Cross Decomposition of the Degree-Constrained Minimum Spanning Tree problem

Han-Suk Sohn
Department of Industrial Engineering, New Mexico State University
Las Cruces, NM 88003, USA

and

Dennis Bricker
Department of Mechanical and Industrial Engineering, University of Iowa
Iowa City, IA 52242, USA

ABSTRACT

As computer communication networks become a prevalent part in our daily life, the importance of efficient design of those networks becomes more evident. One of the critical issues in the network design process is the topological design problem involved in establishing a centralized data communication network with best performance and low costs. It can be recognized as a degree-constrained minimum spanning tree and it has been shown to be NP-hard. The degree-constrained minimum spanning tree problem commonly appears as a subproblem in the design of centralized data communication networks, and so the development of effective algorithms has received much attention in the research literature. To achieve effectiveness in solving degree-constrained minimum spanning tree, a solution algorithm based on cross-decomposition is proposed in this paper. The computational results are analyzed to demonstrate the effectiveness of the proposed algorithm. It shows a great promise in the design of centralized data communication networks.

Keywords: Cross-decomposition, minimum spanning tree, degree-restriction, mixed integer linear programming, network design.

1. INTRODUCTION

The minimum spanning tree problem is a fundamental problem in the design of centralized data communication networks. Many variations of the minimum spanning tree problem occur in the field of communication networks and computer networks. One of them is the degree-constrained minimum spanning tree (DCMST) problem which is concerned with determining a minimum total edge weight spanning tree whose vertices satisfy specified degree restrictions in a given edge-weighted graph *G*. The DCMST problem is of high practical importance. This problem may arise for instance when designing an electrical circuit (Narula and Ho [4]): connect *n* terminals with the minimum amount of wire, where the number of wires incident to terminal *i* can be at most a given number of wires (the degree constraint). The problem of finding a DCMST also arises in many other areas such as transportation, communication, plumbing, sewage, etc (Gavish [2], Narula and Ho [4], Savelsbergh and Volgenant [5], and Kawatra [3]).

2. PROBLEM FORMULATION

Consider the following single-commodity flow formulation of the degree-constrained minimum spanning tree (DCMST) problem (Gavish [2]):

(Problem 1)

$$\text{Minimize} \quad \sum_{i=2}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} c_{ij} X_{ij} \tag{1}$$

$$\text{Subject to} \quad \sum_{\substack{j=1 \\ j \neq i}}^{n} X_{ij} = 1 \qquad i = 2, \cdots, n \tag{2}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} y_{ij} - \sum_{\substack{j=2 \\ j \neq i}}^{n} y_{ji} = 1 \qquad i = 2, \cdots, n \tag{3}$$

$$y_{ij} \leq (n-1) X_{ij} \qquad i = 2, \cdots, n \tag{4}$$

$$\sum_{\substack{i=2 \\ i \neq k}}^{n} X_{ik} + \sum_{\substack{i=1 \\ i \neq k}}^{n} X_{ki} \leq r_k \qquad k = 2, \cdots, n \tag{5}$$

$$y_{ij} \geq 0 \text{ and } X_{ij} = 0 \text{ or } 1 \quad \forall\, i, j \tag{6}$$

where $X_{ij} = 1$ if arc (i, j) is included in the minimum spanning tree and 0 otherwise,
y_{ij} is the flow on the arc (i, j), and given a subset of nodes *S*, for every node *k*, $k \in S$, an upper bound, and
r_k is an upper bound (≥1) imposed on the number of arcs that can be incident to node *k*, where S is a subset of the nodes of the network.
Notationally, if (●) is an optimization problem, we let v(●) be its optimal solution value, $\overline{v}$ (●) its incumbent objective value and F(●) its feasible region.

