Stronger Lower Bounds for the Quadratic Minimum Spanning Tree Problem with Adjacency Costs

Dilson Lucas Pereira [a,1,3], Michel Gendreau [b,4], Alexandre Salles da Cunha [c,2,5]

[a] *CAPES Foundation, Ministry of Education of Brazil, Brasília - DF 70.040-020, Brazil*

[b] *École Polytechnique de Montréal, Montréal - Canada*

[c] *Departamento de Ciência da Computação, Universidade Federal de Minas Gerais, Belo Horizonte - MG, Brazil*

Abstract

We address a particular case of the quadratic minimum spanning tree problem in which interaction costs only apply for adjacent edges. Motivated by the fact that Gilmore-Lawler procedures in the literature underestimate the contribution of interaction costs to compute lower bounds, we introduce a reformulation that allows stronger linear programming bounds to be computed. An algorithm based on dynamic column and row generation is presented for evaluating these bounds. Our computational experiments indicate that the reformulation introduced here is indeed much stronger than those in the literature.

Keywords: quadratic 0-1 programming, spanning trees, column generation

1 Introduction

Given a connected, undirected graph $G = (V, E)$, with $n = |V|$ vertices and $m = |E|$ edges, and a matrix $Q = (q_{ij} \geq 0)_{i,j \in E}$ of interaction costs, the quadratic minimum spanning tree problem (QMSTP) asks for a spanning tree of G whose incidence vector $\mathbf{x} \in \mathbb{B}^m$ minimizes $\sum_{i,j \in E} q_{ij} x_i x_j$. In this work, we focus on a particular case of QMSTP where costs only apply for adjacent edges, i.e., $q_{ij} = 0$ if i and j do not share an endpoint.

To our knowledge, this variant was first studied by Assad and Xu [1], where it was named the adjacent only QMSTP (AQMSTP). The authors proved that both QMSTP and AQMSTP are NP-Hard. Lower bounding procedures and branch-and-bound (BB) algorithms were devised. The general case of the QMSTP was also investigated by Cordone and Passeri [2] and Öncan and Punnen [4]. A BB algorithm is presented in the former, while only a lower bounding procedure is presented in the latter. Applications for the QMSTP and the AQMSTP can be found in hydraulic, communication, and transportation networks [1].

Motivated by the fact that Gilmore-Lawler procedures in the literature underestimate the contribution of interaction costs to compute lower bounds, we introduce a reformulation that allows stronger linear programming (LP) bounds to be computed. An algorithm based on dynamic column and row generation is presented for evaluating them.

2 Formulations

2.1 A QMSTP formulation coming from the literature

In order to formulate QMSTP as a binary linear program, consider binary decision variables $\mathbf{x} = (x_i)_{i \in E}$ to determine whether ($x_i = 1$) or not ($x_i = 0$) an edge $i \in E$ appears in the solution. Consider also linearization variables $\mathbf{y} = (\mathbf{y}_i)_{i \in E}$, $\mathbf{y}_i = (y_{ij})_{j \in E}$, to represent the product $x_i x_j$, $i, j \in E$. We use the following notation throughout the paper. Given $S \subseteq V$, $E(S) \subseteq E$ denotes the edges with both endpoints in S and $\delta(S)$ denotes the edges with only one endpoint in S. If S has only one vertex, say v, we replace $\delta(\{v\})$ by

[1] Dilson Lucas Pereira was funded by CAPES, BEX 2418/11-8.
[2] Alexandre Salles da Cunha is partially funded by CNPq grants 302276/2009-2, 477863/2010-8 and FAPEMIG grant PRONEX APQ-01201-09
[3] Email: `dilson@dcc.ufmg.br`
[4] Email: `michel.gendreau@cirrelt.ca`
[5] Email: `acunha@dcc.ufmg.br`

$\delta(v)$. X denotes the set of all incidence vectors of spanning trees of G and $X_0 = X \cup \{\mathbf{0}\}$.

