Multicast Routing and Wavelength Assignment in WDM Networks with Limited Drop-offs

X.-D. Hu and T.-P. Shuai
Inst. of Applied Math.
Chinese Academy of Sciences
Beijing, China
{xdhu, shuaitp}@mail.amss.ac.cn

Xiaohua Jia
Dept. of Computer Sci.
City Univ. of Hong Kong
Kowloon, Hong Kong
jia@cs.cityu.edu.hk

Mu-Hong Zhang
Dept. of IE and OR
Univ. of California
Berkeley, USA
mhzhang@ieor.berkeley.edu

***Abstract*— In WDM networks with limited drop-offs, the route of a multicast connection consists of a set of light-trees. Each of the light-tree is rooted at the source node and contains no more than a limited number, say k, destination nodes due to the power loss of dropping optical signals off at destination nodes. We call such a light-tree k-drop light-tree. In this paper we study the multicast routing problem of constructing a set of k-drop light-trees that have the minimal network cost. The network cost of a set of light-trees is defined as the summation of the link cost of all the light-trees. We first prove that this problem is polynomial-time solvable for $k = 2$ and NP-hard for $k \geq 3$. We then propose a 4-approximation algorithm for the problem for $k \geq 3$. A wavelength assignment algorithm is also proposed to assign wavelengths to the light-trees of a multicast connection. In the end we give simulation results showing that k-drop multi-tree routing can significantly save not only the network cost but also wavelengths used. Moreover, when $k \geq 5$ its performance is very close to the case where k is infinite (i.e., the case of using a single tree for a multicast connection).**

I. Introduction

Multicast is a point-to-multipoint communication that a node sends data to multiple destinations [4,13]. There are many multicast applications, such as news feeds, video distribution, multimedia conferencing, and so on. It is a challenging job to implement multicast in Wide Area Networks (WANs) due to high complexity of multicast routing [10,13]. Multicast routing is to find a tree rooted from the source and connecting all the destinations. The multicast data will be transmitted from the source and propagated to all destinations along the tree. An important objective of multicast routing is to minimize the *network cost* of the routing tree, which is defined as the sum of costs of all the links in the tree. The problem of finding a tree in a general topology network, such that it connects a set of nodes and has the minimal cost, is known as the Steiner tree problem [2].

To facilitate multicast in wavelength-routed optical networks, the concept of a *light-tree* and the cross-connect architecture of *splitter-and-delivery* were proposed in [1,14]. A light-tree is a point-to-multipoint all optical channel which is rooted from a source and connects multiple destinations. In the absence of wavelength converters, a light-tree would occupy the same wavelength on all links of the tree. Each intermediate node that has more than one child in a light-tree must have a splitter which splits the incoming optical signal into multiple copies outgoing to the child nodes. Each time an optical signal is split, a splitting loss is incurred. If a destination is an intermediate node in the light-tree, a splitting loss is also incurred to drop-off a copy of the signal at the destination. In practice, if a signal is split into n copies, the signal power of at least one copy will be less than or equal to $\frac{1}{n}$ of the original signal power [1]. On the other hand, an optical signal must have a minimum amount of power in order to be dropped-off at a destination or passed to the next down-stream node. Therefore, due to this split loss, it is not possible to drop off data at an arbitrary number of destinations in a single light-tree [9]. A light-tree always has a limited number of destinations that are allowed to drop-off data, and a multicast connection with many destinations needs to use multiple light-trees to transmit data to all destinations.

In this paper, we study how to establish a multicast connection in an optical network under the *k-drop multi-tree model* [9]. Under this model, there is a pre-specified integer k, which limits the maximum number of destinations that optical signals are allowed to drop-off at a light-tree. Multicast routing becomes a problem of finding a set of light-trees such that at most k destination nodes are designated to receive data in each light-tree and every destination node is designated in one of the light-trees in the set. The objective is that the total cost of light-trees in the set is minimal. After the routing, we need to assign a wavelength to each of the produced light-trees in such a way that distinct wavelengths are used for two light-trees if they share a common link. We assume no wavelength converter is used in a light-tree.

The k-drop multi-tree model is a general model for multicast routing. When there is no limit on the number of destinations that optical signals are allowed to drop-off at (i.e., k can be an arbitrary number), it becomes the single-tree model, because all the destinations of a multicast can be reached in a single tree. When no splitting nor drop-off is allowed at switches (i.e., k = 1), it becomes the simple *lightpath model*, where a separate lightpath is required from the source to each of the multicast destinations. Recently some research work has been done on how to establish a multicast connection under the k-drop multi-tree model in optical networks. In [9] the wavelength requirement for a multicast connection in some special topology networks was analyzed, such as in rings, torus, and hypercubes. More analysis and the routing algorithms for minimizing the number of wavelengths in the special topology networks were given in [7]. Finding multi-trees for a multicast connection was also discussed in [16,17]. A greedy algorithm was proposed in [16] to find multi-trees, such that each tree uses the same wavelength (wavelength continuity rule) and the total cost of multi-trees is minimized. The size of a tree is constrained by wavelength continuity rule and it did not consider the limited drop-offs in a tree. Zhang et al. proposed in [17] a set of algorithms to construct a source-based multicast light-forest consisting of one or more multicast trees. The objectives of the algorithms include minimizing the number of wavelengths and the number of hops from the source to each destination. There is no guaranteed performance bound for the algorithms proposed in [16,17] and the performance analysis was done through simulations.

