International Journal of Management Science
Vol 12, No 2, November 2006

A New Mathematical Formulation for Generating a Multicast Routing Tree

Jangha Kang
SCM Group, Entrue Consulting Partners, 25F,
Seoul Finance Center, 84, Taepyungro 1-ga, Chung-gu, Seoul, 100-768, Korea

Donghan Kang*
Samsung Electro-Mechanics, 314 Metan3-dong,
Yeongtong-gu, Suwon, Gyeonggi-do 443-743, Korea

Sungsoo Park
Dept. of Industrial Engineering, Korea Advanced Institute of Science and Technology,
373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, Korea

(Received Jun. 2006; Revised Sep. 2006; Accepted Sep. 2006)

ABSTRACT

To generate a multicast routing tree guaranteeing the quality of service (QoS), we consider the hop constrained Steiner tree problem and propose a new mathematical formulation for it, which contains fewer constraints than a known formulation. An efficient procedure is also proposed to solve the problem. Preliminary tests show that the procedure reduces the computing time significantly.

Keywords: Multicasting, Steiner Tree, Column Generation

1. INTRODUCTION

Multicast technology can be used to enable the use of multimedia applications such as voice and video transmission. It allows the transmission of data from one source node to a selected group of destination nodes. Multicast routes typically use trees, called multicast routing trees, to minimize resource usage such as cost and bandwidth by sharing links. Moreover, the quality of service (QoS) is satisfied by distributing data along a path having no more than a given number of arcs

* Corresponding author, Email: donghan.kang@samsung.com

(hop) between the root node and a terminal node in the routing tree. Thus, a multicast routing tree can be represented as a hop constrained Steiner tree. The problem of getting the tree with minimum arc cost is referred to as the hop constrained Steiner tree problem (HCSTP). The problem can be formally described as follows: Consider a directed connected graph $G=(V,A)$ with a node set $V=\{0,1,\ldots,n\}$ and an arc set A, a nonnegative cost c_a for each arc $a \in A$, a root node 0, a set $T \subseteq V \setminus \{0\}$ of terminal nodes, and a natural number H. We want to find the minimum cost rooted tree spanning 0 and the terminal nodes in T with additional constraints that each path from 0 to any node in T has no more than H arcs. HCSTP is NP-hard because its unconstrained version, the Steiner tree problem, is also known to be NP-hard [1].

Gouveia[1] proposed several formulations including the sub-tour elimination constraints and applied a Lagrangian relaxation approach combined with subgradient optimization to get lower bounds. In 1996, he modeled the problem as a directed multi-commodity flow problem and gave an improved formulation [2]. In 1998, he reformulated the problem with flow variables [3] and provided an empirical evidence that the linear programming relaxation of the formulation, referred to as HDMCF, gave a sharp lower bound on the optimal value. However, it has numerous constraints and variables. The numbers of constraints and variables are about $|V|\times|T|\times H+|A|\times|T|$ and $|A|\times|T|\times H$, respectively, where $|S|$ denotes the cardinality of a set S. As H increases, the size of the formulation and solving time increased considerably. To tackle this problem, Gouveia [3, 4] also derived several Lagrangian relaxation-based procedures for getting lower bounds. However, the lower bounds were dominated by the bounds resulting from the linear programming relaxation of HDMCF.

We present the readers the integer programming model of HDMCF as a reference. For details of the formulation, please refer to [3, 4]. We first introduce the decision variables and notation used in the model.

X_{ij} : a binary variable which indicates whether arc $(i,j) \in A$ is in the minimal spanning/Steiner tree.

Z_{ijqt} : a binary variable which indicates whether arc $(i,j) \in A$ is exactly in position q in the unique path from the root node to node $t \in T$, where $q=1,\ldots,H$.

$I(j)=\{i : i=0,\ldots,n \text{ and } (i,j) \in A\}$.

$O(j)=\{i : i=1,\ldots,n \text{ and } (j,i) \in A\}$.

(HDMCF)

minimize $\sum_{(i,j)\in A} c_{ij}X_{ij}$

subject to

$$\sum_{i\in I(j)} X_{ij} = 1, \qquad j \in T$$

$$\sum_{i\in I(j)} X_{ij} \le 1, \qquad j \in \{1,\dots,n\} \setminus T$$

$$\sum_{j\in I(t)\cup\{t\}} Z_{jkHt} = 1, \qquad t \in T$$

$$\sum_{j\in I(i)} Z_{jiqt} - \sum_{j\in O(i)} Z_{ij,q+1,t} = 0, \qquad i = 1,\dots,n;\ t \in T; q = 2,\dots,H-1$$

$$Z_{0i1t} - \sum_{j\in O(i)} Z_{ij2t} = 0, \qquad (0,i) \in A,\ t \in T$$

$$\sum_{q=1}^{H} Z_{ijqt} \le X_{ij}, \qquad (i,j) \in A,\ t \in T$$

$$Z_{0j1t} \in \{0,1\}, \qquad (0,j) \in A,\ t \in T$$

$$Z_{ijqt} \in \{0,1\}, \qquad (i,j) \in A,\ i \ne 0;\ t \in T,\ q = 2,\dots,H$$

$$\text{or } i = j = t;\ t \in T,\ q = 2,\dots,H$$

$$X_{ij} \in \{0,1\}, \qquad (i,j) \in A$$

In this paper, we propose a new more compact mathematical formulation of the problem and suggest an efficient and fast procedure to solve it.

