CARPATHIAN J. MATH.
22 (2006), No. 1 - 2, 129 - 135

At least version of the generalized minimum spanning tree problem

PETRICĂ C. POP, ANDREI HORVAT MARC and CORINA POP SITAR

ABSTRACT. We consider the at least version of the Generalized Minimum Spanning Tree Problem, denoted by L-GMST, which consists in finding a minimum cost tree spanning at least one node from each node set of a complete graph with the nodes partitioned into a given number of node sets. We describe new integer programming formulations of the L-GMST problem and establish relationships between the polytopes corresponding to their linear relaxations.

1. INTRODUCTION

The minimum spanning tree (MST) problem can be generalized in a natural way by considering instead of nodes, node sets (clusters) and asking for a minimum cost tree spanning *exactly* one node from each cluster. This problem is called the generalized minimum spanning tree problem (GMSTP) and it was introduced by Myung *et al.* [4].

Meanwhile, the GMSTP have been studied by several authors w.r.t. heuristics and metaheuristics, LP-relaxations, polyhedral aspects and approximability, cf., e.g. Feremans, Labbe, and Laporte [3], Feremans [2], Pop, Kern and Still [8, 9] and Pop [5, 6].

Two variants of the generalized minimum spanning tree problem were considered in the literature: one in which in addition to the cost attached to the edges, we have costs attached also to the nodes, called the *prize collecting generalized minimum spanning tree problem* see [7] and the second one which consists in finding a minimum cost tree spanning *at least* one node from each cluster, denoted by L-GMST and introduced by Dror *et al.* [1].

Dror *et al.* [1] showed that the L-GMST problem is NP-hard, introduced three integer programming formulations, two of them have been proved later that there are invalid (see [2]) and examined a number of the heuristic solutions for the problem.

The aim of this paper is to describe new integer programming formulations of the L-GMST problem and establish relationships between the polytopes corresponding to their linear relaxations.

2. DEFINITION OF THE PROBLEM

The at least version of the generalized minimum spanning tree problem (L-GMST) is defined on an undirected graph $G = (V, E)$ with nodes partitioned into

Received: 25.03.2006; In revised form: 05.06.2006; Accepted: 01.11.2006
2000 *Mathematics Subject Classification:* 05C05, 68R10, 90C05, 90C10, 90C27, 90C39.
Key words and phrases: *Minimum spanning tree, generalized minimum spanning trees, integer programming, linear relaxation.*

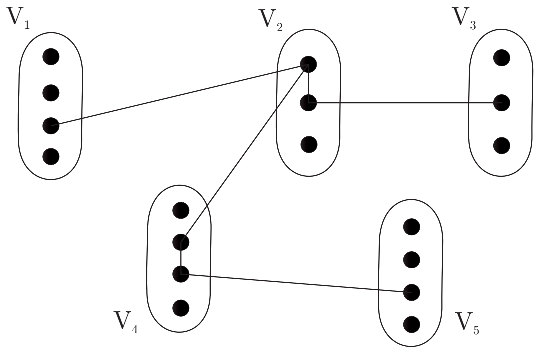

FIGURE 1. Example of a feasible solution of the L-GMST problem, where *at least* one node from each cluster is selected from each cluster

m clusters. Let $|V| = n$ and $K = \{1, 2, \ldots, m\}$ be the index set of the node sets (clusters). Then, $V = V_1 \cup V_2 \cup \ldots \cup V_m$ and $V_l \cap V_k = \emptyset$ for all $l, k \in K$ such that $l \neq k$. We assume that the graph G is complete and each edge $e = \{i, j\} \in E$ has a nonnegative cost denoted by c_{ij}.

The L-GMST problem is the problem of finding a minimum-cost tree spanning a subset of nodes which includes at least one node from each cluster.

3. INTEGER PROGRAMMING FORMULATIONS

The L-GMST problem can be formulated as an integer program in many different ways. For example, introducing the variables $x_e \in \{0,1\}$, $e \in E$ and $y_i \in \{0,1\}$, $i \in V$, to indicate whether an edge $'e'$ respectively a node $'i'$ is contained in the spanning tree, a feasible solution to the L-GMST problem can be seen as a cycle free subgraph, at least one node selected from every cluster and connecting all the clusters. Therefore the L-GMST problem can be formulated as the following 0-1 integer programming problem:

$$\min \sum_{e \in E} c_e x_e$$

$$\text{(3.1)} \quad s.t. \quad y(V_k) \geq 1, \quad \forall\, k \in K = \{1, ..., m\}$$

$$\text{(3.2)} \quad x(E(S)) \leq y(S - i), \ \forall\, i \in S \subset V, 2 \leq |S| \leq n - 1$$

$$\text{(3.3)} \quad x(E) = y(V) - 1$$

$$\text{(3.4)} \quad x_e \in \{0, 1\}, \quad \forall\, e \in E$$

$$\text{(3.5)} \quad y_i \in \{0, 1\}, \quad \forall\, i \in V.$$

where for $S \subseteq V$, the set of edges with both end points from S, denoted by $E(S)$, is defined as usually:

$$E(S) = \{\ e = (i, j) \in E \mid i, j \in S\}.$$

In the above formulation, we use the standard shorthand notations:

$$x(F) = \sum_{e \in F} x_e,\ F \subseteq E \text{ and } y(S) = \sum_{i \in S} y_i,\ S \subseteq V.$$