The following QMSTP formulation was introduced by Assad and Xu [1],

$$\min \sum_{i,j \in E} q_{ij} y_{ij} \tag{1}$$

$$\text{s.t.} \quad \mathbf{x} \in X, \tag{2}$$

$$\mathbf{y}_i \in X_0, \qquad i \in E, \tag{3}$$

$$y_{ii} = x_i, \qquad i \in E, \tag{4}$$

$$\sum_{j \in E} y_{ji} = (n-1)x_i, \qquad i \in E. \tag{5}$$

Consider a relaxation for the QMSTP model above, where constraints (5) are dropped from the formulation. To obtain an optimal solution $(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \mathbb{B}^{m+m^2}$ to such a relaxation, one proceeds as follows. For each $i \in E$, solve

$$\overline{q}_i = \min\{\sum_{j \in E} q_{ij} y_{ij} : \mathbf{y}_i \in X, y_{ii} = 1\}, \tag{6}$$

and let $\overline{\mathbf{y}}_i \in \mathbb{B}^m$ be its solution. Then, find the spanning tree implied by

$$\overline{\mathbf{x}} \in \arg\min\{\sum_{i \in E} \overline{q}_i x_i : \mathbf{x} \in X\}. \tag{7}$$

Once (7) is solved, set $\tilde{\mathbf{x}} = \overline{\mathbf{x}}$ and $\tilde{\mathbf{y}}_i = x_i \overline{\mathbf{y}}_i$, $i \in E$. This process is precisely the well known Gilmore-Lawler [3] procedure, applied for QMSTP.

Assad and Xu [1] devised a Lagrangian relaxation algorithm in which after relaxed, Lagrangian multipliers are attached to constraints (5). A similar scheme was employed by Öncan and Punnen [4]. They relax both (5) and the QMSTP valid inequalities

$$\sum_{i \in \delta(v)} y_{ij} \geq x_j, \qquad j \in E, v \in V. \tag{8}$$

In their approach, multiplier adjustment is only implemented for (8).

Of course, the lower bounding procedure outlined above can be applied to AQMSTP. However, Gilmore-Lawler lower bounds, even with multipliers adjustment, tend to be much weaker when applied to AQMSTP. The reason is the following. When solving (6) for an edge $i = \{u, v\} \in E$, the optimal solution $\overline{\mathbf{y}}_i$ will be implied by a spanning tree in which only one edge adjacent to i is selected, namely the edge $j \in \delta(u) \cup \delta(v)$ with the smallest q_{ij} cost. This fact suggests how underestimated the costs $\overline{q}_i$ may be for the computation of the Gilmore-Lawler lower bound. Next, we present a binary linear programming reformulation for AQMSTP that attempts to overcome this drawback.

2.2 A reformulation for AQMSTP

Let a star of G centered at $v \in V$ be any non-empty subset of edges of $\delta(v)$. Denote by $\mathcal{S}_v$ the set of all stars of G centered at $v \in V$, let $\mathcal{S} = \bigcup_{v \in V} \mathcal{S}_v$, and consider the vector of binary decision variables $\mathbf{t} = (t_H)_{H \in \mathcal{S}}$, used to determine whether $(t_H = 1)$ or not $(t_H = 0)$ a star $H \in \mathcal{S}$ is included in a AQMSTP solution. Consider also the variables $\mathbf{x}$ defined before. A reformulation for AQMSTP is given by

$$
\begin{aligned}
(F_{\text{star}}) \qquad & \min \sum_{H \in \mathcal{S}} q_H t_H && && (9)\\
& \text{s.t.} \quad \sum_{H \in \mathcal{S}_v} t_H = 1, && v \in V, && (10)\\
& \sum_{H \in \mathcal{S}_u : i \in H} t_H = \sum_{H \in \mathcal{S}_v : i \in H} t_H, && i = \{u, v\} \in E, && (11)\\
& x_i = \sum_{H \in \mathcal{S}_v : i \in H} t_H, && i = \{u, v\} \in E, && (12)\\
& \mathbf{x} \in X, && && (13)\\
& \mathbf{t} \in \mathbb{B}^{|\mathcal{S}|}. && && (14)
\end{aligned}
$$

In the reformulation above, $q_H = \sum_{i,j \in H : i \neq j} q_{ij} + (1/2) \sum_{i \in H} q_{ii}$, $H \in \mathcal{S}$. For stating constraints (11) and (12), we assume $u < v$, for an edge $i = \{u, v\} \in E$.