3. CROSS DECOMPOSITION

Dual Decomposition (Lagrangian Relaxation)

Dual decomposition of (Problem 1) is outlined below. The formal Lagrangian dual of (Problem 1) relative to $y_{ij} \le (n\text{-}1)X_{ij}$ is given by (D)

$$
\text{(D)}\quad \underset{u \ge 0}{\text{Maximize}} \left|
\begin{aligned}
&\underset{\substack{x\,\varepsilon\, Z\\ y\, \ge\, 0}}{\text{Minimize}} \sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} c_{ij}X_{ij} \\
&\qquad +\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} u_{ij}\left\{y_{ij}-(n-1)X_{ij}\right\} \\
&\text{Subject to } \sum_{\substack{j=1\\ j\ne i}}^{n} X_{ij}=1 \qquad , i=2,\cdots,n \\
&\qquad \sum_{\substack{j=1\\ j\ne i}}^{n} y_{ij}-\sum_{\substack{j=2\\ j\ne i}}^{n} y_{ji}=1 \\
&\qquad\qquad\qquad , i=2,\cdots,n \\
&\qquad \sum_{\substack{i=2\\ i\ne k}}^{n} X_{ik}+\sum_{\substack{i=1\\ i\ne k}}^{n} X_{ki}\le r_k \\
&\qquad\qquad\qquad , k=2,\cdots,n
\end{aligned}
\right|
$$

$$= \underset{u \ge 0}{\text{Maximize}}\ v\left(\text{DS}(u)\right)$$

i.e., the inner minimization problem (D) is defined as the dual subproblem and the dual master problem can be written as:

$$
(\text{MA}_\text{D})\quad \left|
\begin{aligned}
&\underset{u\ge 0\ ,\ u_0\in R}{\text{Maximize}}\ u_0 \\
&\text{Subject to } u_0 \le \left\{
\begin{aligned}
&\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} c_{ij}X_{ij}^{d} \\
&+\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} u_{ij}\left(y_{ij}-(n-1)X_{ij}^{d}\right)
\end{aligned}
\right\} \\
&\qquad\qquad , \quad d\in \text{D}_{\text{DA}}
\end{aligned}
\right|
$$

$\{x^d, d \in \text{D}_{\text{DA}}\}$ is the set of extreme points of $\text{F}(\text{DS}(u))$.

Primal (Benders') Decomposition

Primal decomposition of (Problem 1) is implemented as follows:

$$
\text{(P)}\ \underset{x\,\varepsilon\, Z}{\text{Minimize}} \left|
\begin{aligned}
&\underset{y\,\ge\, 0}{\text{Minimize}} \sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} 0\, y_{ij} \\
&\qquad +\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} c_{ij}X_{ij} \\
&\text{Subject to } \sum_{\substack{j=1\\ j\ne i}}^{n} y_{ij}-\sum_{\substack{j=2\\ j\ne i}}^{n} y_{ji}=1 \\
&\qquad\qquad , i=2,\cdots,N \\
&\qquad y_{ij}\le (n-1)X_{ij} \\
&\qquad\qquad , i=2,\cdots,N
\end{aligned}
\right|
$$

$$= \underset{x\in Z}{\text{Minimize}}\ v\left(\text{PS}(x)\right).$$

For any fixed value of x, the inner minimization problem (P) is a linear program which is called the primal or Benders' subproblem. By dualizing this linear program (PS(x)) we may rewrite (P) as:

$$
\underset{x\,\in\, Z}{\text{Minimize}} \left|
\begin{aligned}
&\underset{\substack{v\,\ge\, 0\\ u\,\ge\, 0}}{\text{Maximize}} \sum_{i=2}^{n} v_i-\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n}(n-1)X_{ij}u_{ij} \\
&\qquad +\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} c_{ij}X_{ij} \\
&\text{Subject to } v_i-v_j-u_{ij}\le 0 \\
&\qquad , i=2,\cdots,n\ \&\ j=2,\cdots,n \\
&\qquad v_i-u_{ij}\le 0 \\
&\qquad , i=2,\cdots,n
\end{aligned}
\right|
$$

