

Routing and Wavelength Assignment for WDM Multicast Networks

Jingyi He[†] S.-H. Gary Chan[‡] Danny H.K. Tsang[†]

[†]Department of Electrical and Electronic Engineering

[‡]Department of Computer Science

Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong

Abstract—The multicast routing and wavelength assignment (MC-RWA) problem is generally studied with the objective to maximize the number of multicast groups admitted, or equivalently, to minimize the call (or session) blocking probability given a certain number of wavelengths. While this approach is sound, a better objective is to maximize the total number of *users* served (i.e., minimizing the user blocking probability) by allowing part of a multicast group to be admitted. We present for the first time a formulation of the MC-RWA problem with such an objective. The formulation is a nonlinear integer program, which in general is complex to solve. We hence propose a heuristic algorithm based on linear programming (LP). We further develop a simpler MAX-FIRST algorithm, which achieves almost the same performance as the LP algorithm. These algorithms are for static MC-RWA, where the multicast trees are predetermined and cannot be changed during the wavelength assignment. We extend the algorithms to dynamic MC-RWA, where new multicast trees can be built for unserved groups. We finally present upper and lower bounds on the user blocking probability for the static MC-RWA.

I. INTRODUCTION

Wavelength-division multiplexing (WDM) is an effective technique to make use of the large amount of bandwidth in optical fibers to meet the bandwidth requirement of applications. It is generally believed that the next-generation Internet will be largely based on a WDM backbone [1]. Multicast is the transmission of information from one source to multiple destinations simultaneously, i.e., a one-to-many communication technique. Many broadband services such as video conferencing, distance learning, and webcasting employ multicasting for data delivery. The support of multicast in the future WDM networks is therefore essential for these applications. Multicasting in wide area WDM networks has been previously studied in two different contexts. One focuses on migrating IP multicast protocols into optical burst/label switching (OBS/OLS) networks [2], [3], [4], while the other one focuses on the configuration of the WDM layer in wavelength-routed networks and is usually referred to as the multicast routing and wavelength assignment (MC-RWA) problem [5], [6], [7], [8]. In this paper, we will merely consider the latter one.

The MC-RWA problem can be stated as follows: given a limited number of wavelengths and a set of multicast calls, maximize the number of multicast calls admitted, or equivalently, minimize the call (or session) blocking probability, under the constraint that each multicast tree can be assigned only one wavelength (the wavelength continuity or the so-called “light-tree” constraint). The routing and wavelength assignment (RWA) problem consists of two basic subproblems, namely, routing problem and wavelength assignment problem. They can be either coupled or uncoupled. In the uncoupled case, a route (or a tree) is first obtained followed by wavelength assignment. Therefore, the trees cannot be changed during the wavelength assignment. This is usually called static RWA. In the coupled case, on the other hand, the routes are decided according to the state of the wavelength assignment. This is usually called dynamic (or adaptive) RWA.

For the MC-RWA problem stated before, if the whole multicast group cannot be admitted completely, it will be blocked, even though a large portion of the group members may be served. For such appli-

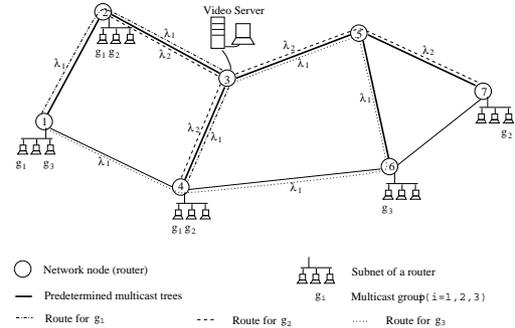


Fig. 1. An example of single-source MC-RWA, with 2 wavelengths and 3 multicast groups.

cations as stored video services, it is more reasonable to serve the part of users that can be admitted rather than block the whole group. This paper addresses the MC-RWA problem with the objective to maximize the total number of served users instead of admitted multicast groups (i.e., minimizing the user blocking probability instead of session blocking probability). We present, for the first time, a formulation of such a problem for the static MC-RWA case, which is a nonlinear integer program. We then study the problem in the context of single-source applications such as stored video services, software distribution, and webcasting.

