Joint Optimization of Scheduling and Multicast Trees by Column-Generation

K.R. Krishnan, David Shallcross, Latha Kant

Joint Optimization of Scheduling and Multicast Trees by Column-Generation

K. R. Krishnan, David Shallcross, and Latha Kant
Telcordia Technologies, Piscataway, NJ 08854
{krk, davids, lkant}@research.telcordia.com

Abstract: **In a wireless network, transmissions from the various nodes have to be scheduled so as to avoid mutual interference. The pattern of interference induced by active transmissions depends on the routes along which the link-transmissions have to be scheduled; i.e., the interference that scheduling has to accommodate depends on the routing. Both routing and scheduling are mechanisms to promote the efficient use of network capacity, and in view of their interdependence, it is important to consider their joint optimization. Considering one without the other can create limitations for each function and is non-optimal. We develop an algorithm based on the column-generation technique of Linear Programming for the joint optimization of routing and scheduling for multicast flows for maximizing network capacity, and demonstrate the benefit of the joint optimization in increased capacity over the case where the routing and scheduling are separately considered.**

Keywords - wireless networks; capacity; routing; scheduling; multicast

1. Introduction

The intrinsic 'broadcast' nature of wireless transmission makes multicasting a natural application for wireless networks. In this paper, we develop an algorithm based on the column-generation technique of Linear Programming for the joint optimization of routing and scheduling for multicast flows for maximizing network capacity, and demonstrate the benefit of the joint optimization in increased capacity over the case where the two functions are separately considered.

In a TDMA (Time Division Multiple Access) wireless network, transmissions from the various nodes have to be *scheduled* so as to avoid mutual interference. Thus, traffic is sent from origin to destination by scheduling the flow on each hop of its route. The pattern of interference created by active transmissions depends on the routes along which the link-transmissions have to be scheduled; i.e., the interference that scheduling has to accommodate depends on the routing. Owing to this connection between routing and scheduling, it is important, for efficient network use, to consider the *joint* optimization of routing and scheduling. Considering one without the other can create limitations for each function and is non-optimal, as was shown in [1] for the case of unicast flows.

The 'capacity' of a wireless network with respect to multicast flows has been investigated in several recent papers [2 - 7], which extend the unicast-capacity results of Gupta and Kumar [8] to derive bounds on multicast flows under certain formulations of what is meant by 'capacity', with assumptions on network topology and loads. In contrast to the notion of capacity formulated in those papers, we work with what might be termed an 'operational' definition of network capacity, stated with reference to the actual traffic demands placed on a given network. According to this 'operational' view, maximizing the capacity of a network means minimizing the fraction of active transmission time needed in the network to carry the specified traffic demands, or, equivalently, *maximizing the inactive fraction of transmission time*. The inactive fraction of time can then be thought of as the 'excess' capacity available in the network after meeting the traffic demands placed on it. Our interest in this paper is in an algorithm for the joint optimization of paths and link-transmission schedules for multicast flows in a network so as to maximize the excess capacity available in the network, using the conventional implementation of multicast in which an intermediate node replicates and forwards the information it receives to successor nodes, and not the more general case of 'network coding'.

A multicast session is described by the triplet (s, D, λ), where s is the source-node, D is the set of destination nodes, and λ is the rate of the flow from s to each of the destinations in D. A route for a multicast flow consists of a 'tree' rooted at s, with leaves at the destination nodes D. The routing we consider allows for the traffic of a multicast session to be split among multiple trees. A direct formulation of the joint optimization problem becomes intractable in various situations in networks of realistic size. If a solution is feasible with a single tree for each multicast session, the optimum choice of tree becomes a mixed-integer program. When multiple trees have to be used for flows, the problem can become intractable owing to the large number of trees one may have to consider. Our approach is to formulate the joint optimization problem as a Linear Program and then investigate the possibility of using column-generation [9] successively to improve on an initial choice of multicast trees for the multicast sessions. In particular, at each step, a solution to the LP allows us to determine, with the use of dual variables, the conditions under which column-generation yields a solution that can improve on the multicast capacity of the initial choice of multicast trees. Thus, the dual variables guide us in the discovery of new candidates for paths for multicast flows in conjunction with the requirements of scheduling, avoiding the need for generating a large number of trees without the assurance that they contribute to improving a solution. We demonstrate the significant capacity gains made by our method in three examples of multicast flows in a 10-node network over the initial case of an optimized transmission schedule for an arbitrary choice of multicast tree.

