Bul. Stiint. Univ. Baia Mare, Ser. B,
Matematică-Informatică, Vol. XVIII(2002), Nr.1, 95 - 104

Dedicated to Costică MUSTĂŢA on his 60^{th} anniversary

APPROXIMATION RESULTS FOR THE GENERALIZED MINIMUM SPANNING TREE PROBLEM

Petrică Claudiu POP

Abstract. We consider the Generalized Minimum Spanning Tree problem denoted by GMST. It is known that the GMST problem is NP-hard. Throughout this paper we distinguish between so-called positive results and negative results in the area of approximation theory. We present an in-approximability result for the GMST problem and under special assumptions we give an approximation algorithm for the problem.

MSC: 90C11, 90C27, 05C05, 90B10.

Keywords: approximation algorithms, combinatorial optimization, generalized minimum spanning trees, LP relaxation.

1.Introduction

The *Generalized Minimum Spanning Tree Problem* is defined on an undirected graph $G = (V, E)$ with the nodes partitioned into m node sets called *clusters*. Let $|V| = n$ and $K = \{1, 2, ..., m\}$ be the node index of the 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 edges are defined only between nodes belonging to different clusters and each edge $e = (i, j) \in E$ has a nonnegative cost c_e.

The GMST is the problem of finding a minimum cost tree spanning a subset of nodes which includes *exactly* one node from each cluster. We will call a tree containing exactly one node from each cluster a *generalized spanning tree*.

The GMST problem was introduced by Myung, Lee and Tcha [6]. Feremans, Labbé and Laporte in [1] present several integer formulations of the GMST problem and compare them in terms of their linear programming relaxations, and in [2] they study the polytope associated with the GMST problem.
The following result was proven by Myung *et al.* in [6].

Theorem 1. *The GMSTP is NP-hard.* ■

Because the GMST problem is NP-hard it is very unlikely that this problem could be solved by a polynomial time algorithm. At the expence of reducing the quality of the solution by relaxing some of the requirements, we can get often speed-up in the complexity. This leads us to the following definition:

Definition 1. *(Approximation algorithms)*
Let X be a minimization problem and $\alpha > 1$. An algorithm APP is called an α-approximation algorithm for problem X, if for all instances I of X it delivers in polynomial time a feasible solution with objective value $APP(I)$ such that

$$APP(I) \leq \alpha OPT(I) \tag{1}$$

where by $APP(I)$ and $OPT(I)$ we denoted the values of an approximate solution and that of an optimal solution for instance I, respectively. ■

The value α is called the *performance guarantee* or the *worst case ratio* of the approximation algorithm APP. The closer α is to 1 the better the algorithm is.

2. A negative result for the GMST problem

For some hard combinatorial optimization problems it is possible to show that they don't have an approximation algorithm unless $\mathcal{P} = \mathcal{NP}$. In order to give a result of this form it is enough to show that the existance of an α-approximation algorithm would allow one to solve some decision problem, known to be $\mathcal{NP}$-*complete*, in polynomial time.

Applying this scheme to the GMST problem we obtain an in-approximability result. This result is a different formulation in terms of approximation algorithms of a result provided by Myung et al. [6] which says that even finding a near optimal solution for the GMST problem is $\mathcal{NP}$-*hard*. The proof of this result similar with the proof provided in [6].

Theorem 2. *Under the assumption $\mathcal{P} \neq \mathcal{NP}$, there is no α-approximation algorithm for the GMST problem.* ■

3. An Approximation Algorithm for Bounded Cluster Size

As we have seen in the previous section there exists no α-approximation algorithm for the GMST problem under the assumption $\mathcal{P} \neq \mathcal{NP}$.

However under the following assumptions:

A1: the graph has bounded cluster size, i.e. $|V_k| \leq \rho$, for all $k = 1, ..., m$

A2: the cost function is strict positive and satisfies the cost function!triangle inequality triangle inequality, i.e. $c_{ij} + c_{jk} \geq c_{ik}$ for all $i, j, k \in V$,

a polynomial approximation algorithm for GMST problem is possible.

In this section under the above assumptions we present an approximation algorithm for the GMST problem with performance ratio 2ρ. The approximation algorithm is constructed following the ideas of Slavik [8] where the Generalized Traveling Salesman Problem and Group Steiner Tree Problem have been treated.

