Discrete Applied Mathematics 164 (2014) 210–224

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

The min-degree constrained minimum spanning tree problem: Formulations and Branch-and-cut algorithm

Leonardo Conegundes Martinez, Alexandre Salles da Cunha *

Departamento de Ciência da Computação, Universidade Federal de Minas Gerais, Belo Horizonte, Minas Gerais, Brazil

ARTICLE INFO

Article history:
Received 30 June 2010
Received in revised form 27 July 2011
Accepted 10 August 2011
Available online 13 September 2011

Keywords:
Combinatorial optimization
Min-degree constrained minimum spanning tree problem
Branch-and-cut
Integer programming formulations

ABSTRACT

Given an edge weighted undirected graph G and a positive integer d, the Min-Degree Constrained Minimum Spanning Tree Problem (MDMST) consists of finding a minimum cost spanning tree of G, such that each vertex is either a leaf or else has degree at least d in the tree. In this paper, we discuss two formulations for MDMST based on exponentially many undirected and directed subtour breaking constraints and compare the strength of their Linear Programming (LP) bounds with other bounds in the literature. Aiming to overcome the fact that the strongest of the two models, the directed one, is not symmetric with respect to the LP bounds, we also presented a symmetric compact reformulation, devised with the application of an Intersection Reformulation Technique to the directed model. The reformulation proved to be much stronger than the previous models, but evaluating its bounds is very time consuming. Thus, better computational results were obtained by a Branch-and-cut algorithm based on the original directed formulation. With the proposed method, several new optimality certificates and new best upper bounds for MDMST were provided.

1. Introduction

Let $G = (V, E)$ be an undirected graph with set of vertices $V = \{1, \ldots, n\}$ and set of edges E ($m = |E|$). Assume that costs $\{c_{ij} \in \mathbb{R}_+ : \{i, j\} \in E\}$ are associated to the edges of E. Given a positive integer d, the Min-Degree Constrained Minimum Spanning Tree Problem (MDMST) consists of finding a minimum cost spanning tree of G, such that each vertex is either a leaf or else has degree at least d in the tree.

MDMST is closely related to the widely known Degree-Constrained Minimum Spanning Tree Problem (DCMST) [7], where an upper bound on the number of edges incident to each vertex is imposed. Contrarily to DCMST which has a long history [7,4–6,11,24,27–29], MDMST was introduced recently by Almeida et al. [2], when the problem was proven to be NP-hard for $4 \le d \le \lfloor \frac{n}{2} \rfloor$. When $d = 2$, MDMST reduces to the Minimum Spanning Tree Problem. In a recent study, MDMST was proven to be NP-hard also when $d = 3$ [3].

According to [21], applications of MDMST are associated with the optimal design of communication networks where the traffic between every pair of nodes is usually very small. Therefore, expensive centralized information devices to be installed in central nodes only pay off if the number of nodes connected to them is greater than a minimum value.

Almeida et al. [2] not only discussed the problem's complexity class but also introduced Integer Programming Formulations based on single and multi-commodity flows and tested Branch and bound (BB) algorithms associated to them. The computational results in [2] suggested that flow formulations for DCMST seem to provide Linear Programming bounds tighter than those given for MDMST by similar formulations. One possible explanation is that the min-degree requirements are modeled through (weak) disjunctive constraints.

* Corresponding author. Fax: +55 3134995858.
E-mail addresses: leocm@dcc.ufmg.br (L.C. Martinez), acunha@dcc.ufmg.br (A.S. da Cunha).

0166-218X/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2011.08.008

Akgún and Tansel [1]presented a new formulation for MDMST based on the Miller et al. [22], Desrochers and Laporte [9]inequalities and on stronger degree enforcing constraints. Compared to the flow based BB in [2], the MTZ-based BB in [1] was capable of solving similar sets of test instances with less computational effort.

In another recent study, Martins and de Souza [21]investigated different Variable Neighborhood Search [23] (VNS) implementations. Their computational results indicated that randomized VNS methods enclosing an enhanced second order repetitive technique produced most of the best upper bounds for MDMST in that study.

