On the NP-Completeness of the Multi-Period Minimum Spanning Tree Problem

Rosklin Juliano Chagas   Alexandre Salles da Cunha
Departamento de Ciência da Computação
Universidade Federal de Minas Gerais, Brasil
Email: `rosklinjuliano@gmail.com`   `acunha@dcc.ufmg.br`

March 2016

Abstract

In this note, we consider the Multi-period Minimum Spanning Tree Problem (MMST), a variant of the well known Minimum Spanning Tree Problem (MST), that consists in the following. Given a connected and undirected graph G and a finite discrete time horizon, one has to schedule the moment in time edges are added to a solution. For each time period, a partial solution consisting of a tree must be built. Once an edge is added to a tree at a given time, it remains in the solution thereafter. At the end of the planning time, the complete solution must be a spanning tree of G. Vertices must be spanned by these trees at times that do not exceed pre-defined dates that depend on each vertex. Edges' costs represent the discounted cash flows of the installation cost at the time of installation plus maintenance costs, from that time until the end of the planning time. The goal is to choose and schedule the installation of the edges, such that the final spanning tree has a minimum cost. The complexity of MMST was not addressed before. We show that, unlike the MST case, the decision version of MMST is $\mathcal{NP}-$Complete.

Keywords: Multi-period Minimum Spanning Tree Problem, NP-Completeness, Combinatorial Optimization

1 Introduction

The Minimum Spanning Tree Problem (MST) [3, 11, 14] is one of the most celebrated problems in combinatorial optimization [12]. MST finds applications in many fields like communications and logistics networks, bioinformatics and production planning just to name a few. Alongside with the matching problem [4], MST is, very likely, the first example in virtually every optimization course, of combinatorial optimization problems that gather desirable properties like the efficient optimization, the efficient convex-hull separation and the existence of integral linear programming formulations.

MST does not have a temporal dimension. In this note, we consider the Multi-period MST (MMST) [9], a variant of MST where, given a connected and undirected graph G and a finite discrete time horizon, one has to schedule the moment in time edges are added to a solution. For each time period, a partial solution consisting of a tree must be built. Once an edge is added to a tree at a given time, it remains in the solution thereafter. At the end of the planning time, the complete solution must be a spanning tree of G. Vertices must be spanned by these trees at times that do not exceed pre-defined dates that depend on each vertex. Edges' costs represent the discounted cash flows of the installation cost at the time of installation plus maintenance costs, from that time until the end of the planning time. The goal is to choose and schedule the installation of the edges, such that the final spanning tree has a minimum cost.

The only study dedicated to MMST that we are aware of is due to Kawatra [9], where an integer programming formulation and a Lagrangian Relaxation algorithm were proposed. The Multi-period Degree Constrained Spanning Tree Problem (MDCMST) and the Multi-period Capacitated Minimum Spanning Tree Problem (MCMST), multi-period analogs to the Degree Constrained Minimum Spanning Tree Problem [2, 13] and the Capacitated Minimum Spanning Tree Problem [5, 15], respectively, received more attention in the literature [1, 6, 8, 10].

Let us formally define MMST. Given a connected and undirected graph $G = (V, E)$, with set of vertices V and edges E, one particular vertex r of V is called the root of the tree. It plays the role of a service provider for the other vertices in V. The planning time is divided into T time periods $\mathcal{T} = \{1, 2, \ldots, T\}$ of equal duration. Each vertex $i \in V$ is assigned to an integer $t_i \in \mathcal{T}$; it is assumed that $t_r = 1$. For each $t \in \mathcal{T}$, $V^t = \{i \in V : t_i \leq t\}$ denotes the subset of vertices that must be connected to r at a time period not later than t. For any subset of edges $M \subseteq E$, denote by $V(M)$ the subset of vertices of V that are incident to any edge in M. Feasible solutions for MMST are named multi-period spanning trees (MPSTs) and correspond to T sets of trees $\{(V(E^t), E^t) : t = 1, \ldots, T\}$ of G, satisfying the following properties:

1. $E^1 \subseteq E^2 \subseteq \cdots \subseteq E^T$: once an edge is added to the tree at a given time period t, it must remain at the solution thereafter.

2. Each subgraph $(V(E^t), E^t)) : t \in \mathcal{T}$ is a tree spanning vertices V^t (and possibly some vertices in $V \setminus V^t$). Therefore, every vertex in V^t must be connected to r by a path in subgraph $(V(E^t), E^t)$, at time $t \in \mathcal{T}$. It follows that $(V(E^T), E^T)$ is a spanning tree of G.

Whenever an edge $\{i, j\}$ is added to a MPST at time period $t \in \mathcal{T}$, a cost $c_{ij}^t \in \mathbb{Z}$ is incurred. Define $E^0 = \emptyset$. The cost of a MPST $\{(V(E^t), E^t) : t \in \mathcal{T}\}$ is $\sum_{t=1}^{T} \sum_{\{i,j\} \in E^t \setminus E^{t-1}} c_{ij}^t$. Among all MPSTs, MMST looks for one with minimum cost.

