 Termination rules

Execution of GA can be terminated when the number of generations exceeds an upper bound (*ng*) specified by the user.
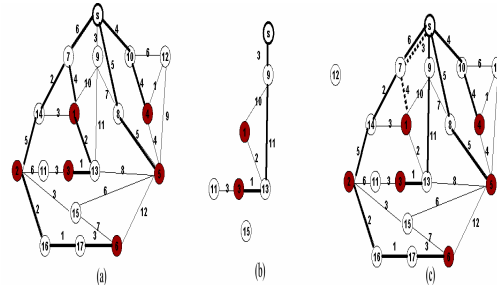


Fig. 6. Result by applying Farthest-First heuristic initial algorithm.

## 4. Experimental Results

In order to evaluate the performance of the proposed genetic algorithm, the algorithm has been implemented and applied it to solve problems that were randomly generated. The results of these experiments are reported below. In all the experiments, the implementation was

conducted in C, and all the experiments were run on a personal computer (PC) with a Pentium III 1GHZ CPU and 256MB RAM.

The GA has been run on general network with different nodes (n=20, 40, 60), and with different number of destination nodes ($|D|$ = 4, 6, or 8) and with $\alpha$=10. The link costs are randomly selected from the set of integer {1, 2, …, 100}. For these tests, the sets of multicast requests are randomly selected from a connected network. The parameters of GA are crossover probability =1.0, population size $Pp_{opulation}$=1200, parent size $N_{parent}$=200, the maximum number of generation is 1000. The experimental results are shown in Table 1. The heading of "IA" refers to the cost by applying and the heading of "GA" refers to the value obtained by performing the proposed genetic algorithm. The heading of "reduce" defines as "(IA-GA)/GA×100%". Test results show that the proposed GA obtained good performance. GA can reduce the cost in average 9.38% of the initial solution.

## 5. Conclusions

In this paper, the multicast problem on WDM networks without wavelength conversion and static traffic is modeled and studied. Since finding the optimal solution problem is an NP-hard problem, a genetic algorithm is proposed to solve this problem. The simulation results show that the proposed genetic algorithm is robust for this problem.

## References

[1] M. Jeong, Y. Xiong, H. C. Cankaya, M. Vandenhoute, and C. Qiao, "Efficient multicast schemes for optical burst-switched WDM networks," in *Proc. IEEE International Conference on Communications* (*ICC*), pp. 1289-1294, 2000.

[2] B. Mukherjee, *Optical Communication Networks*. New York: Mc-Graw-Hill, 1997.

[3] 2 J. Beasly, "An SST-based algorithm for the Steiner problem in graphs," *Networks*, vol. 19, pp. 1-16, 1989.

[4] C. Qiao, M. Jeong, A. Guha, X. Zhang, and J. Wei, "WDM multicasting in IP over WDM networks," in *Proc. Int. Conf. Network Protocol* (*ICNP*), Nov. 1999, pp. 89-96.

[5] X. Zhang, J. Wei, and C. Qiao, "On fundamental issues in IP over WDM multicast," in *Int. Conf. Computer Communications and Networks* (*IC3N*), Oct. 1999, pp. 84-90.

[6] R. Malli, X. Zhang, and C. Qiao, "Benefits of multicasting in all-optical networks," in *SPIE Proceedings, All Optical Networking*, Nov. 1998, pp. 209-220.

[7] J. Holland, *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press (Ann Arbor), 1975.

[8] Aijun Ding and Gee-Swee Poo, "A survey of optical multicast over WDM networks," *computer communications* 26 2003, 1193-200.

[9] N. Shimamoto, A. Hiramatu, and K. Yamasaki, "A dynamic routing control based on a genetic algorithm," 1993 *IEEE International Conference on Neural Network*, 1993, pp. 1123-1128.

[10] Ren-Hung Hwang, Wei-Yuan Do, and Shyi-Chang Yang, "Multicast Routing Based on Genetic Algorithms," *Journal of Information Science and Engineering*, Vol. 16, No. 6, Nov. 2000, pp. 885-901.

[11] Jang-Jiin Wu, Ren-Hung Hwang and Hsueh-I Lu, 2000, "Multicast Routing with Multiple QoS Constraints in ATM Networks," *Information Sciences: an International Journal*, Vol. 124, pp. 29-57.

Table 1. Comparison of experimental result by GA and heuristic algorithm.

| $n$ | $\|D\|$ | $\alpha$ | IA | GA | reduce |
|---|---|---|---|---|---|
| 20 | 4 | 10 | 123 | 111 | 10.81% |
| 20 | 6 | 10 | 134 | 123 | 8.94% |
| 20 | 8 | 10 | 167 | 147 | 13.61% |
| 40 | 4 | 10 | 167 | 144 | 15.97% |
| 40 | 6 | 10 | 184 | 164 | 12.20% |
| 40 | 8 | 10 | 208 | 195 | 6.67% |
| 60 | 4 | 10 | 201 | 194 | 3.61% |
| 60 | 6 | 10 | 237 | 221 | 7.24% |
| 60 | 8 | 10 | 256 | 243 | 5.35% |
| average | | | | | **9.38%** |