$$= \underset{x\in Z}{\text{Minimize}}\ v\left(\text{PS}_\text{D}(x)\right)$$

Primal mater problem can be written as:

$$
(\text{MA}_\text{P})\quad \left|
\begin{aligned}
&\underset{x\in Z\ ,\ x_0\in R}{\text{Minimize}}\ x_0 \\
&\text{Subject to } x_0 \ge \left\{
\begin{aligned}
&\sum_{i=2}^{n} v_i^{d}-\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n}(n-1)X_{ij}u_{ij}^{d} \\
&+\sum_{i=2}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} c_{ij}X_{ij}
\end{aligned}
\right\} \\
&\qquad\qquad , d\in \text{D}_{\text{PA}}
\end{aligned}
\right|
$$

$$\sum_{\substack{j=1 \\ j\neq i}}^{n} X_{ij} = 1 \qquad , i = 2,\cdots,n$$

$$\sum_{\substack{i=2 \\ i\neq k}}^{n} X_{ik} + \sum_{\substack{i=1 \\ i\neq k}}^{n} X_{ki} \le r_k \qquad , k = 2,\cdots,n$$

In the above formulation, $\{v^d$ & u^d, $d \in D_{PA}\}$ is the set of extreme points of $F(PS_D(x))$. The constraints of the primal master problem are called the Benders' cuts or primal cuts and are generated by the dual solutions of the primal subproblem.

Note that if $F(PS_D(x))$ is unbounded, in other words, if u goes to infinity with *(n-1)Xu* finite (i.e., PS((x)) is infeasible), then add the regularity constraint (i.e., $\sum v_i \le M$, where M is very large number) to the problem and solve ($PS_D(x)$) again.

Cross Decomposition

Cross decomposition is a hybrid of Benders' decomposition and Lagrangian relaxation, in which the subproblem of each algorithm serves the purpose of the master problem of the other. That is, Benders' subproblem receives x^k from the Dual subproblem rather than from the Benders' master problem. Likewise, the Dual subproblem receives the necessary Lagrangian multipliers u^k from the Benders' subproblem, rather than from the Dual master problem. The complete details of the cross decomposition algorithm are shown as below.

Step 1: Initialize

Initialize the iteration counter k to zero. Set ξ to 1 where ξ is a counter that is used in testing for "stalling", i.e., a failure to converge. Set $\overline{v}_P = \overline{u}_0 = (+\infty)$, where $\overline{u}_0$ is the incumbent upper bound on the objective in (D) as given by the dual master problem and $\overline{v}_P$ is the incumbent (primal) objective value of (P). Set $\overline{v}_D = \overline{x}_0 = (-\infty)$, where $\overline{x}_0$ is the incumbent lower bound on the primal objectives as yielded by the primal master problem and $\overline{v}_D$ is the incumbent (dual) objective value of (D). Select initial values for the Lagrangian multipliers u^0 ($\ge$0).

Step 2: Dual Subproblem

(a) Increment the iteration counter k by 1. Solve ($DS(u^k)$). Let x^k be an optimal solution to the Lagrangian relaxation of (P) corresponding to u^k.

(b) Let $\overline{v}_D$ denote the incumbent lower bound on the primal objective function as yielded by Lagrangian relaxation. If the current relaxation yields a lower bound that is higher than the incumbent, i.e., if $\overline{v}_D < v(DS(u^k))$ then update $\overline{v}_D = v(DS(u^k))$ and set ξ to 1. Otherwise, increment ξ by 1. Check for optimality: If $\overline{v}_D \ge \overline{v}_P$, stop; $\overline{x}$ is an optimal solution of (P).

(c) Convergence Test CT_P. If $\xi = 4$, the algorithm has stalled; go to step (4b). If $\xi < 4$, set $x^{k+1} = x^k$ and go on to step 3.

Comment: When $\xi = 4$, one wants to avoid using the values for the complicating variables that have been yielded by the current relaxation. Instead, the primal master problem is solved.