Several reasons justify our interest on studying the complexity class of MMST. Although some discussions on why the greedy algorithm of Kruskal [11] does not work for the multi-period case can be found in some studies [10, 9], the question about the existence of a polynomial time algorithm, eventually of a different nature, was never formally addressed before. Unlike the case of the decision version of MDCMST and MCMST, whose complexity status follows from the $\mathcal{NP}-$completeness of their single period versions (DCMST and CMST), similar conclusions cannot be drawn for MMST, since MST, on the contrary, is polynomially solvable. On the practical side, a polynomial time algorithm for MMST, provided it existed, could be a better alternative to the approximate MMST solution approach introduced in [9]. Since MDCMST and MCMST can be seen as a MMST with side constraints, polynomial time algorithms for the latter would imply relaxations for the former two problems, over which one could efficiently optimize. Such an algorithm could be used for providing Lagrangian Relaxation lower bounds for MDCMST and MCMST, after relaxing and dualizing the side constraints. From a polyhedral point of view, the MMST $\mathcal{NP}$-Completeness proof provided here leaves almost no hope of having a complete compact characterization of the convex hull of its integer feasible solutions.

In Section 2, we provide the main result in the paper, a simple yet elegant polynomial time reduction of the decision version of Steiner Tree Problem [7] to the decision version of MMST, D-MMST. In Section 3, we conclude the paper with directions for future research.

2 A reduction from the Steiner Tree Problem

Consider D-MMST and D-STP, respectively the decision versions of MMST and of the Steiner Tree Problem [7]:

D-MMST:
Instance: a connected and undirected graph $G = (V, E)$, a set of time periods $\mathcal{T} = \{1, \dots, T\}$, integers $\{t_i \in \mathcal{T} : i \in V\}$, a root vertex $r \in V : t_r = 1$, costs $\{c^t_{ij} \in \mathbb{Z} : \{i, j\} \in E, t \in \mathcal{T}\}$, an integer K.
Question: Does a MPST with cost at most K for such an instance exist ? D-STP :
Instance: an undirected graph $G = (V, E)$, costs $\{w_{ij} \in \mathbb{Z} : \{i, j\} \in E\}$, set of terminal vertices $R \subseteq V$ and integer K.
Question: Does a tree of G that spans R, with cost at most K exist ?

Theorem 1 *D-MMST is $\mathcal{NP}$-Complete.*

Proof. 2 *Take a D-MMST instance whose answer is yes and a MPST $\{(V(E^t), E^t) : t \in \mathcal{T}\}$ with cost at most K, for that instance. Checking the cost of such a MPST and whether each subgraph $\{(V(E^t), E^t) : t \in \mathcal{T}\}$ satisfies the feasibility conditions (1)-(2) defined in the previous section can be carried out in $O(T|E|)$ time. D-MMST thus belongs to $\mathcal{NP}$.*

We assume that G, the graph that defines the instance for D-STP, has one connected component that spans R, so the set of Steiner Trees of G is non-empty. If G itself is not connected, pick any vertex $i \in R$ and add to G edges connecting i to one vertex in every connected component of G, other than that spanning R. Denote by E' the set of added edges. For every edge $\{i, j\} \in E'$, set $w_{ij} = K + 1$. Redefine $E \leftarrow E \cup E'$. Note that now (V, E) is a connected graph. Note that these operations can be carried out in polynomial time.

Given a D-STP instance defined by $(V, E), R, K$ and $\{w_{ij} \in \mathbb{Z} : \{i, j\} \in E\}$, an instance for D-MMST is defined by the same graph (V, E), $\mathcal{T} = \{1, 2\}$ and costs $\{c^1_{ij} = w_{ij} : \{i, j\} \in E\}$, $\{c^2_{ij} = 0 : \{i, j\} \in E\}$. Define $t_i = 1 : i \in R$, $t_i = 2 : i \in V \setminus R$ and pick $i \in V : t_i = 1$ to be the root node r.

Let $\{(V(E^1), E^1), (V(E^2), E^2)\}$ be a MPST for the D-MMST instance, whose cost is at most K. Note that $E^1 \cap E' = \emptyset$, otherwise the cost of this MPST would be at least $K + 1$. Define a Steiner Tree $(\hat{V}, \hat{E})$ as: $\hat{E} = E^1$ and $\hat{V} = V(E^1)$. Since E^1 spans V^1, $(V(E^1), E^1)$ is a tree and by construction $V^1 = R$, it follows that $(\hat{V}, \hat{E})$ is a Steiner Tree of G. Its cost is $\sum_{\{i,j\} \in \hat{E}} w_{ij} = \sum_{\{i,j\} \in E^1} c^1_{ij} = \sum_{t=1}^{2} \sum_{\{i,j\} \in E^t \setminus E^{t-1}} c^t_{ij} \leq K$ (recall that we defined $E^0 = \emptyset$).