We show in Fig. 1 an example of a single-source video system. The video server stores all the video contents of interest to many users distributed in the network. The videos are delivered to the end users by means of multicast. All the nodes are wavelength routers. There are 2 wavelengths in the system, λ_1 and λ_2 , and 3 multicast groups, g_1 , g_2 , and g_3 . We show in the figure the multicast groups requested by the end hosts in the network: there are 3 users joining g_1 , 3 users joining g_2 , and 2 users joining g_3 . Since g_1 and g_2 have more group members than g_3 , in order to maximize the number of users served, we may first serve them (completely) using the two wavelengths. If a group is served only if it can be served completely, g_3 can not be served because no wavelength can be used to reach both node 1 and node 6, where the two group members reside. However, if partial accommodation is allowed, we can use λ_1 to reach node 6 along the predetermined multicast tree, and therefore serve one more user. Furthermore, if retreeing is allowed (i.e., dynamic MC-RWA), we can serve both of the users using λ_1 along the new multicast tree for g_3 , as illustrated by the dotted line.

As far as we know, there has been no work addressing the MC-RWA problem specifically for single-source applications. Given the complexity of the nonlinear integer program, we study the problem starting from a heuristic algorithm based on linear programs (LPs). We then propose a simple MAX-FIRST algorithm, which always serves the group with the maximum number of users that can be served. We also extend the algorithms to dynamic MC-RWA. In order to efficiently study the performance of static MC-RWA, we also provide the lower and upper bounds on the user blocking probability of the static MC-RWA.

This work was supported, in part, by the Areas of Excellence (AoE) on Information Technology funded by the University Grant Council in Hong Kong (AoE98/99.EG01), and by the Direct Allocation Grant of Hong Kong Research Grant Council (DAG99/00.EG25).

The rest of this paper is organized as follows. In Sect. II, we present a general formulation for the MC-RWA problem with the objective to minimize the user blocking probability. In Sect. III, the LP and MAX-FIRST algorithms are proposed and compared via simulation. We present the bounds on the user blocking probability of the static MC-RWA in Sect. IV and conclude in Sect. V.

II. FORMULATION OF THE MC-RWA PROBLEM

The network is represented by an undirected graph $G = (V, E)$, where V is the set of nodes with $|V| = N$ and E is the set of links with $|E| = L$. Denote the set of multicast trees for all the calls as $\{T_1, T_2, \dots, T_M\}$ (known *a priori* given the tree-building algorithm), where M is the total number of multicast calls. Since a node can have users in different multicast groups in its subnet, we define a *virtual node* for each of the existing multicast groups at a real node, representing all the users of that group in the real node's subnet. Denote N_k the number of virtual nodes in T_k , $k = 1, 2, \dots, M$, and the total number of virtual nodes $N' = \sum_{k=1}^M N_k$. We index the virtual nodes and denote r_i the route reaching virtual node i from its source, and u_i the number of users it represents, for $i = 1, 2, \dots, N'$. The set of wavelengths is denoted by $\{\lambda_1, \lambda_2, \dots, \lambda_W\}$, where W is the total number of wavelengths. We further define the following matrices. Let $A = (a_{ij})$ be the $N' \times W$ indication matrix, where

$$a_{ij} = \begin{cases} 1, & \text{if } r_i \text{ is assigned } \lambda_j, \\ 0, & \text{otherwise.} \end{cases}$$

Rewrite A as a column of submatrices:

$$A = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_M \end{pmatrix},$$

where A_k is an $N_k \times W$ matrix that represents the wavelength assignment of T_k , $k = 1, 2, \dots, M$. Let $B = (b_{ij})$ be the $N' \times L$ indication matrix, where

$$b_{ij} = \begin{cases} 1, & \text{if link } j \text{ is on } r_i, \\ 0, & \text{otherwise.} \end{cases}$$