The rest of the paper is organized as the follows. In Section 2 we formulate the minimal multicast routing problem in k-drop multi-tree model. In Section 3 we propose a polynomial-time optimal algorithm for the case of $k = 2$. In Section 4, we show the problem is NP-hard for $k \geq 3$ and propose a 4-approximation algorithm to solve the problem. In Section 5, we propose a wavelength assignment algorithm for the multi-trees for a multicast connection. In Section 6, we conduct an extensive simulations to study the performance of the proposed algorithms. Finally, Section 7 concludes the paper.

II. Preliminaries

A. Problem Specification

In this paper, we assume bidirectional transmission (but the approach and analysis are both applicable for unidirectional case). That is, for two nodes (switches/routers) A and B in the network, there are two links between them, one carries transmission from A to B while the other from B to A.

A multicast connection is represented by $< s, D >$, where s is the source node from which data is sent to a set of destination nodes D. Under the multi-tree model, at most k destination nodes are allowed to receive the data in a light-tree, where parameter k is dependent on the power budget of light transmission.

B. Problem Formulation

According to the above specification and assumption of our problem, we model the network under consideration as an edge-weighted graph $G(V, E)$, where vertex-set V is the set of nodes in the network representing switches/routers and edge-set E is the set of links between nodes. Weight function $c : E \rightarrow \Re^+$ represents the network cost of using a particular edge. We assume that the weight function c is additive over the links in a path $p(u, v)$ between u and v, i.e.,

$$c(p(u,v)) \equiv \sum_{a \in p(u,v)} c(a).$$

For the simplicity of presentation, we denote by $p_{G'}(u, v)$ the shortest path from u to v in subgraph G' of G.

We define a *k-drop tree* as a tree in G such that in the tree at most k nodes in D are designated to receive the data. In addition, we define a *k-drop multi-tree routing* of $< s, D >$, denoted by $R(s, D; k) = \{T_i \,|\, i\}$, as a set of k-drop trees T_i's such that every destination in D must be designated to receive the data in a T_i in $R(s, D; k)$ for some i. Let $m \equiv \lceil |D|/k \rceil$, then the number of k-drop trees in $R(s, D; k)$ is $|R(s, D; k)| \geq m$. Two k-drop trees in $R(s, D; k)$ may share a common edge, which will not cause any problem during data transmission under the wavelength division multiplexing (WDM) technology [3,15].

When multicasting, data is transmitted over the multiple k-drop trees from the source to reach all the destinations. Data is transmitted through each edge in a k-drop tree exactly once, the network cost of multicasting data can then be defined as the total network cost of k-drop trees in $R(s, D; k)$, i.e.,

$$c\big(R(s,D;k)\big) \equiv \sum_{T_i \in R(s,D;k)} c(T_i).$$

In this paper, we study how to, given a multicast connection $< s, D >$ and a positive integer k, find a k-drop multi-tree routing $R(s, D; k)$ of minimum network cost. We refer this problem as k-MTR (*k-drop Multi-Tree Routing*) problem. Although the optimization of wavelength usage is not expressed explicitly as an objective, it is achieved implicitly. This is because a $R(s, D; k)$ with less network cost tends to have less number of trees and less number of links in a tree, and thus have less chances for trees to share a common links (which results in less number of wavelengths required).

III. Optimal Solution to k-MTR for $k \leq 2$

When $k = 1$, the optimal solution to the k-MTR problem consists of $|D|$ shortest paths from source s to each of $|D|$ destinations (which is the case of the lightpath model). It can be found in polynomial-time. In the following, we will show how to find an optimal solution to k-MTR for $k = 2$.

When k=2, each k-drop tree in the multicast routing can contain no more than two destinations. The basic idea is to introduce an auxiliary graph G' so that the k-MTR problem in the original graph G can be reduced to the *minimum weighted matching problem* [12] in G', which can be solved in polynomial-time. A *matching* of a graph is a set of edges such that no vertex in the graph is incident to two edges in the set. It is further called *complete* if every graph is incident to an edge in the matching. The minimum weighted matching problem is to, given an edge-weight graph that has complete matchings, find a complete matching whose weight is minimal, where the weight of a matching is the total weight of edges in the matching.

Given a multicast connection $< s, D >$ on network G, where $D = \{d_1, \cdots, d_{|D|}\}$. For each pair of destinations (d_i, d_j), compute the minimum Steiner tree from s to destinations d_i and d_j, denoted by $T(s, d_i, d_j)$. Note that $T(s, d_i, d_j)$ consists of at most three paths, $p_G(d_i, u)$, $p_G(d_j, u)$, and $p_G(s, u)$, jointed at some node $u \in V$. (When u is d_i or d_j, $T(s, d_i, d_j)$ consists of two paths.) Hence computing

$T(s, d_i, d_j)$ is to find a node $u \in V$ such that
$c(p_G(d_i, u)) + c(p_G(d_j, u)) + c(p_G(s, u)) =$
$\min_{v \in V}\{c(p_G(d_i, v)) + c(p_G(d_j, v)) + c(p_G(s, v))\}$.

After computing the Steiner tree for each pair of destinations, we introduce a new graph $G'(D \cup \{s_1, \cdots, s_{|D|}\}, E')$. Each s_i is a pseudo node of s associated with destination d_i. There is an edge between d_i and d_j for $i \neq j$ and the weight of the edge $w(d_i, d_j) = c(T(s, d_i, d_j))$, i.e., the cost of minimum Steiner tree from source s to destinations d_i and d_j. There is an edge between s_i and s_j for $i \neq j$ and its weight is zero. There is an edge between s_i and d_i for each i and its weight $w(s_i, d_i) = c(p_G(s, d_i))$, i.e, the cost of shortest path from source s to destination d_i. There is no edge between d_i and s_j for $i \neq j$.

