T U M

I N S T I T U T F Ü R I N F O R M A T I K

The Complexity of Computing Graph-Approximating Spanning Trees

Matthias Baumgart and Hanjo Täubig

TUM-I0822
Juli 08

T E C H N I S C H E U N I V E R S I T Ä T M Ü N C H E N

TUM-INFO-07-I0822-0/1.-FI

Druck: Institut für Informatik der
Technischen Universität München

The Complexity of Graph-Approximating Spanning Trees

Matthias Baumgart and Hanjo Täubig

Institut für Informatik, Technische Universität München,
Boltzmannstraße 3, D-85748 Garching, Germany
`baumgart@in.tum.de`

Abstract

This paper deals with the problem of computing a spanning tree of a connected undirected graph $G = (V, E)$ minimizing the sum of distance differences of all vertex pairs $u, v \in V$ which are connected by an edge $\{u, v\} \in E$. We show that the decision variant of this optimization problem is NP-complete with respect to the L_p norm for arbitrary $p \in \mathbb{N}$. For the reduction, we use the well known NP-complete problem Vertex Cover.

1 Introduction

1.1 Motivation

Recently, there has been some interest in problems related to simplification of graphs (with respect to the number of edges, i.e. network sparsification), which intended to thin out the graph while retaining certain network characteristics (such as the distances between node pairs or the centrality measures of the nodes). The aim was to reduce the complexity of a given graph in order to simplify computations of network problems or to feature a concise visualization of a complex network with its most important structural properties (which makes the network more amenable to visual examination).

If this simplification is carried to an extreme, we would require the resulting graph to be a spanning tree, since the elements of this graph class have a minimum number of edges among all connected subgraphs and they offer a variety of beneficial properties which can be exploited for fast network algorithms. A couple of concrete applications, e.g. in systems biology, can be found in [4].

In this paper, we study the problem of computing a spanning tree of a graph, that minimizes, in its simplest form, the sum of the distances between all pairs of nodes, that were connected by an edge in the original graph. Actually, we consider a more general form, where the sum is computed of p-th powers of the respective distances (or distance differences), i.e., the calculation is made with respect to the L_p-norm.

1.2 Related Work

The problem is related to a couple of other problems, the most similar of which is the problem of computing distance-minimizing or distance-approximating spanning trees (DMST, DAST, [4]). In contrast to the setting in this paper, the DMST and DAST problems consider the distances of *all* vertex pairs (instead of only pairs connected by single edges in the original graph). Both problems (DMST and DAST) were shown to be NP-complete for all norms L_p, $p \in \mathbb{N}$. For both problems a fixed-edges variant was introduced in [4], where the input includes a set of fixed edges that have to appear in any admissible solution. For this fixed-edges version of DAST and arbitrary L_p-norms, there is no constant-factor approximation unless P = NP.

The simplest case of the DMST problem, i.e., DMST using the L_1-norm, is equal to the Simple Network Design Problem introduced in [8] as well as the problem of computing a Minimum Average Distance (MAD) Tree [2]. Moreover, this problem is equivalent to the DAST problem with respect to the L_1-norm.

In the more general form of the Network Design Problem, we are given a weighted undirected graph and want to compute a connected subgraph, that respects a certain budget constraint (regarding the sum of the edge weights) and minimizes the sum of all shortest path lengths. Of course, this problem was also shown to be NP-complete [8].

A similar relationship exists between GAST and the problem of computing a minimum fundamental cycle basis. Again, we are given a weighted undirected graph. The aim is to compute a spanning tree (or the respective cycle basis), that causes a minimum sum of the weights of all fundamental cycles (induced by the edges of the spanning tree). Deo et al. [3] have shown NP-completeness of this problem. Galbiati and Amaldi [5] proposed an $2^{\mathcal{O}(\sqrt{\log n \log\log n})}$-approximation algorithm for arbitrary graphs. Their approach used a related problem introduced by Hu, namely the Minimum Communication Cost Spanning Tree Problem [7], which was shown to be approximable within the same factor by Peleg and Reshef [10].

While all these problems are NP-complete, there is also an example for polynomial computability of distance-sum-related spanning trees, namely the problem of computing a Minimum Diameter Spanning Tree [1, 9, 6].

2 Preliminaries

Throughout this paper, we assume that graphs $G = (V, E)$ are always simple, undirected, connected, and unweighted. The adjacency matrix of G is denoted by A_G. For two vertices $u, v \in V$, the distance between u and v in G is defined as the length of a shortest path between u and v in G. This length is denoted by $d_G(u, v)$. We define the distance matrix D_G by $D_G[i, j] = d_G(v_i, v_j)$. Obviously, D_G is a symmetric matrix with non-negative entries. Furthermore, for any given spanning tree T of G, it holds $D_T[i, j] \geq D_G[i, j]$ for all $v_i, v_j \in V$.