Previous Work

One finds various approaches in the literature to the problem of finding paths in a wireless network for multicast traffic and for scheduling hop-by-hop transmissions. One approach is to determine [10, 11], for each multicast session, a suitable directed tree in the network, rooted at the source, whose leaf-nodes are the destinations of the multicast. It is possible that nodes other than the source-node and the destination-nodes also appear in the tree, serving as intermediate relay nodes. Thus, the multicast tree is a Steiner tree in the network. Algorithms, exact or heuristic, can be used for determining 'minimum-cost' Steiner trees, where, for example, 'cost' could be the total Euclidean length of links in the tree. However, selecting paths without reference to the need to schedule transmissions to minimize interference could lead to non-optimal choices, as was shown in [1] for the case of unicast flows. Another approach is to select, at regular intervals, the multicast tree to use *from a set of pre-specified trees* [4], so as to minimize queue lengths in buffers at the time of selection. A third approach is not to bother with constructing a tree at all [5], but rely, instead, on network connectivity and a transmission scheme that uses 'minislots' within each time-slot to relay each packet to its various destinations in bounded time. The paper [7] considers a protocol for scheduling hop-by-hop multicast transmission in which a sending node is allowed to transmit whenever the number of its non-interfering receivers exceeds a certain threshold.

After developing the traffic equations for multicast flows in Section 2, we consider in Section 3 the formulation of the constraints introduced by the need for scheduling transmissions to avoid interference. However, there is an important difference between a unicast flow and a multicast flow as to what constitutes interference. Wireless transmission is inherently broadcast in its nature, since a single transmission by a node can be 'heard' by many neighbors, which, in fact, could be a *helpful* feature in directing multicast flows. This point is discussed in Section 3 and leads to the definition of a 'multicast link'– a set of links belonging to a multicast tree and incident at a common node. Their simultaneous transmissions advance the multicast flow and do not constitute interference with one another, and can be treated as a single 'link' for the purpose of the multicast.

Then, in Section 4, we first examine a Linear Program formulation for optimizing the transmission schedule when the routing is pre-specified, and take up the joint optimization of routing and transmission schedule in Section 5. In Section 6, we show the application of column-generation and dual variables to the joint optimization problem, and state the proposed algorithm in Section 7. Section 8 presents numerical results for several examples to show the effectiveness of the proposed method, and our conclusions are summarized in Section 9.

2. Traffic Flow Equations and Constraints

Suppose there are N nodes, L links, and n multicast sessions in the network. The traffic λ_k of multicast session k uses τ_k different trees $[t_{k1},\cdots,t_{k\tau_k}]$, with the traffic being divided among those trees in the proportions $\{\alpha_{k1},\cdots,\alpha_{k\tau_k}\}$, with $\sum_{j=1}^{\tau_k}\alpha_{kj}=1$.

Let

$T=\sum_{k=1}^{n}\tau_k$ = total number of multicast trees used in the network.

The mapping of the traffic from the multicast trees onto links of the network is specified by the following expression for load f_l on link l:

$$f_l=\sum_{k=1}^{n}\sum_{j=1}^{\tau_k}\lambda_k\alpha_{kj}\delta_{lj}^k,\qquad l=1,\dots,L$$

$$\text{where } \delta_{lj}^k=\begin{cases}1, & \text{if link } l \text{ is part of the } j^{th} \text{ tree of multicast session } k\\ 0, & \text{otherwise}\end{cases}\qquad (1).$$

We put (1) in matrix form with the aid of the following definitions:

Let

$$\begin{aligned} f&=[f_1,\dots,f_L]^T\\ \alpha^k&=[\alpha_{k1},..,\alpha_{k\tau_k}]^T,\ k=1,\dots,n\\ \alpha&=[\alpha_{11},..,\alpha_{1\tau_1},\alpha_{21},..,\alpha_{2\tau_2}\cdots\alpha_{n1},..,\alpha_{n\tau_n}]^T \end{aligned}.$$

For $k=1,\dots,n$ define the $L\times\tau_k$ matrix $B^k=(b_{lj}^k)\equiv(\lambda_k\delta_{lj}^k)$.