Rewrite B as a column of submatrices:

$$B = \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_M \end{pmatrix},$$

where B_k is a $N_k \times L$ matrix that represents the connections of T_k , $k = 1, 2, \dots, M$. $B_k^T A_k$ is then a $L \times W$ matrix indicating the wavelength usage of T_k on all the links, for $k = 1, 2, \dots, M$. To ensure that no two paths having shared links while belonging to different groups are assigned the same wavelength, all the $M \times L \times W$ matrices must not have overlapping non-zero elements, i.e., if the (i, j) element of $B_k^T A_k$ is non-zero, then the (i, j) elements of all the other $M - 1$ matrices must all be zeros. Since each multicast tree can be assigned at most one wavelength, $\mathbf{1}_{N_k}^T A_k$ must be a row vector of at most one non-zero element, where $\mathbf{1}_n$ denotes an n -element column vector of all 1s. (Similarly, we will use $\mathbf{1}_{L \times W}$ to denote an $L \times W$ matrix of all 1s.) We introduce the following two functions: $f(\mathbf{X})$ is a function that replaces the non-zero elements in matrix \mathbf{X} with 1s, and $g(\mathbf{x})$ is a

function that counts the number of non-zero elements in vector \mathbf{x} . We then formulate the static MC-RWA problem as follows:

$$\begin{aligned} \text{Find:} & \quad a_{ij}; \\ \text{To maximize:} & \quad \sum_{i=1}^{N'} \left(u_i \sum_{j=1}^W a_{ij} \right); \\ \text{Subject to:} & \quad a_{ij} \in \{0, 1\}; \\ & \quad \sum_{k=1}^M f(B_k^T A_k) \leq \mathbf{1}_{L \times W}; \\ & \quad g(\mathbf{1}_{N_k}^T A_k) \in \{0, 1\}, \quad \text{for } k = 1, \dots, M. \end{aligned}$$

The solution of the above formulation serves the maximum number of users by allowing partial multicast trees to be accommodated. But the optimization problem is hard to solve, not only because of the nonlinearity of the program (which comes from the function $f(\mathbf{X})$ and $g(\mathbf{x})$), but also because of the large number of the decision variables when the number of virtual nodes is large. Therefore, we present two heuristics as given in the following section.

III. MC-RWA FOR A SINGLE-SOURCE SYSTEM

We now study the MC-RWA problem in the context of single-source applications. For static MC-RWA, given the multicast requests, the multicast trees can be calculated using any existing algorithm such as KMB algorithm, minimum spanning tree (MST) heuristic, or the shortest path tree (SPT) heuristic [9]. We use the SPT heuristic, because it is simple and offers low delay from the server to the user. We first present two heuristic algorithms for minimizing the user blocking probability in the static MC-RWA problem. The first one is based on linear programming, while the second one is based on the observation that the group of the most users should be served first. We then extend the algorithms to dynamic MC-RWA. We finally present our simulation results.

A. Linear Program (LP) Algorithm

The basic idea of this algorithm is that we maximize the number of users that can be served by first accommodating only complete multicast groups, then trying to serve as many users as possible by allowing partial accommodation. The algorithm is based on two integer linear programs (ILPs), which are shown in the following. We reuse some of the symbols in Sect. II with some minor modifications. Let $A = (a_{ij})$ be the $M \times W$ indication matrix, where

$$a_{ij} = \begin{cases} 1, & \text{if } T_i \text{ is assigned } \lambda_j, \\ 0, & \text{otherwise.} \end{cases}$$

Let $B = (b_{ij})$ be the $M \times L$ indication matrix, where

$$b_{ij} = \begin{cases} 1, & \text{if link } j \text{ is on } T_i, \\ 0, & \text{otherwise.} \end{cases}$$

Let u_i be the number of users in group i . We have **Program 1**:

$$\begin{aligned} \text{Find:} & \quad a_{ij}; \\ \text{To maximize:} & \quad \sum_{i=1}^M \left(u_i \sum_{j=1}^W a_{ij} \right); \\ \text{Subject to:} & \quad a_{ij} \in \{0, 1\}; \\ & \quad \sum_{j=1}^W a_{ij} \in \{0, 1\}; \\ & \quad (B^T A)_{ij} \in \{0, 1\}. \end{aligned}$$

This ILP is for complete group accommodation. The constraint $\sum_{j=1}^W a_{ij} \in \{0, 1\}$ means that a multicast tree can be assigned at most one wavelength. The constraint $(B^T A)_{ij} \in \{0, 1\}$ ensures that no two multicast trees having shared links are assigned the same wavelength.