For simplicity we used the notation $S - i$ instead of $S \setminus \{i\}$. In the above formulation, constraints (3.1) guarantee that from every cluster we select at least one node, constraints (3.2) eliminate all the subtours and finally constraint (3.3) guarantees that the selected subgraph has $y(V) - 1$ edges.

This formulation, introduced by Feremans *et al.*, is called the *generalized subtour elimination formulation* since constraints (3.2) eliminate all the cycles.

We denote the feasible set of the linear programming relaxation of this formulation by P_{sub}, where we replace the constraints (3.4) and (3.5) by $0 \leq x_e, y_i \leq 1$, for all $e \in E$ and $i \in V$.

We may replace the subtour elimination constraints (3.2) by connectivity constraints, resulting in the so-called *generalized cutset formulation*:

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e x_e \\ s.t. \quad & (3.1), (3.3), (3.4), (3.5) \text{ and} \\ & x(\delta(S)) \geq y_i + y_j - 1,\ \forall\ i \in S \subset V, j \notin S \qquad (3.6) \end{aligned}$$

where for $S \subseteq V$, the *cutset*, denoted by $\delta(S)$, is defined as usually:

$$\delta(S) = \{\ e = (i,j) \in E \mid i \in S,\ j \notin S\}.$$

We denote the feasible set of the linear programming relaxation of this formulation by P_{cut} . The following property holds:

Proposition 3.1. $P_{sub} \subset P_{cut}$.

Proof. Let $(x, y) \in P_{sub}$ and $i \in S \subset V$ and $j \notin S$. Since $E = E(S) \cup \delta(S) \cup E(V \setminus S)$, we get

$$\begin{aligned} x(\delta(S)) &= x(E) - x(E(S)) - x(E(V \setminus S)) \\ &\geq y(V) - 1 - y(S) + y_i - z(V \setminus S) + y_j = y_i + y_j - 1. \end{aligned}$$

□

Our next model, the so-called *generalized multicut formulation*, is obtained by replacing simple cutsets by multicuts. Given a partition of the nodes $V = C_0 \cup C_1 \cup \ldots \cup C_k$, we define the multicut $\delta(C_0, C_1, ..., C_k)$ to be the set of edges connecting different C_i and C_j. The generalized multicut formulation for the L-GMST problem is:

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e x_e \\ s.t. \quad & (3.1), (3.3), (3.4), (3.5) \text{ and} \\ & x(\delta(C_0, C_1, ..., C_k)) \geq \sum_{j=0}^{k} y_{i_j} - 1,\ \forall\ C_0, C_1, ..., C_k \text{ node partitions} \\ & \text{of } V \text{ and } \forall\ i_j \in C_j \text{ for } j = 0, 1, ..., k. \qquad (3.7) \end{aligned}$$

Let P_{mcut} denote the feasible set of the linear programming relaxation of this model. Clearly, $P_{mcut} \subseteq P_{cut}$.

Proposition 3.2. $P_{sub} = P_{mcut}$.

Proof. It suffices to prove that $P_{sub} \subseteq P_{mcut}$ and $P_{mcut} \subseteq P_{sub}$. This boils down to showing that a pair (x, y) satisfies constraint (3.2) if and only if it satisfies constraint (3.7).

Let $(x, z) \in P_{sub}$ and $C_0, C_1, ..., C_q$ be a partition of the nodes. Since $E = E(C_0) \cup E(C_1) \cup ... \cup E(C_q) \cup \delta(C_0, C_1, ..., C_q)$, we get

$$\begin{aligned} x(\delta(C_0, C_1, ..., C_q)) &= x(E) - x(E(C_0)) - x(E(C_1)) - ... - x(E(C_q)) \geq \\ &\geq y(V) - 1 - y(C_0) + y_{i_0} - y(C_1) + y_{i_1} - ... - y(C_q) + y_{i_q} \\ &= \sum_{j=0}^{q} y_{i_j} - 1 \end{aligned}$$

where $i_j \in C_j$, $j = 0, 1, ..., q$.