Step 3: Primal Subproblem

Increment the iteration counter: $k = k + 1$. Solve the primal subproblem ($PS(x^k)$) with x^k as the values for the complicating variables. Let x^k and u^k be the optimal primal and dual solutions, respectively. The solution of ($PS(x^k)$) yields a solution that is feasible in (P). If the upper bound yielded by the current restriction of (P) is lower than $\overline{v}_P$, i.e., if $\overline{v}_P > v(PS(x^k))$ then update $\overline{v}_P = (PS(x^k))$ and the incumbent primal solution $\overline{x} = x^k$ and set ξ to 1. Otherwise, increment ξ by 1. Check for termination: Stop if $\overline{v}_P \le \overline{v}_D$ or $\overline{v}_P \le \overline{x}_0$ with x^k then being an optimal solution for (P). i.e., $\overline{x}$ is an optimal solution of (P).

Convergence Test CT_D: If $\xi = 4$, go to step (4a) in order to get new Lagrangian multipliers. If ξ <4, then obtain the multipliers from the optimal dual solution, i.e., set $u^{k+1} = u^k$ and go to step 2.

Step 4: Master System

(a) Solve the dual master problem (MA_D) to get multipliers u^{k+1} and an upper bound on (D), i.e., $\overline{u}_0$. Let ($\overline{u}_0$, u^{k+1}) be an optimal solution. Set $\xi = 0$. If $\overline{u}_0 > \overline{v}_D$, go to step 2. Otherwise set ξ to 1.

(b) Solve the primal master problem (MA_D) to obtain values for the complicating variables x^{k+1} and a lower bound on the objective in (P), i.e., $\overline{x}_0$. Let ($\overline{x}_0$, x^{k+1}) be an optimal solution. Set ξ to 0. Check for termination: Stop if $\overline{x}_0 > \overline{v}_P$ with x^k, stop $\overline{x}_0$ is an optimal solution of (P). Otherwise go to step 3.

The significance of the test as to whether ξ equals 4 or not derives from a result demonstrated by Van Roy [1983], namely, there cannot be a replication within a series of 4 iterations executed solely between steps 2 and 3 of the algorithm. In other words, it is sufficient (for a convergent cross decomposition algorithm) that, within a series of four consecutive subproblems, there is just one improvement of the incumbent primal or dual solution. If a convergence test fails we can solve either master problem, (DM) or (PM), in order to restore the convergence.

4. NUMERICAL RESULTS

The purpose of this research is to develop a useful algorithm that can solve the addressed capacitated minimum spanning tree problem. The proposed algorithm in the preceding chapter was programmed in Mosel language for Xpress-MP and run on an IBM Pentium III CPU 1.66GHz supported by Computer Systems Support (CSS) at the University of Iowa. In order to compare the performance of the implemented algorithm with the performance of another algorithm reported in the literature (i.e., the Lagrangian dual algorithm of Volgenant [7]), 'CRD40#' data sets which are a class of 40-node (excluding central node) symmetric instances from OR-Library (Beasley [1]) have been used. Note that, in the Lagrangian relaxation (LR) algorithm, a subgradient method was used to search for the Lagrangian multiplier. Based upon prior computational experience and common usage, a scalar for the step size factor in subgradient optimization for Lagrangian dual algorithm is determined by starting with an initial value of 2 and reducing it by a factor of 1/1.1 whenever the dual subproblem solution has failed to increase within 10 iterations. Four different values for maximum degree (2, 3, 5, and 10) were used. For each parameter value, 10 test problems were solved for CRD40# instance (CRD4001 through CRD40010) and 5 test problems were solved for CRD80# instance (CRD801 through CRD805).