Let $B=[B^1\vdots\cdots\cdots\vdots B^n]$, an $L\times T$ matrix.

Then (1) can be rewritten as

$$f=\sum_{k=1}^{n}B^k\alpha^k=B\alpha \qquad (2).$$

We represent the set of constraints

$$\sum_{j=1}^{\tau_k}\alpha_{kj}=1,\quad k=1,\dots,n$$

in the matrix form

$$H\alpha=E \qquad (3),$$

where H is an $n\times T$ matrix and E is an n - vector of 1's.

3. Transmission Schedule

We let S denote the set of independent link-sets considered for use in the network, where the links of an independent link-set can all be active at the same time without causing mutual interference. Let $|S| = M$. The problem of generating all independent sets is itself NP-hard, and in the numerical examples of a 10-node network presented later, we used the set of *all maximal independent link-sets* as a 'reasonably complete' set.

Let ϕ_s be the fraction of a frame-duration for which independent link-set s is active, with $\sum_{s=1}^{M}\phi_s \le 1$ as a necessary condition of feasibility of the given loads in the given network under the given routing. With the frame-duration as the unit of time, the fraction of time a given link l is active is the sum of the active durations of each independent link-set to which the link belongs. Thus, if g_l denotes the maximum rate at which link l transmits whenever it is active, we have the constraint

$$f_l \le \sum_{s=1}^{M} y_{ls}\phi_s$$

$$\text{where } y_{ls} = \begin{cases} g_l, & \text{if } l \in \text{link-set } s \\ 0, & \text{otherwise} \end{cases}$$

or, in matrix-form,

$$f \le Y\phi \qquad (4).$$

A. Notion of Interference for Unicast and Multicast Flows

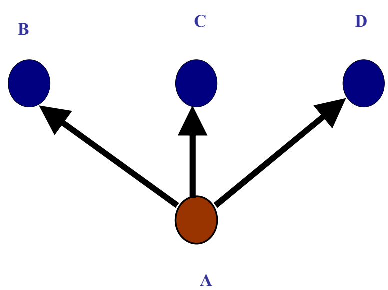


Figure 1. Interference for Unicast and Multicast Flows

We note that there is an important difference between a unicast flow and a multicast flow with respect to interference among link-transmissions. For example, in Figure 1, Node A has three incident links. If A is transmitting three separate *unicast* flows to B, C, and D respectively, these transmissions have to be scheduled for three different time-slots, since, in our model, a node can only handle a single flow in each time slot, and hence only ONE of the links (A→B), (A→C), and (A→D) can be active at any given time. On the other hand, the transmission of a *multicast* flow by a node on two or more of its incident links is part of the broadcast mechanism by which the multicast flow reaches its destinations, and does NOT constitute interference. Hence, such transmissions can be simultaneous, i.e., for a multicast transmission from node A to nodes B, C, and D, all three links (A→B), (A→C), and (A→D) can be scheduled for the same time-slot. Such a set of links with a common transmitter is defined to be a 'multicast link', and *can be treated as a single link for the purpose of a multicast*. Thus, interference for multicasts is a less restrictive condition than for unicasts.

4. CAPACITY MAXIMIZATION WITH GIVEN MULTICAST TREES

Given a set of multicast sources in the network with traffic rates $[\lambda_1,\dots,\lambda_n]$, we view the problem of capacity-maximization as the problem of supporting the traffic of the given sources while *minimizing* the active time of the network $\sum_{s=1}^{M}\phi_s = C^T\phi$, where the M - vector C is given by $C^T = [1,1,\dots,1]$. Thus, we may look upon $\left[1-\sum_{s=1}^{M}\phi_s\right]$, the proportion of time the links are inactive, as a measure of *spare* capacity in the network beyond that required to carry the given loads $[\lambda_1,\dots,\lambda_n]$, which is the operational definition of capacity that we adopt in this paper. Thus, for a *given* choice of multicast trees $[t_{11},\cdots,t_{1\tau_1},\dots t_{n1},\cdots,t_{n\tau_n}]$ the optimization of transmission schedule for capacity-maximization can be formulated as the LP:

Minimize $C^T\phi$, by choice of $\{\phi,\alpha\}$,
subject to the constraints

$$\begin{aligned} C^T\phi &\le 1 \\ Y\phi - B\alpha &\ge 0 \\ H\alpha &= E \\ \phi \ge 0,\quad & \alpha \ge 0 \end{aligned} \qquad (5).$$

5. JOINT OPTIMIZATION OF TRANSMISSION SCHEDULE AND MULTICAST TREES

When one has to determine the optimum choice of multicast trees for the traffic rather than just choose *from a given set of trees*, the problem becomes much harder and is now a mixed-integer program. Rather than taking a direct approach to this intractable problem, we seek to use the technique of 'column generation' in which, starting from a small initial set of 'columns' of the constraint matrix (in our case, the columns correspond to multicast trees), we use dual variables to guide a search for a column that would offer improvements to the objective function over the existing columns. The method also has the advantage of making it clear when no such improvement is possible by augmentation of the existing columns.

The dual of the LP (5) is constructed in terms of the following dual variables:

u	a scalar
$v = [v_1,\dots,v_L]^T$:	an L - vector ,
$w = [w_1,\dots,w_n]^T$:	an n - vector

and is given by

$$
\begin{aligned}
&\text{Maximize}[u + \sum_{j=1}^{n} w_j] \\
&\text{by choice of } \{u, v, w\}, \text{subject to} \\
&Cu + Y^T v \le C \\
&-B^T v + H^T w \le 0 \\
&u \le 0 \\
&v \ge 0 \\
&w \text{ unrestricted}
\end{aligned} \qquad (6).
$$

When the primal and dual LPs are both feasible, they have the same optimum objective value, and the solution of the LP also produces values for the dual variables $\{v, w\}$.

The next step is to use the dual variables to define a condition that has to be satisfied by any new column (multicast tree) that can offer an improvement in the objective of the primal LP.

6. Column Generation

Suppose that we solve the primal LP (5) and its associated dual LP (6). Let the optimum values of the variables be denoted with the superscript *. Then, the 'cost' of the optimum solution of (5) is given by

$$\text{Minimum cost} = \sum_{s=1}^{M} \phi_s^* \qquad (7).$$

Suppose that we add another multicast tree t for a specific multicast session k, $1 \le k \le n$. Of course, this would require also that the condition

$$\sum_{j=1}^{\tau_k} \alpha_{kj} = 1 \text{ be modified to}$$

$$\sum_{j=1}^{\tau_k} \alpha_{kj} + \alpha_{kt} = 1$$

Let the structure of this new multicast tree be denoted by the column-vector $b = [b_1, \cdots, b_L]^T$, where

$$b_i = \begin{cases} 1, & \text{if link } i \text{ belongs to the tree } t \\ 0, & \text{otherwise} \end{cases}$$

The new B-matrix corresponding to the addition of tree t is given by $\left[B^1 \vdots B^2 \vdots \cdots B^{k-1} \vdots B^k \vdots \lambda_k b \vdots \cdots B^n\right]$, and the corresponding new H matrix has a new column $h = [h_1, \cdots, h_n]^T$ in the position $\left[\sum_{j=1}^{k} \tau_j + 1\right]$, with

$$h_i = \begin{cases} 1, & \text{for } i = k \\ 0, & \text{for } i \ne k \end{cases}$$

From the interpretation of the dual variables [9] determined by the solution to (6), it follows that the addition of tree t can lead to a smaller cost than (7) only if

$$
\begin{aligned}
&\lambda_k \sum_{i=1}^{L} v_i^* b_i - w_k^* < 0 \\
&\text{i.e., } \sum_{i=1}^{L} v_i^* b_i < \frac{w_k^*}{\lambda_k},
\end{aligned} \qquad (8).
$$

Condition (8), in fact, implies two requirements:

1. $w_k^* > 0$
2. We are able to construct a tree for multicast session k with link weights v_i^* such that $\sum_{i=1}^{L} v_i^* b_i < \frac{w_k^*}{\lambda_k}$

In the absence of degeneracy [9], when these two requirements are met, the addition of multicast tree t for multicast session k offers an improvement over the solution (7). In particular, if, at some point, it is found that no tree can be constructed that meets the criterion (8), we know that an optimal solution has been found (however, the guarantee of optimality does not hold if inexact methods were used to decide that such a tree does not exist).