Conversely, let $(x, y) \in P_{mcut}$, $i \in S \subset V$ and consider the inequality (3.7) with $C_0 = S$ and with $C_1, ..., C_k$ as singletons whose union is $V \setminus S$. Then

$$x(\delta(S, C_1, ..., C_k)) \geq \sum_{j=0}^{k} y_{i_j} - 1 = y_i + y(V \setminus S) - 1,$$

where $i \in S \subset V$. Therefore

$$\begin{aligned} x(E(S)) &= x(E) - x(\delta(S, C_1, ..., C_k)) \\ &\leq y(V) - 1 - y_i - y(V \setminus S) + 1 = y(S - i). \end{aligned}$$

□

4. FORMULATIONS FOR THE DIRECTED PROBLEM

In order to achieve a tighter LP-relaxation, we transform the original problem defined on an undirected graph into a problem defined on a directed graph, obtaining the *at least version of the generalized minimum arborescence problem*.

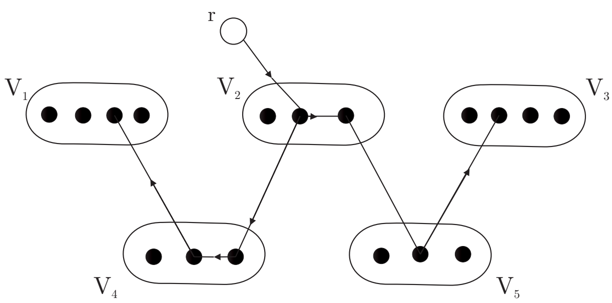

FIGURE 2. Example of a feasible solution of the L-GMST problem, in the case when the underling graph is directed

The directed graph $D = (V_d, A)$ is characterized by the vertex set $V_d = V \cup \{r\}$ which consists of the vertices of the input graph G and an artificial root vertex r and by the arc set A which consists of opposite arcs (i, j) and (j, i) for each edge

$e = (i, j) \in E$ having the same weight as the edge $(i, j) \in E$, plus a set of arcs from the root node to each node from the vertex set V, having the cost equal to 0.

The directed version of the L-GMST problem is defined on the directed graph $D = (V_d, A)$ and consists of determining a minimum cost arborescence which includes at least one node from every cluster.

In addition to the variables already introduced, we introduce the variables $w_a \in \{0, 1\}$, $a \in A$ to indicate whether an arc $'a'$ is contained in the spanning arborescence.

We consider first a *directed generalized cutset formulation* of the L-GMST problem.

$$\begin{aligned}
& \min && \sum_{e \in E} c_e x_e \\
& s.t. && y(V_k) \geq 1, && \forall\, k \in K = \{1, ..., m\} \\
& && w(A) = y(V_d) - 1 \\
(4.8)\quad & && w(\delta^-(S)) \geq y_i, && \forall\, i \in S \subseteq V,\ r \notin S \\
(4.9)\quad & && \sum_{i \in V} x_{ri} = 1 \\
(4.10)\quad & && w_{ij} + w_{ji} = x_e\ , && \forall\, e = (i, j) \in E \\
(4.11)\quad & && x, y, w \in \{0, 1\}.
\end{aligned}$$

where for $S \subseteq V$, the *set of arcs entering S*, denoted by $\delta^-(S)$, is defined as usually:

$$\delta^-(S) = \{(i, j) \in A \mid i \notin S,\ j \in S\}.$$

In this model constraints (4.8) and (4.9) guarantee the existence of a path from the artificial root node to any other selected node which includes only the selected nodes.

Let P_{dcut} denote the projection of the feasible set of the linear programming relaxation of this model into the (x, y)-space.

We introduced now a formulation of the L-GMST problem based on branchings. Consider, as in the previous formulation, the digraph $D = (V_d, A)$. We define the *branching model* of the L-GMST problem to be:

$$\begin{aligned}
& \min && \sum_{e \in E} c_e x_e \\
& s.t. && y(V_k) \geq 1, && \forall\, k \in K = \{1, ..., m\} \\
& && w(A) = y(V_d) - 1 \\
& && \sum_{i \in V} x_{ri} = 1 \\
(4.12)\quad & && w(A(S)) \leq y(S - i), && \forall\, i \in S \subset V, 2 \leq |S| \leq n - 1 \\
(4.13)\quad & && w(\delta^-(V_k)) \geq 1\ , && \forall\, k \in K \\
& && w_{ij} + w_{ji} = x_e\ , && \forall\, e = (i, j) \in E \\
& && x, y, w \in \{0, 1\}.
\end{aligned}$$

Let P_{branch} denote the projection of the feasible set of the linear programming relaxation of this model into the (x, y)-space. Obviously, $P_{branch} \subseteq P_{sub}$.

