JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 25, 83-103 (2009)

Heuristic Algorithms for Finding Light-Forest of Multicast Routing on WDM Network

DER-RONG DIN
Department of Computer Science and Information Engineering
National Changhua University of Education
Changhua City, 500 Taiwan
E-mail: deron@cc.ncue.edu.tw

Wavelength Division Multiplexing (*WDM*) is an important technique to make use of the large amount of bandwidths in optical fibers to meet the bandwidth requirements of applications. In this paper, two new models (*MWDCRP* and *MCRP*) of multicast routing on WDM networks are studied. In these models, it is assumed that each switch on WDM network can perform 'drop,' 'continue' and 'drop and continue' operations. In MWDCRP, given the multicast request and the delay constraint, the goal is to find a minimal wavelength *light-forest* to route the multicast request under the delay constraint. In MCRP, the objective cost of the multicast routing problem has two components: one is the transmitting cost, the other is the number of used wavelengths. Given the WDM network and the multicast request, the goal is to find a minimal cost *light-forest* to route the multicast request. Since these problems are NP-hard, four heuristic algorithms (named as *Maximal-Delay-First* (*MDF*), *miNimal-Delay-First* (*NDF*), *Farthest-Greedy* (*FG*), and *Nearest-Greedy* (*NG*)) are proposed to solve these problems. Simulation results demonstrate that the proposed algorithms can generate good solutions.

Keywords: light-forest, multicast routing, wavelength division multiplexing (WDM), heuristic algorithm, shortest-path based

1. INTRODUCTION

1.1 WDM

Wavelength Division Multiplexing (*WDM*) is an important technique to make use of the large amount of bandwidths in optical fibers to meet the bandwidth requirements of applications. There is a growing consensus that the next generation Internet will employ an IP-over-WDM architecture [1]. In this architecture, IP routers are attached to the optical cross-connects (OXCs) and IP links are realized by *light-paths* in a WDM network. Communication requests are transmitted by sending packets on a dedicated wavelength of the light-path between the source node's transmitter and the destination node's receiver [2, 3]. Because there is no electro-optic (E/O) or optic-electronic (O/E) conversion in OXCs, all-optical networks greatly increase the throughput capacity [4]. In addition, because the network requires not only transmission line capacity enhancement, but also cross-connect node processing capability enhancement. The WDM should be used in combination with wavelength routing [5, 6].

In wavelength routing, data signals are carried on a unique wavelength from a source node to a destination node passing through nodes where the signals are optically

Received April 13, 2007; revised August 10, 2007 & May 12, 2008; accepted June 26, 2008.
Communicated by Takeshi Tokuyama.

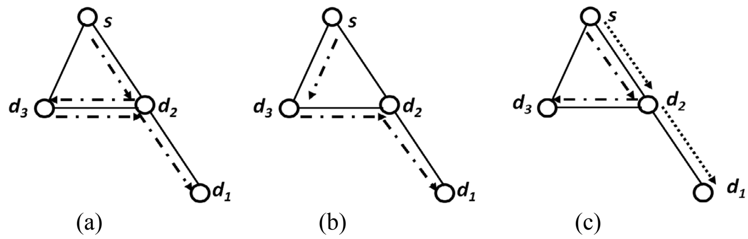
routed and switched without regeneration in the electrical domain. When a physical network is given and connections among the nodes in the network are required, an optical path (*light-path*) with a dedicated wavelength for each required connection should be established. The *Routing and Wavelength Assignment* (*RWA*) problem is a problem to select suitable paths and wavelengths among the many possible choices for the required connections [7].

If sufficient network resources are available at the time the request arrives, the routing path is established; otherwise the call request is blocked. If the nodes in WDM networks do not have wavelength converters then as a consequence the path must maintain the same wavelength throughout the entire path. In the WDM research community, this is known as the *wavelength continuity constraint* [5]. By practical limitations on transmission technology, the number of available wavelengths on a fiber is restricted. So, a good solution to the RWA problem is important to increase the efficiency of the WDM networks.

In order to solve various applications on WDM networks, mechanisms must be developed to handle not only point-to-point communications but also *multicast*. Multicast is the transmission of information from one source to multiple destinations simultaneously, for example, a one-to-many communication technique. Many broadband services such as video conferences, distance learning, and web casting employ multicast for data delivery. The support of multicast in future WDM networks is essential for these applications. Thus, issues concerning supporting multicast on WDM networks need to be studied.

1.2 Multicast

Recently, several researchers studied the multicast problem over WDM network [8-13]. Several comprehensive surveys have been done in [13-15]. Several researchers studied the multicast problem on WDM with all or sparse *multicast capable* (*MC*) [8, 12]. But the drawback of the problem studied in [8, 12] is that the MC node is expensive. To overcome this drawback, Ali and Deogun have proposed a low-cost novel architecture called *Tap-and-Continue* (*TaC*), shown in Fig. 1, for realizing multicast [9]. This architecture provides a natural evolution from current unicast cross-connects and is based on *tapping devices*. The proposed device can reduce the cost of MC cross-connects at the expense of more fiber links used in the routing structure. In the TaC cross-connect, optical signals are passed through a set of *Tap-and-Continue Modules* (*TCMs*). In a TCM, an extremely small fraction of the input signal is tapped and forwarded to the local station. The remaining power on the order of 99.9%, is switched to any one of the other outputs [9].



(a) (b) (c)

Fig. 1. (a) Example of trial; (b) Light-path; (c) Light-trees.

In [9], Ali and Deogun have defined and formulated the problem of routing a multicast session in a network equipped only TaC cross-connects, *Multiple-Destination Minimum-Cost Trail* (*MDMCT*) problem and showed the NP-completeness by reducing the well-known *Directed Hamiltonian Path* [16] problem to it. Informally, the routing optimization problem is that of finding an optimal *trail* that starts from a source node and visits all nodes in a nonempty set D of destinations with the objective of minimizing the number of directed edged traversed. A heuristic algorithm *multiple-destination trail* (*MDT*) was proposed to find the feasible routing of the multicast request. In [9], authors also showed that MDMCT problem always has a feasible solution in bidirectional graph. In [9], the route from source to destination is constructed as a *trial*, but this is impractical. For example, consider the network shown in Fig. 1 (a), where node s is the source and nodes d_1, d_2, and d_3 are the destinations. The trial $s \to d_2 \to d_3 \to d_2 \to d_1$ goes through d_3 by way of passing directed link (d_2, d_3) and immediately forwards the light back to d_2 through the reverse link (d_3, d_2), which is impractical. Moreover, consider the example for the path $s \to d_3 \to d_2 \to d_1$ (as shown in Fig. 1 (b)), this path uses one wavelength but the delay between source and the farthest destination may be too large.

In the WDM networks with TaC nodes, the trial can be replaced by a *light-tree* which consists of light-paths $s \to d_2 \to d_3$ and $s \to d_2 \to d_1$, as shown in the Fig. 1 (c). After this replacement, the delay between the source and the farthest destination can be reduced. But on the other hand, the wavelength usage is greater than the original trial. From this, there is a trade-off between cost and wavelength usage.

In [10] and [11], another version of multicast problems on the WDM network model was studied. Nodes in the WDM network are further subdivided into four types [10, 11]: *Drop and Continue* node (*DaC*-node), *Wavelength Conversion* node (*WC*-node), *Light-Splitting* node (*split*-node), and *Virtual Source* (*VS*). A *DaC*-node, which is the same as the TaC node in [9], is capable of tapping a small amount of optical power from the wavelength channel and transmitting the remainder. The VS is capable of both split-node and WC-node. The VS plays a key role in the construction of a multicast forest. A heuristic algorithm is proposed in [11] to construct the *multicast tree* to route the multicast request.

Light-tree [17] is a point-to-multipoint extension of light-path aiming to provide an underlying infrastructure for multicast in the optical network. Light-tree scheme uses extensively the light splitting capability of each node. Like the light-path, there is no O/E/O conversion at any intermediate node on a light-tree. It assumes that all nodes have adequate light splitting capacity, which makes up the full light splitting network. At each branching point, the beam of laser light will be split into a certain number of sub-beams, which turns a light-path into a light-tree. If all nodes in a network are MC nodes, one light-tree may be sufficient for routing data to all destinations; otherwise, a set of light-trees, aggregated as a light-forest, may be required for the network with sparse light splitting in which some of the nodes are MC nodes. In this case, several light-trees rooted at the same source node, is to be used, forming a *light-forest* (or multicast forest) [18].