Problem ID	LR		
	BUB	BLB	Gap (%)
crd4001	589	479	18.76
crd4002	589	486	17.53
crd4003	601	491	18.24
crd4004	552	452	18.09
crd4005	575	480	16.53
crd4006	617	470	23.82
crd4007	595	473	20.50
crd4008	615	465	24.39
crd4009	581	462	20.48
crd40010	613	472	23.00

Table 1. 40-node problem (r = 2)

Problem ID	XD		
	BUB	BLB	Gap (%)
crd4001	539	498	7.61
crd4002	570	496	12.98
crd4003	559	516	7.69
crd4004	523	512	2.10
crd4005	543	504	7.18
crd4006	617	498	19.29
crd4007	576	497	13.72
crd4008	597	492	17.59
crd4009	568	499	12.15
crd40010	597	498	16.58

Table 2. 40-node problem (r = 2)

Problem ID	% BUB-BLB	
	LR	XD
crd801	31.95	16.25
crd802	32.76	20.98
crd803	29.76	20.72
crd804	35.01	22.23
crd805	29.08	22.70

Table 3. 80-node problem (r = 2)

Problem ID	% BUB-BLB	
	LR	XD
crd801	0	0
crd802	0	0
crd803	0	0
crd804	22.97	10.12
crd805	0	0

Table 4. 80-node problem (r = 3)

According to the result, we would say that it is not necessary to consider a degree constraint of value greater than r = 2 for the CRD40# cases (See Tables 1 and 2). That is, for the CRD40# instances, the unconstrained minimum spanning tree is feasible with respect to degree constraints if we consider $r > 2$.

It is also interesting that, for the case r = 3, the unconstrained minimum spanning tree is optimal for all CRD80# instances except for CRD804 (See Table 4).

For the case of maximum degree 2, the computational results from both approaches are not quite satisfactory (See Table 3). However, we note that performances of both approaches are significantly improved as the maximum degree increases. Also, it is clear that, for both CRD40# and CRD80# problems, the cross decomposition algorithm performs exceedingly well in terms of the ratio of the best upper bound and the best lower bound. The cross decomposition algorithm compares very favorably with the performance of the Lagrangian dual algorithm.

5. CONCLUSION

An implementation of cross decomposition algorithm for DCMST problem has been described and the corresponding computational results have been analyzed. In order to compare the performance of the implemented algorithm with the performance of another algorithm reported in the literature (namely, the Lagrangian dual algorithms of Volgenant [7]), we tested our algorithm on the benchmark problems downloaded from OR-Library (Beasley [1]). The excessive computational times are disappointing but the use of algorithms which take better advantage of the problem structure, especially that of the dual subproblem, provides a large potential for improvement. The proposed algorithm was seemed to converge more rapidly than the Lagrangian dual algorithm for the DCMST problem for the case of maximum degree 2.

6. REFERENCES

[1] J. E. Beasley, "OR-Library: Distributing Test Problems by Electronic Mail", **Journal of the Operational Research Society,** Vol. 41, No. 11, 1990, pp. 1069-1072 , http://mscmga.ms.ic.ac.uk/jeb/orlib/capmstinfo.html.

[2] B. Gavish, "Topological design of centralized computer networks: Formulation and algorithms", **Networks,** Vol. 12, 1982, pp. 355-377.

[3] R. Kawatra, "A multiperiod degree constrained minimal spanning tree problem", **European Journal of Operational Research,** Vol. 143, 2002, pp. 53-63.

[4] S.C. Narula and C.A. Ho, "Degree-constrained minimum spanning tree", **Computors & Operational Research,** Vol. 7, 1980, pp. 239-249.

[5] M. Savelsbergh and T. Volgenant, "Edge exchanges in the degree-constrained minimum spanning tree problem", **Computors & Operational Research,** Vol. 12, 1985, pp. 341–348.

[6] T. J. Van Roy, "Cross decomposition for mixed integer programming", **Mathematical Programming,** Vol. 25, 1983, pp. 46-63.

[7] A. Volgenant, "A Lagrangean approach to the degree constrained minimum spanning tree problem", **European Journal of Operational Research,** Vol. 39, 1989, pp. 325-331.