

A Genetic Algorithm for Multicast Routing under Delay Constraint in WDM Network with Different Light Splitting*

MING-TSUNG CHEN AND SHIAN-SHYONG TSENG

Department of Computer and Information Science

National Chiao Tung University

Hsinchu, 300 Taiwan

Because optical WDM networks will become a realistic choice for buildings backbones, multicasting in the WDM network should be supported for various network applications. In this paper, a new multicast problem, *Multicast Routing under Delay Constraint Problem (MRDCP)*, routing a request with delay bound to all destinations in a WDM network with different light splitting is solved by genetic algorithms (GAs), where different light splitting means that nodes in the network can transmit one copy or multiple copies to other nodes by using the same wavelength. The MRDCP can be reduced to the *Minimal Steiner Tree Problem (MSTP)* which has been shown to be NP-Complete. We propose a destination-oriented representation to represent chromosomes, three general genetic operators (*selection*, *crossover*, and *mutation*), four types of operators (*Chromosome Crossover*, *Individual Crossover*, *Chromosome Mutation*, and *Individual Mutation*). Four mutation heuristics (*Random Mutation (RM)*, *Cost First Mutation (CFM)*, *Delay First Mutation (DFM)*, and *Hybrid Mutation (HM)*) are employed in the GA method. Finally, experimental results show that our solution model can obtain a near optimal solution.

Keywords: genetic algorithm, multicast routing, WDM network, delay constraint, splitting degree, NP-hard

1. INTRODUCTION

Optical networks [1] are high-capacity telecommunications networks, based on optical technologies and optical components, which provide routing, grooming, and restoration at the wavelength level as well as wavelength-services. The technology of WDM (*Wavelength Division Multiplexing*) networks [2], based on optical wavelength-division multiplexing on an optical fiber to form multiple-communication channels at different wavelengths, provides connectivity among optical components to make optical communication meet the increasing demands for high channel bandwidth and low communication delay. To transmit data between a source and a destination in a WDM network, a *light-path* that connects the two nodes should be established.

The growth in network applications, such as video conferencing, video on demand system, real-time control systems, on-line shopping, gaming, stock exchange, and so on, has generated new requirements for communication models such as *multicast communication*. *Multicast* is a routing problem of sending a message or data from a single source

Received April 16, 2003; revised November 4, 2003; accepted February 25, 2004.

Communicated by Chu-Sing Yang.

* This work was partially supported by MOE Program for Promoting Academic Excellence of Universities under grant No. 89-E-FA04-1-4, High Confidence Information Systems.



to multiple destinations. To support multicast communication in *WDM* networks, nodes in the *WDM* network may have *light splitting capability* that is used to split an optical signal from an input port into multiple signals to output ports without electrical conversions. Also, a routing-tree used to transmit a request could be constructed. The *light-tree* [3] is a special routing-tree made by configuring nodes in a physical topology that occupies the same wavelength in the tree links. Therefore, each branch node of the light-tree should be an optical switch with light splitting capability. These branch nodes, called *MC* (*multicast capable*) nodes are usually more expensive to build for a *WDM* network than those without, which are called as *MI* (*multicast incapable*) nodes, due to their complex architecture [1]. Furthermore, the two important measurements (communication-cost and wavelength) for evaluating the performance of routing-tree are usually considered in *WDM* networks for *QoS* (*Quality of Service*). Another measurement, transmission-delay, will come arise in the problem of multicasting in *WDM* networks. Considering these different measurements, several protocols [4] and algorithms [5-8] have been proposed for traditional networks to solve various problems. Recently, the multicast routing problem in *WDM* networks with sparse light splitting, was proposed and solved by X. Zhang *et al.* [9]. Also, other research into multicast routing with wavelength conversion [10] or with delay bound [11] was proposed. The *Multicast Routing under Delay Constraint Problem* (*MRDCP*) is to find a *light-forest* with minimal *multicast cost* which is a sum measured of communication-cost and wavelength on a *WDM* network with different light splitting such that a request can be routed under delay bound. This problem was first proposed and solved by using several heuristics in [13], where the *light-forest* is a set of light-trees in which each light-tree needs one wavelength to route the multicast request. The new problem is NP-Complete because this problem can be reduced to the *Minimal Steiner Tree Problem* (*MSTP*) which is NP-Complete [20]. The two characteristics of a *WDM* network are, nodes with different light splitting and a request with delay bound, and are considered simultaneously in [13].

In this paper, the *MRDCP* will be solved by using *genetic algorithms* (*GA*) with four types of operators (*Chromosome Crossover*, *Individual Crossover*, *Chromosome Mutation*, and *Individual Mutation*) and four mutation heuristics (*Random Mutation*, *Cost First Mutation*, *Delay First Mutation*, and *Hybrid Mutation*). The goal is to find a routing-tree with minimal multicast cost which satisfies a delay constraint and a destination constraint defined in [13]. Finally, the obtained routing-tree can be separated into a light-forest by the *Separating_Step* [13].

The remainder of this paper is organized as follows. Section 2 presents related work on the multicast routing problem. In section 3, we formally define the problem. The background of GAs is discussed in section 4. In section 5, the GA framework for *MRDCP* and four heuristics for finding the optimal solution of the problem are proposed. Section 6 and section 7 give simulation results and conclusions, respectively.

2. RELATED WORK

Several related research papers [9-13] recently concerned multicast routing in *WDM* networks. The multicast routing problem with sparse light splitting was proposed and solved by X. Zhang *et al.* [9], where the network has two types of nodes, an *MC* node



with unrestricted light splitting capability and an *MI* node in which all nodes do not have the capability of wavelength conversion. Four routing algorithms, Reroute-to-Source, Reroute-to-Any, Member-First, and Member-Only, were proposed to find a light-forest. When constructing the light-forest, nodes connecting to an *MI* node are rerouted by different rerouting algorithms. Nevertheless, the communication-cost of a light-forest and the number of wavelengths were not considered in the process of constructing the light-forest; that is, the minimizations of communication-cost and wavelength of the light-forest were not considered in [9]. For these different algorithms, they can find different light-forests with different numbers of wavelengths by experiments in a randomly generated network with 13 nodes.

The research reported in [10] proposed the *VS-rooted* approach to solve the multicast routing in *WDM* networks with a sparse virtual source (*VS*), where the *VS* is a node with wavelength conversion and light splitting capability. The *VS* node can transmit an incoming signal to any number of output links on any wavelength. This approach works in two phases, one which partitions the network into regions based on the vicinity of *VS* nodes, and one which generates a multicast tree by connecting the source and *VS* nodes containing a region with some destination. The paper claims that the approach can obtain a significant reduction in setup time, the utilization of wavelength and the number of links. Nevertheless, how to find a light-forest with minimal communication cost, minimum number of wavelengths, and minimum cost of wavelength conversion were not discussed. It must be noted that the wavelength assignment, minimizing communication cost, minimizing number of wavelengths, and the request with delay bound were not considered in [9, 10].

The research discussing routing a request with delay bound was proposed by X. H. Jia *et al.* [11]. Under the assumption that each node in the network must have light splitting capability, two integrated algorithms for finding a routing tree from a source to all destinations and assigning a wavelength to the generated routing tree, has two advantages: the number of wavelengths used is small and the network cost of the routing tree is low. In [12], B. Chen proposed an efficient approximation algorithm to solve a similar problem but without requesting a delay bound. The proposed heuristic takes into account both the cost of using wavelength on links and the cost of wavelength conversion. The problem, *Multicast Routing under Delay Constraint Problem (MRDCP)*, was proposed and solved first by using several heuristics in [13]. It must be noted that the problem involves a request with delay bound, *WDM* network with sparse light splitting capability, and minimizing multicast cost which is a linear combination of communication-cost and number of wavelengths of the light-forest. The new problem is NP-Complete and is more difficult than previous proposed problems [9-12]. Nevertheless, papers [9, 11, 13] do not take wavelength assignment into consideration.

3. FORMULATION

A *WDM* network is represented by a weighted graph $G(V, E)$, where the node set V represents the optical nodes (switches or routers), and the edge set E represents the optical links between nodes. The number of nodes in the *WDM* network is $|V| = n$. Each link is composed of two oppositely directed fibers. For each link e , $c(e)$ and $d(e)$ are associ-



ated with edge e to represent the communication-cost and the transmission-delay, respectively. $\theta(v) \geq 1$ represents the *light splitting capacity* of node $v \in V$ and is the number of copies of requests that can be forwarded to other nodes. Therefore, v can transmit $\theta(v)$ copies to other nodes concurrently using the same wavelength.

In [13], a multicast request represented by $r(s, D = \{d_1, d_2, \dots, d_m\}, \Delta)$ goes from a certain source $s \in V$ passes through several nodes. All destinations in the set $D \subseteq V - \{s\}$ could be visited, where $|D| = m$. The transmission-delay of all the light-paths between s and the destinations must be bounded by Δ .