In this paper, the multicast routing problem on WDM networks, whose nodes are TaCs, is studied. Given the WDM network and a multicast request, the goal is to find a *light-forest* which consists of one or more light-trees rooted at the multicast source and to their destinations, in such a way that the objective can be minimized. The usual goals to the multicast routing optimization problem [19] in traditional networks are the minimization of: (1) the path delay [20] due to blocking and (2) the cost of the path tree to reach the destinations. Often, the total cost minimization of the routing tree of networks with data replication is equivalent to the classical NP-hard problem: the *minimum Steiner tree* [16]. Two WDM multicast problems are studied in this paper. Because finding optimal solutions to these NP-hard problems are impractical, and that performing exact searches for optimal solutions are impractical due to exponential growth in execution time. In this paper, for each problem model, two heuristic algorithms are proposed to construct a light-forest for a given multicast session so that the multicast data can be delivered to all the members of the session.

The rest of this paper is organized as follows. In section 2, the basic assumptions and formulations of the problems are given. For the MWDCRP, two heuristic algorithms (MDF and NDF) are proposed and described in section 3. In section 4, for the MCRP problem, two heuristic algorithms (*Farthest-Greedy* and *Nearest-Greedy*) are proposed. Experimental results are given in section 5. Finally, conclusions are given in section 6.

2. PROBLEM FORMULATION

2.1 Network Model

In this paper, all nodes in a WDM network are *TaC* nodes. The functions of the *TaC* nodes are listed as follows [21]:

- *drop only*: When the locally attached router is a destination and there is no need to forward a copy to any downstream node.
- *continues only*: When the locally attached router is not a destination and there is a down-stream destination.
- *drop and continue*: When the locally attached router is a destination and there is a down-stream destination.

For the given WDM network, the unique source node has the function that it is possible to send multiple "copies" to the same output using different wavelengths along different paths. That is, the source node of the multicast request has multiple transmitters, and therefore packets can be transmitted to as many children as needed when constructing a multicast tree, rooted at it. Similarly, a source can transmit to its children on different wavelengths using different transmitters.

2.2 Problem Definition

A WDM network is represented by an undirected and connected graph $G(V, E, c, d, w)$, where V is the node-set of G which represents 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 two directions. Each edge e in E is associated with cost $c(e)$ and delay $d(e)$ which represent the communication cost and delay of edge e, respectively. Let w be the number of wavelengths provided by the WDM network. Functions $c(P(u, v))$ and $delay(P(u, v))$ are additive over the edges of a light-path $e \in P(u, v)$ between two nodes u and v, as shown in Eqs. (1) and (2):

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

$$delay(P(u,v)) = \sum_{\forall e \in P(u,v)} d(e). \tag{2}$$

In this paper, a multicast request $r(s, D)$ (or $r(s, D, MAX\_DELAY)$) is given, the request is for setting up a multicast channel from the source node s to a group of destinations $D = \{d_1, d_2, \ldots, d_{|D|}\}$, where $s \in V$, $s \notin D$, and $D \subseteq V$. For the given WDM network $G(V, E, c, d, w)$, it can be viewed as the union of a set of *wavelength networks* $G_k(V, E)$, $k = 1, 2, \ldots, w$. Let $T_k(s, D_k)$ be the light-tree for the request $r(s, D_k)$ on the kth wavelength network, where k is in $\{1, 2, \ldots, w\}$ and $D_k \subseteq D$. That is, the set D of destinations can be partitioned into several disjoint subsets, $D = \bigcup_{k=1,2,\ldots,w} D_k$ and $D_i \cap D_j = \emptyset$, for $i \neq j$; $i, j \in \{1, 2, \ldots, w\}$. Light-tree is a tree united by a set of light-paths with same source node on the same wavelength network. Let $T = \bigcup_{k=1,2,\ldots,w} T_k$ denote the *light-forest* for the request $r(s, D)$. The cost of a light-tree $T_k(s, D_k)$ on the kth wavelength network is defined as the sum of the cost of all edges in the light-tree $T_k(s, D_k)$. It can be formally defined as

$$c(T_k(s, D_k)) = \sum_{\forall e \in T_k(s, D_k)} c(e). \tag{3}$$

The delay of a light-tree $T_k(s, D_k)$ on the kth wavelength network is defined as the maximum of the delay of all destinations in the light-tree $T_k(s, D_k)$. It can be formally defined as

$$delay(T_k(s, D_k)) = \max_{\forall v \in D_k} delay(P(s, v)). \tag{4}$$

Light-forset is a tree united by a set of light-trees with same source node on different wavelength networks. Similarly, the cost and delay of the light-forest T are defined as

$$c(T) = \sum_{k=1}^{w} c(T_k(s, D_k)), \tag{5}$$

$$delay(T) = \max_{k=1,2,\ldots,w} delay(T_k(s, D_k)). \tag{6}$$

In this paper, the input optical signal of a TaC node can only be forwarded to an output port leading to its child. A TaC node in WDM network can perform a "drop and continue" function to 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. Therefore the degree of node, expect the source node, of light-tree on each wavelength network is less than or equal to 2 and this is denoted as the *light-splitting constraint*.

Another important goal of the model is to minimize the number of wavelengths to serve the multicast request without causing wavelength conflicts. Since wavelength is one of the most important resources in the WDM network, if the number of used wavelengths can be reduced, the consequent multicast requests can find the light-tree or light-forest easily. Thus, the blocking probability of the multicast request can be reduced. Otherwise, the multicast request will be blocked or rejected. In the equation described above, if the number of wavelengths used to route the multicast request are not restricted, the routing path may tend to consume more wavelengths.

In this paper two problems named the *Minimal Wavelength Delay Constraint Routing Problem* (*MWDCRP*) and the *Minimal Cost Routing Problem* (*MCRP*) are studied.

2.2.1 MWDCRP

In MWDCRP, given a WDM network $G(V, E, c, d, w)$ and a multicast request $r(s, D, MAX\_DELAY)$, where MAX_DELAY is the delay constraint of each routing path $P(s, d_i)$. The goal is to find the routing trees (or tree) and assign wavelengths to them such that the number of the used wavelengths can be minimized under the delay constraint. The cost function $c(e)$ of each link e in WDM is ignored. Let $y_k = 1$, if wavelength k is used by the light-forest T; $y_k = 0$, otherwise. Define $V(T)$ be the set of nodes of the tree T. The objective cost can be formulated as:

$$\sum_{k=1}^{w} y_k \tag{7}$$

subject to $D \subseteq V(T)$ and $delay(T) \leq MAX\_DELAY$. (8)

In the past, the *constrained Steiner tree problem* of constructing low cost trees with bounded end-to-end delay constraints in general network was studied by Kompella *et al.* [22] known to be NP-complete. Moreover, the routing and wavelength assignment (RWA) problem in WDM network is also NP-complete [7]. In MWDCRP, both the routing paths and the wavelength assignments of the source and destinations should be considered, and the *light-splitting constraint* is used to constrain the routing paths. Thus, the RWA in WDM network are special cases of MWDCRP. This means that the MWDCRP in WDM networks is an *NP-hard* problem.

2.2.2 MCRP

In MCRP, given a multicast request $r(s, D)$ and a WDM network $G(V, E, c, d, w)$, the goal is to find the routing trees (or tree) and assign wavelengths to them such that the objective cost can be minimized. The objective cost has two components: the first one is the cost of the light-forest $c(T)$ and the other is the number of used wavelengths. In the WDM networks, two paths must be assigned different wavelengths if their routes share a common link. Define binary variables y_k, for wavelength $k = 1, 2, \ldots, w$; $y_k = 1$, if wavelength k is used by the light-forest T; $y_k = 0$, otherwise. Thus, the total objective cost of the MCRP can be defined in Eq. (9):

$$c(r(s, D)) = c(T) + \alpha \times \sum_{k=1}^{w} y_k, \tag{9}$$

where α is the ratio of the cost of light-forest to that of the wavelengths used.

To reduce the routing cost, rerouting the path to destination on a whole new wavelength network is a good approach, but this may increase the number of used wavelengths. Wavelength is an important resource of WDM, thus, there is a tradeoff between the choices of a routing path on used and new wavelengths. In the real applications, the value of α may significantly affect the final routing solution. If set larger α, it may prefer to find the longer routing path on the same wavelength network instead of shorter routing path on a new wavelength network. Otherwise, it may prefer to find the shorter routing path on a new wavelength network. The value of the α can be set by the network manager and is depended on the number of wavelengths provided by the WDM network or the traffic pattern. If the number of wavelengths provided by the WDM network is large or the network traffic is low, the α should be set to a smaller value (about a positive real that is less than one half of the summation of cost of all edges in the network); otherwise, α should be set to a larger value (about a real that is one to two times of the summation of cost of all edges). Because both the routing paths and the wavelength assignments are considered in the multicast problem in WDM networks, the multicast routing optimization problem in general networks is a special case of MCRP. Moreover, for the WDM multicast routing problems studied in [18, 23] which were NP-complete are also special cases of MCRP, this means that the MCRP in WDM networks is an *NP-hard* problem.