2. A NEW MATHEMATICAL FORMULATION

We formulate the problem as a directed multi-commodity flow model with path variables. Before introducing the formulation, we first define a *feasible path* r as a directed path from the root node to a terminal node using at most H arcs. Let $R(t)$ be the set of feasible paths to a terminal node t, and $R(t;a) \subseteq R(t)$ be the set of paths in $R(t)$ containing $a \in A$. To formulate the problem, we introduce two types of variables: x_a and y_{rt}. The binary variables x_a, $a \in A$ indicate whether arc a is contained in the Steiner tree and the binary variables y_{rt}, $r \in R(t)$, $t \in T$ indicate whether path r from 0 to t is realized in the Steiner tree.

The mathematical formulation of HCSTP can be stated as follows:

(MP)

minimize $\sum_{a\in A} c_a x_a$

subject to

$$x_a - \sum_{r\in R(t;a)} y_{rt} \geq 0, \qquad a\in A, t\in T, \tag{1}$$

$$\sum_{r\in R(t)} y_{rt} = 1, \qquad t\in T, \tag{2}$$

$$x_a \in \{0,1\}, \qquad a\in A, \tag{3}$$

$$y_{rt} \in \{0,1\}, \qquad t\in R(t), \ t\in T. \tag{4}$$

Constraints (1) ensure that if there are realized paths containing arc a, the arc is contained in the Steiner tree. Constraints (2) ensure that one feasible path to $t\in T$ is realized in the Steiner tree.

A linear programming (LP) relaxation of the formulation can be obtained by relaxing integrality restrictions in (3) and (4). Let HDMCFL be the LP relaxation of HDMCF and let MPL be that of MP. MPL gives the same lower bound on the optimal value as that of HDMCFL. This can be easily proved by showing that for any given feasible solution to MPL, we can construct a feasible solution to HDMCFL with the same objective value, and vice versa. We leave details of the proof to the readers.

MP has only $(|A|+1)\times|T|$ constraints. However, it has exponentially many variables related to the feasible paths. Nevertheless, we can solve the LP relaxation efficiently using the column generation technique [5].

3. SOLVING MPL

We propose an efficient procedure for solving MPL based on the column generation technique. Suppose that a subset $\overline{R}(t)\subseteq R(t)$ is given for each $t\in T$. Then, replacing $R(t)$ by $\overline{R}(t)$ in MPL yields a restricted formulation, referred to as MPL', whose solution is usually suboptimal to MPL. Let α_{ta}, $\forall t\in T$, $a\in A$, and β_t, $\forall t\in T$ be the dual variables associated with the constraints (1) and (2), respectively. Then, the constraints of the dual problem of MPL' are

$$\sum_{t\in T} \alpha_{ta} \leq c_a, \qquad a\in A, \tag{5}$$

$$\beta_t \leq \sum_{a\in L(r)} \alpha_{ta}, \qquad t\in \overline{R}(t), \ t\in T,$$

$$\alpha_{ta} \geq 0, \ t\in T, \qquad a\in A,$$

where $L(r)$ is the set of arcs constituting path r. Interested readers can refer to Chvatal [5] for details of the dual problem of an LP problem. Let $(\bar{\alpha}, \bar{\beta})$ be an optimal solution to the dual problem of MPL'. Then, it is also optimal to the dual problem of MPL if $\bar{\beta}_t \le \sum_{a \in L(r)} \bar{\alpha}_{ta}$, for all $r \in R(t)$ and $t \in T$ since $(\bar{\alpha}, \bar{\beta})$ always satisfies the constraints (5). So, we can write the optimality condition for MPL as follows:

$$\min\left\{\sum_{a \in L(r)} \bar{\alpha}_{ta} : r \in R(t)\right\} \ge \bar{\beta}_t, \quad \forall \quad t \in T.$$

Hence, the procedure to solve MPL using the column generation method can be outlined as follows: We solve MPL' using the simplex method [5] which yields an optimal solution to MPL' together with an optimal dual solution $(\bar{\alpha}, \bar{\beta})$. Now, using $(\bar{\alpha}, \bar{\beta})$, we search for a feasible path r for each $t \in T$, the addition of which to MPL' may result in a decrease in the optimal objective value of MPL'. If no such feasible paths exist, then the solution at hand is an optimal solution to MPL. Otherwise, we add the generated paths to MPL', and then repeat the above process.