Assume that there are q paths in $G(V, E)$ between any two nodes u and v and let each path be represented by $P_i(u, v) = \langle w_1^i, w_2^i, \dots, w_{l_i-1}^i, w_{l_i}^i \rangle$, where $w_1^i = u, w_{l_i}^i = v$, and $e_{w_j^i, w_{j+1}^i}$ represents a link between nodes w_j^i and w_{j+1}^i , and l_i is the number of nodes in $P_i(u, v)$ for $1 \leq i \leq q$ and $1 \leq j \leq l_i - 1$. $\mathbf{P}(u, v) = \{P_i(u, v) \mid 1 \leq i \leq q\}$ is used to represent the set of all light-paths between nodes u and v . The communication-cost and the transmission-delay of the path $P_i(u, v)$ described by $c(P_i(u, v))$ and $d(P_i(u, v))$ are represented as

$$\text{communication-cost : } c(P_i(u, v)) = \sum_{e \in P_i(u, v)} c(e) = c(e_{w_1^i, w_2^i}) + c(e_{w_2^i, w_3^i}) + \dots + c(e_{w_{l_i-1}^i, w_{l_i}^i})$$

$$\text{transmission-delay : } d(P_i(u, v)) = \sum_{e \in P_i(u, v)} d(e) = d(e_{w_1^i, w_2^i}) + d(e_{w_2^i, w_3^i}) + \dots + d(e_{w_{l_i-1}^i, w_{l_i}^i})$$

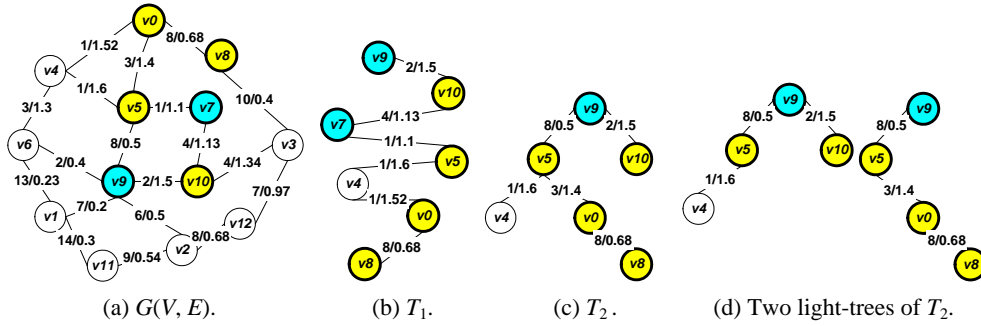


Fig. 1. The WDM network $G(V, E)$ and routing-trees for $r(v_9, \{v_0, v_5, v_8, v_{10}\}, 3.3)$.

Example 1: The graph $G(V, E)$ shown in Fig. 1 (a) is used to represent a WDM network with 13 nodes, where nodes v_7 and v_9 are MC nodes. Each link in the graph is associated with a value-pair “ a/b ”, where a and b are the communication-cost and the transmission-delay of a link, respectively. For a given request, $r(v_9, \{v_0, v_5, v_8, v_{10}\}, 3.3)$, on the WDM network, trees T_1 and T_2 , shown in Figs. 1 (b) and (c), are two possible routing-trees for r , where v_9 is the source, v_0, v_5, v_8 , and v_{10} are destinations, and the delay bound is 3.3 time units. T_1 will need 1 wavelength, 17 (i.e., $2 + 4 + 1 + 1 + 1 + 8$) communication-cost units, and 7.53 (i.e., $1.5 + 1.3 + 1.1 + 1.6 + 1.52 + 0.68$) time units, but this is not *feasible* because 7.53 time units are greater than the bound 3.3. Nevertheless, because v_5 is not an MC node and the out-degree of v_5 is 2, T_2 will need 2 wavelengths for routing

the request from v_5 to v_4 and from v_5 to v_0 . Two light-trees T_2 , shown in Fig. 1 (d), need 11 ($8 + 1 + 2$) and 19 ($8 + 3 + 8$) communication-cost units, and 2.1 ($0.5 + 1.6$) time units and 2.58 ($0.5 + 1.4 + 0.68$) time units, respectively. We may conclude that T_2 shown in Fig. 1 (c) is feasible because the maximum 2.58 time units is smaller than 3.3. \square

Among light-paths in $\mathbf{P}(u, v)$, two critical light-paths, *critical cost light-path* (CCLP) whose communication-cost is minimal and *critical delay light-path* (CDLP) whose transmission delay is minimal, can be denoted as $P^c(u, v)$ and $P^d(u, v)$, respectively, where

$$\text{critical cost light-path (CCLP): } P^c(u, v), \text{ where } c(P^c(u, v)) = \min_{1 \leq i \leq q} c(P_i(u, v))$$

$$\text{critical delay light-path (CDLP): } P^d(u, v), \text{ where } d(P^d(u, v)) = \min_{1 \leq i \leq q} d(P_i(u, v))$$

The given lp light-paths, $P(u_1, v_1)$, $P(u_2, v_2)$, ..., and $P(u_{lp}, v_{lp})$, can be combined into a graph, $\bigcup_{i=1}^{lp} P(u_i, v_i)$. A routing tree $MSpT^c(\bigcup_{i=1}^{lp} P(u_i, v_i), D)$ [13] can be obtained by applying *Prim's MSpT (Minimum Spanning Tree) algorithm* [14] to find the *MSpT* with minimum sum of communication costs, and by eliminating all leaf nodes which do not belong to the destination set D .

Example 2: Consider the four light-paths, $P_1(v_9, v_0) = \langle v_9, v_6, v_4, v_5, v_0 \rangle$, $P_2(v_9, v_5) = \langle v_9, v_5 \rangle$, $P_3(v_9, v_{10}) = \langle v_9, v_2, v_{12}, v_3, v_{10} \rangle$, $P_4(v_9, v_8) = \langle v_9, v_{10}, v_3, v_8 \rangle$, and $D = \{v_0, v_5, v_{10}, v_8\}$ shown in Fig. 2 (a). The graphs, $T = P_1(v_9, v_0) \cup P_2(v_9, v_5) \cup P_3(v_9, v_{10}) \cup P_4(v_9, v_8)$ and $MSpT^c(T, D)$, are shown in Figs. 2 (b) and (c), respectively. \square

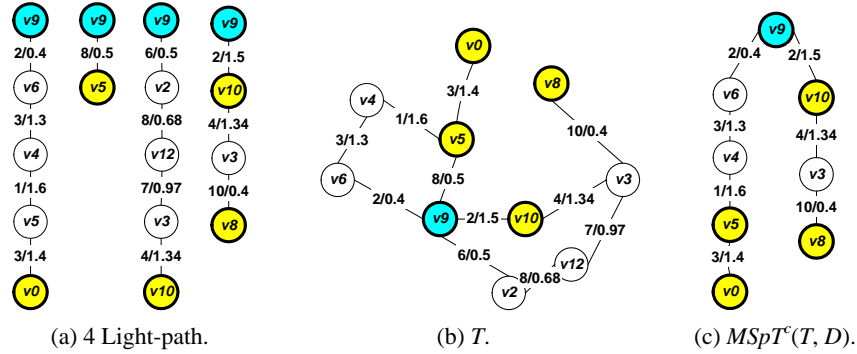


Fig. 2. Sample of $MSpT^c(T, D)$.

Given a routing tree T for the $r(s, D, \Delta)$, assume that the root s of T has τ sub-trees, $ST_1, ST_2, \dots, ST_\tau$. Let $\omega(T)$ represent the number of minimum required wavelengths of T , and $\omega(T) = \max_{1 \leq i \leq \tau} \omega(ST_i)$. In [13], $\omega(T)$ can be defined recursively as



$$\omega(T) = \begin{cases} 1 & T \text{ having a root node only} \\ \max\left(\left\lceil \frac{\sum_{1 \leq i \leq \tau} \omega(ST_i)}{\theta(s)} \right\rceil, \varpi(T)\right) & \text{otherwise} \end{cases} \quad (1)$$

Because the total communication cost of an edge is directly proportional to the number of used wavelengths of this edge and ST_i needs $\omega(ST_i)$ wavelengths, $\omega(ST_i)$ wavelengths in e_{s,s_i} will be required to route the request. Therefore, the total communication cost and the transmission delay for routing a request from s to each destination in sub-tree ST_i should be $\omega(ST_i) \cdot c(e_{s,s_i}) + c(ST_i)$ and $d(ST_i) + d(e_{s,s_i})$, respectively, where s_i is the root of ST_i . The communication cost and transmission delay of T are described recursively as:

$$c(T) = \sum_{1 \leq i \leq \tau} (\omega(ST_i) \cdot c(e_{s,s_i}) + c(ST_i)) \quad (2)$$

$$d(T) = \max_{1 \leq i \leq \tau} (d(ST_i) + d(e_{s,s_i})) \quad (3)$$

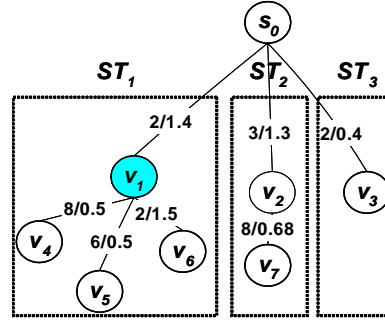


Fig. 3. Sample of routing-tree T .