2.3 Light-splitting Constraint

Because a TaC node in WDM network cannot serve as a branching node of the multicast tree and a VS node can splitting signal to any outgoing link, the degree of node of light-tree on each wavelength network should satisfy this constraint denoted as the *light-splitting constraint*. A key observation is that, due to the light-splitting constraint of the WDM node, a single light-tree may not be sufficient to transmit the multicast message to all destinations in a multicast session. Consider the example shown in Fig. 2, a WDM network with 18 nodes and a multicast request $r(s, D) = r(s, \{1, 2, 3, 4, 5, 6\})$ are given. The number near the edge represents the cost of edge. When the shortest path heuristic algorithm is used to find the shortest tree from s to D, node 7 is used to forward data to nodes 1, 2, and 3 as the bold lines show in Fig. 2. This shortest path tree violates the *light-splitting constraint*. To solve this problem, path $P(s, 3)$ or $P(s, 2)$ may be reassigned another wavelength. Once destinations 2 and 3 are reassigned to other wavelengths, source

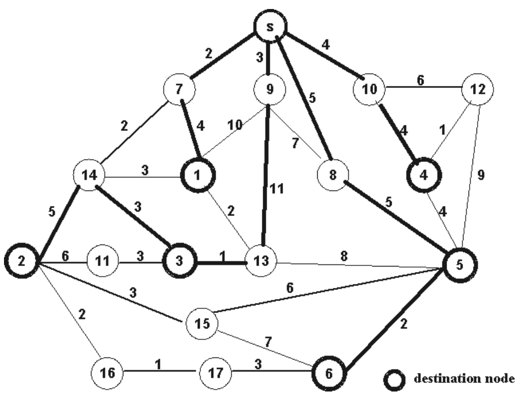


Fig. 2. Example 1 and the shortest path tree P.

node s needs to send out three "copies" on link (s, 7), and three wavelengths are needed on link (s, 7) in a wavelength-routed network. Alternate, paths $P(s, 2)$ and $P(s, 3)$ may be rerouted to satisfy the light-splitting constraint. So, rerouting algorithms should be developed. Note that, for the same multicast request, using different heuristics will likely result in light-forests with different costs.

3. HEURISTIC ALGORITHMS FOR MWDCRP

Given a multicast request, the goal is to find a light-forest such that the number of used wavelengths can be minimized under the delay constraint. It is worth noting that the cost $c(e)$ of link e on WDM network is ignored. Because the MWDCRP is NP-hard, in this section, two heuristic algorithms are proposed to solve it. These proposed algorithms are *shortest-path based*, that is, first, the shortest path algorithm is used to find the minimal delay paths from source to all destinations. Then these paths are modified to satisfy the delay and light-splitting constraints.

Given the WDM network $G(V, E, c, d, w)$ and the multicast request $r(s, D, MAX\_DELAY)$, for each destination node d_i in D find the minimal delay path $P(s, d_i)$ from source node s to d_i in G. Then, paths $P(s, d_i)$, $i = 1, 2, \ldots, D$ are combined to form P, such that $P = \bigcup_{i=1}^{|D|} \{P(s, d_i)\}$. It is worth noting that P should be a tree. If $delay(P) > MAX\_DELAY$ then the light-forest which satisfies the delay constraint cannot be found. In this paper, it is assumed that there exists a feasible solution which satisfies the delay constraint. If P satisfies the light-splitting constraint and $delay(P) \leq MAX\_DELAY$, then P uses only one wavelength and will be the optimal *light-tree* of the multicast request. Otherwise, two heuristic algorithms are used to modify the tree P to light-tree or light-forest which satisfies the light-splitting constraint.

Before introducing the proposed tree-modified algorithm, some notations are introduced and stated here. Let $Edge(P(s, d_i))$ be the set of edges that compose the path $P(s, d_i)$ and all the edges with at least one of its endpoints, not be s, be on $P(s, d_i)$. Let $G' = G \backslash Edge(P(s, d_i))$ be the remaining graph by removing edges and nodes in $Edge(P(s, d_i))$. For example, assume G be the network shown in Fig. 2. 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 $v_1, v_2, \ldots, v_{degree(s)}$ be the nodes in P which are directly adjacent to the source node s, the sub-tree rooted at v_i is represented by $PT(v_i)$, for $i = 1, 2, \ldots, v_{degree(s)}$. This is due to the routing tree in a wavelength network can not contain cycle(s) and must satisfy the light-splitting constraint. Thus, it is easy to find that if there is more than one destination on the leaves in one of the sub-trees, only one of these destinations can be selected to route by this sub-tree and the other destinations should be rerouted. Let $dest(v_i)$ and $mindest(v_i)$ be the destination node with maximal delay and minimal delay in sub-tree $PT(v_i)$, respectively. Let $Live(dest(v_i))$ ($Live(mindest(v_i))$) be the set of edges whose one endpoint is $dest(v_i)$ ($Live(mindest(v_i))$) and the other endpoint is in G'. Let *UNREACH* be the set of all destinations that have not been routed and which are sorted in the descending order according to the minimal delay of path from source s to destinations in G. Let $G_1, G_2, \ldots, G_w$ be the w *wavelength networks* of a WDM network and initially $G_1 = G_2 = \ldots = G_w = G$.

To determine the rerouting paths of the destinations in the leaves, two heuristic methods are developed. They are *Maximal-Delay-First* (*MDF*) and *miNimal-Delay-First* (*NDF*).

3.1 *Maximal-Delay-First* (*MDF*)

For each sub-tree $PT(v_i)$, $i = 1, 2, \ldots, v_{degree(s)}$, only the destination node $v \in PT(v_i)$ with maximal delay $delay(P(s, v)) \geq \max_{u \in PT(v)}\{delay(P(s, u))\}$ is routed by the path $P(s, v)$ on $PT(v_i)$, the others should be rerouted.

Consider the network shown in Fig. 2 (assume weight of edge $e \in E$ represent the $d(e)$), they are three destinations in the sub-tree $PT(7)$ with nodes 1, 2, and 3. The maximal delay destination is the node 2, thus the routing paths to destination nodes 1 and 3 should be rerouted. Similarly, the maximal delay destinations in the sub-trees $PT(8)$ and $PT(10)$ are nodes 6 and 4, respectively. Now, the problem is "how to find the reroute paths?" because the routed paths and nodes used by the maximal delay path in each sub-tree cannot be used twice in the same wavelength network. Moreover, the rerouting paths together with the existing paths cannot form a cycle or cycles or violate the light-splitting constraint. Thus, the path $P(s, v)$ to the maximal delay destination and the $Edge(P(s, v))$ in each sub-tree should be removed from G to construct G'. For each subtree $PT(v_i)$, a subgraph is constructed by $G' \cup Live(dest(v_i))$. It is worth noting that the delay of the routing path, which is called the *extended path*, routed through node $dest(v_i)$ should be included in computing the delay of the path $P(s, dest(v_i))$.

According to above discussions, the rerouting path can be determined to be more complex. The Maximal-Delay-First (MDF) Algorithm is described as follows:

Algorithm Maximal-Delay-First (MDF)
Step 1: Let $k = 1$, $G_1 = G_2 = \ldots = G_w = G$. For each destination d_i in D find the minimal delay path $P(s, d_i)$. Combine the 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 $i = 1, 2, \ldots, degree(s)$, find the maximal delay destination node $dest(v_i)$, the routing path $P(s, dest(v_i))$ and the $Edge(P(s, dest(v_i)))$. Modify G_1' by removing edges in $Edge(P(s, dest(v_i)))$, $i = 1, 2, \ldots, degree(s)$ from G_1 and construct *UNREACH*. For each maximal delay destination $dest(v_i)$ in subtree $PT(v_i)$, find $Live(dest(v_i))$. While UNREACH is non-empty and k is less than or equal to w, perform step 3.

Step 3: Take the maximal delay destination in *UNREACH*, say v, find the minimal delay paths in the set of subgraphs $SG = \{G_z'$ and $G_z' \cup Live(dest(v_i))$, for all v_i, $z = 1, 2, \ldots, k\}$. Find the minimal delay path in the path set *SG*. If a constraint-satisfied path is found, assign path $P(s, v)$ to the corresponding wavelength z; subtract $Edge(P(s, v))$ from G_z'; remove the destination v and other destinations passed by path $P(s, v)$ from *UNREACH*; and update corresponding $Live(dest(v_i))$. Otherwise, increasing k by one, if k is greater than w will return FALSE.