After assigning wavelengths to the complete multicast trees by solving **Program 1**, we next allow partial accommodation to make full use

of the free wavelengths. We update the wavelength usage information on each link and build an updated topology for each wavelength by removing from the initial network topology the links on which the wavelength has been used. Let M' denote the number of unserved groups, and reindex these groups from 1 to M' . By checking the multicast trees with the topology on each wavelength, we get an $M' \times W$ matrix $C = (c_{ij})$, where c_{ij} is the number of users that can be served if group i is assigned λ_j . Similar to matrix A , we define an $M' \times W$ matrix $A' = (a'_{ij})$, where

$$a'_{ij} = \begin{cases} 1, & \text{if group } i \text{ is assigned } \lambda_j, \\ 0, & \text{otherwise.} \end{cases}$$

We then formulate a new ILP (**Program 2**):

$$\begin{aligned} \text{Find:} & \quad a'_{ij}; \\ \text{To maximize:} & \quad \sum_{i=1}^{M'} \sum_{j=1}^W c_{ij} a'_{ij}; \\ \text{Subject to:} & \quad a'_{ij} \in \{0, 1\}; \\ & \quad \sum_{j=1}^W a'_{ij} \in \{0, 1\}; \\ & \quad \sum_{i=1}^{M'} a'_{ij} \in \{0, 1\}. \end{aligned}$$

The constraint $\sum_{j=1}^W a'_{ij} \in \{0, 1\}$ means that a group can be assigned at most one wavelength. The constraint $\sum_{i=1}^{M'} a'_{ij} \in \{0, 1\}$ means that a wavelength can be assigned to only one group. **Program 2** is solved iteratively. After each iteration, matrix C can be updated by updating the topology and user information. The iteration stops until C becomes an all-zero matrix, which means that no more users can be served. In this way, we can ensure that no wavelength is wasted on the predetermined multicast trees.

By combining these two ILPs, we propose the following LP algorithm:

```
Solve Program 1;
Initialize the matrix  $C$ ;
while ( $C$  is not all-zero) {
  Solve Program 2;
  Check the fixed trees to update  $C$ ;
}
```

It tries to maximize the total number of users served by the predetermined multicast trees using a two-step approach, and makes full use of the wavelengths in each step. Therefore, it is a near-optimal solution for the formulation given in Sect. II in the context of single-source applications.

B. MAX-FIRST Algorithm

The above algorithm requires solving ILPs, which is currently still not very efficient. We hence propose the following simpler approach, and will show that it performs almost exactly the same as the LP algorithm. The basic idea of this approach is to do the assignment iteratively and first serve the group having the largest number of users that can be served using certain wavelength at each round. We call this approach the MAX-FIRST algorithm, which is detailed as follows.

We use the same matrix C as defined in **Program 2**, which maintains the number of users that can be served if a group is assigned a certain wavelength. At the beginning of each iteration, we check the $M' \times W$ matrix C . If C is an all-zero matrix, which means that no more users can be served, the algorithm stops. Otherwise, find the largest element of C : c_{ij}^{max} . Assign λ_j to group i , and update the topology on λ_j and the unserved users in group i . Then delete row i and column j from C , and find the next largest element. Repeat until $c_{ij}^{max} = 0$, which happens under the following three circumstances: i) all the M' groups have been served (maybe partially), hence all the M' rows are deleted and C is empty; ii) all the W wavelengths have been assigned, hence all the