Considering the new graph G', edge $(d_i, d_j) \in E'$ corresponds to the minimum Steiner tree from s to destinations d_i and d_j, and edge $(s_i, d_i) \in E'$ corresponds to the shortest path from source s to destination d_i. That is, each edge in G', except edges (s_i, s_j), corresponds to a k-drop tree for $k \leq 2$. Thus a k-drop multi-tree routing $R(s, D; k)$ corresponds to a set of edges in G' such that each destination in D is incident to exactly one edge in the set. Therefore, using pseudo nodes s_i (so that G' has complete matchings) and setting weight zero to edges between them, the problem of finding a set of 2-drop trees that include all destinations and have the minimal total cost becomes the problem of finding a minimum weighted complete matching in G' whose cost is minimal. The algorithm is formally presented below.

Algorithm A *Minimum Matching Based Algorithm*

(A1) Compute the shortest path $p_G(u, v)$ for each node pair u and v in V.
(A2) Compute the minimum Steiner tree $T(s, d_i, d_j)$ for each destination pair d_i and d_j in D.
(A3) Construct new graph $G'(D \cup \{s_1, \cdots, s_{|D|}\}, E')$.
(A4) Find the minimum weighted matching M in G'.
(A5) Output the set of 2-drop routing trees from M.

Theorem 1 *Algorithm A solves the 2-MTR in time $O(|V|^3 + |D|^4)$.*

Proof In Step A1, the shortest path between each node pair in V can be found in time $O(|V|^3)$. In Step A2, the minimum Steiner tree for each node pair in D can be produced in time $O(|D|^2|V|)$. In Step A3, constructing the auxiliary graph requires no extra time after Step A1-2 are finished. In Step A4, the minimum weighted matching problem in G' can be formulated as a linear programming problem and thus solved by suing the primal-dual method in time $O(|D|^4)$ (refer to [12]). In Step A5, having found the matching M, the 2-MTR for $< s, D >$ can be obtained by substituting each edge $(d_i, d_j) \in M$ by the Steiner tree $T(s, d_i, d_j)$, and each edge $(s_i, d_i) \in M$ by the shortest path $p_G(s, d_i)$. It costs linear time to $|D|$. The proof is then finished. ■

IV. Approximate Solution to k-MTR for $k \geq 3$

In general the k-MTR problem is NP-hard, because the Steiner tree problem can be reduced to it by setting $k \geq |D|$.

In this section we will propose an approximation algorithm with a guaranteed performance ratio. The basic idea is to first produce a directed trail of low cost including all nodes in $D \cup \{s\}$, and then break it into m small trails on which at most k nodes in D are designated to receive data, in the end for each small trail make a k-drop tree constituting of s and those designated nodes in D. The directed trail can be obtained by constructing a Hamilton circuit of low cost in an auxiliary graph whose vertex-set is $D \cup \{s\}$.

Algorithm B *Hamilton Circuit Based Algorithm*

(B1) Construct an auxiliary complete edge-weighted graph G_a of $D \cup \{s\}$. For $u, v \in D \cup \{s\}$ the weight of edge (u, v) is $c(p_G(u, v))$.
(B2) Construct a Hamilton circuit H_c of G_a by using Christonfides' method [5,12].
(B3) Obtain a directed trail $\mathcal{T}$ of $D \cup \{s\}$ in G by substituting each edge in H_c by the shortest path between its two endpoints of the edge in G, $\mathcal{T} = (s \to d_1 \to \cdots \to d_{|D|-1} \to d_{|D|} \to s)$.
(B4) Partition $\mathcal{T}$ into m subtrails $\mathcal{T}_i$ for $i = 0, 1, \cdots m-1$ such that $d_{ik+1}, d_{ik+2}, \cdots, d_{ik+k}$ are designated in $\mathcal{T}_i$ to receive the data. For each i, find v_i in $\mathcal{T}_i$ which is closest to source s.
(B5) Construct a k-drop tree T_i designating k destinations $d_{ik+1}, d_{ik+2}, \cdots, d_{ik+k}$ in subtrail $\mathcal{T}_i$, i.e., $T_i := \mathcal{T}_i \cup p_G(s, v_i)$.
(B6) Output $R_B(s, D; k) := \{T_0, T_1, \cdots, T_{m-1}\}$.

Because in Step B1 the auxiliary graph G_a is a complete graph and the weight function defined on its edges satisfies triangular inequality, in Step B2 Christonfides' method can be employed to construct a Hamilton circuit of G_a. The following lemma comes directly from the well-known result due to Christonfides [5].

Lemma 1 *For any given multicast connection $< s, D >$ on G, the Hamilton circuit H_c of G_a produced at (B1-2) has cost at most 3/2 times that of the minimum Hamilton circuit of G_a.*

We now prove that Algorithm B has a constant guaranteed performance ratio in the worst case analysis. To do this, we need the following lemma. Given a multicast connection $< s, D >$, let $R_{opt}(s, D; k)$ be an optimal k-routing and $c(R_{opt}(s, D; k))$ be its cost.