The process of applying MDF algorithm to the example of Fig. 2 (assume *MAX_*

DELAY = 15) is illustrated as follows: First, remove edges in *Edge*(*P*(*s*, 2)), *Edge*(*P*(*s*, 6)), and *Edge*(*P*(*s*, 4)) from G_1, that is, find $G_1' = G_1$-*Edge*(*P*(*s*, 2))-*Edge*(*P*(*s*, 6))-*Edge*(*P*(*s*, 4)). The remaining graph is shown in Fig. 3 (a), where *UNREACH* = {3, 1}. Because destination 3 is the maximal delay destination in UNREACH, the path $P^1(s, 3)$ from *s* to 3 in G_1' is $s \to 9 \to 13 \to 3$ with a delay of 15 and the extended path $P^1(2, 3)$ is $2 \to 11 \to 3$ with a delay of 18 (9 + 9 = 18) (as shown in Fig. 3 (b)). The extended path violated the delay constraint, thus, destination 3 is routed by $s \to 9 \to 13 \to 3$. Further, after removing *Edge*($P^1(s, 3)$) from G_1', because there is no path can be found from source to the destination 1, another wavelength network (G_2) is needed to reroute the path. Thus, consider the wavelength graph $G_2 = G$, the minimal delay path from *s* to 1 is $s \to 7 \to 1$ with the delay of 6. Final, the light-forest with 2 used wavelengths is shown in Fig. 3 (c).

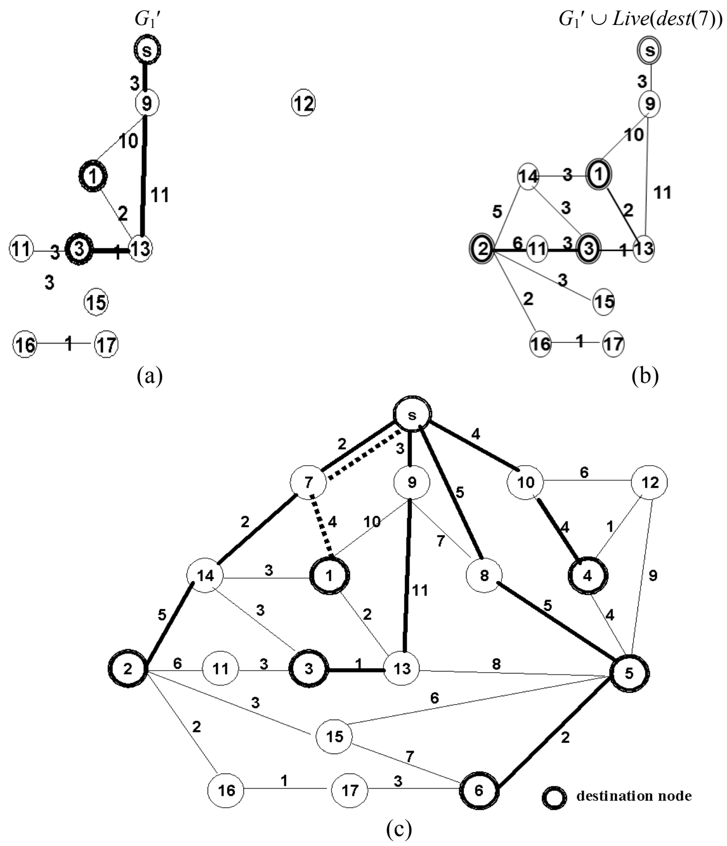


Fig. 3. (a) G_1'; (b) $G_1' \cup Live(dest(7))$; (c) Final result by applying the MDF algorithm.

3.2 miNimal-Delay-First (NDF)

Another rerouting heuristic algorithm is the *miNimal-Delay-First* (*NDF*). There are two differences between the miNimal-Delay-First and the Maximal-Delay-First algorithms. First, in step 1, for each sub-tree $PT(v_i)$ $i = 1, 2, \ldots, v_{degree(s)}$, the *leaf destination v* (denoted *mindest*(v_i)) with the minimal delay (*delay*(*P*(*s*, *v*)) ≤ $\min_{u \in PT(v_i)}$\{*delay*(*P*(*s*, *u*))\}, *u* is the leaf node in *PT*(v_i) is routed by the path *P*(*s*, *v*) on *PT*(v_i), and the other destinations should be rerouted. Second, in step 3, for the destinations in UNREACH, the destination with the minimal delay has higher priority to be rerouted. Because keeping the route from source to the minimal delay destination can be more beneficial to reach more destinations which satisfy the delay-constraint. That is, this can make full use of the `tape and continuous' property of TaC node and can reach more destinations in a route on the same wavelength network.

For the example shown in Fig. 2, after removing edges in *Edge*(*P*(*s*, 1)), *Edge*(*P*(*s*, 6)), and *Edge*(*P*(*s*, 4)) from *G*, the result graph G_1' is shown in Fig. 4 (a) and the UNREACH is equal to {3, 2}. Because the destination 3 is the node with the minimal delay, the next step is tried to reroute destination 3, that is, find the minimal delay path $P^1(s, 2)$ from *s* to 3 in G_1'. The path found in G_1' is $s \rightarrow 9 \rightarrow 13 \rightarrow 3$ with a delay of 15 as shown in Fig. 4 (a). The extended path found in $G_1' \cup$ *Live*(*mindest*(7)) is $1 \rightarrow 13 \rightarrow 3$ with a delay of 9 (6 + 3) as shown in Fig. 4 (b). The extended path found in $G_1' \cup$ *Live*(*mindest* (8)) is $6 \rightarrow 17 \rightarrow 16 \rightarrow 2 \rightarrow 13 \rightarrow 3$ with a delay of 26 (12 + 14) as shown in Fig. 4 (c). Thus, the extended path $1 \rightarrow 13 \rightarrow 3$ satisfies the delay constraint. Further, the destination 2 is routed by $s \rightarrow 7 \rightarrow 14 \rightarrow 2$ with the delay of 9 on G_2. The final result by applying NDF Algorithm to Fig. 2 is shown in Fig. 4 (d).

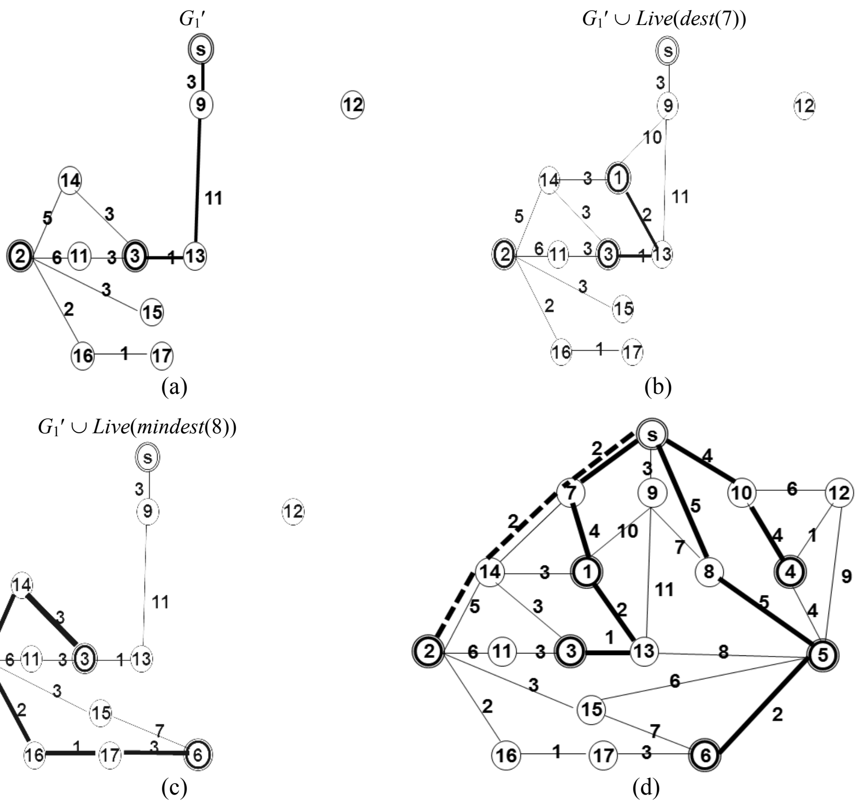


Fig. 4. Results by applying the NDF algorithm.

4. HEURISTIC ALGORITHMS FOR MCRP

Given a multicast request $r(s, D)$, the goal is to find the light-forest such that the total cost (consists of cost of light-tree and cost of wavelength) can be minimized. It is worth noting that the delay $d(e)$ of link e on WDM is ignored. Since the MCRP is NP-hard, in this section, two heuristic algorithms are proposed to solve it. These proposed algorithms are also *shortest-path based*. Obviously, the algorithms MDF and NDF described in the previous section which is proposed to solve the MWDCRP can be modified to solve the MCRP. Moreover, MDF and NDF algorithms cannot be applied directly because they did not take the number of used wavelengths and the ratio α into consideration. Two modified algorithms *Farthest-Greedy* and *Nearest-Greedy* are presented in this section.

4.1 Farthest-Greedy

For each sub-tree $PT(v_i)$, $i = 1, 2, \ldots, v_{degree(s)}$, only the farthest destination v with $c(P(s, v)) \geq \max_{\forall u \in PT(v_i)}\{c(P(s, u))\}$ is routed by the path $P(s, v)$ on $PT(v_i)$ and the others should be rerouted. This is denoted as the *Farthest-first Greedy* (FG). Since the objective function $c(r(s, D)) = c(T) + \alpha \times \sum_{k=1}^{w} y_k$ consists of two components: the first one is the cost of the light-tree for routing to the destinations; the second one is the weighted cost of wavelength used. To reduce the routing cost, reroute the path to destination on a whole new wavelength network is a good approach, but this will increase the cost of wavelength used. Thus, there is a tradeoff between the choice of routing path on used and new wavelength network.