3.1 An integer programming formulation of GMST problem

We define for each edge $\{i, j\}$ and each node i the binary variables:

$$x_{ij} = \begin{cases} 1 & \text{if edge } \{i,j\} \text{ is included in the selected subgraph} \\ \\ 0 & \text{otherwise} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if node } i \text{ is included in the selected subgraph} \\ \\ 0 & \text{otherwise} \end{cases}$$

The GMST problem can be formulated as the following integer programming problem:

Problem IP1:

$$\begin{array}{llll} Z_1 = \min & \sum_{e \in E} c_e x_e & & (1) \\ s.t. & z(V_k) = 1, & \forall\, k \in K = \{1, ..., m\} & (2) \\ & x(\delta(S)) \geq z_i, & \forall\, i \in S, \forall S \subset V \text{ such that for} & (3) \\ & & \text{some cluster } V_k, \text{ with } k \in K,\ S \cap V_k = \emptyset & (4) \\ & x(E) = m - 1 & & (5) \\ & x_e \in \{0, 1\}, & \forall\, e \in E & (6) \\ & z_i \in \{0, 1\}, & \forall\, i \in V. & (2) \end{array}$$

We use here the standard shorthand notations: for every subset S of V, $E(S) = \{(i,j) \in E | i, j \in S\}$, $x(E(S)) = \sum_{e \in E(S)} x_e$, $y(S) = \sum_{j \in S} y_j$ and as usual the cutset $\delta(S)$ is defined by

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

Condition (2) guarantees that a feasible solution contains exactly one vertex from every cluster. Condition (3) guarantees that any feasible solution is a connected subgraph. Condition (4) simply assures that any feasible solution has m-1 edges and due to the fact that the cost function is non-negative this constraint is redundant.

Consider now the linear programming relaxation of the integer programming formulation of the GMST problem. In order to do that, we simply replace conditions (5) and (6) in IP1 by new conditions:

$$0 \leq x_e \leq 1, \quad \text{for all } e \in E \tag{7}$$
$$0 \leq z_i \leq 1, \quad \text{for all } i \in V. \tag{8}$$

3.2 An Approximation Algorithm for GMST problem

We assume that the assumptions A1 and A2 hold.

The algorithm for approximating the optimal solution of the GMST problem is as follows:

- **Algorithm "Approximate the GMST problem"**

Input: A complete graph $G = (V, E)$ with strictly positive cost function on the edges satisfying the triangle inequality, and with the nodes partitioned into the clusters $V_1, ..., V_m$ with bounded size, $|V_k| \leq \rho$.

Output: A tree $T \subset G$ spanning some vertices $W' \subset V$ which includes exactly one vertex from every cluster, that approximates the optimal solution to the GMST problem.

1. Solve the linear programming relaxation of the problem IP1 and let $(z^*, x^*, Z_1^*) = ((z_i^*)_{i=1}^n, (x_e^*)_{e\in E}, Z_1^*)$ be the optimal solution.

2. Set $W^* = \left\{ i \in V | z_i^* \geq \frac{1}{\rho} \right\}$ and consider $W' \subset W^*$ with the property that W' has exactly one vertex from each cluster, and find a minimum spanning tree $T \subset G$ on the subgraph G' generated by W'.

3. Output $APP = cost(T)$ and the generalized spanning tree T.

Even though the linear programming relaxation of the problem IP1 has exponentially many constraints, it can still be solved in polynomial time either using ellipsoid method with a min-cut max-flow oracle [3] or using Karmakar's algorithm [5] since the linear programming relaxation can be formulated "compactly" (the number of constraints polynomially bounded) see [7].

3.3 Auxiliary results

In order to establish upper bounds on the performance ratio of the above algorithm, we now present some auxiliary results. Let now $W \subset V$ and consider the following linear program:

Problem LP2:

$$Z_2^*(W) = \min \quad \sum_{e \in E} c_e x_e \tag{3}$$

$$\text{s.t.} \quad x(\delta(S)) \geq 1, \; S \subset V, \text{ s.t. } W \cap S \neq \emptyset \neq W \setminus S \tag{9}$$

$$x(\delta(i)) = 0, \; i \in V \setminus W \tag{10}$$

$$0 \leq x_e \leq 1, \; e \in E. \tag{11}$$

Replacing constraints (11) with the integrality constraints $x_e \in \{0,1\}$, the formulation obtained is the formulation of the minimum tree spanning the subset of nodes $W \subset V$

Consider the following relaxation of the problem LP2.