Example 3: The root s_0 of the routing tree T shown in Fig. 3 has three sub trees ($\tau = 3$), ST_1 , ST_2 , ST_3 . In T , node v_1 is an MC node with $\theta(v_1) = 2$ and the others are MI nodes. We can obtain $\varpi(ST_1) = 1$, $\omega(ST_1) = \max(\lceil \frac{1+1+1}{\theta(v_1)} \rceil, 1) = 2$, and $\varpi(ST_2) = 1$, $\omega(ST_2) = 1$, and $\omega(ST_3) = 1$. Therefore, $\varpi(T) = \max_{1 \leq i \leq 3} \omega(ST_i) = 2$, $\omega(T) = \max(\lceil \frac{\sum_{1 \leq i \leq 3} \omega(ST_i)}{\theta(s_0)} \rceil, \varpi(T)) = \max(\lceil \frac{1+2+1}{1} \rceil, 2) = 4$. The communication costs and the transmission delays of ST_1 , ST_2 , and ST_3 can be computed by $c(ST_1) = 1 \cdot 8 + 1 \cdot 6 + 1 \cdot 2 = 16$, $d(ST_1) = \max(0.5, 0.5, 1.5) = 1.5$, $c(ST_2) = 8$, $d(ST_2) = 0.68$, $c(ST_3) = 0$, and $d(ST_3) = 0$, respectively. Therefore, $c(T) = (\omega(ST_1) \cdot c(e_{s_0,v_1}) + c(ST_1)) + (\omega(ST_2) \cdot c(e_{s_0,v_2}) + c(ST_2)) + (\omega(ST_3) \cdot c(e_{s_0,v_3}) + c(ST_3)) = (2 \cdot 2 + 16) + (1 \cdot 3 + 8) + (1 \cdot 2 + 0) = 33$ and $d(T) = \max(d(ST_1) + d(e_{s_0,v_1}), d(ST_2) + d(e_{s_0,v_2}), d(ST_3) + d(e_{s_0,v_3})) = \max(1.5 + 1.4, 0.68 + 1.3, 0 + 0.4) = 2.9$ is obtained. \square



In a *WDM* network, the communication cost and wavelength are two important resources. For a network with restricted wavelength bandwidth, the route needing fewer wavelengths such that the request can be routed to all destinations is more important; otherwise, a route with lower communication cost is desired. The two measurements would be used to evaluate a routing-tree, but it is difficult to decide which one is the more important. Therefore, the *weight ratio* α is defined as the ratio of the two measurements. The *multicast cost function* f used to calculate the *multicast cost* of the routing-tree T is defined as

$$f(T) = c(T) + \alpha \omega(T) \quad (4)$$

In our problem, the *WDM* network does not provide the capability of wavelength conversion, and a light tree arriving at different destinations can be used to route the request by using some wavelength, so a routing tree T for the $r(s, D, \Delta)$ can be *separated* into $\omega(T)$ light-trees, $T_L^1, T_L^2, \dots, T_L^{\omega(T)}$; that is, $Separating_Step(T) = \{T_L^i \mid 1 \leq i \leq \omega(T)\}$, $T = \bigcup_{i=1}^{\omega(T)} (T_L^i)$, $c(T) = \sum_{i=1}^{\omega(T)} c(T_L^i)$, $d(T) = \max_{1 \leq i \leq \omega(T)} d(T_L^i)$, $f(T) = \sum_{i=1}^{\omega(T)} f(T_L^i)$, and $\omega(T_L^i) = 1$ for $1 \leq i \leq \omega(T)$, where the *Separating_Step* can be found in [13]. Therefore, a *light forest* $\Gamma = \{T_L^1, T_L^2, \dots, T_L^{\omega(T)}\}$ of T is a set of light trees separated from T , and each T_L^i can be used to route the request to destination set $D_i \subset D$.

The light-forest $\Gamma = \{T_L^1, T_L^2, \dots, T_L^{\omega(T)}\}$ of T for $r(s, D, \Delta)$ is *feasible* if it satisfies three constraints, *destination constraint*, *delay constraint*, and *degree constraint* formulated as

$$(1) \text{ destination constraint: } D = \bigcup_{i=1}^{\omega(T)} D_i \quad (5)$$

$$(2) \text{ delay constraint: } d(T_L^i) \leq \Delta \quad (6)$$

$$(3) \text{ degree constraint: } \omega(T_L^i) = 1 \quad (7)$$

where D_i is a destination set routed by T_L^i , $T_L^i \in \Gamma$, and $1 \leq i \leq \omega(T)$.

A routing tree T is a *candidate* if it satisfies the *delay* and *destination constraints*. It should be noted that a candidate does not necessarily satisfy *degree constraint* because the candidate could be separated into a feasible light-forest by the *Separating_Step* [13]. An *optimal candidate* means that the candidate has minimal multicast cost. In the definition of the multicast cost function, when $\alpha = 0$ has been chosen, the optimal candidate will have a minimal sum of communication costs; on the other hand, if α has a large value, the optimal candidate will have minimal wavelength consumption. Therefore, if $\alpha > 1$, the effect of wavelength usage may be greater than the communication cost; otherwise, communication-cost is the major concern.

Therefore, once an efficient candidate is found, a feasible light-forest can easily be obtained by separating this candidate; e.g., the candidate needs two wavelengths as shown in Fig. 1 (c), which could be separated into two light trees as shown in Fig. 1 (d), and the two light trees can be merged into the original candidate. It can be seen that the multicast cost of the light forest consisting of the two light trees is equivalent to the multicast cost of the candidate.



Two special cases, a network which can light by setting all light splitting capacities of nodes to ∞ and a network which can split sparse light by setting all light splitting capacities of nodes to 1 or ∞ to route a request with and without delay bound, were proposed in [11] and [9], respectively. In this paper, we solve the more generalized problem *MRDCP* by using a GA method. Given a *WDM* network $G(V, E)$ with different light splitting and a request $r(s, D, \Delta)$, find a feasible light-forest Γ such that $f(\Gamma)$ is minimal. The detailed description is as follows.

4. CONCEPT OF GA

In [15] the GA search space consists of all possible solutions to the problem. A solution in the search space is called an *individual* whose genotype is composed of a set of *chromosomes* which are represented by sequences of 0s and 1s. These chromosomes of individuals could dominate phenotypes of individuals. Each individual has an associated objective function called *fitness*. A good individual is the one who has a high or low fitness value depending upon the problem (maximization or minimization). The strength of a chromosome in the individual is represented by its *fitness value*, and the chromosomes of the individuals are carried to the next generation. A set of individuals with associated fitness values is called *population*. This population at a given stage of GA is referred to as a *generation*. The best individual in a generation (i.e., the individual with the best fitness value) is discovered. The general GA proceeds as follows.

Genetic Algorithm()

```

Begin
  Initialize population;
  while (not terminal condition) do
    Begin
      choose parents from population; /* Selection/Reproduction */
      construct offspring by combining parents; /* Crossover */
      optimize (offspring); /* Mutation */
      If suited (offspring) then
        replace worst fit (population) by better offspring; /* Survival of the fittest */
    End;
  End.

```

There are three main components in the while loop:

- (1) The process of selecting good individuals from the current generation who are to be carried to the next generation is called *selection/reproduction*.
- (2) The process of shuffling two randomly selected strings (chromosomes) in the two individuals to generate new offspring is called *crossover*. Sometimes one or more bits of a chromosome are complemented to generate a new offspring. This process of complementation is called *mutation*.
- (3) The replacement of the worst performing individuals based on fitness value is called *replacement*.



The population size is finite in each generation of GA, which implies that only relatively fit individuals in generation j will be carried to generation $j + 1$. The power of GA comes from the fact that the algorithm converges rapidly to an optimal or near optimal solution. The iterative process is terminated when the solution reaches the optimum value [16]. The three genetic operators, namely selection, crossover and mutation are discussed in the next section.

4.1 Selection/Reproduction

Since the population size in each generation is limited, only a finite number of good individuals will be copied into the mating pool, depending on their fitness values. The individuals with higher fitness values contribute more copies to the mating pool than those with lower fitness values. This can be achieved by assigning a proportionately higher probability of copying an individual that has a higher fitness value. The Selection/reproduction uses the fitness values of the individuals obtained after evaluating the objective function. It uses a biased roulette wheel [16, 17] for the selection of individuals to be taken into the mating pool. It ensures that highly fit individuals (with high fitness value) have a higher number of offspring in the mating pool. Each individual I_i in the current generation is allotted a roulette wheel slot with size in proportion to Pr_i to its fitness value. This proportion Pr_i can be defined as follows. Let $Of(I_i)$ be the actual fitness value of the individual I_i in a generation, $Sum = \sum_{I_i \text{ in population}} Of(I_i)$ be the sum of the fitness values of all individuals in the generation, and let $Pr_i = Of(I_i)/Sum$. When the roulette wheel is spun, there is a greater chance that a better individual will be copied into the mating pool because a good individual occupies a larger area on the roulette wheel.

4.2 Crossover

This operator involves two steps. First, from the mating pool, the two individuals are selected at random for mating, and second, the crossover point c is selected uniformly at random in the interval $[1, l]$ for each pair of chromosomes in the two chosen individuals, where l is the length of the chromosome. Two new chromosomes called *offspring-chromosome* are then obtained by swapping all characters between positions $c + 1$ and n . Two new individuals, called *offspring*, who own the *offspring chromosomes* with different genotypes, are also obtained. This can be shown using two chromosomes P and Q each of length $n = 6$ bit vectors.

Chromosome P: $\langle u_1, u_2, u_3, u_4, u_5, u_6 \rangle$

Chromosome Q: $\langle v_1, v_2, v_3, v_4, v_5, v_6 \rangle$,

where u_i and v_j are fixed length bit vectors for $1 \leq i, j \leq 6$.