Consider the network shown in Fig. 2, there are three destinations in the sub-tree $PT(7)$ with nodes 1, 2, and 3. The farthest (maximal cost) destination is node 2, thus the routing paths to destination nodes 1 and 3 should be rerouted. The farthest destinations in the sub-trees $PT(8)$ and $PT(10)$ are node 6 and 4, respectively. Let k be an integer which indicates the maximal index of the currently used wavelength network. To find a rerouting path of a destination v, $k + 1$ ($\leq w$) wavelength graphs ($G_1, G_2, \ldots, G_k, G_{k+1}$) are considered in the proposed algorithm. Let $P^i(s, v)$ be the minimal cost path from source s to destination v in the wavelength graph G_i, the cost of the path $P^i(s, v)$ is defined as $c(P^i(s, v))$, $i = 1, 2, \ldots, k + 1$ ($\leq w$) and is determined as follows:

- If $P^i(s, v)$ passes destinations, $\{v_1, v_2, \ldots, v_k\}$ in *UNREACH* on G_i, then the costs of $P(s, v_1), P(s, v_2), \ldots, P(s, v_k)$ on G should be subtracted from the cost of path $P^i(s, v)$. That is, $c(P^i(s, v)) = c(P^i(s, v)) - c(P(s, v_1)) - c(P(s, v_2)) - \ldots - c(P(s, v_k))$.
- If $P^i(s, v)$ uses the wavelength graph the $(k + 1)$th ($\leq w$), where i is equal to $k + 1$, then the cost of path $P^i(s, v)$ should be increased by α.
- If path from s to v cannot be found in the wavelength network G_i, then $c(P^i(s, v)) = \infty$.

After finding and computing the paths and costs of the possible paths, the path with the minimal cost is selected to be the rerouting path. If i is equal to $k + 1$ then increasing k by 1. Find the set $Edge(P^i(s, v))$ and remove edges in $Edge(P^i(s, v))$ from G_i.

In MCRP, the similar notations defined in MWDCRP are used, let $dest(v_i)$ be the destination node with maximal cost in sub-tree $PT(v_i)$. Let $Live(dest(v_i))$ be the set of edges whose one endpoint is $dest(v_i)$ and the other endpoint is in G'. For each subtree $PT(v_i)$, a subgraph is constructed by $G' \cup Live(dest(v_i))$. It is worth noting that the cost of the routing path, which is called the *extended path*, routed through node $dest(v_i)$ should be included in computing the cost of the path $P(s, dest(v_i))$. This rerouting process is continually executed until *UNREACH* is empty.

The details of the Farthest-First Greedy (FG) algorithm are described as follows:

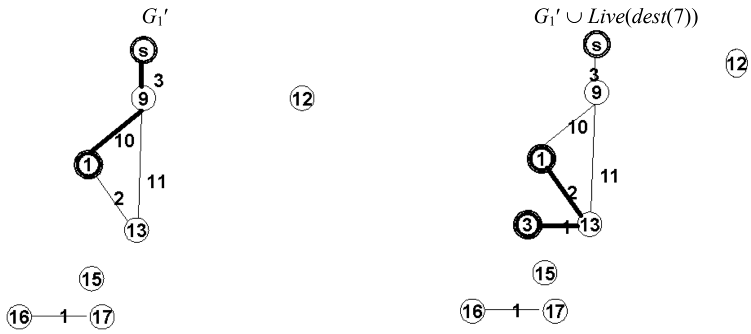
Algorithm Farthest-First Greedy (FG)
Step 1: Let $k = 1$, $G_1' = G_2' = \ldots = G_w' = G$. For each destination d_i in D find the minimal cost path $P(s, d_i)$ on G_1. Combine the paths to form a routing tree P. If P satisfies the light-splitting constraint, then DONE. Otherwise perform step 2.

Step 2: For each sub-tree $PT(v_i)$ on G_1, rooted at v_i, $i = 1, 2, \ldots, degree(s)$, find the farthest destination node $dest(v_i)$, the routing path $P^1(s, dest(v_i))$ and the $Edge(P^1(s, dest(v_i)))$. Modify G_1' by removing $Edge(P^1(s, dest(v_i)))$, $i = 1, 2, \ldots, degree(s)$ from G_1' and construct *UNREACH*. While *UNREACH* is non-empty, perform steps 3 and 4.

Step 3: Take the farthest destination in *UNREACH*, say v, find the minimal cost paths in the set of subgraphs $SG = \{G_z', G_z' \cup Live(dest(v_i))$, for all v_i, $z = 1, 2, \ldots, k + 1\}$.

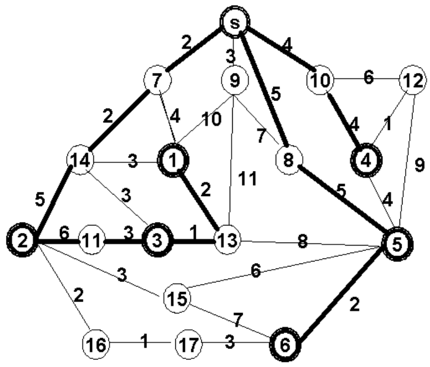
Step 4: Find the minimal cost path in the path set *SG*. If the path is found, assign path $P(s, v)$ to the corresponding wavelength z; subtract $Edge(P(s, v))$ from G_z'; remove the destination v and other destinations passed by path $P(s, v)$ from *UNREACH*; and update $Live(dest(v_i))$. Otherwise, increasing k by one, if k is greater than w will return FALSE.

Assume $\alpha = 4$, consider the example shown in Fig. 2. After performing steps 1 and 2 of the Farthest-first Greedy algorithm, the result is shown in Fig. 5 (a). Because destination 3 is the farthest node in *UNREACH*, step 3 tries to find the minimal cost path from s to 3. In G_1 the path is $s \rightarrow 9 \rightarrow 13 \rightarrow 3$ with the cost of 15. In $G_1' \cup Live(dest(7))$, the extended path is $2 \rightarrow 11 \rightarrow 3$ with the cost of 9. Similarly, in G_2' the path is $s \rightarrow 7 \rightarrow 14 \rightarrow 3$ with the cost of 11 ($7 + \alpha$). It is clear that the extended path $c(P^1(2, 3))$ is the minimal cost path, thus, destination 3 is routed in the wavelength graph G_2.



(a) (b)

Fig. 5. Final result by applying the Farthest-Greedy algorithm.



(c)

Fig. 5. (Cont'd) Final result by applying the Farthest-Greedy algorithm.

Then, to find the minimal cost routing path for destination 1, those paths on G_1, $G_1 \cup Live(dest(7))$ and G_2 are considered. They are path $P^2(s, 1) = s \to 9 \to 1$ with a cost of 13, extended path $P^1(3, 1) = 3 \to 13 \to 1$ with a cost of 3 and path $P^3(s, 1) = s \to 7 \to 1$ with a cost of 10 ($6 + \alpha$). Obviously, the minimal cost path is the extended path $P^1(3, 1)$. Thus, the final result by applying the Farthest-first Greedy algorithm to Fig. 2 is shown in Fig. 5 (c).

4.2 Nearest-Greedy

Another rerouting heuristic developed to solve the MCRP is the Nearest-Greedy. There are two differences between the Nearest-Greedy and the Farthest-Greedy. First, in step 2, for each sub-tree $PT(v_i)$ rooted at v_i for $i = 1, 2, \ldots, v_{degree(s)}$, the minimal cost leaf destination v ($c(P(s, v)) \leq \min_{u \in PT(v_i)} c(P(s, u))$, where u is the leaf destination in $PT(v_i)$) is routed by the path $P(s, v)$ on $PT(v_i)$ and the others should be rerouted. This is due to keeping the route from source to the nearest destination can be more beneficial to reach more destinations. That is, this can make full use of the 'tape and continuous' property of TaC node and can reach more destinations in a route on the same wavelength network. Second, in step 3, for the destinations in UNREACH, the minimal cost destination has the highest priority to be rerouted.

In MCRP, the similar notations defined in MWDCRP are used, let $mindest(v_i)$ be the destination node with minimal cost in sub-tree $PT(v_i)$. Let $Live(mindest(v_i))$ be the set of edges whose one endpoint is $mindest(v_i)$ and the other endpoint is in G'. Assume $\alpha = 4$, consider the example shown in Fig. 2, after performing steps 1 and 2 of Algorithm Nearest-Greedy, the result is the same as the result performed by steps 1 and 2 of NDF (as shown in Fig. 4 (a)) excepted that the weight of link represents the cost but not the delay of link. Because destination 3 is the destination with minimal cost in UNREACH, step 3 tries to find the minimal cost path $P^1(s, 3)$.