Problem LP3:

$$Z_3^*(W) = \min \quad \sum_{e \in E} c_e x_e$$

$$\text{s.t.} \quad x(\delta(S)) \geq 1, \; S \subset V, \text{ s.t. } W \cap S \neq \emptyset \neq W \setminus S$$

$$0 \leq x_e, \quad e \in E. \tag{12}$$

Thus we omitted constraint (10) and relaxed constraint (11).

The following result is a straightforward consequence of the parsimonius property (see [4]) if we choose $r_{ij} = 1$, if $i, j \in W$, and 0 otherwise, and $D = V \setminus W$.

Lemma 1. *The optimal solution values to problems LP2 and LP3 are the same, that is*

$$Z_2^*(W) = Z_3^*(W).$$

■

Consider the following problem:

Problem IP4:

$$Z_4 = \min \quad \sum_{e\in E} c_e x_e$$

$$\text{s.t.} \quad x(\delta(S)) \geq 1, \ S \subset V, \text{ s.t. } S \neq \emptyset \neq V \tag{13}$$

$$x_e \in \{0,1\}, \quad e \in E. \tag{14}$$

Clearly, it is the integer programming formulation of the MST (minimum spanning tree) problem. Let LP4 be the LP relaxation of this formulation, that is we simply replace the constraint (14) by the constraint $0 \leq x_e \leq 1, \ forall e \in E$.

Denote by Z_4^* the value of the optimal solution of the LP4. The following known result for minimum spanning trees holds:

Proposition 1.

$$L^T(V) \leq (2 - \frac{2}{|V|})Z_4^*.$$

where $L^T(V)$ denotes the cost of the minimum spanning tree on V.

Proof: *See for example [8].* ■

Let $W \subset V$, then Proposition 5 can be easily modified to obtain:

Proposition 2.

$$L^T(W) \leq (2 - \frac{2}{|W|})Z_2^*(W).$$

Proof: *Let (x_e) be a feasible solution to LP2. If $e \notin E(W) = \{(i,j) \mid i,j \in W\}$ implies $x_e = 0$ and using Proposition 5 we prove the inequality.* ■

4. Performance Bounds

Let $(y^*, x^*, Z_1^*) = ((y_i^*)_{i=1}^n, (x_e^*)_{e\in E}, Z_1^*)$ be the optimal solution to the LP relaxation fo the GMST problem. Define

$$\hat{x}_e = \rho x_e^*$$

$$\widehat{y}_i = \begin{cases} 1 & \text{if } y_i^* \geq \frac{1}{\rho} \\ 0 & \text{otherwise} \end{cases}$$

$W = \left\{ i \in V \,|\, y_i^* \geq \frac{1}{\rho} \right\} = \{ i \in V \,|\, \widehat{y}_i = 1 \}$. Because we need only one vertex from every cluster we delete extra vertices from W and consider $W' \subset V$ such that $|W'| = $ m and W' consists of exactly one vertex from every cluster.

Since LP1 is the LP relaxation of the problem IP1, we have

$$Z_1^* \leq Z_1$$

Now let us show that $(\widehat{x}_e)_{e \in E}$ is a feasible solution to LP4. Indeed, $\widehat{x}_e \geq 0$ for all $e \in E$, hence condition (12) is satisfied. Let $S \subset V$ be such that $W' \cap S \neq \phi \neq W' \setminus S$ and choose some $i \in W' \cap S$. Hence $\widehat{y}_i = 1$ and $y_i^* \geq \frac{1}{\rho}$. Then we have

$$\sum_{e \in \delta(S)} \widehat{x}_e = \rho \sum_{e \in \delta(S)} x_e^* \geq \rho y_i^* \geq \rho \frac{1}{\rho} = 1$$

by definition of $\widehat{x}_e$ and the fact that the x_e^* are solution to LP1. Hence the $\widehat{x}_e$ satisfy constraint (9) in LP3.

Therefore,

$$\begin{aligned} APP &= L^T(W') \leq (2 - \frac{2}{|W'|}) Z_3^* = (2 - \frac{2}{|W'|}) Z_4^* \leq (2 - \frac{2}{|W'|}) \sum_{e \in E} c_e \widehat{x}_e \\ & (2 - \frac{2}{|W'|}) \rho \sum_{e \in E} c_e x_e^* = (2 - \frac{2}{|W'|}) \rho Z_1^* \\ &\leq (2 - \frac{2}{|W'|}) \rho Z_1 = (2 - \frac{2}{|W'|}) \rho OPT. \end{aligned}$$

And since $W' \subset V$, that is m $= |W'| \leq |V| = $ n, we have proved the following.