Let crossover point c be the third vector from the left. Bit vectors between 4 and 6 are swapped and bit vectors between 1 and 3 remain unchanged. Then the two offspring-chromosomes can be obtained as follows,



Offspring-chromosome R: $\langle u_1, u_2, u_3, v_4, v_5, v_6 \rangle$

Offspring-chromosome S: $\langle v_1, v_2, v_3, u_4, u_5, u_6 \rangle$

4.3 Mutation

The combined operation of reproduction and crossover may sometimes lose potentially useful information from the chromosome. To overcome this problem, the mutation implemented by randomly complementing a bit (0 to 1 and vice versa) in some bit vector is introduced to ensure that good chromosomes are not permanently lost.

5. GA FOR MRDCP

This section describes the details of the GA developed to solve the MRDCP in WDM networks.

5.1 Chromosomal Coding Scheme

In our model, the *haploid chromosome* [17] is used to represent an individual's genotype. Since our problem is to find a candidate consisting of light paths between a source and all destinations, we employ a destination-oriented coding scheme that each individual consists of m chromosomes using positive integer numbers to represent m light paths between the source and m destinations. For an arbitrary k , the *chromosome* $C_i^k = \langle w_1^k, w_2^k, \dots, w_{|C_i^k|}^k \rangle$ in the individual $I_i = \langle C_i^1, C_i^2, \dots, C_i^m \rangle = \prod_{1 \leq k \leq m} C_i^k$ represents a light path from $w_1^k = s$ to $w_{|C_i^k|}^k = d_k$, where $|C_i^k|$ is the number of nodes, $d_k \in D$, and w_j^k is a node in C_i^k for $1 \leq j \leq |C_i^k|$. A subset of a chromosome is called a *gene*. $c(C_i^k) = \sum_{j=1}^{|C_i^k|-1} c(e_{w_j^k, w_{j+1}^k})$ and $d(C_i^k) = \sum_{j=1}^{|C_i^k|-1} d(e_{w_j^k, w_{j+1}^k})$ represent the communication cost and the transmission delay of C_i^k , respectively. In our model, because a routing-tree can be presented by an individual, the routing-tree viewed as the phenotype of I_i can be represented by $MSpT^c(\bigcup_{k=1}^m C_i^k, D)$.

Example 4: The four light paths from v_9 to v_0, v_5, v_{10} , and v_8 for $r(v_9, \{v_0, v_5, v_{10}, v_8\}, 3.3)$ shown in Fig. 2 (a) can be used to represent the chromosomes of the same individual I_j as follows:

$$C_j^1 = \langle v_9, v_6, v_4, v_5, v_0 \rangle, |C_j^1| = 5$$

$$C_j^2 = \langle v_9, v_5 \rangle, |C_j^2| = 2$$

$$C_j^3 = \langle v_9, v_2, v_{12}, v_3, v_{10} \rangle, |C_j^3| = 5$$

$$C_j^4 = \langle v_9, v_{10}, v_3, v_8 \rangle, |C_j^4| = 4$$

$$I_i = \langle C_j^1, C_j^2, C_j^3, C_j^4 \rangle = \prod_{1 \leq k \leq 4} C_j^k$$

□



Because a chromosome indicates a light-path between a source and a destination, each edge of the light path needs to correspond directly to a physical optical fiber in a WDM network. Nevertheless, when the evolution is in progress, the two new offspring chromosomes will be obtained by concatenating two genes of the parent's chromosomes in a crossover or mutation. The concatenation causes the offspring to own a *disturbed chromosome* that contains a nonexistent fiber-link. Therefore, an operation, *cat*, is used to concatenate and repair the disturbed chromosome to prevent the new offspring chromosome from using an invalid optical link. Formally, the concatenation of two genes, $g_i^{x_1, x_2} = \langle w_{x_1}^k, w_{x_1+1}^k, \dots, w_{x_2}^k \rangle \in C_i^k$ for $1 \leq x_1, x_2 \leq |C_i^k|$ and $g_j^{y_1, y_2} = \langle \bar{w}_{y_1}^k, \bar{w}_{y_1+1}^k, \dots, \bar{w}_{y_2}^k \rangle \in C_j^k$ for $1 \leq y_1, y_2 \leq |C_j^k|$ can be defined as

$$cat(g_i^{x_1, x_2}, g_j^{y_1, y_2}) = \begin{cases} \langle w_{x_1}^k, w_{x_1+1}^k, \dots, w_{x_2}^k, \bar{w}_{y_1+1}^k, \bar{w}_{y_1+2}^k, \dots, \bar{w}_{y_2}^k \rangle & \text{if } w_{x_2}^k = \bar{w}_{y_1}^k \\ \langle w_{x_1}^k, w_{x_1+1}^k, \dots, w_{x_2}^k, \bar{w}_{y_1}^k, \bar{w}_{y_1+1}^k, \dots, \bar{w}_{y_2}^k \rangle & \text{if } w_{x_2}^k \neq \bar{w}_{y_1}^k \text{ and } e_{w_{x_2}^k, \bar{w}_{y_1}^k} \in E, \\ cat(cat(g_i^{x_1, x_2}, P_x(w_{x_2}^k, \bar{w}_{y_1}^k)), g_j^{y_1, y_2}) & \text{if } w_{x_2}^k \neq \bar{w}_{y_1}^k \text{ and } e_{w_{x_2}^k, \bar{w}_{y_1}^k} \notin E \end{cases}$$

where $P_x(w_{x_2}^k, \bar{w}_{y_1}^k) \in \mathbf{P}(w_{x_2}^k, \bar{w}_{y_1}^k)$, called as a *repaired-gene*, is a randomly-chosen light-path between $w_{x_2}^k$ and $\bar{w}_{y_1}^k$. Because a chromosome describes a light-path from a source to some destination, the concatenation would appear on the two genes belonging to two chromosomes which route to the same destinations. Therefore, in the above definition, $g_i^{x_1, x_2}$ and $g_j^{y_1, y_2}$ need to belong to two chromosomes C_i^k and C_j^k with the same k , respectively.

Example 5: Suppose that two genes $\langle v_9, v_2 \rangle$ and $\langle v_3, v_8 \rangle$ are given. In Fig. 1 (a), $\langle v_9, v_2 \rangle$ and $\langle v_3, v_8 \rangle$ can not be concatenated into $\langle v_9, v_2, v_3, v_8 \rangle$ because the link (e_{v_2, v_3}) between v_2 and v_3 does not exist in $G(V, E)$. Therefore, some repaired-gene $\langle v_2, v_{12}, v_3 \rangle$ should be chosen arbitrarily from $\mathbf{P}(v_2, v_3)$ and so

$$\begin{aligned} cat(\langle v_9, v_2 \rangle, \langle v_3, v_8 \rangle) &= cat(cat(\langle v_9, v_2 \rangle, \langle v_2, v_{12}, v_3 \rangle), \langle v_3, v_8 \rangle) \\ &= cat(\langle v_9, v_2, v_{12}, v_3 \rangle, \langle v_3, v_8 \rangle) \\ &= \langle v_9, v_2, v_{12}, v_3, v_8 \rangle \end{aligned}$$

□

5.2 Crossover Operator

There are two types of crossover operators which are used randomly in the development of this GA method: (1) *Chromosome Crossover (CC)*, and (2) *Individual Crossover (IC)*. Suppose that the two individuals $I_i = \prod_{1 \leq k \leq m} C_i^k$ and $I_j = \prod_{1 \leq k \leq m} C_j^k$, and chromosomes $C_i^k = \langle w_1^k, w_2^k, \dots, w_{|C_i^k|}^k \rangle$ in I_i and $C_j^k = \langle \bar{w}_1^k, \bar{w}_2^k, \dots, \bar{w}_{|C_j^k|}^k \rangle$ in I_j are given for some k , where $w_1^k = \bar{w}_1^k = s$ and $w_{|C_i^k|}^k = \bar{w}_{|C_j^k|}^k = d_k \in D$. The two operators are defined as follows.

- *Chromosome Crossover (CC)*: Because the first nodes of C_i^k and C_j^k have the same source, two crossover points x and y ($x \leq y$) will be selected randomly from 2 to



$\min(|C_i^k|, |C_j^k|)$. These two offspring chromosomes \hat{C}_i^k and \hat{C}_j^k are described as follows.