In G_1', the path $P^1(s, 3)$ is $s \to 9 \to 13 \to 3$ with the cost of 15 (Fig. 6 (a)). In $G_1' \cup Live(mindest(7))$, the extended path $1 \to 13 \to 3$ with a cost of 3 is found (Fig. 6 (b)). In $G_1' \cup Live(mindest(8))$, the extended path $6 \to 17 \to 16 \to 2 \to 14 \to 3$ with the cost of 14 is found. Note that the destination 2 is passed by the path, thus the cost $c(P^1(s, 3)) = 7$

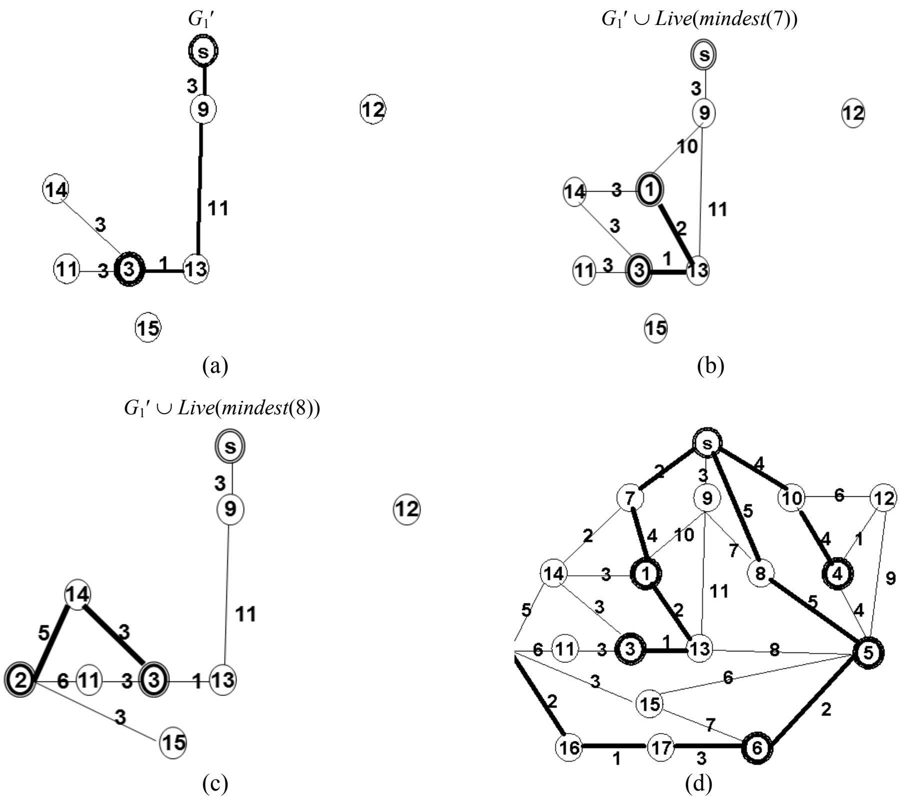


Fig. 6. Final result by applying the Nearest-Greedy algorithm.

should be subtracted the extended path $c(P^1(s, 3)) = 8$, therefore the extended path is the cost of 7. Similarly, in G_2 the path $P^2(s, 3)$ is $s \to 7 \to 14 \to 3$, $c(P^2(s, 3)) = 2 + 2 + 3 + \alpha = 11$. Obviously, the extended path $1 \to 13 \to 3$ is the minimal cost path, thus, destination 3 is routed on wavelength graph G_1.

After performing algorithm, the routing paths for destination 2 are $P^2(s, 2) = s \to 7 \to 14 \to 2$ with cost 11 (4 + 7), extended path $P^1(3, 2) = 3 \to 14 \to 2$ with a cost of 8 (Fig. 6 (c)), extended path $P^1(6, 2) = 6 \to 17 \to 16 \to 2$ with a cost 6. Thus, extended path $P^1(6, 2) = 6 \to 17 \to 16 \to 2$ is the minimal cost path. The final result by applying Nearest-Greedy to Fig. 2 is shown in Fig. 6 (d).

5. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed heuristic algorithms, they were implemented and applied 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 512MB RAM.

To test the efficiency of the proposed algorithms, the *Genetic Algorithm* (*GA*) proposed in [24] is used. Genetic algorithms have been trusted as a class of general-purpose search strategies that strike a reasonable balance between exploration and exploitation. A WDM network with n = 100 nodes, m = 1208 edges and w = 10 wavelengths was used to determine the parameters of GA. The parameters of GA were determined by performing ten times and the average results are used for the comparison. The parameters for running GA are: population size = 2000, crossover probability = 1.0, mutation probability = 0.3. The best solution of GA after running ten times is used to compare.

5.1 Results of MWDCRP

For the MWDCRP, the proposed algorithms have been run on several randomly generated and connected networks with different nodes (n is in {100, 200, 300}), and with different number of destination nodes ($|D|$ is in {10, 20, 30, 40, 50}) and with different delay constraint (*MAX_DELAY*) in {40, 30, 20}. The link delays are randomly selected from the set of integer {1, 2, …, 20}. For these tests, the sets of multicast requests are randomly generated on a random connected network.

The experimental results of the MDF and NDF algorithms are shown in Table 1. The 'ratio' columns are computed by (wavelength of algorithm / wavelength of GA × 100%). Observe the result shown in Table 1, the **MDF** gets the ratio 106.20% and better than the result of *NDF* (112.91%).

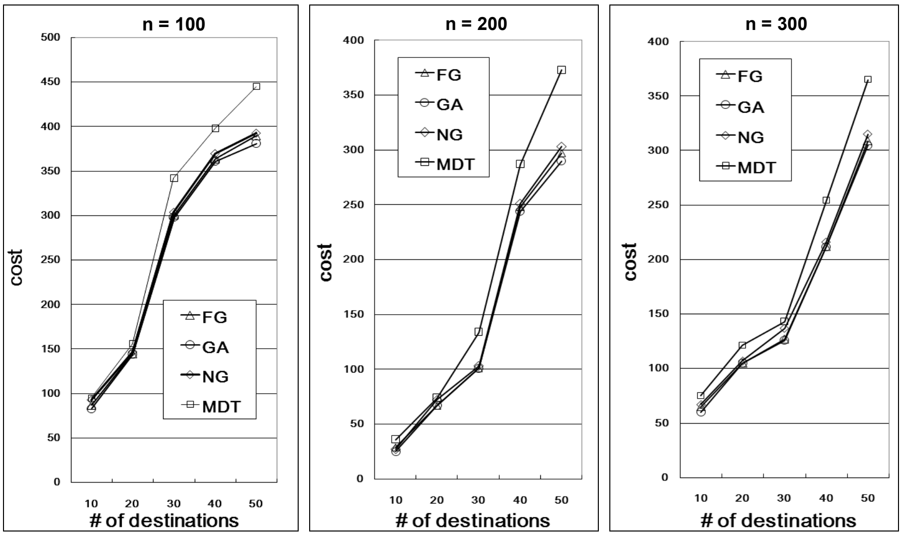
Table 1. Experimental results of algorithms MDF & NDF.

		MAX_DELAY=20					MAX_DELAY=30					MAX_DELAY=40				
n	$\|D\|$	MDF		NDF		GA	MDF		NDF		GA	MDF		NDF		GA
		wav.	ratio	wav.	ratio	wav.	wav.	ratio	wav.	ratio	wav.	wav.	ratio	wav.	ratio	wav.
100	10	3	100%	3	100%	3	4	100%	5	125%	4	5	125%	5	125%	4
100	20	6	100%	7	117%	6	8	100%	9	112%	8	9	100%	10	111%	9
100	30	12	120%	13	130%	10	13	108%	14	117%	12	14	108%	14	108%	13
100	40	13	108%	14	117%	12	14	100%	15	107%	14	16	100%	17	106%	16
100	50	15	107%	16	114%	14	18	106%	18	106%	17	19	106%	18	100%	18
200	10	2	100%	2	100%	2	2	100%	2	100%	2	3	100%	3	100%	3
200	20	3	100%	4	133%	3	4	100%	4	100%	4	5	125%	5	125%	4
200	30	6	120%	7	140%	5	6	100%	7	117%	6	7	100%	7	100%	7
200	40	8	100%	9	113%	8	9	100%	9	100%	9	9	100%	10	111%	9
200	50	11	110%	11	110%	10	12	109%	11	100%	11	12	100%	12	100%	12
300	10	2	100%	2	100%	2	2	100%	2	100%	2	3	150%	3	150%	2
300	20	4	133%	4	133%	3	4	100%	4	100%	4	4	100%	5	125%	4
300	30	6	100%	6	100%	6	6	100%	7	117%	6	6	100%	8	133%	6
300	40	7	100%	8	114%	7	8	100%	8	100%	8	9	113%	9	113%	8
300	50	10	111%	11	122%	9	11	110%	11	110%	10	11	110%	12	120%	10
average		107.33%		116.22%			102.22%		107.37%			109.05%		115.13%		