To check whether there are feasible paths to be added to MPL', we should solve the following problem for each $t \in T$: minimize $\sum_{a \in L(r)} \bar{\alpha}_{ta}$ subject to $r \in R(t)$. This problem is referred to as SP(t). Let $r(t)$ be an optimal solution to SP(t). If $\sum_{a \in L(r(t))} \bar{\alpha}_{ta}$ is less than $\bar{\beta}_t$, then $r(t)$ violates the optimality condition for MPL. Therefore, $r(t)$ should be added to MPL'. We can see that SP(t) is the hop constrained shortest path problem, which can be solved by a modified Bellman-Ford algorithm in $O(|A| \times H)$ time; Interested readers can refer to Gouveia [3] for details of the algorithm.

Although MPL has relatively small number of constraints, the number of constraints needed in solving MPL can be reduced further by applying row generation method. The procedure of row generation is similar to that of column generation. Let us consider a subset $A(t) \subseteq A$ for each $t \in T$. Suppose that MPL' contains all constraints (2) and some constraints (1) corresponding to $a \in A(t)$, $t \in T$. Then, the row generation procedure can be outlined as follows: After new columns are added to MPL', we solve MPL' with the simplex method. Then, we get an optimal solution $(\bar{x}, \bar{y})$ to MPL'. Since, not all of constraints (1) are included in MPL', there may exist some constraints that are violated by $(\bar{x}, \bar{y})$. If no such constraints exist, we search for new columns to be added to MPL'. Otherwise, we add the violated constraints to MPL', and then repeat the above process. If no columns and hence no constraints are generated, the optimal solution to MPL' is also optimal to MPL.

After solving MPL, if the resulting optimal solution violates the integrality constraints, we should make more efforts to find an optimal integer solution to MP. We can find an optimal integer solution using the branch-and-bound algorithm incorporated with the column generation and row generation methods. Interested readers can refer to Park *et al.* [6] for details of the algorithm.

4. COMPUTATIONAL EXPERIMENTS

We generate a network, which consists of 81 nodes, 480 arcs, and 10 terminal nodes by the procedure proposed by Gouveia [3]. We performed tests for MP and HDMCF varying the hop limit H from 3 to 30. We used CPLEX 7.0 callable library routines for the dual simplex algorithm as an LP solver and all tests were run on a Pentium 500MHz PC with 256Mbytes main memory. Computational results are illustrated in Figure 1. It shows that the computing time of our algorithm remains in low levels, but that of HDMCF increases rapidly as H increases. Moreover, we found that the final MPL' has 978 variables and 2933 constraints but HDMCF has 115888 variables and 28010 constraints when H is 30. This shows that MP has a small number of constraints and that the column and row generation method is really efficient in reducing the size of LP.

In each test, we got an optimal integer solution without performing any branch-and-bound procedure for both formulations: MP, HDMCF, which indicates that they give tight lower bounds on the optimal value.

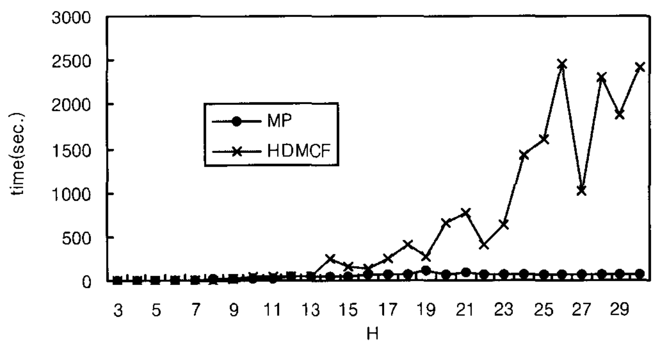

Figure 1. Computational Results

5. CONCLUSION

We have presented a new mathematical formulation of the hop constrained Steiner tree problem, together with an efficient method to solve it. Although our formulation and the Gouveia's formulation (HDMCF) yield the same LP bounds, preliminary tests show that our algorithm results in much smaller computing time than that of HDMCF when H is large. This validates the practical use of our algorithm for generating a multicast routing tree in real networks.

REFERENCES

[1] Gouveia, L., "Using the Miller-Tucker-Zemlin constraints to formulate a minimal spanning tree problem with hop constraints," *Computers and Operations Research* 22, 9 (1995), 959-970.

[2] Gouveia, L., "Multicommodity flow models for spanning trees with hop constraints," *European Journal of Operational Research* 95 (1996), 178-190.

[3] Gouveia, L., "Using variable redefinition for computing lower bounds for minimum spanning and Steiner trees with hop constraints," *INFORMS Journal on Computing* 10, 2 (1998), 180-188.

[4] Gouveia, L. and C. Requejo, "A new Lagrangean relaxation approach for the hop-constrained minimum spanning tree problem," *European Journal of Operational Research* 132 (2001), 539-552.

[5] Chvatal, V., *Linear programming,* Freeman, New York, 1983.

[6] Park, K., S. Kang, and S. Park, "An integer programming approach to the bandwidth packing problem," *Management Science* 42, 9 (1996), 1277-1291.