Constraints (10) guarantee that exactly one star centered at v must be selected. Edges in this star connect v to its neighbors in the tree we are looking for. Constraints (11) impose that, if an edge $\{u, v\}$ appears in the star centered at v, it must also appear in the star centered at u. Constraints (12) couple $\mathbf{x}$ and $\mathbf{t}$ variables. Finally, constraints (13) impose that the set of selected edges must imply a spanning tree of G.

3 Algorithms

We assume that X is represented by the extreme points of the integral polytope implied by $\{\sum_{i \in E} x_i = n - 1,\ 0 \leq x_i \leq 1, i \in E\}$ and the exponentially many subtour elimination constraints (SEC)

$$\sum_{i \in E(S)} x_i \leq |S| - 1, \quad S \subset V, |S| > 2. \tag{15}$$

Formulation F_{star} thus involves exponentially many rows and columns, one for each star centered at $v \in V$. We now describe how its LP bounds are evaluated by a dynamic generation of columns and cutting planes (15). To that aim, assume that a restricted set of constraints (15) and stars are present in a restricted linear programming master program (RLMP) that derives from F_{star}, after replacing (14) by $\mathbf{t} \geq \mathbf{0}$. New columns and cuts are added to RLMP in rounds of column and row generation. We solve the first RLMP and then proceed with column generation. When no stars with negative reduced cost are found, we separate (15). If violated SECs are found, they are appended into a new reinforced RLMP, which is re-optimized. A new round of column generation then follows. The procedure stops when no violated cuts nor columns with negative reduced cost are found. At that moment, the LP relaxation of F_{star} is solved.

The separation of (15) is conducted by means of the exact separation algorithm of Padberg and Wolsey [5]. For solving the pricing problems, assume that optimal dual variables $\hat{\boldsymbol{\alpha}} = (\hat{\alpha}_v)_{v \in V}$, $\hat{\boldsymbol{\beta}} = (\hat{\beta}_i)_{i \in E}$, and $\hat{\boldsymbol{\gamma}} = (\hat{\gamma}_i)_{i \in E}$, respectively assigned to constraints (10), (11), and (12), are available after RLMP is solved. For a given $v \in V$, the associated pricing problem consists in finding a star $H \in \mathcal{S}_v$ that violates the constraint

$$q_H - \hat{\alpha}_v - \sum_{i=\{v,u\} \in H} \hat{\beta}_i + \sum_{i=\{u,v\} \in H} \hat{\beta}_i + \sum_{i=\{u,v\} \in H} \hat{\gamma}_i \geq 0. \tag{16}$$

For each $v \in V$, let $\mathbf{w}_v = (w_{vi})_{i \in \delta(v)}$ be a vector of binary variables. The pricing problem for $v \in V$ is formulated as the following unconstrained quadratic binary program

$$\min\{ \sum_{i,j \in \delta(v)} q'_{ij} w_{vi} w_{vj} : \mathbf{w}_v \in \mathbb{B}^{|\delta(v)|} \} - \hat{\alpha}_v. \tag{17}$$

For i and $j \in \delta(v)$, the following costs q'_{ij} are used. If $i \neq j$, $q'_{ij} = q_{ij}$. If $i = j$, let $\eta(v)$ be the neighbor of v in i. We have two cases: (i) if $v > \eta(v), q'_{ii} = q_{ii} + \hat{\beta}_i + \hat{\gamma}_i$ and (ii) if $v < \eta(v)$, $q'_{ii} = q_{ii} - \hat{\beta}_i$.