Theorem 3. *The performance ratio of the algorithm "Approximate GMST problem" for approximating the optimum solution to the GMST problem satisfies:*

$$\frac{APP}{OPT} \leq (2 - \frac{2}{n}) \rho.$$

■

One can easily generalize the algorithm and its analysis to the case when, in addition to distances between edges, there is a cost, say d_i, associated with each vertex $i \in V$.

In this case the GMST problem can be formulated as the following integer program:

$$OPT = \min \sum_{e \in E} c_e x_e + \sum_{i \in V} d_i z_i$$
$$s.t. \quad (4.2) - (4.6).$$

Suppose that $(\overline{x}, \overline{z})$ is an optimal solution. Then the optimal value OPT of this integer program consists of two parts:

$$L_{OPT} := \sum_{e \in E} c_e \overline{x}_e \text{ and } V_{OPT} := \sum_{i \in V} d_i \overline{z}_i.$$

Under the same assumptions A1 and A2, the algorithm for approximating the optimal solution of the GMST problem in this case, is as follows:

1. Solve the linear programming relaxation of the previous integer program and let $(z^*, x^*) = ((z_i^*)_{i=1}^n, (x_e^*)_{e \in E})$ be the optimal solution.

2. Set $W^* = \left\{ i \in V | z_i^* \geq \frac{1}{\rho} \right\}$ and consider $W' \subset W^*$ with the property that W' has exactly one vertex from each cluster, and find a minimum spanning tree $T \subset G$ on the subgraph G' generated by W'.

3. Output $APP = ecost(T) + vcost(T)$ and the generalized spanning tree T.

where by $ecost(T)$ and $vcost(T)$ we denoted the cost of the tree T with respect to the edges, respectively to the nodes.

Regarding the performance bounds of this approximation algorithm, using the same auxiliary results and defining $(\hat{x}_e, \hat{z}_i)$ as we did at the beginning of this subsection, the following inequalities hold:

$$L^T(W') \leq \rho(2 - \frac{2}{n}) L_{OPT},$$

$$V^T(W') \leq \rho V_{OPT},$$

where $L^T(W')$, $V^T(W')$ denote the cost of the tree T spanning the nodes of W' with respect to the edges, respectively to the nodes and as before $W' \subset W^* = \left\{ i \in V \mid z_i^* \geq \frac{1}{\rho} \right\}$, such that $|W'| = m$ and W' consists of exactly one vertex from every cluster.

For the approximation algorithm proposed in this case, the following holds:

$$\begin{aligned} APP &= L^T(W') + V^T(W') \leq \rho(2 - \frac{2}{n})L_{OPT} + \rho V_{OPT} \\ &\leq \rho(2 - \frac{2}{n})(L_{OPT} + V_{OPT}) = \rho(2 - \frac{2}{n})OPT. \end{aligned}$$

REFERENCES

[1] C. Feremans, M. Labbe and G. Laporte, A comparative Analysis of Several Formulations for the Generalized Minimum Spanning Tree Problem, Technical Report IS-MG 22, Universite Libre de Bruxelles, Belgium, 2000.

[2] C. Feremans, M. Labbe and G. Laporte, Polyhedral Analysis of the Generalized Minimum Spanning Tree Problem, Technical Report IS-MG 1, Universite Libre de Bruxelles, Belgium, 2000.

[3] M. Grotschel, L. Lovasz and A. Scrijver, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica, 1, 169-197, 1981.

[4] M.X. Goemans and D.J. Bertsimas, Survivable networks, linear programming relaxations and the parsimonious property. Mathematical Programming, 60: 145-166, 1993.

[5] N.K. Karmarkar, A new polynomial-time algorithm for linear programming. Combinatorica, 4: 373-395, 1984.

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

[7] P.C. Pop, U. Faigle, W. Kern, G. Still, Relaxation Methods for the Generalized Minimum Spanning Tree Problem, submitted.

[8] P. Slavik, On the Approximation of the Generalized Traveling Salesman Problem, submitted.

Received: 11.07.2002

Faculty of Mathematical Sciences, University of Twente
P.O. Box 217, 7500 AE Enschede, The Netherlands
E-mail: pop_petrica@yahoo.com

Department of Mathematics and Computer Science
North University of Baia Mare, Str. Victoriei nr. 76
4800 Baia Mare ROMANIA
E-mail: pop_petrica@yahoo.com