5.2 Results of MCRP

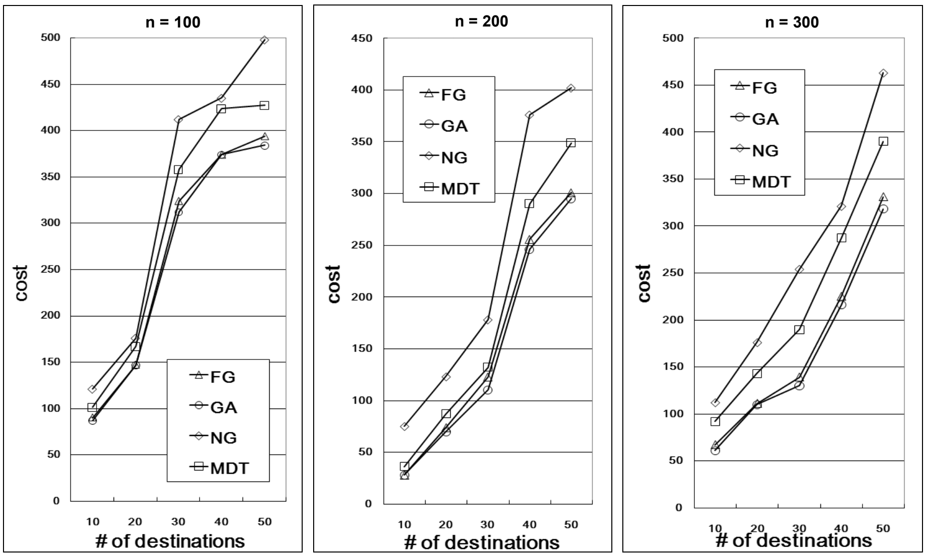
For the MCRP, the proposed algorithms have been run on several randomly generated and connected networks with different nodes (n is in {100, 200, 300}), and with different number of destination nodes ($|D|$ is in {10, 20, 30, 40, 50}) and with different ratio (α) in {50, 100, 150}. The link costs are randomly selected from the set of integer {1, 2, …, 20}. For these tests, the sets of multicast requests are randomly generated on a random connected network.

Moreover, to examine the performance of the proposed algorithms, the *Multiple-destination Trial* (MDT) heuristic algorithm proposed in [9] was implemented and used for comparisons. The average result of GA, the results of FG, NG and MDT algorithms for different values of ratio α are shown in Figs. 7, 8 and 9, respectively.



(a) $n = 100$. (b) $n = 200$. (c) $n = 300$.

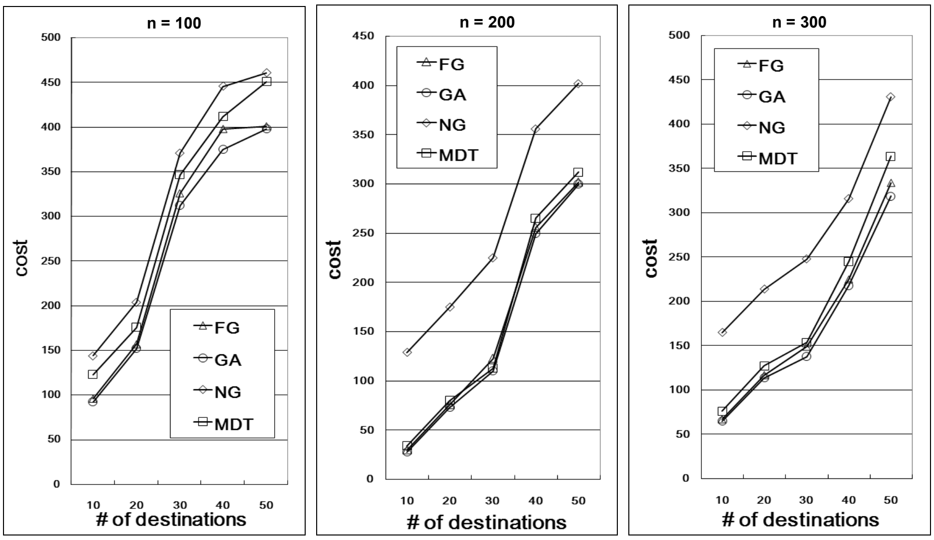
Fig. 7. Comparison of algorithms on the MCRP with $\alpha = 50$.



(a) $n = 100$. (b) $n = 200$. (c) $n = 300$.

Fig. 8. Comparison of algorithms on the MCRP with $\alpha = 100$.

The experimental results of the FG and NG, MDT, and GA algorithms are shown in Table 2. The 'ratio' columns are computed by (cost of algorithm / cost of GA × 100%) and the 'CPU seconds' columns demonstrate the CPU time in seconds of the algorithm. Observe the result shown in Table 2, the **Farthest-Greedy** gets the ratio 103.70% and better than the results of *Nearest-Greedy* (105.89%) and *Multiple-destination Trial* (119.41%).



(a) $n = 100$. (b) $n = 200$. (c) $n = 300$.

Fig. 9. Comparison of algorithms on the MCRP with $\alpha = 150$.

Table 2. Experimental results of algorithms FG & NG.

α=50												
n	$\|D\|$	F-G			N-G			MDT			GA	
		cost	CPU seconds	ratio	cost	CPU seconds	ratio	cost	CPU seconds	ratio	cost	CPU seconds
100	10	87	0.012	104.82%	93	0.012	112.05%	95	0.017	114.46%	83	2.123
100	20	145	0.014	100.69%	147	0.014	102.08%	156	0.017	108.33%	144	2.345
100	30	301	0.017	101.01%	304	0.017	102.01%	342	0.017	114.77%	298	2.876
100	40	364	0.023	100.83%	370	0.023	102.49%	398	0.023	110.25%	361	3.129
100	50	390	0.032	102.36%	393	0.032	103.15%	445	0.023	116.80%	381	3.657
200	10	29	0.124	116.00%	27	0.124	108.00%	36	0.124	144.00%	25	4.234
200	20	67	0.154	100.00%	72	0.154	107.46%	74	0.124	110.45%	67	4.673
200	30	101	0.187	100.00%	103	0.187	101.98%	134	0.124	132.67%	101	5.164
200	40	248	0.231	101.64%	251	0.231	102.87%	287	0.124	117.62%	244	5.543
200	50	297	0.266	102.41%	303	0.266	104.48%	373	0.124	128.62%	290	5.908
300	10	65	0.187	108.33%	67	0.187	111.67%	75	0.183	125.00%	60	7.431
300	20	105	0.213	100.00%	107	0.213	101.90%	121	0.183	115.24%	105	7.901
300	30	127	0.243	100.79%	137	0.243	108.73%	143	0.183	113.49%	126	8.321
300	40	212	0.284	100.00%	216	0.284	101.89%	254	0.183	119.81%	212	8.791
300	50	309	0.311	101.31%	315	0.311	103.28%	365	0.183	119.67%	305	9.165
average				102.68%			104.94%			119.41%		

α=100												
n	$\|D\|$	F-G			N-G			MDT			GA	
		cost	CPU seconds	ratio	cost	CPU seconds	ratio	cost	CPU seconds	ratio	cost	CPU seconds
100	10	90	0.012	103.45%	89	0.012	102.30%	101	0.017	116.09%	87	2.123
100	20	147	0.014	100.00%	150	0.014	102.04%	167	0.017	113.61%	147	2.345
100	30	324	0.017	103.85%	324	0.017	103.85%	358	0.017	114.74%	312	2.876
100	40	374	0.023	100.00%	382	0.023	102.14%	423	0.023	113.10%	374	3.129
100	50	394	0.032	102.60%	401	0.032	104.43%	427	0.023	111.20%	384	3.657
200	10	28	0.124	100.00%	31	0.124	110.71%	36	0.124	128.57%	28	4.234
200	20	74	0.154	105.71%	78	0.154	111.43%	87	0.124	124.29%	70	4.673
200	30	123	0.187	111.82%	116	0.187	105.45%	132	0.124	120.00%	110	5.164
200	40	256	0.231	104.07%	271	0.231	110.16%	290	0.124	117.89%	246	5.543
200	50	301	0.266	102.03%	310	0.266	105.08%	349	0.124	118.31%	295	5.908
300	10	67	0.187	109.84%	69	0.187	113.11%	92	0.183	150.82%	61	7.431
300	20	111	0.213	100.91%	123	0.213	111.82%	143	0.183	130.00%	110	7.901
300	30	139	0.243	106.92%	143	0.243	110.00%	190	0.183	146.15%	130	8.321
300	40	225	0.284	104.17%	231	0.284	106.94%	287	0.183	132.87%	216	8.791
300	50	331	0.311	104.09%	325	0.311	102.20%	390	0.183	122.64%	318	9.165
average				103.96%			106.78%			124.02%		