Lemma 2 *Let $d_{i'}$ be the destination node in trail $\mathcal{T}_i$ that is the closest to s. Then*

$$\sum_{i=0}^{m-1} c(p_G(s, d_{i'})) \leq c(R_{opt}(s, D; k)).$$

Proof Suppose that the optimal k-MTR $R_{opt}(s, D; k)$ has trees, $T_1^*, T_2^*, \cdots, T_N^*$, where each T_i^* is a k-drop tree and $N \geq m$. We construct an auxiliary weighted bipartite graph $B(X, Y)$, where $X = \{\mathcal{T}_i \,|\, i = 0, 1, \cdots, m-1\}$ and $Y = \{T_i^* \,|\, i = 1, \cdots, N\}$. There exists an edge $(\mathcal{T}_i, T_j^*)$ in $B(X, Y)$ if and only if $\mathcal{T}_i$ and T_j^* designate $\alpha \geq 1$ destination nodes in common and the weight of the edge is $w(\mathcal{T}_i, T_j^*) = \alpha$.

Now we prove, by using Hall's Theorem (refer to [12]), that $B(X,Y)$ has a perfect matching such that each $\mathcal{T}_i$ is incident to an edge in the matching. Suppose, by contradiction, that there exists a subset $X_0 \subseteq X$ such that X_0's neighbor set $Y_0 \subseteq Y$, which consists of vertices adjacent with some vertices in X_0, satisfies $|Y_0| \leq |X_0| - 1$. Since each T_i designates at most k destination nodes and each of them is designated in exactly one optimal k-drop tree, then the total weight of edges incident to T_i is at most k. For each T_j^* we have the same result. Now for $X' \subseteq X$ and $Y' \subseteq Y$, let $w(X')$ and $w(Y')$ denote the total weights of edges incident to some $\mathcal{T}_i \in X'$ and $T_j^* \in Y'$, respectively. Then $w(Y') \leq k|Y'|$, this implies $w(X_0) \leq w(Y_0) \leq k|Y_0|$. In addition, we have

$$w(X \setminus X_0) \leq k|X \setminus X_0| = k(m - |X_0|).$$

Hence we obtain the following contradiction.

$$\begin{aligned} |D| &= w(X_0) + w(X \setminus X_0) \\ &\leq k|Y_0| + k(m - |X_0|) \\ &\leq k(m-1) < |D|. \end{aligned}$$

Therefore, there exists a desired matching. Without loss of generality, we denote this matching by $M = \{(\mathcal{T}_i, T_i^*) \,|\, i\}$. This means that for each i there exists a destination node designated in both $\mathcal{T}_i$ and T_i^*. Thus the cost of T_i^* is not less than the cost of the shortest path from s to that common designated destination node, which, by the definition of $d_{i'}$, is not less than the cost of the shortest path from s to $d_{i'}$, i.e., $c(T_i^*) \geq c(p_G(s, d_{i'}))$. To sum up this inequality over i, we obtain the desired inequality. The proof is finished. ■

Theorem 2 *Given a multicast connection* $< s, D >$ *and* $k \geq 3$, *Algorithm B produces a k-MTR* $R_B(s,D;k)$ *in time* $O(|D||V|^2)$ *whose cost is at most four times that of the optimal k-MTR* $R_{opt}(s,D;k)$.

Proof Let H_{opt} be the minimum Hamilton circuit of G_a. Then we have

$$2c(R_{opt}(s,D;k)) \geq c(H_{opt}), \tag{1}$$

since two $R_{opt}(s,D;k)$s correspond a Hamilton circuit of G_a. In addition, by Lemma 1 and inequality (1), we have

$$c(H_c) \leq \frac{3}{2} c(H_{opt}) \leq 3c(R_{opt}(s,D;k)). \tag{2}$$

Thus by the rules of Algorithm (B5) and Lemma 2, we have

$$\begin{aligned} c(R_B(s,D;k)) &= \sum_{i=0}^{m-1} c(T_i) \\ &= \sum_{i=0}^{m-1} c(\mathcal{T}_i) + \sum_{i=0}^{m-1} c(p_G(s, v_i)) \\ &< c(\mathcal{T}) + c(R_{opt}(s,D;k)) \\ &= c(H_c) + c(R_{opt}(s,D;k)) \\ &\leq 4c(R_{opt}(s,D;k)) \quad \text{by (2).} \end{aligned}$$

Now consider the running time of Algorithm B. First, notice that the shortest path between a pair of vertices in G can be found in time $O(|V|^2)$, thus the auxiliary graph G_a at (B1) can be constructed in time $O(|D||V|^2)$. Secondly, the Hamilton circuit H_c of G_a can be produced in time $O((|D|+1)^3)$ (refer to [12]). Thirdly, the directed trail T at (B3) and its partition into m subtrails at (B4) can be obtained in time $O(|V|)$. In the end, every k-drop tree can be produced in time $O(|V|^2)$. Therefore, Algorithm B outputs $R_B(s,D;k)$ in time $O(|D||V|^2)$. The proof is finished. ■

When $m = 1$ the k-MTR problem becomes the Steiner tree problem in networks, which has an approximation algorithm with performance ratio less than 2 [4]. When $m = 2$, Algorithm B can be modified slightly so that its approximation ratio 4 could be reduced.

Corollary 1 *For the case of* $m = 2$, *there is a polynomial-time algorithm that produces a k-MTR whose cost is at most three times that of the optimal k-MTR and the minimum Steiner tree of* $D \cup \{s\}$.