Case 1. $x = y$

$$\begin{aligned}\hat{C}_i^k &= \text{cat}(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, \langle \bar{w}_x^k, \bar{w}_{x+1}^k, \dots, \bar{w}_{|C_j^k|}^k \rangle) \\ \hat{C}_j^k &= \text{cat}(\langle \bar{w}_1^k, \bar{w}_2^k, \dots, \bar{w}_{x-1}^k \rangle, \langle w_x^k, w_{x+1}^k, \dots, w_{|C_i^k|}^k \rangle)\end{aligned}$$

Case 2. $x < y$

$$\begin{aligned}\hat{C}_i^k &= \text{cat}(\text{cat}(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, \langle \bar{w}_x^k, \bar{w}_{x+1}^k, \dots, \bar{w}_y^k \rangle), \langle w_{y+1}^k, w_{y+2}^k, \dots, w_{|C_i^k|}^k \rangle) \\ \hat{C}_j^k &= \text{cat}(\text{cat}(\langle \bar{w}_1^k, \bar{w}_2^k, \dots, \bar{w}_{x-1}^k \rangle, \langle w_x^k, w_{x+1}^k, \dots, w_y^k \rangle), \langle \bar{w}_{y+1}^k, \bar{w}_{y+2}^k, \dots, \bar{w}_{|C_j^k|}^k \rangle)\end{aligned}$$

- *Individual Crossover (IC)*: Randomly select two crossover points x and y ($x \leq y$) from 1 to m . These two offspring \hat{I}_i and \hat{I}_j are described as follows:

Case 1. $x = y$

$$\begin{aligned}\hat{I}_i &= \langle C_i^1, \dots, C_i^{x-1}, C_j^x, \dots, C_j^m \rangle \\ \hat{I}_j &= \langle C_j^1, \dots, C_j^{x-1}, C_i^x, \dots, C_i^m \rangle\end{aligned}$$

Case 2. $x < y$

$$\begin{aligned}\hat{I}_i &= \langle C_i^1, \dots, C_i^{x-1}, C_j^x, \dots, C_j^y, C_j^{y+1}, \dots, C_i^m \rangle \\ \hat{I}_j &= \langle C_j^1, \dots, C_j^{x-1}, C_i^x, \dots, C_i^y, C_j^{y+1}, \dots, C_j^m \rangle\end{aligned}$$

5.3 Mutation

There are two types of mutation operators used in the development of this GAs: (1) *Chromosome Mutation (CM)*, and (2) *Individual Mutation (IM)*. The two operators are defined as follows:

- *Chromosome Mutation (CM)*: Randomly select two mutation points x and y ($x \leq y$) from 2 to $|C_i^k|$. The mutated chromosome \hat{C}_i^k of C_i^k is presented as follows.

Case 1. $x = y$

$$\hat{C}_i^k = \text{cat}(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, P(w_x^k, w_{|C_i^k|}^k)), \text{ where } P(w_x^k, w_{|C_i^k|}^k) \in \mathbf{P}(w_x^k, w_{|C_i^k|}^k)$$

Case 2. $x < y$

$$\begin{aligned}\hat{C}_i^k &= \text{cat}(\text{cat}(\langle w_1^k, w_2^k, \dots, w_{x-1}^k \rangle, P(w_x^k, w_y^k)), \langle w_{y+1}^k, w_{y+2}^k, \dots, w_{|C_i^k|}^k \rangle), \\ &\text{ where } P(w_x^k, w_y^k) \in \mathbf{P}(w_x^k, w_y^k)\end{aligned}$$

- *Individual Mutation (IM)*: Randomly select two mutation points x and y ($x \leq y$) from 1 to m . The mutated \hat{I}_i and \hat{I}_j is as follows:



Case 1. $x = y$

$$\hat{I}_i = \langle C_i^1, \dots, C_i^{x-1}, P(s, d_x), C_i^{x+1}, C_i^m \rangle, \text{ where } P(s, d_x) \in \mathbf{P}(s, d_x)$$

Case 2. $x < y$

$$\hat{I}_j = \langle C_i^1, \dots, C_i^{x-1}, P(s, d_x), P(s, d_{x+1}), \dots, P(s, d_y), C_i^{y+1}, \dots, C_i^m \rangle, \\ \text{where } P(s, d_z) \in \mathbf{P}(s, d_z) \text{ for } x \leq z \leq y.$$

The mutation not only ensures the population against permanent fixation at any particularity locus but also spoils the better chromosome. Since a chromosome is used to represent a light-path, the mutation operator implies that another new light-path (chromosome) would be established by randomly choosing a node called a *mutation node* in the chromosome (light-path), and rerouting the mutation node to another node called a *rerouting node*, where the *rerouting node* can be decided by using different *rerouting approaches*. In our GA model, the four different mutation heuristics, *Random Mutation (RM)*, *Cost First Mutation (CFM)*, *Delay First Mutation (DFM)*, and *Hybrid Mutation (HM)* which provide different rerouting approaches are described as follows:

- (1) *Random Mutation (RM)*: The rerouting node is chosen irregularly from the neighboring nodes.
- (2) *Cost First Mutation (CFM)*: The heuristic of *CFM* gives the population more evolutionary pressure so that the rerouting node is chosen according to the mutation probability of the link. The purpose is to decrease communication-cost in multicast cost function. All mutation probabilities of light-paths between the mutation node u and all neighborhood nodes are computed according to their communication costs such that the link with high communication-cost will have lower mutation probability. Therefore, a neighborhood node with high mutation probability has a greater chance to be chosen than those with lower mutation probability. In *CFM*, the mutation probability of the link between u and v_k , $Pr_{CFM}(u, v)$, is defined as:

$$Pr_{CFM}(u, v) = \frac{\frac{1}{c(P^c(u, v))}}{\sum_{x \in V} \frac{1}{c(P^c(u, x))}}$$

- (3) *Delay First Mutation (DFM)*: The *DFM* is similar to the *CFM* except for the definition of mutation probability. In *DFM*, the link with high transmission delay has a lower mutation probability. Therefore, the mutation probability of the link between u and v_k , $Pr_{DFM}(u, v)$, is decided by their transmission delay and defined as:

$$Pr_{DFM}(u, v) = \frac{\frac{1}{d(P^d(u, v))}}{\sum_x \frac{1}{d(P^d(u, x))}}$$



- (4) *Hybrid Mutation (HM)*: The *HM* is hybridizing the *RM*, *CFM*, and *DFM* to construct an intelligent routing approach according to the *feature* of chromosome, the *feature* includes two states, one is whether or not the transmission-delay greater than Δ the second is whether or not the communication-cost is minimum. For a given chromosome C_i^k , the *HM* adopts the appropriate mutation heuristic from *RM*, *CFM* or *DFM* according to the following rules:

$$\begin{aligned} \text{Adopt CFM} & : \text{if } c(C_i^k) > c(P^c(s, d_k)) \text{ and } d(C_i^k) \leq \Delta \\ \text{Adopt DFM} & : \text{if } d(C_i^k) > \Delta \\ \text{Adopt RM} & : \text{if } c(C_i^k) = c(P^c(s, d_k)) \text{ and } d(C_i^k) \leq \Delta \end{aligned}$$

5.4 Fitness Function Definition

Generally, GAs use fitness function to evaluate all individuals in a population and achieve the goal of finding optimal results. According to the phenotype of individual I_j , the corresponding routing-tree T_{I_j} , $T_{I_j} = MSPT^c(\bigcup_{k=1}^m C_j^k, D)$, could be decided by the chromosomes in I_j . It is not necessarily true that the corresponding routing-tree is a light-tree or a candidate. Since the goal is to find a feasible light-forest with minimizing multicast cost, the multicast cost function $f(T_{I_j})$ can be viewed as a fitness function associated with each individual. The object is *minimizing* $f(T_{I_j})$.

In our encoding schema, because each chromosome represents a light-path from a source to each destination, the destination constraint (5) in section 3 is always satisfied. If the delay constraint (6) in section 3 is considered in this fitness function, we would have a complex problem formulation. Also, the individual reflects that it a feasible solution, which is not required while breeding of chromosome in the GA. Thus, we need to attach a penalty by using a *penalty function* to the fitness function in the event that the individual is not a feasible solution. Nevertheless, because the demanded wavelength for the corresponding routing tree can be computed in (1) and separated into a light-forest by *Separating_Step* in [13], the degree constraint (7) is not discussed in the penalty function but in multicast cost function (4). Therefore, considering the delay constraint, the formulation above can be rewritten in another form:

$$\text{Minimize fitness}(T_{I_j}) = f(T_{I_j}) + \beta \cdot \text{Penalty}(T_{I_j}), \text{ subject to}$$

$$d(T_{I_j}) \leq \Delta$$

$$\text{where } \text{Penalty}(T_{I_j}) = \begin{cases} f(T_{I_j}) \cdot \exp\left(\frac{d(T_{I_j}) - \Delta}{\Delta}\right) & \text{if } d(T_{I_j}) > \Delta, \\ 0 & \text{otherwise} \end{cases}$$

and β is the penalty weight which should be greater than m .

In the Selection/Reproduction stage, each individual I_j has a probability $Pr(I_j)$ of being selected as the parent that is disproportional to their fitness because our object is to minimize the fitness value. Therefore, the $Pr(I_j)$ can be redefined as



$$Pr(I_j) = \frac{\frac{1}{fitness(T_{I_j})}}{\sum_{I_x \text{ in population}} \frac{1}{fitness(T_{I_x})}}$$

5.5 Chromosome Repair

Because the *cat*-operator needs to be used in crossover and mutation, the light path of the chromosome may be inefficient for containing a cyclic sub-path. Suppose that the repaired gene in Example 5 is $\langle v_2, v_1, v_9, v_{10}, v_3 \rangle$. The *cat* ($\langle v_9, v_2 \rangle, \langle v_3, v_8 \rangle$) = $\langle v_9, v_2, v_1, v_9, v_{10}, v_3, v_8 \rangle$ would contain a cyclic sub path $\langle v_9, v_2, v_1, v_9 \rangle$. Using the following *Chromosome-Repair* procedure, $\langle v_9, v_2, v_1, v_9, v_{10}, v_3, v_8 \rangle$ could be reduced $\langle v_9, v_{10}, v_3, v_8 \rangle$.