W columns are deleted and C is empty; or iii) the unserved groups just cannot be served by the wavelengths left, hence all elements are zero. Then check the group and topology information, and update the matrix C for the next iteration. The MAX-FIRST algorithm is summarized as follows:

```
Initialize matrix  $C$ ;
while ( $C$  is not all-zero) {
  do {
    Find the largest element  $c_{ij}^{max}$ ;
    Assign  $\lambda_j$  to group  $i$ ;
    Update the topology on  $\lambda_j$  and the
      unserved users in group  $i$ ;
    Exclude row  $i$  and column  $j$  from  $C$ ;
  }
  while ( $c_{ij}^{max} \neq 0$ );
  Check the fixed trees to update  $C$ ;
}
```

C. Dynamic MC-RWA

If fixed multicast trees are used in wavelength assignment, it is possible that no more users can be served even though there are usable wavelengths, simply because the users are unreachable along the predetermined trees. The waste of wavelengths can be avoided if new multicast trees are dynamically built based on the updated topologies on the usable wavelengths. Both the LP algorithm and the MAX-FIRST algorithm can be easily modified to allow retreeing. This is achieved by building new shortest path trees on each available wavelength in updating matrix C , instead of just checking the fixed trees. We call these two extensions the LP with retreeing algorithm and MAX-FIRST with retreeing algorithm, respectively.

D. Illustrative Simulation Results

We evaluate the proposed algorithms on the NSFNET backbone network as shown in Fig. 2. There are 14 nodes and 21 bi-directional links in the network, with the labels corresponding to the delay on each link. Among the 14 nodes, two have nodal degree 2, two have nodal degree 4, and the rest have nodal degree 3. In our simulation, a source node (i.e., the node to which the video server attaches) is first randomly chosen. The users are generated as follows. A node (excluding the source node) may have users (in any group) in its subnet with probability α , the ‘‘active’’ probability of the node.¹ An active node may have users from one to infinity, according to the geometric distribution:

$$\gamma_n = (1 - \beta)\beta^{n-1}, \quad \text{for } n = 1, 2, \dots, \quad (1)$$

where γ_n is the probability that a node has n users, and β is a parameter which determines the density of the users in a node’s subnet (the average number of users per active node, denoted by μ , is simply $\frac{1}{1-\beta}$). The source multicasts its data via multiple multicast groups, and a multicast group is chosen by a user with probability p_i , $i = 1, \dots, M$, with $\sum_{i=1}^M p_i = 1$ and $p_1 \geq \dots \geq p_M$ (p_i is also known as the ‘‘popularity’’). The popularity of the multicast groups follows the Zipf distribution in our study, which is defined as $p_i = \frac{c}{i^z}$, $i = 1, \dots, M$, where $z = 0.729$ and c is a constant determined by $\sum_{i=1}^M p_i = 1$ [10]. Our simulation shows that if a source node has a larger nodal degree, the performance in terms of blocking probability is better, while picking different nodes of the same nodal degree as the source node does not make much difference (not shown here). Hence we fix the source node at MI (of nodal degree 3: $D = 3$) for the following results.

¹The users in the subnet of the source node are not considered, because they do not count for the routing and wavelength assignment problem.

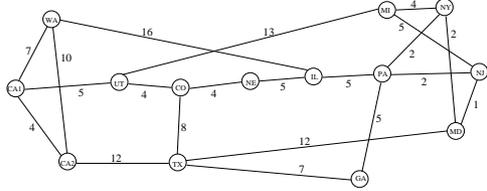


Fig. 2. NSFNET backbone network topology used in the simulation.

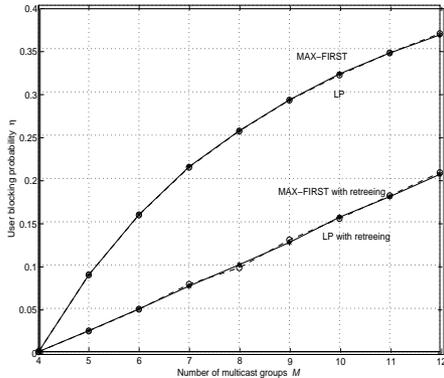


Fig. 3. η versus M for all the four algorithms ($W = 4$).