7. Algorithm for Choice of Multicast Trees

We now describe the overall procedure followed in applying column-generation in the example to be presented:

1. First, enumerate the set of independent link-sets to be used in the network. For the 10-node network shown below, we, in fact, generated all the *maximal* independent link-sets.

2. Determine an initial set of multicast trees for the given multicast demands, using unit 'cost' on all links in this initial step. In later iterations, the 'costs' assigned to links will be prescribed by the dual variables on the LP solution of the previous iteration. The 'cost' of a path is the sum of the 'costs' of the links on the path.

   Since finding a minimal Steiner tree is itself an NP-hard problem, we use the following simple heuristic algorithm. The same algorithm is used later when constructing trees as part of the column-generation step.

   Algorithm for Construction of Steiner Tree

   To find a low-total-weight tree connecting a single source to multiple destinations:
   While there is a destination not yet connected, find a minimum-cost path from the tree constructed so far to a destination not yet connected, and add this path to the tree.

3. Repeat the following steps (a), (b), (c), and (d) for a specified maximum number of iterations:

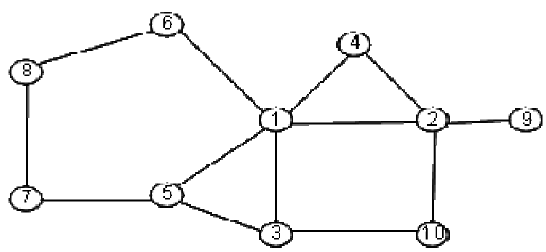
   a) Solve the LP (5) with the given independent link-sets and the current choice of multicast trees.

b) Extract the dual variables. Assign new link-costs and multicast tree cost-thresholds from the dual variables, in accordance with (8).

c) For each multicast demand, search for low-cost multicast trees (using the algorithm above), and add to the current set of multicast trees any new trees found that have total costs less than the cost-threshold in (8) for the corresponding multicast demand.

d) If no trees are found below the required cost-threshold (or if the number of iterations has reached the pre-set limit), exit the loop, and take the solution to be completed.

8. RESULTS



Link-bandwidth = 10 Mb/s for all links
Traffic Rate for Multicast Sessions = 2 Mb/s

Figure 2. 10-Node Network used in Example

The algorithm was applied to the 10-node network shown below in Figure 2, for three different cases of multicast traffic demands. The bandwidths of all the links were taken to be 10 Mb/sec and the traffic rate of each multicast session in the network was taken to be 2Mb/sec.

The multicast traffic sessions used in the three cases are as follows:

Case I
Multicast T1 = [Node 10 → Nodes (6, 9)]
Case II
Multicast T2 = [Node 7 → Nodes (2, 6)]
Case III
Multicast T3 = T1 + T2 = [Node 10 → Nodes (6, 9)] + [Node 7 → Nodes (2, 6)]

In each case, the process of column-generation was started by solving the LP (5) with an arbitrary tree for each multicast, and then repeating the LP after adding trees that were found admissible by the criterion (8), making use of the dual variables corresponding to the LP solution. The iterative procedure stops either when no new trees can be found that meet (8) or when the number of iterations has reached a pre-set limit (30, in our example).

As noted earlier, a measure of the spare capacity in the network is given by $\left[1-\sum_{s=1}^{M}\phi_s\right]$, which is the metric that we seek to maximize. The results obtained for the three cases are shown in Table 1. The effectiveness of the method is borne out by the difference between the initial spare capacity (corresponding to the initial choice of a multicast tree) and the final spare capacity that produced at the end of the column-generation iterations. Table 1 also shows, for each case, the number of iterations taken to reach the final solution, as well as the number of trees and the number of independent link-sets that are used in that solution.