Proof At (B4) of Algorithm B, after partitioning $\mathcal{T}$ into 2 subtrails $\mathcal{T}_0$ and $\mathcal{T}_1$, we construct a k-MTR consisting of two k-drop trees,

$$\begin{aligned} T_1 &= \{s \to d_1 \to d_2 \to \cdots \to d_{k-1} \to d_k\}, \\ T_2 &= \{s \to d_{|D|} \to d_{|D|-1} \to \cdots \to d_{k+2} \to d_{k+1}\}. \end{aligned}$$

By applying the same argument used in the proof of Theorem 2, we deduce $c(R_B(s,D;k)) \leq 3c(R_{opt}(s,D;k))$. Moreover, since the minimum Hamilton circuit of G_a has cost less than two times that of the minimum Steiner tree T_{opt} of $D \cup \{s\}$ in G, we have $c(R_B(s,D;k)) \leq 3c(T_{opt})$. ■

V. WAVELENGTH ASSIGNMENT FOR MULTI-TREES

In this section, we consider how to assign a wavelength to each of k-drop trees such that two trees are assigned with distinct wavelengths if they share a common link. This problem can be reduced to the *vertex-coloring problem* of graphs as follows: Construct a graph $G'(V', E')$ such that V' is a set of k-drop trees and there is an undirected edge in E' between two nodes in V' in graph G' if the corresponding k-drop trees share a common physical link in $G(V,E)$.

Since the vertex-coloring problem is NP-hard which has no polynomial-time algorithm with a constant approximation performance ratio [2], we use a simple heuristic based on the sequential coloring approach proposed in [11].

Algorithm C *Wavelength Assignment Algorithm*

(C1) Construct graph $G'(V', E')$.
(C2) Choose a vertex in V' that has the least degree.
(C3) Find a maximal set of vertices in current V' that are not adjacent to the selected vertex and there is no edge between any pair of vertices in the set.
(C4) Assign a wavelength to the vertices in the set and removes them from V'.
(C5) Repeat (C2-4) until every vertex V' is assigned a wavelength.

Theorem 3 *Given a k-MTR* $R(s,D;k)$ *that has* m *k-drop trees, Algorithm C assigns wavelengths to it in time* $O(m^2|V|^3)$.

Proof At Step C1, $G'(V', E')$ can be produced in time $O(m^2|V|^2)$, because T_i has at most $(|V| - 1)$ edges. In the loop of Step C2-5, all vertices in V' are assigned a color one by one, and two vertices are assigned the same color if and only if they are not adjacent. Choosing a vertex of the least degree and finding a maximal set both can be done by checking adjacency among vertices in V'. Thus Algorithm C can finish wavelength assignment in time $O(m^2|V|^3)$. ■

VI. Simulation Study

In the previous sections, we have proposed algorithms for constructing a k-MTR and allocating wavelengths to them. At the same time we have made theoretical analysis of their performances, respectively. In this section, we will simulate the proposed algorithms in various network environment.

A. Simulation Model

In the simulations we use two network topologies. One is the backbone of NSFnet [3]. It consists of 14 nodes representing 14 states in the USA. The network cost of a link joining two states is the driving distance between them. The other is generated by using the approach introduced in [15] to emulate wide-area sparse networks deliberately. 100 nodes are distributed randomly over a rectangular coordinate grid. Each node is placed at a location with integer coordinates. A link between two nodes u and v is added by using 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 $0 < \lambda, \gamma \leq 1$. Larger values of λ produce graphs with higher link densities, while small values of γ increases the density of short links relative longer ones. In our simulations, γ and λ both are set to 0.9. As a result, the nodes in generated graphs have average degrees of 6.83. Cost function c on an edge in the generated graphs is the distance between its two end nodes on the rectangular coordinated grid.

The multicast connections are generated randomly. The source node s and set D of destination nodes are randomly picked up from the nodes in the networks.

There are three objectives to conduct simulation work:

(i) To determine how much k-MTR can save the network cost. For this purpose, we use the cost of Steiner trees as a performance benchmark. A Steiner tree is the optimal tree (in terms of minimal network cost) for a multicast connection for the case when k is infinite (i.e., a Steiner tree contains all destinations). Since it is NP-hard to compute Steiner trees, we use a simple 2-approximation algorithm proposed in [4]. It works as follows: (1) construct a complete weighted graph G_a whose vertex-set is $\{s\} \cup D$ and each edge (u, v) in G_a is the shortest path between u and v in original graph G; (2) compute a minimum spanning tree T_{min} on G_a; and (3) obtain a Steiner tree T_S by substituting each edge in T_{min} by the corresponding shortest path in G.

(ii) To see the advantages of k-MTR over other multi-drop routing models, we use the *k-drop Multi-Path Routing* (k-MPR for short) as a performance benchmark. k-MPR model [8,9] is for the networks where switching nodes do not have splitter to split signals but are still drop-off capable. A k-drop path is a path that is from the source to some destinations in which at most k destinations are designated to receive the data. In the k-MPR model, multicast routing is to find a set of k-drop paths such that every destination is designated to receive data in a k-drop path in the set and the total cost is minimal. We compare our method for k-MTR with an algorithm for the k-MPR model, which is a 4-approximation algorithm proposed in [8]. It is based on the Steiner tree algorithm described in the above (i).

(iii) To see how much k-MTR can reduce the number of wavelengths used. For this purpose, we use the lightpath model (i.e., $k = 1$) as a performance benchmark. In the lightpath model, a route of a multicast connection is a tree of lightpaths (*lightpath-tree* for short), where there is a lightpath from the source to each of the destinations. We simply use the shortest path from the source to a destination as a lightpath.

We simulate the network cost and the number of wavelengths used against two parameters: $|D|$ the number of destination nodes and the drop number k in k-MTR and k-MPR. The results presented in the figures are the mean values of 50 simulation runs.