We consider a system of 4 wavelengths ($W = 4$). We always assume that $W \leq M$, because all the groups will always be completely served if otherwise, and hence there is no wavelength assignment issue. Our baseline system has 8 multicast groups ($M = 8$), with each node (except the source node) equally likely to be active and nonactive ($\alpha = 0.5$), and on average 10 users per active node ($\mu = 10$). We study the user blocking probability (η), which is defined as the ratio of the number of blocked users to the total number of users generated.

We first study the performance of the four algorithms we proposed with the interest in η versus M . For the LP algorithms, **Program 1** and **Program 2** are solved by a linear program solver called *lp_solve*. Figure 3 shows the simulation results, from which two important observations can be made. First, wavelength assignment with retreeing (i.e., dynamic MC-RWA) substantially outperforms its counterpart (the user blocking probability is reduced by more than half). Second, the MAX-FIRST algorithms perform almost exactly the same as the LP algorithms, no matter retreeing is used or not. This is because using linear programming contributes to better performance only when wavelengths can be reused for groups having no common links. However, it is not very likely for this to happen in a single-source system. In the remainder of this section, we will only focus on the MAX-FIRST algorithms.

Figure 4 shows η versus μ . As μ is reasonably large, the user blocking probability remains almost unchanged. This is expected because when the number of users is large, their distribution in different multicast groups conforms more strictly with the popularity distribution, and hence the results are steady. In contrast, when μ is small, the users actually generated are more likely in the popular groups, which are served with priority. Therefore, the blocking probability is lower.

Figure 5 shows the results of η versus α . As α increases, i.e., each node is more likely to be active, the blocking probability of the retreeing scheme increases much faster than the fixed-tree counterpart. In other words, the performance gain from retreeing drops as the users become more densely populated in the network. This is because it is less likely to have free wavelengths to do retreeing after serving the first W groups.

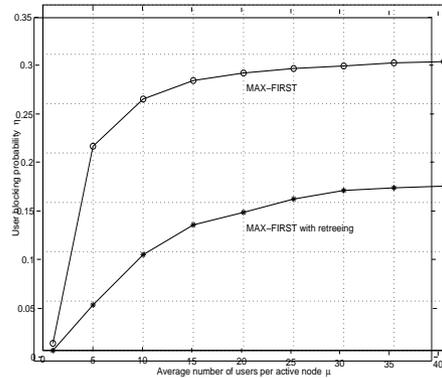


Fig. 4. η versus μ for MAX-FIRST with and without retreeing ($W = 4, M = 8$).

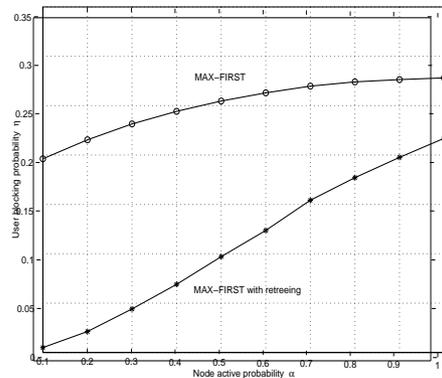


Fig. 5. η versus α for MAX-FIRST with and without retreeing ($W = 4, M = 8$).

The results shown so far all allow partial tree accommodation, while the improvement of this compared with serving only complete trees is also of interest. Figure 6 shows the blocking performance of the MAX-FIRST with retreeing algorithm, with and without partial accommodation. It is observed that allowing partial tree accommodation have better performance than serving only complete trees. Although the absolute reduction in user blocking probability is not large, we still see a 10% to 20% performance improvement of partial accommodation compared with complete accommodation.

IV. BOUNDS FOR STATIC MC-RWA

In this section, we derive the lower and upper bounds on the user blocking probability for the wavelength assignment with fixed multicast trees. Denote D the number of outgoing links (i.e., the nodal degree) of the source node, then the D outgoing links span the remaining $N - 1$ nodes in the network via the predetermined multicast trees. Denote the number of nodes spanned by link j as n_j . Then a user is down link j with probability $q_j = n_j / (N - 1)$, $j = 1, \dots, D$, because the users are uniformly distributed in the network.