The following result holds:

Proposition 4.3. $P_{branch} = P_{dcut} \cap P_{sub}$.

Proof. First we prove that $P_{dcut} \cap P_{sub} \subseteq P_{branch}$.

Let $(x, z) \in P_{dcut} \cap P_{sub}$. Using the connectivity constrained (4.8) for $S = V_k$ and knowing that we have to span all the clusters implies that constrained (4.13) is fulfilled. Therefore $(x, z) \in P_{branch}$.

We show that $P_{branch} \subseteq P_{dcut} \cap P_{sub}$.

It is obvious that $P_{branch} \subseteq P_{sub}$, therefore it remains to show $P_{branch} \subseteq P_{dcut}$.

Let $(x, y) \in P_{branch}$.

Now we show that $w(\delta^-(l)) \leq y_l$, for $l \in V_k, k \in K$.

Take $V^l = \{i \in V \mid (i, l) \in \delta^-(l)\}$ and $S^l = V^l \cup \{l\}$, then $w(\delta^-(l)) = w(A(S^l))$ and choose $i_l \in V^l$.

$$\begin{aligned} 1 &\leq \sum_{l \in V_k} w(\delta^-(l)) = \sum_{l \in V_k} w(A(S^l)) \leq \sum_{l \in V_k} y(S^l \backslash i_l) \\ &= \sum_{l \in V_k} y_l + \sum_{l \in V_k} \sum_{j \in V^l \backslash i_l} y_j = 1 + \sum_{l \in V_k} \sum_{j \in V^l \backslash i_l} y_j. \end{aligned}$$

Therefore, for all l there is only one $i_l \in V^l$ with $y_{i_l} \neq 0$ and

$$w(\delta^-(l)) = w(A(S^l)) \leq y(S^l \backslash i_l) = y_l.$$

For every $i \in S \subset V$

$$w(A(S)) = \sum_{i \in S} w(\delta^-(i)) - w(\delta^-(S)) \leq y(S - i),$$

which implies that:

$$\begin{aligned} w(\delta^-(S)) &\geq \sum_{i \in S} w(\delta^-(i)) - y(S) + y_i \\ &= \sum_{i \in S} \left[1 - \sum_{l \in V_k \backslash \{i\}} w\left(\delta^-(l)\right) \right] - y(S) + y_i \\ &\geq \sum_{i \in S} \left[1 - \sum_{l \in V_k \backslash \{i\}} y_l \right] - y(S) + y_i = y(S) - y(S) + y_i = y_i. \end{aligned}$$

□

REFERENCES

[1] Dror, M., Haouari, M. and Chaouachi, J., *Generalized Spanning Trees*, European Journal of Operational Research, 120, 583-592, (2000)

[2] Feremans, C., *Generalized Spanning Trees and Extensions*, PhD thesis, Universite Libre de Bruxelles, Belgium, (2001)

[3] Feremans, C., Labbe, M. and Laporte, G., *A Comparative Analysis of Several Formulations of the Generalized Minimum Spanning Tree Problem*, Networks **39**, Issue 1, 29-34, (2002)

[4] Myung, Y. S., Lee, C. H., Tcha, D. W. *On the Generalized Minimum Spanning Tree Problem. Networks*, **26**, 231-241 (1995)

[5] Pop, P. C., *The Generalized Minimum Spanning Tree Problem*, PhD thesis, University of Twente, The Netherlands, (2002)

[6] Pop, P. C., *New Models of the Generalized Minimum Spanning Tree Problem*, Journal of Mathematical Modelling and Algorithms, **3**, Issue 2, 153-166, (2004)

[7] Pop, P. C., *On the Prize-Collecting Generalized Minimum Spanning Tree Problem*, to appear in Annals of Operations Research

[8] Pop, P. C., Kern, W. and Still, G., *Approximation Theory in Combinatorial Optimization. Application to the Generalized Minimum Spanning Tree Problem*, Revue d'Analyse Numerique et de Theorie de l'Approximation, Tome 34, No. 1, 93-102, (2005)

[9] Pop, P. C., Kern, W. and Still, G., *A New Relaxation Method for the Generalized Minimum Spanning Tree Problem*, European Journal of Operational Research, 170, 900-908, (2006)

NORTH UNIVERSITY OF BAIA MARE, FACULTY OF SCIENCES
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE
VICTORIEI 76, 430122 BAIA MARE, ROMANIA
E-mail address: pop_petrica@yahoo.com
E-mail address: hmandrei@yahoo.com

BABEŞ-BOLYAI UNIVERSITY,
FACULTY OF ECONOMICS,
CLUJ-NAPOCA, ROMANIA
E-mail address: sitarcorina@yahoo.com