B. Analysis of Simulation Results

Fig. 1-4 are the results in the NSFnet. Fig. 1 and Fig. 3 show the network costs and the number of wavelengths used by lightpath-tree, Steiner tree, k-MTR and k-MPR against the drop number k, respectively. Two sets of results are displayed. The solid line is for the case of 13 destinations and the dashed line is for 7 destinations. Notice that the network cost and the number of wavelengths used by lightpath-tree and Steiner tree do not vary with the drop number k, i.e., they are constant (shown as horizontal lines). Fig. 2 and Fig. 4 show the network costs and the number of wavelengths used by these four routing methods against the number of destinations when $k = 4$, respectively. Notice that the network costs and the number of wavelengths used by all routing methods increase proportionately as the number of the destinations increases. Fig. 5-8 show the parallel results of randomly generated network.

In the simulation, when $k = 2$, k-MTR and k-MPR are produced by using methods described in the proofs of Theorem 1 of [8] and Algorithm A. Both of them have minimal costs in each case, respectively. When $k \geq 3$, k-MTR and k-MPR are produced by applying the Steiner tree based 4-approximation algorithm of [8] and Algorithm B, respectively. In Fig. 1, Fig. 3, Fig. 5, and Fig. 7, accordingly we do not join the results of $k = 2$ with the results of $k = 3$. From those eight figures we can draw the following conclusions.

(1) The network cost of lightpath-tree is about two times that of k-MTR. These ratios are independent of the size of a multicast connection and very stable. This can be observed from Fig. 1-2 and Fig. 5-6. In addition, the network cost of Steiner tree routing is about two times that of k-MTR. In fact, when k becomes large enough, the average performance of

our algorithm is much better than the guaranteed ratio 4. This can be observed from Fig. 1 and Fig. 5.

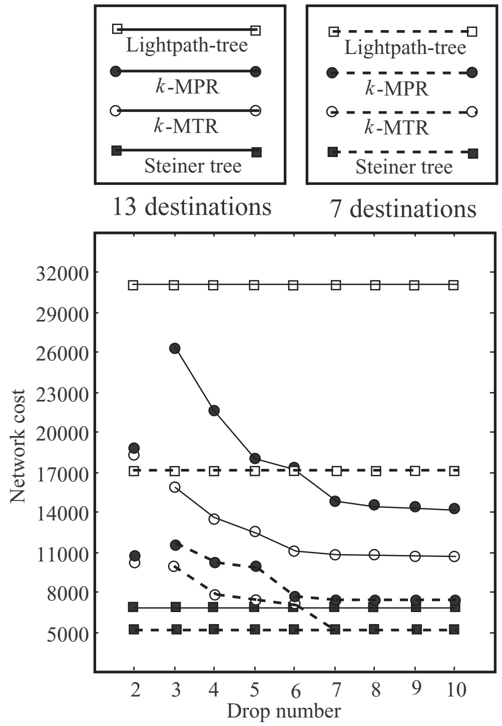

Fig. 1. Network costs vs. drop numbers in NSFnet.

(2) The network costs of k-MTR decrease as k increases. However, increasing k is not very effective in decreasing the network cost. This can be seen from Fig. 1 and Fig. 5. The reason behind this interesting result is that when k becomes larger, although k-MTR will consist of less number of trees, each of them will become bigger so that it includes more destinations. This make each tree more costly.

(3) The number of wavelengths used by lightpath-tree is about four times that of k-MTR. This ratio is independent of the size of multicast connection and very stable. This can be observed from Fig. 4 and Fig. 8.

(4) The number of wavelengths used by k-MTR decreases as k increases. However, increasing k is not very effective in reducing the network cost for k-MTR. This can be seen from Fig. 3 and Fig. 7. The reason is that when k becomes larger, a tree can contain more destinations, which makes a bigger tree. Therefore, trees with larger k would have more chances to share links among them, which prevent them from using the same wavelength.

(5) k-MTR is more effective in saving the wavelengths than the network cost, although it is designed for achieving minimal cost. The reason is that k-MTR of less network cost consists of less trees and less links for a tree (averagely), which results less chance of wavelength conflict (i.e., two trees share a common link, but assigned with the same wavelength).

(6) In general, k-MTR uses considerably less network cost and wavelengths than k-MPR. The reason is that k-MPR is

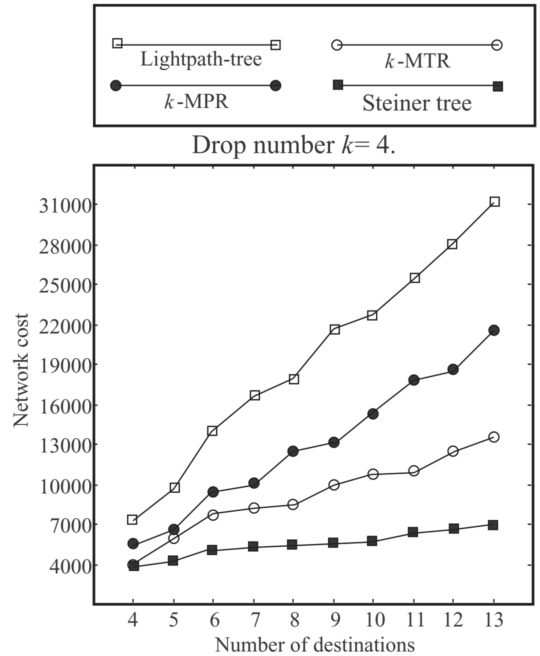

Fig. 2. Network cost vs. number of destinations in NSFnet.