The upper bound can be easily derived as follows. Since there are W wavelengths, we can always completely serve the W most popular groups using them. Hence, the worst case is that none of the remaining groups can be served (even partially). The upper bound of the blocking probability is therefore:

$$UB = 1 - \sum_{i=1}^W p_i. \quad (2)$$

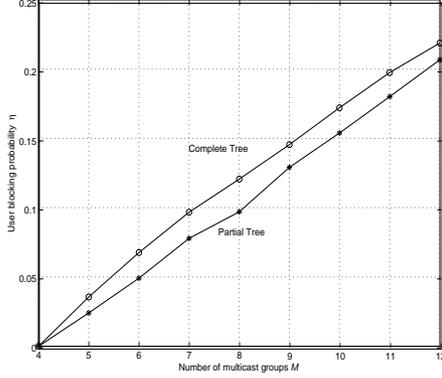


Fig. 6. Comparison of η versus M for complete tree and partial tree accommodation.

The upper bound is achieved when each of the W most popular groups is served by all the outgoing links of the source node. An implicit assumption for (2) is that group i has p_i of the total number of users, which is true on average. The following derivation of the lower bound is also based on this assumption.

We define a channel as a wavelength on an outgoing link of the source node, and there are totally $W \times D$ channels in the system. If there are free channels left after assigning wavelengths to the W most popular groups, we may serve the users in the other groups. The lower bound of the user blocking probability is achieved if we assume that the most popular one of the unserved groups can be completely served as long as a free channel exists. In order to derive the lower bound, we consider the probability that the W most popular groups only use part of the outgoing links of the source node. We first define and derive the probability that group i is served by d_i outgoing links: $\alpha_i^{(d_i)}$, where $d_i = 1, \dots, D$ and $i = 1, \dots, W$. Assume group i has m_i users, then we have the following constraints: $\sum_{k=1}^{d_i} m_i^{(k)} = m_i$ and $m_i^{(k)} \geq 1$, where $m_i^{(k)}$ is the number of users down link j_k , for $j_k = 1, \dots, D$ and $k = 1, \dots, d_i$. The two constraints are denoted by S for simplicity. There are $\binom{D}{d_i}$ different combinations of d_i links chosen from the total D outgoing links. Let $U^{(d_i)}$ be the set of all the combinations, with each element being a set of the indices of the links in the combination: $J = (j_1, \dots, j_{d_i})$. For the combination $(m_i^{(1)}, \dots, m_i^{(d_i)})$, there are $\binom{m_i}{m_i^{(1)} \dots m_i^{(d_i)}}$ ways to distribute the users, where $\binom{m_i}{m_i^{(1)} \dots m_i^{(d_i)}}$ is the multinomial coefficient defined as $\binom{m_i}{m_i^{(1)} \dots m_i^{(d_i)}} = \frac{m_i!}{m_i^{(1)}! \dots m_i^{(d_i)}!}$. The probability that group i is served by d_i outgoing links is therefore

$$\alpha_i^{(d_i)} = \sum_{J \in U^{(d_i)}} \left(\sum_S \left(\binom{m_i}{m_i^{(1)} \dots m_i^{(d_i)}} \prod_{k=1}^{d_i} q_{j_k}^{m_i^{(k)}} \right) \right). \quad (3)$$

For $d_i = 1, \dots, D$ and $i = 1, \dots, W$, there are D^W different combinations of the d_i s, which are denoted by a set S' . The probability with which a combination happens is $\prod_{i=1}^W \alpha_i^{(d_i)}$. The number of free channels left after assigning wavelengths to the first W groups is $DW - \sum_{i=1}^W d_i$. In the best case, each of these channels is used to serve a complete group, hence the total number of groups that can be served (including the first W groups) is $I = \min(M, (DW - \sum_{i=1}^W d_i))$, where $\min(x, y)$ means the smaller one of x and y . The lower bound of the user blocking probability is therefore simply

$$LB = 1 - \sum_{S'} \left(\prod_{i=1}^W \alpha_i^{(d_i)} \sum_{i=1}^I p_i \right), \quad (4)$$