To show the converse, let $(\hat{V}, \hat{E})$ be a Steiner tree with cost at most K. Note that $E' \cap \hat{E} = \emptyset$, otherwise $\sum_{\{i,j\} \in \hat{E}} w_{ij} \geq K + 1$ would apply. We construct a MPST $\{(V(E^1), E^1), (V(E^2), E^2)\}$ from $(\hat{V}, \hat{E})$ as follows. Set $E^1 = \hat{E}$. In doing so, all vertices in V^1 are connected by a tree at time 1. If $V = V(E^1)$, set $E^2 = E^1$. Otherwise, define E^2 as the union of E^1 and any subset of edges in $E \setminus E^1$ such that $(V(E^2), E^2)$ is a spanning tree of G. E^2 must exist, otherwise (V, E) would not be connected. Note that conditions (1)-(2) are clearly satisfied by $\{(V(E^1), E^1), (V(E^2), E^2)\}$ and its cost is $\sum_{t=1}^{2} \sum_{\{i,j\} \in E^t \setminus E^{t-1}} c_{ij}^t = \sum_{\{i,j\} \in E^1} c_{ij}^1 = \sum_{\{i,j\} \in \hat{E}} w_{ij} \leq K$.

The constructions above can be implemented in $O(E)$ time. Thus, if we have a yes certificate for one problem in polynomial time, so have we for the other, and the proof is complete. □

3 Future research

In the light of the $\mathcal{NP}-$Completeness of D-MMST, MMST does not provide relaxations for MDCMST and MCMST over which one should expect to optimize in polynomial time. It is equally unlikely that one may end up with a compact linear integer programming formulation for the convex hull of feasible solutions for MMST. Nevertheless, the polyhedral structure of MMST may provide valid inequalities that can be used in exact solution algorithms not only for MMST, but also to MDCMST and MCMST. In particular, valid inequalities for MMST could be used to strengthen the MDCMST and MCMST formulations in [1, 8, 10]. We plan to investigate good characterizations of the convex hull of feasible solutions for these problems, starting with valid inequalities for MMST.

References

[1] A. S. da Cunha. Algorithms for the multi-period degree constrained minimum spanning tree problem. *Proceedings of the 2nd International Symposium on Combinatorial Optimization*, 1:131–134, 2012.

[2] A. S. da Cunha and A. Lucena. Lower and upper bounds for the degree-constrained minimum spanning tree problem. *Networks*, pages 55–66, 2007.

[3] J. Edmonds. Matroids and the greedy algorithm. *Mathematical Programming*, 1:127–136, 1971.

[4] J. Edmonds and E. Johnson. Matching: A well-solved class of integer linear programs. *Proc Calgary Int Conf Combinatorial Struct their Appl, R.K. Guy et al. (Editor), Gordon and Breach, New York*, pages 89–92, 1970.

[5] B. Gavish. Formulations and algorithms for the capacitated minimal directed tree problem. *Journal of the Association for Computing Machinery*, 30:118–132, 1983.

[6] A. Junaidi, Wamiliana, D. Sakethi, and E. Baskoro. Computational aspects of greedy algorithm for solving the multiperiod degree constrained minimum spanning tree problem. *J. Sains MIPA, Edisi Khusus Tahun*, 14(1):1–6, 2008.

[7] R. Karp. Reducibility among combinatorial problems. *In R. E. Miller and J . W. Thatcher (eds.) Complexity of Computer Computations*, New York:85–103, 1972.

[8] R. Kawatra. A multiperiod degree constrained minimal spanning tree problem. *European Journal of Operational Research*, 143:53–63, 2002.

[9] R. Kawatra. A multiperiod minimal spanning tree problem. *The Sixth IASTED International Multi-Conference on Wireless and Optical Communications*, 2006.

[10] R. Kawatra and D. Bricker. A multiperiod planning model for the capacitated minimal spanning tree problem. *European Journal of Operational Research*, 121:412–419, 2000.

[11] J. Kruskal. On the shortest spanning tree of a graph and the travelling salesman problem. *Proceedings of the American Mathematical Society*, 7:48–50, 1956.

[12] M. Mares. The saga of minimum spanning trees. *Computer Science Review*, 2(3):165–221, 2008.

[13] S. Narula and C. Ho. Degree-constrained minimum spanning tree. *Computers & Operations Research*, 7(4):239–249, 1980.

[14] R. Prim. Shortest connection networks and some generalizations. *Bell System Technological Journal*, 36:1389–1401, 1957.

[15] E. Uchoa, R. Fukasawa, J. Lysgaard, A. Pessoa, M. Aragão, and D. Andrade. Robust branch-cut-and-price for the capacitated minimum spanning tree problem over a large extended formulation. *Math. Programming*, 112:443–472, 2008.