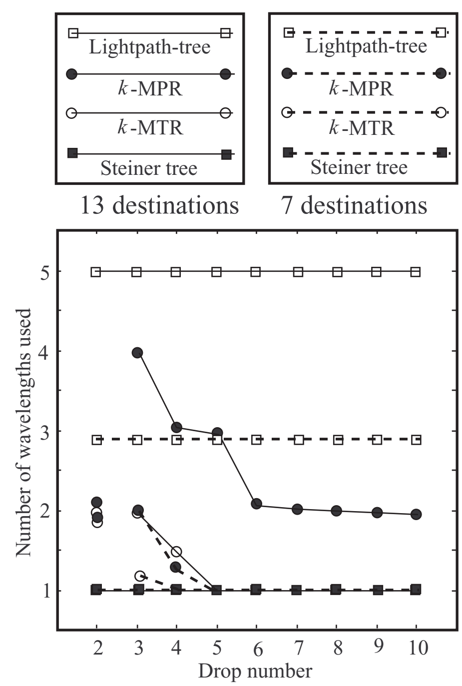

Fig. 3. Number of wavelengths vs. drop numbers in NSFnet.

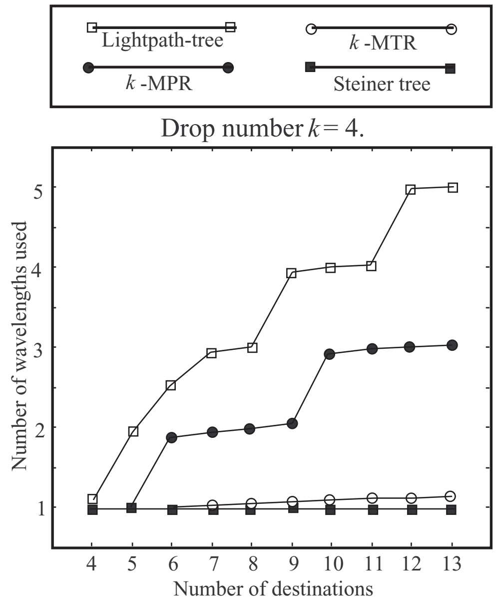

Fig. 4. Number of wavelengths vs. number of destinations in NSFnet.

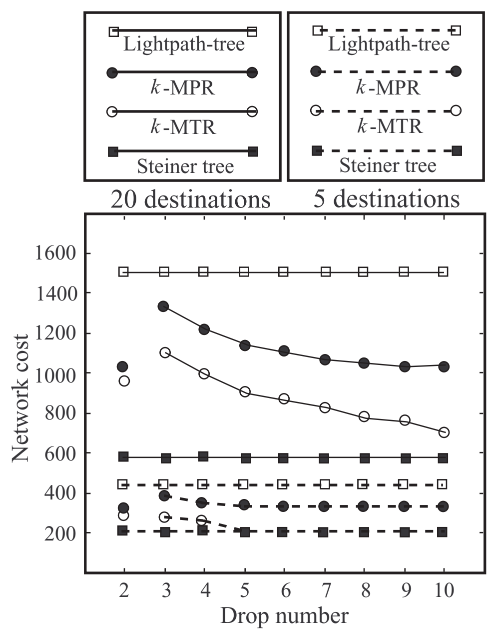

Fig. 5. Network costs vs. drop numbers in random networks.

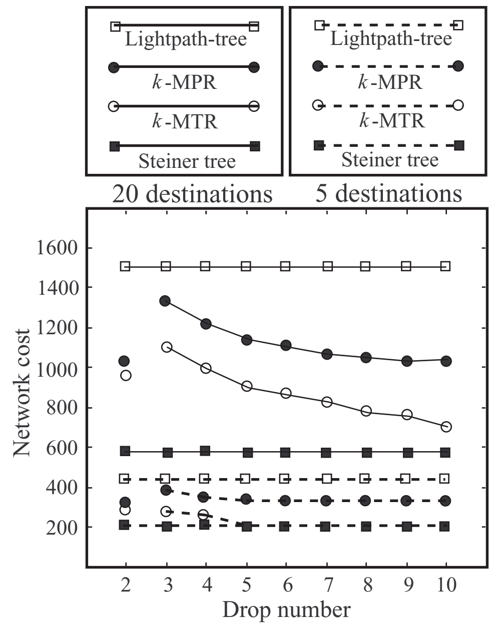

Fig. 6. Network costs vs. number of destinations in random networks.

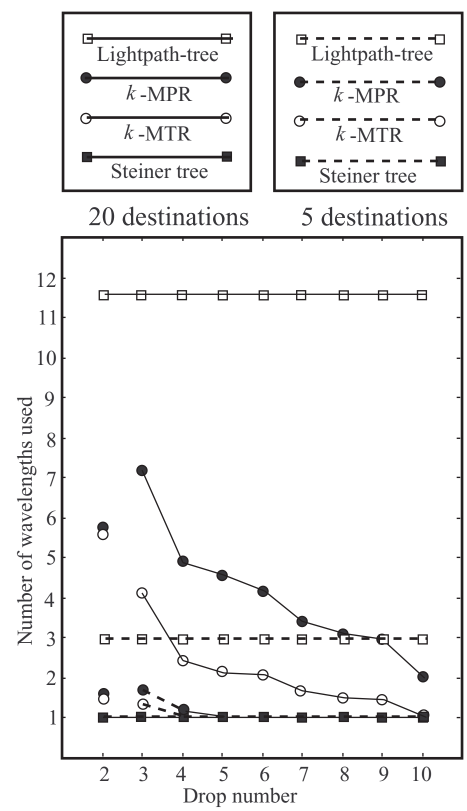

Fig. 7. Number of wavelengths vs. drop number in random networks.