α=150												
n	$\|D\|$	F-G			N-G			MDT			GA	
		cost	CPU seconds	ratio	cost	CPU seconds	ratio	cost	CPU seconds	ratio	cost	CPU seconds
100	10	96	0.012	104.35%	94	0.012	102.17%	123	0.017	133.70%	92	2.123
100	20	157	0.014	103.29%	154	0.014	101.32%	176	0.017	115.79%	152	2.345
100	30	326	0.017	104.49%	328	0.017	105.13%	347	0.017	111.22%	312	2.876
100	40	398	0.023	106.13%	386	0.023	102.93%	412	0.023	109.87%	375	3.129
100	50	401	0.032	100.75%	410	0.032	103.02%	451	0.023	113.32%	398	3.657
200	10	30	0.124	107.14%	30	0.124	107.14%	34	0.124	121.43%	28	4.234
200	20	76	0.154	104.11%	78	0.154	106.85%	80	0.124	109.59%	73	4.673
200	30	123	0.187	111.82%	118	0.187	107.27%	113	0.124	102.73%	110	5.164
200	40	256	0.231	102.40%	274	0.231	109.60%	265	0.124	106.00%	250	5.543
200	50	302	0.266	100.67%	312	0.266	104.00%	312	0.124	104.00%	300	5.908
300	10	67	0.187	103.08%	74	0.187	113.85%	76	0.183	116.92%	65	7.431
300	20	117	0.213	102.63%	123	0.213	107.89%	127	0.183	111.40%	114	7.901
300	30	149	0.243	107.97%	147	0.243	106.52%	154	0.183	111.59%	138	8.321
300	40	225	0.284	103.21%	235	0.284	107.80%	245	0.183	112.39%	218	8.791
300	50	334	0.311	104.70%	325	0.311	101.88%	364	0.183	114.11%	319	9.165
average				104.45%			105.82%			112.94%		

Moreover, the results show that MDT is the fastest algorithm. Both the CPU time is seconds spent by algorithms FG and NG are faster than GA. On the other hand, the memory resources needed by the FG and NG are smaller than that of GA. That is, FG and NG algorithms are suitable implemented in the real environment.

6. CONCLUSIONS

In this paper, the *Minimal Cost Routing Problem* (*MCMRP*) and the *Minimal Wavelength Delay Constraint Routing Problem* on WDM networks with *Tap-and-Continue* (TaC) nodes are defined and studied. For the MCRP, a new cost model which consists of the wavelength usage and communication cost is defined. The objective is to minimize the sum of the cost of used wavelengths and the communication cost of the *light-forest*. For the MWDCRP, for a given multicast request, a light-forest is found such that the number of used wavelengths is minimized under the delay constraint. Specifically, the formulations for the WDM multicast routing problem are given. Because these problems are NP-hard, for each problem two heuristic algorithms are proposed to solve it, they are *Maximal-Delay-First* (*MDF*), *miNimal-Delay-First* (*NDF*), *Farthest-Greedy* (*FG*), and *Nearest-Greedy* (*NG*). In the proposed heuristics, first, shortest path algorithm is performed to fine the shortest path tree, then the rerouting techniques are used to find the constraint-satisfy light-tree. Simulation results demonstrate that the proposed algorithms can generate good solutions near the solution obtained by performing the genetic algorithm.

ACKNOWLEDGMENT

The author is very grateful to anonymous reviewers for their helpful suggestions and constructive comments.

REFERENCES

1. J. Y. Wei, "Advances in the management and control of optical internet," *IEEE Journal on Selected Areas in Communications*, Vol. 20, 2002, pp. 768-785.
2. K. H. Liu, *IP over WDM*, John Wiley & Sons, Ltd., New York, 2002.
3. W. Hu, R. Q. Hu, and Y. Qian, "Wavelength retuning in a WDM mesh network with survivable traffic grooming," in *Proceedings of International Conference on Performance*, *Computing*, *and Communications*, 2006, pp. 529-536.
4. B. Mukherjee, *Optical WDM Networks*, Springer, New York, 2006.
5. T. Lee, K. Lee, and S. Park, "Optimal routing and wavelength assignment in WDM ring networks," *IEEE Journal on Selected Areas in Communications*, Vol. 18, 2000, pp. 2146-2154.
6. N. Nagatsu, S. Okamoto, and K. Sato, "Optical path cross-connect system scale evaluation using path accommodation design for restricted wavelength multiplexing," *IEEE Journal of Selected Areas in Communications*, Vol. 14, 1996, pp. 893-902.

7. L. Wuttisittikulkij and M. J. O'Mahony, "Design of efficient and practical algorithm for wavelength assignment in multi-wavelength ring transport networks," in *Proceedings of IEEE GLOBECOM*, 1997, pp. 571-575.
8. W. Xingwei, H. Meijiaa, Y. Xiushuangb, and H. Mina, "A QoS multicast routing algorithm based on tabu-hierarchy genetic algorithm in IP/DWDM optical Internet," in *Proceedings of the SPIE*, *The International Society for Optical Engineering*, *Network Architectures*, *Management*, *and Applications III*, Vol. 6022, 2005, pp. 802-808.
9. M. Ali and J. S. Deogun, "Cost-effective implementation of multicasting in wavelength-routed networks," *IEEE Journal of Lightware Technology*, Vol. 18, 2000, pp. 1628-1638.
10. N. Sreenath, C. S. R. Murthy, and G. Mohan, "Virtual source based multicast routing in WDM optical networks," *Photonic Network Communications*, Vol. 3, 2001, pp. 213-226.
11. N. Sreenath, N. K. M. Reddy, G. Mohan, and C. S. R. Murthy, "Virtual source based multicast routing in WDM networks with sparse light splitting," *IEEE Workshop on High Performance Switching and Routing*, 2001, pp. 141-145.
12. X. Zhang, J. Wei, and C. Qiao, "On fundamental issues in IP over WDM multicast," in *Proceedings of International Conference on Computer Communications and Networks*, 1999, pp. 84-90.
13. A. Dinga and G. S. Poo, "A survey of optical multicast over WDM networks," *Computer Communications*, Vol. 26, 2003, pp. 193-200.
14. Y. Zhou and G. S. Poo, "Optical multicast over wavelength-routed WDM networks: a survey," *Optical Switching and Networking*, Vol. 2, 2005, pp. 176-197.
15. C. S. R. Murthy and M. Gurusamy, *Optical Multicast Routing-WDM Optical Networks: Concepts*, *Design*, *and Algorithms*, Chapter 8, Prentice Hall Inc, New Jersey, 2002.
16. M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, New York, 1979.
17. L. H. Sahasrabuddhe and B. Mukherjee, "Light-trees: Optical multicasting for improved performance in wavelength-routed networks," *IEEE Communications Magazine*, Vol. 37, 1999, pp. 67-73.
18. X. Zhang, J. Wei, and C. Qiao, "Constrained multicast routing in WDM networks with sparse light splitting," *Journal of Lightwave Technology*, Vol. 18, 2000, pp. 1917-1927.
19. C. Oliveira and P. Pardalos, "A survey of combinatorial optimization problems in multicast routing," *Computers and Operations Research*, Vol. 32, 2005, pp. 1953-1981.
20. A. T. Haghighat, K. Faez, M. Dehghan, A. Mowlaei, and Y. Ghahremani, "GA-based heuristic algorithms for bandwidth-delay-constrained least-cost multicast routing," *Computer Communications*, Vol. 27, 2004, pp. 111-127.
21. C. Qiao, X. Zhang, and L. Zhou, "Scheduling all-to-all connections in WDM rings," in *Proceedings of International Conference on Social Optical Engineering*, 1994, pp. 970-979.
22. V. P. Kompella, J. C. Pasquale, and G. C. Polyzos, "Multicast routing for multimedia communication," *IEEE*/*ACM Transactions on Networking*, Vol. 1, 1993, pp. 286-292.

23. D. Yang and W. Liao, "Design of light-tree based logical topologies for multicast streams in wavelength routed optical networks," in *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies* (*INFOCOM*), 2003, pp. 32-41.
24. D. R. Din, "Genetic algorithms for finding minimal cost light-forest of multicast routing on WDM network," *Journal of Artificial Evolution and Applications*, Vol. 2008, 2008, http://www.hindawi.com/GetArticle.aspx?doi=10.1155/2008/536913.

Der-Rong Din (丁德榮) received M.S. and Ph.D. degrees in Computer and Information Science from National Chiao Tung University in 1993 and 2001, respectively. Now, he is on the faculty of Department of Computer Science and Information Engineering at National Changhua University of Education. His current research interests are in WDM networking, mobile communication, parallel compiler and algorithms.