Chromosome-Repair ($C = \langle u_1, u_2, \dots, u_{|C_i^k|} \rangle$)
 // C is a chromosome, l is the number of links in C .
 // *starting*(e) and *end*(e) are starting node and end node of edge e in the order of light-path.
 {
 1. $e_x = e_1$, //where $ex = e_{u_x, u_{x+1}}$
 2. while($e_x \neq e_1$)
 3. if ($\exists e_y \in C, \exists e_x \neq e_y$ and *starting*(e_x) = *starting*(e_y))
 4. then
 5. removing e_z from C for $x < z \leq y$
 6. replacing e_x with $e_{starting(e_x), end(e_y)}$
 7. else
 8. $e_x = e_{x+1}$
 9. end while-loop
 }

5.6 Replacement Strategy

This subsection discusses a method used for creating a new generation after crossover and mutation are carried out on the individuals of the previous generation. There have been several replacement strategies proposed in the literature; a good discussion can be found in [17]. The most common strategies involve to probabilistically replacing the poorest performing individuals of the previous generation. The *elitist strategy* appends the best performing individual of a previous generation to the current population, thereby ensuring that the individual with the best fitness value always survives to the next generation.

The algorithm developed here combines both of the concepts mentioned above. Each offspring generated after crossover is added to the new generation if it has a better fitness value than both of its parents. If the fitness value of an offspring is better than only one of the parents, then we select an individual randomly from the better parent and the offspring. If the offspring is worse than both parents then either of the parents is se-



lected at random for the next generation. This ensures that the best individual is carried to the next generation, while the worst is not carried to the succeeding generations.

5.7 Termination Rules

The execution of GA can be terminated using any one of the following rules:

- R1: when the average and maximum fitness values are above a predetermined threshold;
- R2: when the average and maximum fitness values of strings in a generation become the same; or
- R3: when numbers of generations exceed an upper bound specified by the user.

The best value for a given problem can be obtained from a GA when the algorithm is terminated using R2 [17]. However, R3 is chosen in this paper.

6. SIMULATION

Our work focuses on how to find a near optimal light forest such that *destination*, *delay*, and *degree constraints* are satisfied. The approach used in this simulation to evaluate the performance of our GA model proposed in the previous sections can be referenced in Waxman [18]. In the approach, there are n nodes in the networks, which are distributed randomly over a rectangular grid, and are placed on integer coordinates. For a network topology generated for experimenting, each link between two nodes u and v is added with the probability function $P(u, v) = \lambda \exp(-p(u, v)/\gamma\delta)$, where $p(u, v)$ is the distance between u and v , δ is the maximum distance between any two nodes, and $0 < \lambda, \gamma \leq 1$. In the probability function, a larger value of λ produces networks with high link densities, while small value of γ increases the densities of short links relative longer ones. In our simulations, we use $\lambda = 0.7$, $\gamma = 0.9$, and size of rectangular grid = 50.

To reduce the complexity of the problem, the cost function c of link (u, v) in the network is the distance between u and v on the rectangular coordinate grid, and delay function d of link (u, v) is generated randomly between 0.1 and 3. For each request $r(s, D, \Delta)$, Δ is generated randomly and may be a small value such that it is impossible to find a candidate satisfying the delay constraint, or a huge value such that any routing-tree including the source and all destinations is a candidate. Therefore, to generate a request with a reasonable value of delay bound and to prevent the delay bound from being a huge value or a small value, we will choose the value according the maximum of the *CDLPs* between the source and all destinations. When the delay bound is set to have the value of the maximum, the candidate will be restricted to include the *CDLP* with maximum transmission delay. That is, setting Δ to be equal to some time of the maximum can give more discussion on the efficiency of GA method. In our observation, the depth of a candidate is greater than 5 for networks with more than 30 nodes; furthermore, the depth may be greater than 10 for some special request. Therefore, we assume that Δ equals to 1.2 times the maximum in the following experiments. Two different WDM networks, net_1 with 30 nodes ($n = 30$) and net_2 with 100 nodes ($n = 100$), are used to test the GA model. The experimental requests are generated randomly to simulate different requests



with a different number of destinations for net_1 and net_2 , i.e., the notation of “ $m = 4$ ” is used to represent a request with 4 destinations equal.

Several experiments consisting of three parts, efficiency of GA model, comparisons between GA, 3-Phase Model (3PM) [13], and *Integer Linear Programming (ILP)*, and comparisons among 4 mutation heuristics are described as follows. It is obvious that the number of population size and the number of generations will affect the execution time when evolution is in progress. The effects of population size (PS), generations, mutation probability (MP), crossover probability (CP), average multicast cost, and average CPU time for two networks net_1 and net_2 will discuss in the first part. According to experimental results in the first part, we will give a suggestion about suitable values for PS , generations, MP , and CP in the second part and the third part of this paper.

6.1 Efficiency of GA Model

Because the new routing problem is NP-Complete [20], the efficiency of GA is difficult to estimate, but can be compared with the well-known *Minimal Steiner Tree (MST)* problem. The *MRDCP* can be reduced to the *MST* problem by ignoring the delay constraint and the degree constraint. The candidate with minimum multicast cost could be equivalent to the Steiner Tree with minimum communication cost by setting $\alpha = 0$. The four types of comparison between the GA method and the *Minimal Distance Network Heuristic (MDNH)* [19] for *MST* can be discussed by choosing $\alpha = 0$ and $\beta = m$.

6.1.1 Effect of population size

The average number of required generations to obtain the routing-trees with equal or less multicast cost than the routing-trees obtained by *MDHN* [19] for different population sizes are shown in Figs. 4 (a) and (b). They were made by choosing m from $\{4, 5, \dots, 10\}$, PS from $\{100, 200, \dots, 500\}$ for net_1 and PS from $\{300, 400, \dots, 700\}$ for net_2 , $MP = 0.3$, and $CP = 1.0$. Therefore, we find that the GA model can always obtain a better routing-tree than *MDHN*'s. We observe the larger value for PS can reduce the number of generations, but it increases the execution time. Therefore, for the two networks, net_1 and net_2 , the population sizes are chosen to be 200 and 300 so that the GA achieve a better solution with less execution time.

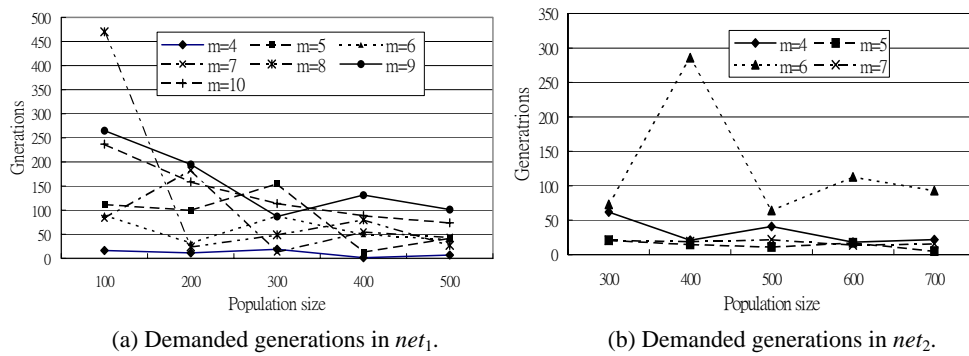


Fig. 4. Demanded generations for different populations.



6.1.2 Effects of generations

Because experiments in GA are time-consuming, it is necessary to examine the effects of generations in GA. We set $MP = 0.3$, $CP = 1.0$, maximum of generations is 1000 ($MG = 1000$), and m is selected from $\{4, 5, 6, 7, 8, 9, 10\}$. The experimental results of net_1 on $PS = 200$ and net_2 on $PS = 300$ shown in Figs. 5 (a) and (b) describe the relationship between generation and average multicast cost for different requests. The promoted percentage of multicast cost comparing GA with MDHN is shown in Fig. 6. From these results, we see that GA can find the best solutions at 600 generations at most for net_1 and net_2 . Nevertheless, when the number of generations is higher than 700, the averages of multicast cost are static and the computation for more generations is wasteful. Therefore, 2000 is chosen as the maximum number of generations ($MG = 2000$) for net_1 and net_2 .

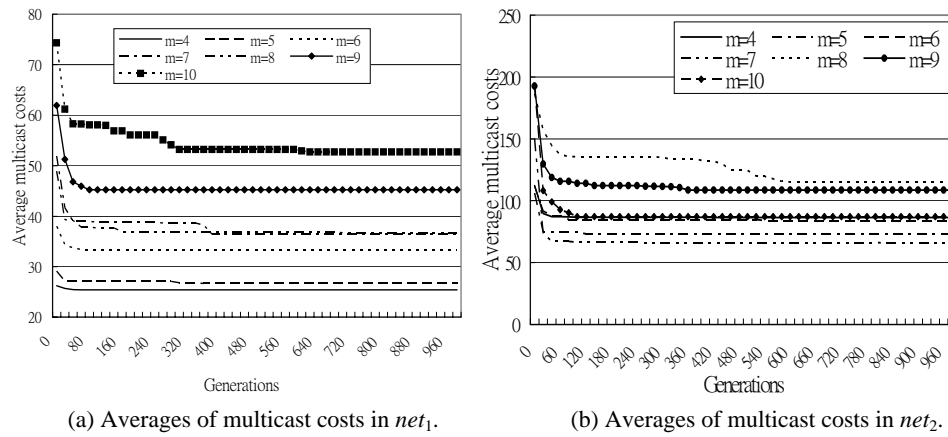


Fig. 5. Averages of multicast costs for different generations.