3.1 Solving the pricing problems

To solve the pricing problem assigned to $v \in V$, we first linearize (17) and then implement a cut-and-branch algorithm. To that aim, consider the introduction of linearization variables $\mathbf{z}_v = (\mathbf{z}_{vi})_{i\in\delta(v)}$, where $\mathbf{z}_{vi} = (z_{vij})_{j\in\delta(v):j>i}$, $i \in \delta(v)$ ($z_{vji} := z_{vij}$ if $j > i$). A linear integer program (IP) for the pricing problem is

$$\min \sum_{i,j\in\delta(v):i\neq j} q'_{ij} z_{vij} + \sum_{i\in\delta(v)} q'_{ii} w_{vi} - \hat{\alpha}_v \tag{18}$$

$$\text{s.t.} \quad z_{vij} \leq w_{vi}, \qquad i < j \in \delta(v), \tag{19}$$

$$z_{vij} \leq w_{vj}, \qquad i < j \in \delta(v), \tag{20}$$

$$w_{vi} + w_{vj} - 1 \leq z_{vij}, \qquad i < j \in \delta(v), \tag{21}$$

$$(\mathbf{w}_v, \mathbf{z}_v) \in \mathbb{B}^{|\delta(v)|} \times \mathbb{R}_+^{(|\delta(v)|^2-|\delta(v)|)/2}. \tag{22}$$

Note that $q'_{ij} \geq 0$ for $i \neq j \in \delta(v)$ and that we are interested in stars with negative reduced cost. Given any integer feasible solution $(\overline{\mathbf{w}}_v, \overline{\mathbf{z}}_v)$ for (18)-(22), if for a $i \in \delta(v)$, $\overline{w}_{vi} = 1$ and $q'_{ii} \geq 0$ or $\sum_{j\in\delta(v):j\neq i}(q'_{ij} + q'_{ji})\overline{w}_{vj} \geq |q'_{ii}|$, then setting $\overline{w}_{vi} = 0$ cannot yield a solution with worse cost. This observation implies the optimality cut

$$\sum_{j\in\delta(v):j\neq i} (q'_{ij} + q'_{ji}) z_{vij} \leq |q'_{ii}| w_{vi}. \tag{23}$$

If, for a set $S \subseteq (\delta(v) \setminus \{i\})$, $\sum_{j\in S}(q'_{ij} + q'_{ji}) \geq |q_{ii}|$, we also have the cut

$$w_{vi} + \sum_{j\in S} w_{vj} \leq |S|. \tag{24}$$

Our cut-and-branch algorithm for the resolution of (18)-(22) is initialized with (23) and some cuts of type (24), obtained as follows. For each $i \in \delta(v)$ with $q'_{ii} < 0$ we sort the elements of $\delta(v) \setminus \{i\}$ as $\{e_1, ..., e_{|\delta(v)|-1}\}$ in a non-decreasing order of their costs $q'_{ij} + q'_{ji}$. Then, let $\overline{k}$ be the minimum k for which $\sum_{l=1}^{k}(q'_{ie_l} + q'_{e_l i}) \geq |q_{ii}|$. We add $w_{vi} + \sum_{l=1}^{\overline{k}} w_{vj} \leq \overline{k}$ and also

$$\sum_{j\in\delta(v)\setminus\{i\}} z_{vij} \leq (\overline{k} - 1) w_{vi} \tag{25}$$

to the formulation. It should be remarked that these cuts are not globally valid. However, at least one optimal solution for (18)-(22) is satisfied by them.

4 Computational experiments and future work

Instances considered in our computational study for AQMSTP were generated as follows. For each QMSTP instance in [4], one corresponding AQMSTP instance was generated by setting $q_{ij} = 0$, whenever edges i and j do not share one endpoint. Our computational experiments were conducted with a 3GHz Intel Core 2 Duo machine, with 5GB of RAM memory, under Linux operating system. The IP solver CPLEX 12 was used to solve the pricing problems. CPLEX LP solver was also used to solve each RLMP.