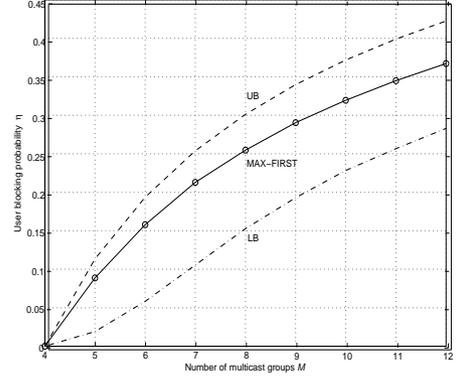


Fig. 7. The upper and lower bounds on the user blocking probability for static MC-RWA and the actual simulation result.

which is now a function of the m_i s. Let m denote the total number of users in the network. Based on our assumption, $m_i = m p_i$ with $m = (N - 1) a \mu$ being the average value of the total number of users. The rounding errors are ignored here. We compare the performance of the MAX-FIRST algorithm with the bounds in Fig. 7, which shows that the bounds are reasonably tight.

V. CONCLUSIONS

We have studied the multicast routing and wavelength assignment (MC-RWA) problem in this paper, with a special focus on minimizing the user blocking probability instead of the session blocking probability for single-source applications. We present for the first time a formulation for the static MC-RWA with such an objective. Since the formulation is a nonlinear integer program which is complex to solve, we propose a near-optimal solution of the problem using a two-step approach based on linear programming. We also propose a simpler MAX-FIRST algorithm, which always serves the group with the maximum number of users that can be served. These algorithms have also been extended to dynamic MC-RWA, where new multicast trees can be built for the unserved groups. Simulation results show that the simple MAX-FIRST algorithms can achieve almost the same performance as the more complex LP algorithms, and hence are suggested for such systems. While retreeing can substantially improve the performance compared with using fixed trees, partial tree accommodation can further reduce the user blocking probability. Reasonably tight bounds on the user blocking probability in the case of static MC-RWA have also been given.

REFERENCES

- [1] B. Mukherjee, *Optical Communication Networks*. McGraw-Hill, 1997.
- [2] C. Qiao, M. Jeong, A. Guha, X. Zhang, and J. Wei, "WDM multicasting in IP over WDM networks," in *Proc. International Conference on Network Protocols (ICNP)*, pp. 89–96, 1999.
- [3] X. Zhang, J. Wei, and C. Qiao, "On fundamental issues in IP over WDM multicast," in *Proc. International Conference on Computer Communications and Networks (IC3N)*, pp. 84–90, 1999.
- [4] M. Jeong, Y. Xiong, H. C. Cankaya, M. Vandenhouste, and C. Qiao, "Efficient multicast schemes for optical burst-switched WDM networks," in *Proc. IEEE International Conference on Communications (ICC)*, pp. 1289–1294, 2000.
- [5] G. Sahin and M. Azizoglu, "Multicast routing and wavelength assignment in wide area networks," in *Proc. SPIE Vol. 3531*, pp. 196–208, 1998.
- [6] R. K. Pankaj, "Wavelength requirements for multicasting in all-optical networks," *IEEE/ACM Transactions on Networking*, vol. 7, no. 3, pp. 414–424, 1999.
- [7] Y. Sun, J. Gu, and D. H. K. Tsang, "Routing and wavelength assignment in all-optical networks with multicast traffic," in *Proc. ITC-16*, 1999.
- [8] L. H. Sahasrabudde and B. Mukherjee, "Light-trees: Optical multicasting for improved performance in wavelength-routed networks," *IEEE Communications Magazine*, vol. 37, no. 2, pp. 67–73, 1999.
- [9] F. K. Hwang, D. S. Richards, and P. Winter, *The Steiner Tree Problem*. Elsevier Science Publishers B. V., 1992.
- [10] A. Dan, D. Sitaram, and P. Shahabuddin, "Dynamic batching policies for an on-demand video server," *ACM/Springer Multimedia Systems*, vol. 4, pp. 112–121, June 1996.