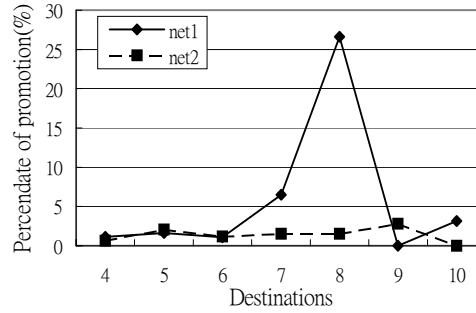


Fig. 6. Percentages of multicast cost promotion.

6.1.3 Effects of mutation probability

To examine the effects of the mutation probability (MP) of GAs, the same primitive individuals in the stage of population initialization are used for different test cases with



different MP s. We set $PS = 200$, $CP = 1.0$, $MG = 1000$, and selected MP from $\{0, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 1.0\}$. For 5 runs, the results shown in Figs. 7 (a) and (b) describe the averages of CPU time and the average numbers of generations for different MP s in net_1 , respectively. For these results, the better value of MP is different for different requests, and we found that $MP = 0.2$ can be chosen to gain better performance. Therefore, in our following experiments, $MP = 0.2$ is chosen.

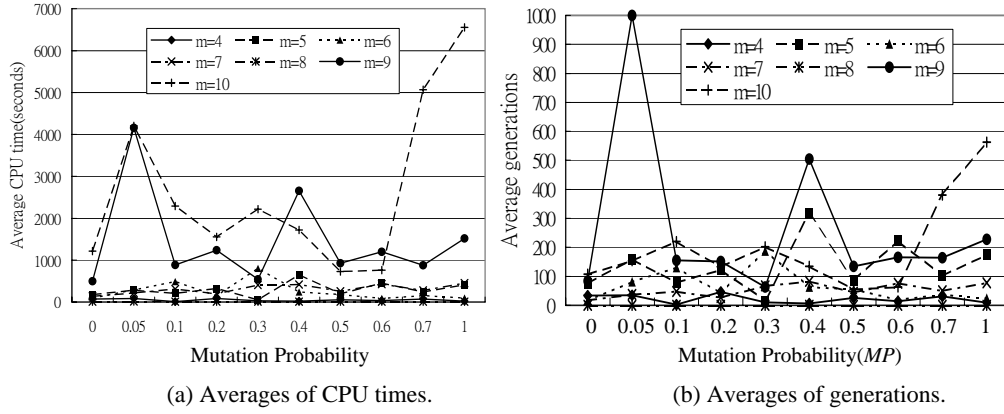


Fig. 7. Averages of CPU times and generations in net_1 .

6.1.4 Effect of crossover probability

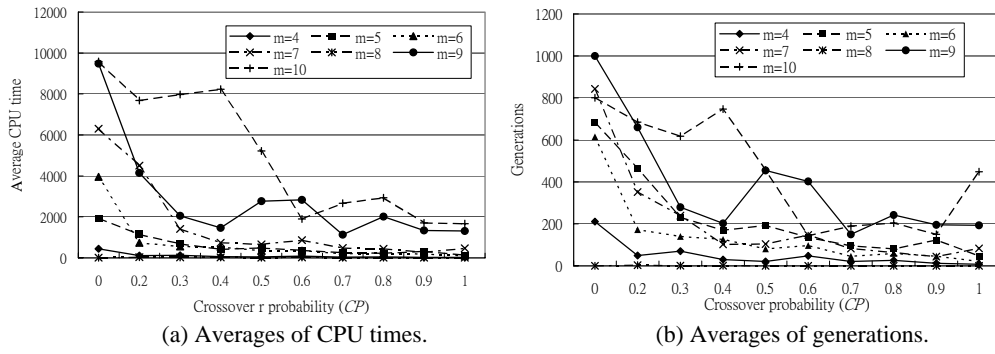
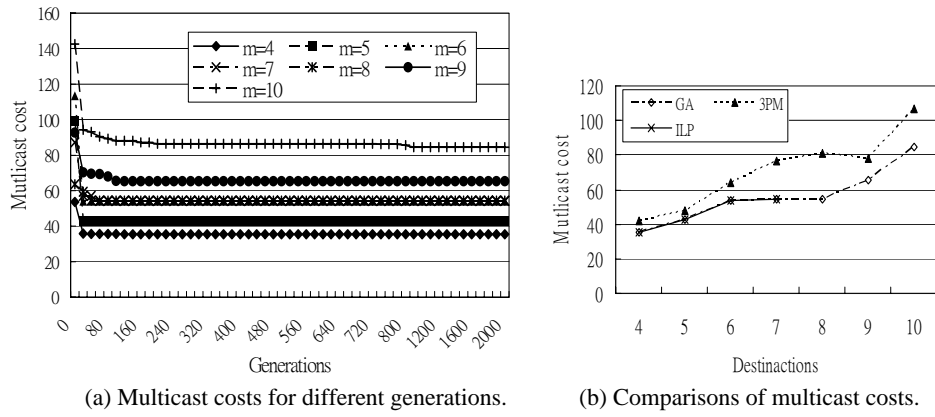
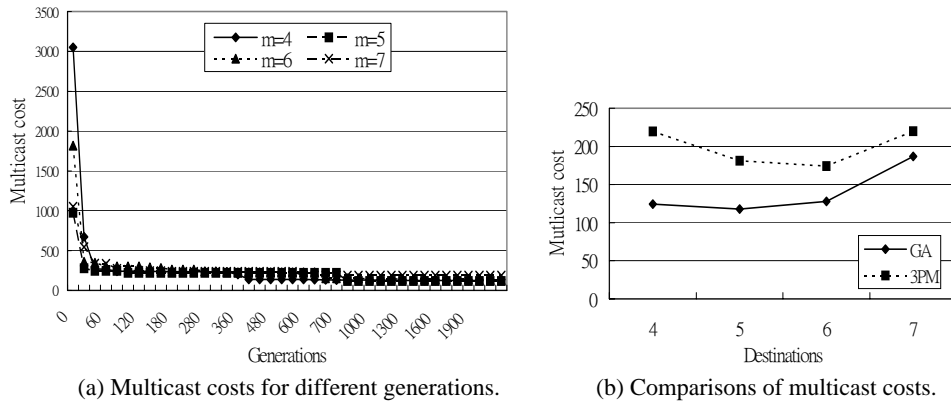
To examine the effect of the crossover probability (CP) of GAs, we set $PS = 200$, $MP = 0.2$, $MG = 1000$, and selected CP from $\{0.0, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$. After 5 runs, the results shown in Figs. 8 (a) and (b) describe the averages of CPU time and the averages of generations for different CP s in net_1 , respectively. For these results, the better value of CP is different for different requests, and we found that $CP = 1.0$ can be chosen to gain better performance. Therefore, in our following experiments, $CP = 1.0$ is chosen to gain the optimal solution.

In brief, the population sizes are chosen to be 200 and 300 for net_1 and net_2 , and $MG = 2000$, $MP = 0.2$, and $CP = 1.0$. Nevertheless, these parameter values might not be suitable for any given network.

6.2 Comparisons between GA, 3PM, and ILP

The experimental results of multicast costs for different generations and comparisons of multicast costs between GA, 3-Phase Model (3PM) [13], and ILP model for different requests are shown in Figs. 9 (a) and (b) for net_1 , and in Figs. 10 (a) and (b) for net_2 , respectively. Using the ILP model implemented by the linear programming tool ILOG's CPLEX, an optimal solution for routing the request with fewer than 8 destinations in net_1 can be found, but the optimal solution for routing the request with more than 8 destinations in net_1 or routing a request in net_2 can not be found in affordable execution time.



Fig. 8. Averages of CPU times and generations in net_1 .Fig. 9. Comparisons of multicast costs in net_1 .Fig. 10. Comparisons of multicast costs in net_2 .

As shown in Fig. 9 (a), the GA can find an equivalent light forest to the light forest found by ILP. The experimental results of promotion percentages of multicast cost between GA and 3PM and CPU time for GA, 3PM, and *ILP* are shown in Figs. 11 (a) and (b) for different requests, respectively. According to these results, we observe that GA can always find a better solution than 3PM, but the computation time is high, and the solution found by GA may be an optimal solution. Also, GA can improve the promotion percentages of multicast cost by more than 19.86% for *net*₁ and 29.94% for *net*₂. Nevertheless, because the execution time is proportional to the number of destinations in requests, the reduction in execution time is an important challenge for the GA method.