TABLE 1. MULTICAST CAPACITY WITH COLUMN-GENERATION

Case	**Spare Capacity =** $\left[1-\sum_{s=1}^{M}\phi_s\right]$			**# Iterations**	**# Independent Link-Sets**	**# Trees**
	Initial	***Final***	***Increase***			
I	0.4	0.60	0.20 (50%)	18	4	2
II	0.2	0.63	0.43 (215%)	18	7	3
III	0.0	0.36	0.36 (∞)	30	13	10

9. SUMMARY

The need to schedule link transmissions in wireless networks to avoid interference among links leads to limitations on the capacity of the network that are quite separate from the inherent capacity of the links themselves. The choice of routing also determines how well the capacities of various paths are put to use in carrying traffic. Scheduling and routing interact in their effect on capacity – links are scheduled so as to avoid interference with other links, while the interference pattern that scheduling has to accommodate depends on traffic routing. As a result, for efficient use of the network, routing and scheduling should be considered together: considering one without the other is non-optimal. This was shown in [1] for the case of unicast flows.

In this paper, we have dealt with the case of multicast flows. Here, the joint optimization involved is that of scheduling and the determination of multicast routes (Steiner trees, with root at multicast source and leaf-nodes at destinations). A direct formulation of this problem leads to an intractable mixed-integer program. We have proposed an iterative method, based on LP duality theory and column-generation, in which, in each iteration, we derive conditions for higher capacity to be achievable by a new choice of tree. If such a tree cannot be found, the current solution is guaranteed to be optimal. This guarantee of optimality holds when exact methods are used to check for the existence of trees that meet the required conditions. If non-exact methods have to be used for tree-generation (since tree-generation is itself a hard problem), optimality at termination is not guaranteed. In practice, one also uses a pre-set bound on the number of iterations. In any case, at each step, we know whether the next step can yield a better solution or not. Our examples show the efficacy of the method and the significant capacity-gain that results from joint consideration of routing and scheduling.

ACKNOWLEDGEMENT

This research was carried out by collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

REFERENCES

[1] K.R. Krishnan, D. Shallcross, L. Kant, "Joint optimization of transmission schedule and routing for network capacity", MILCOM 2008, San Diego, November 2008.

[2] S. Shakkottai, X. Liu, and R. Srikant, "The Multicast Capacity of Large Multihop Wireless Networks", MobiHoc '07, Montreal, Canada, September 2007.

[3] X-Y Li, S-J Tang, and O. Frieder, "Multicast Capacity for Large Scale Wireless Ad Hoc Networks", MbiCom'07, Montreal, Canada, September 2007.

[4] S. Sarkar and L. Tassiulas, "A Framework for Routing and Congestion Control for Multicast Information Flows", IEEE Transactions on Information Theory, Vol. 48, No, 10, October 2002.

[5] S. Toumpis and A. J. Goldsmith, "Performance Bounds for Large Wireless Networks with Mobile Nodes and Multicast Traffic", International Workshop on Wireless Ad-Hoc Networks, May-June 2004.

[6] O. Chaker and J. Conan, "Multicast Capacity of Wireless Ad Hoc Networks with Hierarchical Routing", IEEE Vehicular Technology Conference, VTC2007, Dublin, Ireland, April 2007.

[7] P. Chaporkar and S. Sarkar, "Wireless Multicast: Theory and Approaches", IEEE Trans. Information Theory, Vol. 51, No. 6, June 2005.

[8] P. Gupta and P.R. Kumar, "The capacity of wireless networks", IEEE Transactions on Information Theory, March 2000, 46(2), pp. 388-404.

[9] M. S. Bazaraa and J. J. Jarvis, Linear Programming and Network Flows (Chapter 7), John Wiley & Sons, 1977.

[10] Penttinen, "Efficient multicast tree algorithm for ad hoc networks", Proc. 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems, pp.519-521, Fort Lauderdale, Florida, October 2004.

[11] Penttinen, "Minimum cost multicast trees in ad hoc networks", Proceedings of the 2006 IEEE International Conference on Communications (ICC 2006), Istanbul, Turkey, June 2006.