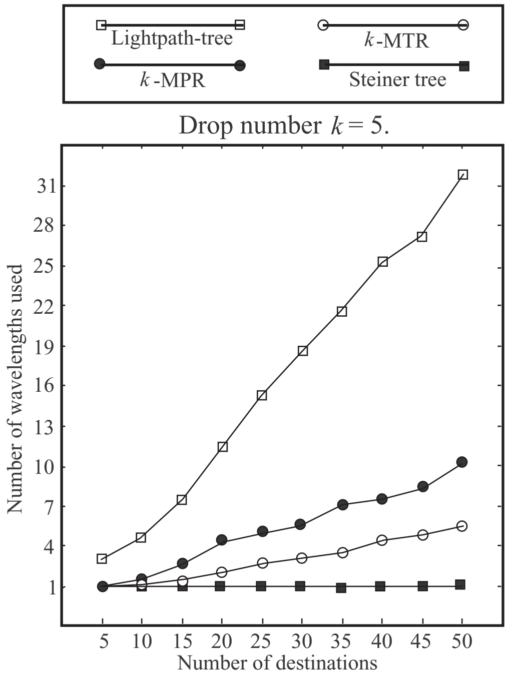

Fig. 8. Number of wavelengths vs. number of destinations in random networks.

a special case of k-MTR since every k-drop path can be considered as a k-drop tree.

VII. Conclusions

In this paper, we have studied multicast routing problem under the k-drop multi-tree model. We have proposed a polynomial-time optimal algorithm for the cases when $k \leq 2$ and a 4-approximation algorithm for the cases when $k \geq 3$. The extensive simulation studies show that our proposed algorithm can save considerable amount of network cost and wavelengths than the lightpath-tree method, and the performance is close to the Steiner tree method when k becomes large enough. Simulation results also show that the k-drop multi-tree model (k-MTR) generally outperforms the k-drop multi-path model (k-MPR).

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant No. 60273071,70221001, 60373012. The author Xiaohua Jia also works with Computing School, Wuhan Univ, China.

References

[1] M. Ali and J. S. Deogun, Power-efficient design of multicast wavelength-routed networks, *IEEE Journal on Selected Areas in Communications*, 18 (10) (2000), 1852-1862.

[2] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, *Complexity and Approximatio: Combinatorial Optimization Problems and Their Approximability Properties*, Springer-Verlag: Berlin and Heidelberg, 1999.

[3] D. Banerjee and B. Mukherjee, A practical approach for routing and wavelength assignment in large wavelength-routed optical networks, *IEEE Journal on Selected Areas in Communications*, 14 (5) (1996), 903-908.

[4] K. Bharath-Kumar and J. M. Jaffe, Routing to multiple destinations in computer networks, *IEEE Transactions on Communications*, 31 (3) (1983), 343-351.

[5] N. Christonfides, Worst-case analysis of a new heuristic for the traveling salesman problem, *Technical Report*, GSIA, Carnegie-Mellon University, 1976.

[6] V. Chvátal, A greedy heuristic for the set-covering problem, *Mathematics of Operations Research*, 4 (3) (1979), 233-235.

[7] S.-X. Gao, X.-H. Jia, X.-D. Hu, and D.-Y. Li, Wavelength requirements and routing for multicast connections in lightpath and light-tree models of WDM networks with limited drops, *IEE Proceedings on Communications*, 148 (6) (2001), 363-367.

[8] J. Gu, X.-D. Hu, and M.-H. Zhang, Algorithms for multicast connection under multi-path routing model, *Information Processing Letters*, 84 (1) (2002), 31-39.

[9] R. L. Hadas, Efficient collective communication in WDM networks with a power budget, in proceedings of *IEEE International Conference on Computer Communication Network*, (2000), 612-616.

[10] Xiaohua Jia, X.-D. Hu, M. Lee, D.-Z. Du, and J. Gu, Optimization of Wavelength Assignment for QoS Multicast in WDM Networks, *IEEE Trans on Communications*, 49(2)(2001), 341-350.

[11] D. W. Matula, G. Marble, and J. D. Isaacson, Graph coloring algorithms, in *Graph Theory and Computing*, R. C. Read, Ed., New York and London: Academic, 1972, 109-122.

[12] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[13] L. H. Sahasrabuddhe and B. Mukherjee, Multicast routing algorithms and protocols: a tutorial, *IEEE Network*, 14 (1) (2000), 90-102.

[14] L. H. Sahasrabuddhe and B. Mukherjee, Light-trees: optical multicasting for improved performance in wavelength-routed networks, *IEEE Communications Magazine*, 37 (2) (1999), 67-73.

[15] B. M. Waxman, Routing of multipoint connections, *IEEE Journal on Selected Areas in Communications*, 6 (1988), 1617-1622.

[16] Guoliang Xue and R. Banka, Bottom-up construction of dynamic multicast trees in WDM networks, *Proceedings of IEEE International Conference on Performance, Computing, and Communications Conference*, 2003, 49 -56.

[17] X. Zhang, J. Wei, and C. Qiao, Constrained multicast routing in WDM networks with sparse light splitting, *Proceedings of 19th IEEE Conference on Computer Communications*, March 26-30, 2000.