In Table 1, we compare three lower bounds for AQMSTP. The first two are Lagrangian relaxation bounds: $Lag_{(5)}$ (obtained by [1], by relaxing (5)) and $Lag_{(8)}$ (obtained in [4], by relaxing (8)). The last bound is LP(F_{star}), the LP relaxation bound implied by F_{star}. All lower bounds coming from the literature were evaluated by our own implementation of their procedures. In the first three columns of the table, we provide n, m, and ub, an upper bound on the optimum (obtained by running a simple greedy heuristic). For each lower bounding scheme, we provide the lower bound lb, the duality gap (computed with ub) and the time t, in seconds, needed to evaluate the bound. In the table we present results only for some selected instances; the ones with the smallest and the largest duality gaps, for each size n, m.

Compared to the bounds in the literature, all of them being of the Gilmore-Lawler type, LP(F_{star}) is remarkably stronger. The computational effort involved in the evaluation of LP(F_{star}), however, is much higher.

We plan to implement a BB algorithm based on formulation F_{star}, and compare it to existing algorithms. One line of research is to investigate ways of speeding up the resolution of the pricing subproblems, which have structure to be explored. That could be accomplished, for example, by reinforcing (18)-(22) with valid inequalities for the boolean quadric polytope.

References

[1] A. Assad and W. Xu. The quadratic minimum spanning tree problem. *Naval Research Logistics (NRL)*, 39(3):399–417, 1992.

[2] R. Cordone and G. Passeri. Heuristic and exact approaches to the quadratic minimum spanning tree problem. In *Seventh Cologne-Twente Workshop on Graphs and Combinatorial Optimization*, 2008.

[3] Y. Li, P.M. Pardalos, K.G. Ramakrishnan, and M.G.C. Resende. Lower bounds

Inst.			$Lag_{(5)}$			$Lag_{(8)}$			LP(F_{star})		
n	m	ub	lb	gap	t	lb	gap	t	lb	gap	t
10	45	205	136.26	33.52	0	122.55	40.21	0.53	205	0	1.22
10	45	257	145.14	43.52	0	135.56	47.24	0.52	242.8	5.52	1.38
20	190	304	101.51	66.6	0.01	103.16	66.06	8.26	302	0.65	83.17
20	190	386	128.9	66.6	0.01	138.7	64.06	8.36	357.1	7.48	82.3
30	435	416	137.84	66.86	0.06	139.88	66.37	46.84	402.92	3.14	2214.73
30	435	473	130.16	72.48	0.06	130.29	72.45	44.26	425.33	10.07	2024.5
10	45	17608	15502.1	11.95	0	15149	13.96	0.53	17608	0	1.66
10	45	17046	14984.27	12.09	0	14386.66	15.6	0.52	15921	6.59	1.32
20	190	22724	15539.27	31.61	0.01	15726	30.79	8.25	22678.5	0.2	192
20	190	21181	13270.65	37.34	0.02	12764.42	39.73	8.25	19688	7.04	209.48
30	435	28059	17963.14	35.98	0.07	18040.18	35.7	46.52	28020.75	0.13	22487.21
30	435	27553	14742.15	46.49	0.02	15027.62	45.45	45.77	25472.25	7.55	18272.15
10	45	538	305	43.3	0	322.29	40.09	0.53	486.89	9.49	2.16
10	45	512	293.88	42.59	0	270.5	47.16	0.51	368.2	28.08	1.84
20	190	17889	17799.27	0.5	0	17187	3.92	0.07	17889	0	0.11
20	190	17703	14884.6	15.92	0	13056.5	26.24	0.09	17468	1.32	0.16
30	435	945	490.93	48.04	0.07	506.09	46.44	60.31	783.61	17.07	2522.87
30	435	1071	507.03	52.65	0.06	497.17	53.57	44.52	776.12	27.53	2512.09

Table 1
Comparison of lower bounds

for the quadratic assignment problem. *Annals of Operations Research*, 50:387–410, 1994.

[4] T. Öncan and A. P. Punnen. The quadratic minimum spanning tree problem: A lower bounding procedure and an efficient search algorithm. *Computers and Operations Research*, 37(10):1762–1773, 2010.

[5] M. W. Padberg and L. A. Wolsey. Trees and cuts. In *Combinatorial Mathematics Proceedings of the International Colloquium on Graph Theory and Combinatorics*, volume 75, pages 511–517. 1983.