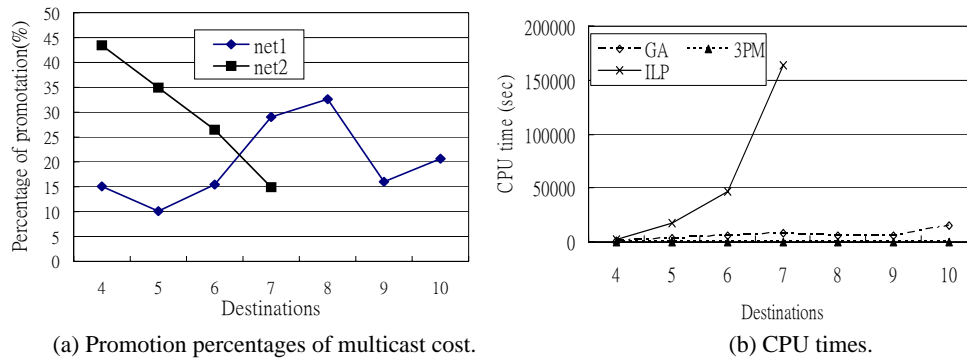


Fig. 11. Promotion percentages and CPU times for different requests.

6.3 Comparisons Among Four Mutation Heuristics

In this paper, four mutation heuristics are proposed to solve the *MRDCP*. Using the same primitive individuals in the stage of population initialization, comparisons of multicast costs for different generations and CPU times for different requests among the four mutation heuristics are shown in Figs. 12 to 13 for *net*₁. According to these experimental results for execution time analysis, we observe that the convergence of *CFM* is quicker than the other three heuristics and that *HM* needs less execution time.

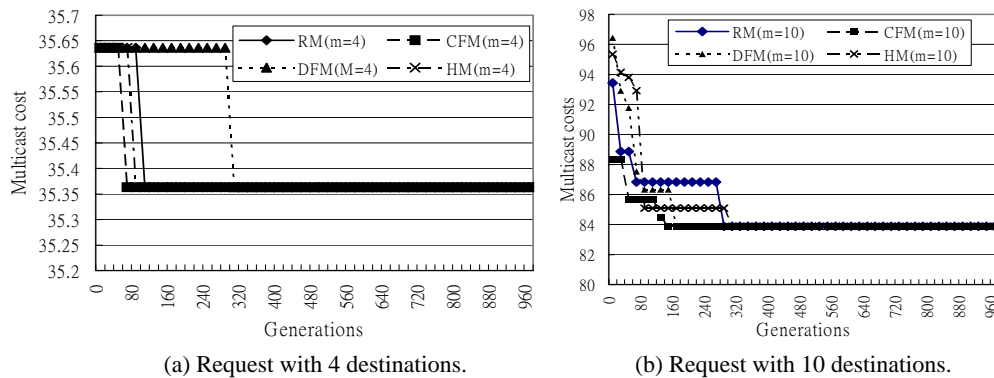


Fig. 12. Multicast costs for different generations in *net*₁.



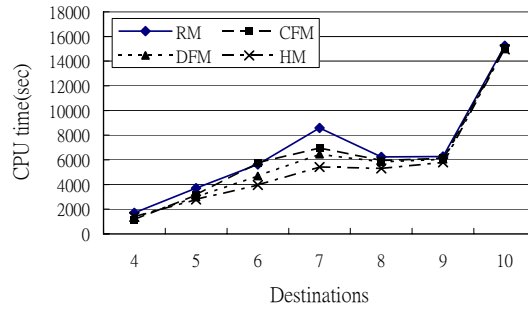


Fig. 13. CPU times for different requests in net_1 .

7. CONCLUSIONS

In this paper, a new formulation and a new multicast routing problem under delay constraint in *WDM* network with different light splitting (*MRDCP*) are studied. In the GA method, we also propose a destination-oriented representation used to represent a routing tree; three general genetic operators, selection, crossover, and mutation, were employed. Four types of operators (*Chromosome Crossover*, *Individual Crossover*, *Chromosome Mutation*, and *Individual Mutation*) and four mutation heuristics (*Random Mutation (RM)*, *Cost First Mutation (CFM)*, *Delay First Mutation (DFM)*, and *Hybrid Mutation (HM)*) are employed in our model. Experimental results indicate that the GA method can obtain a better solution than 3PM.

Because a *WDM* network, which has the capability of wavelength conversion, will provide more flexibility for routing requests, the cost of wavelength conversion needs to be evaluated by a fitness function to find an efficient light forest due to the overhead of wavelength conversion. Nevertheless, for a *WDM* network providing sparse wavelength conversion, an extra constraint describing a node with or without wavelength conversion needs to be included. Therefore, the problem is more difficult and may be solved by modifying the coding schema, crossover operator, and mutation operator. We are now trying to refine our solution model to solve two problems, routing a request in the network with sparse wavelength and routing multiple requests currently in the network without different light splitting.

REFERENCES

1. P. E. Green, *Fiber-Optic Networks*, Cambridge, MA, Prentice-Hall, 1992.
2. E. Lowe, "Current European WDM development trends," *IEEE Communications Magazine*, Vol. 36, 1998, pp. 46-50.
3. L. H. Sahasrabudde and B. Mukherjee, "Light-trees: optical multicasting for improved performance in wavelength-routed networks," *IEEE Communications Magazine*, Vol. 37, 1999, pp. 67-73.
4. D. Kosiur, *IP Multicasting: The Complete Guide to Interactive Corporate Networks*, Wiley Computer Publishing, 1998.



5. A. Ballardie, "Core based tree (CBT) multicast routing architecture," Internet RFC 2201, 1996.
6. B. K. Kadaba and J. M. Jaffe, "Routing to multiple destinations in computer network," *IEEE Transactions on Communications*, Vol. COM-31, 1983, pp. 343-351.
7. G. N. Rouskas and I. Baldine, "Multicast routing with end-to-end delay and delay variation constraints," *IEEE Journal on Selection Areas in Communications*, Vol. 15, 1997, pp. 346-356.
8. R. M. Karp, "Reducibility among combinational problems," in R. E. Miller and J. W. Thatcher (eds.), *Complexity of Computer Computations*, Plenum Press, New York, 1972, pp. 85-103.
9. X. Zhang, J. Wei, and C. Qiao "Constrained multicast routing in WDM networks with sparse light splitting," in *Proceedings of IEEE INFOCOM 2000*, Vol. 3, 2000, pp. 1781-1790.
10. N. Sreenath, N. K. M. Reddy, G. Mohan, and C. S. R. Murthy, "Virtual source based multicast routing in DWM networks with sparse light splitting," in *Proceedings of IEEE Workshop on High Performance Switching and Routing*, 2001, pp. 141-145.
11. X. H. Jia, D. Z. Du, and X. D. Hu, "Integrated algorithm for delay bounded multicast routing and wavelength assignment in all optical networks," *Computer Communications*, Vol. 24, 2001, pp. 1390-1399.
12. B. Chen and J. Wang, "Efficient routing and wavelength assignment for multicast in WDM networks," *IEEE Journal on Selection Areas in Communications*, Vol. 20, 2002, pp. 97-109.
13. M. T. Chen and S. S. Tseng, "Multicast routing under delay constraint in WDM network with different light splitting," in *Proceedings of International Computer Symposium (ICS 2002)*, Vol. 1, 2002, pp. 272-279.
14. R. C. Prim, "Shortest connection networks and some generations," *Bell System Technical Journal*, Vol. 36, 1957, pp. 1389-1401.
15. D. R. Din and S. S. Tseng "A genetic algorithm for solving dual-homing cell assignment problem of the two-level wireless ATM network," *Computer Communication*, Vol. 25, 2002, pp. 1536-1547.
16. E. Rolland, "Abstract heuristic search methods for graph partitioning," Ph.D. Dissertation, The Ohio State University, Columbus, Ohio, 1991.
17. L. Davis, (ed.), *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991.
18. B. M. Waxman, "Routing of multipoint connections," *IEEE Journal on Selection Areas in Communications*, Vol. 6, 1988, pp. 333-342.
19. L. Kou, G. Markowsky, and L. Berman, "A fast algorithm for Steiner trees," *Acta Informatica*, Vol. 15, 1981, pp. 141-145.
20. R. M. Karp, "Reducibility among combinational problems," in R. E. Miller and J. W. Thatcher (eds.), *Complexity of Computer Computations*, Plenum Press, New York, 1972, pp. 85-103.





Ming-Tsung Chen (陳明崇) received the B.S. degree in Mathematics from the National Cheng Kung University in 1988, and the M.S. degree in Computer Science and Information Engineering from the National Chiao Tung University in 1990. Since October 1990, he has served in the Chunghwa Telecom Co., Ltd. He is currently working toward the Ph.D. degree in Computer and Information Science from the National Chiao Tung University. His currently research interests are in algorithm, mobile communication, WDM networking, and data mining.



Shian-Shyong Tseng (曾憲雄) received his Ph.D. degree in Computer Engineering from the National Chiao Tung University in 1984. Since August 1983, he has been on the faculty of the Department of Computer and Information Science at National Chiao Tung University, and is currently a Professor there. From 1988 to 1992, he was the Director of the Computer Center National Chiao Tung University. From 1991 to 1992 and 1996 to 1998, he acted as the Chairman of Department of Computer and Information Science. From 1992 to 1996, he was the Director of the Computer Center at Ministry of Education and the Chairman of Taiwan Academic Network (TANet) management committee. In December 1999, he founded Taiwan Network Information Center (TWNIC) and is now the Chairman of the board of directors of TWNIC. He is also the President of SIP/ENUM Forum Taiwan and the Chair of organizing committee of Taiwan Internet Content Rating Foundation. His current research interests include data mining, expert systems, computer algorithm, and Internet-based applications.