In this work, we improve on the theoretical and computational results presented in a conference version of this paper [20], in several ways. We discuss two Integer Programming formulations for MDMST based on exponentially many (directed and undirected) subtour breaking constraints. The formulations are compared with others in the literature, in terms of the strength of their Linear Programming relaxations, both theoretically and computationally. Aiming to overcome the fact that the strongest of the two models, the directed one, is not symmetric with respect to the LP bounds, we also presented a symmetric compact reformulation, devised with the application of an Intersection Reformulation Technique [12] to the directed model. The reformulation proved to be much stronger than the previous models, but evaluating its bounds is very time consuming. Thus, better computational results were obtained by a Branch-and-cut algorithm [25] based on the original directed formulation. With the proposed method, 24 new optimality certificates and 49 new best upper bounds for MDMST were provided.

The remainder of the paper is organized as follows. In Section 2, we present the formulations, the reformulation by intersection and we address the quality of their lower bounds from a theoretical perspective. In Section 3, we discuss how the Linear Programming bounds of the proposed models were evaluated and compare them, from a computational standpoint, with other bounds in the literature. A Branch-and-cut implementation and extensive computational experiments are discussed in Section 4. We end the paper in Section 5, offering some conclusions and directions for future investigations.

2. Integer programming formulations

Generally, it is possible to formulate Integer Programming problems using different choices of variables and constraints [19]. For the MDMST case, such choices must guarantee that the feasible solutions are cycle-free and connected (imposing *the spanning tree enforcing constraints*) and that the minimal degree requirements are satisfied (imposing *the degree enforcing constraints*).

To define the spanning tree representation of the MDMST solutions by Integer Programs, many different approaches are available, including formulations based on network flow arguments [2], on Miller–Tucker–Zemlin inequalities [1] and on undirected (packing) and directed subtour breaking constraints [20].

One advantage of formulations based on network flows and on Miller–Tucker–Zemlin constraints is their compactness. However, the huge sizes of multi-commodity flow formulations (involving $O(n^2 + nm)$ constraints and $O(nm)$ variables) usually preclude their direct use to provide Linear Programming (LP) bounds in Branch and bound algorithms, unless a decomposition approach (for example, a Lagrangian Relaxation method [10,14]) is applied. On the other hand, formulations based on single commodity flows and on Miller–Tucker–Zemlin subtour breaking constraints usually provide weaker LP lower bounds. Next, we investigate formulations for MDMST based on exponentially many subtour breaking constraints and compare their strength to other formulations for MDMST.

2.1. Formulations for MDMST involving exponentially many subtour breaking constraints

A canonical way to formulate MDMST as an Integer Program is to introduce binary variables $\{z_{ij} \in \mathbb{B} : \{i, j\} \in E\}$ to select tree edges ($z_{ij} = 1$, if edge $\{i, j\} \in E$ is chosen, $z_{ij} = 0$ otherwise), and binary variables $\{y_i \in \mathbb{B} : i \in V\}$ to choose its leaf implying vertices ($y_i = 1$ if i is a leaf node, $y_i = 0$ otherwise).

Let us assume that, given $S \subseteq V$, $E(S) := \{\{i, j\} \in E : i, j \in S\}$ denotes the set of edges of E that have both endpoints in S and that $\delta(S) = \{\{i, j\} \in E : i \in S, j \notin S\}$. Accordingly, given $i \in V$, $\delta(\{i\})$ defines the subset of edges of E that are incident to i. For any $M \subseteq E$, define $z(M) := \sum_{\{i,j\}\in M} z_{ij}$.

An Integer Programming formulation for MDMST is given by:

$$w = \min \left\{ \sum_{\{i,j\}\in E} c_{ij}z_{ij} : (z, y) \in P_u \cap (\mathbb{B}^m, \mathbb{B}^n) \right\}, \tag{1}$$

where polyhedron $P_u \subset \mathbb{R}_+^{n+m}$ is defined by:

$$z(E) = n - 1, \tag{2}$$

$$z(E(S)) \le |S| - 1, \quad S \subset V,\ S \ne \emptyset, \tag{3}$$

$$z_{ij} \ge 0, \quad \forall \{i, j\} \in E, \tag{4}$$

$$z(\delta(\{i\})) \ge 1 + (d - 1)(1 - y_i), \quad \forall i \in V, \tag{5}$$

$$z(\delta(\{i\})) \le 1 + (n - 2)(1 - y_i), \quad \forall i \in V, \tag{6}$$

$$0 \le y_i \le 1, \quad \forall i \in V. \tag{7}$$