Let $A \in \mathbb{N}^{n\times m}$ and $B \in \mathbb{N}^{n\times m}$ be two $n \times m$-matrices. We denote by $C = A \circ B$ the matrix we obtain by performing a multiplication element by element, i.e. $C[i, j] = A[i, j] \cdot B[i, j]$ for all pairs (i, j). This entrywise product of two matrices of equal dimension is also known as the Hadamard or Schur product.

For evaluating a matrix $A \in \mathbb{N}^{n\times n}$, we use the L_p-norm which is defined as

$$\|A\|_{L_p} = \left(\sum_{i=1}^{n} \sum_{j=1}^{n} A[i,j]^p \right)^{1/p}$$

for all $1 \le p < \infty$.

3 The 2-Hitting-Set Gadget

For the reduction, we use the Vertex Cover problem, which is well known to be NP-complete. To avoid confusion between 'vertices' and 'edges' of the instance of Vertex Cover and of the constructed graph, we use the less common terminology of the equivalent 2-Hitting Set (2HS) problem, i.e. 'literals' (vertices) and 'clauses' (edges).

Problem: 2-Hitting Set (2HS).
Input: A triple $(\mathcal{C}, \mathcal{S}, k)$ consisting of a family $\mathcal{C} = \{C_1, \dots, C_m\}$ of 2-element subsets of a set $\mathcal{S} = \{s_1, \dots, s_n\}$ and a number $k \in \{1, \dots, n\}$.
Question: Is there a subset $\mathcal{S}' \subseteq \mathcal{S}$ such that $|\mathcal{S}'| \le k$ and the set $C_\mu \cap \mathcal{S}'$ is not empty for each $\mu \in \{1, \dots, m\}$?

A subset $\mathcal{S}' \subseteq \mathcal{S}$ having the required properties is called an *admissible solution* to a 2HS instance $(\mathcal{C}, \mathcal{S}, k)$. For a given 2HS instance, we define the graph $G(\mathcal{C}, \mathcal{S})$ (similar to [4]) as follows:

- For each $s_\mu \in \mathcal{S}$, $\mu \in \{1, \dots, n\}$, we define a *literal gadget* G_μ consisting of two *connection vertices* v_μ and v'_μ. Both vertices are connected by the so-called *elongation path* $(v_\mu, e_1^\mu, \dots, e_{m+1}^\mu, v'_\mu)$ of length $m+2$ and the so-called *literal path* $(v_\mu, l_1^\mu, \dots, l_m^\mu, v'_\mu)$ of length $m+1$.
- For each $\mu \in \{1, \dots, n-1\}$, we connect the literal gadgets G_μ and $G_{\mu+1}$ by adding an edge $\{v'_\mu, v_{\mu+1}\}$.
- Additionally, we introduce a vertex v'_0 which is connected to the first literal gadget G_1 by the edge $\{v'_0, v_1\}$.
- For each $\mathcal{C}_\mu = \{s_\nu, s_\kappa\}$, we define a *clause path* of length $2n(m+2)$ that connects the vertices l_μ^ν and l_μ^κ and a *safety path* of length $2n(m+2)$ that connects the vertices v'_0 and l_μ^ν whereas we assume w.l.o.g. that $\nu < \kappa$.

In Figure 1, there is an illustration of the graph representation $G(\mathcal{C}, \mathcal{S})$ for the 2HS instance $(\mathcal{C}, \mathcal{S}, k)$ with $\mathcal{S} = \{s_1, s_2, s_3, s_4\}$ and $\mathcal{C} = \{\{s_1, s_3\}, \{s_2, s_4\}, \{s_1, s_4\}, \{s_3, s_4\}\}$.

Lemma 1. *Let $(\mathcal{C}, \mathcal{S}, k)$ be an instance of* 2HS. *Then we have $d_{G(\mathcal{C},\mathcal{S})}(v'_0, v'_n) = n(m+2)$. Moreover, there exists an admissible solution $\mathcal{S}' \subseteq \mathcal{S}$ of size $|\mathcal{S}'| \le k$ if and only if there exists a spanning tree T of $G(\mathcal{C}, \mathcal{S})$ containing all edges in the clause paths such that $d_T(v'_0, v'_n) \le d_{G(\mathcal{C},\mathcal{S})}(v'_0, v'_n) + k$.*

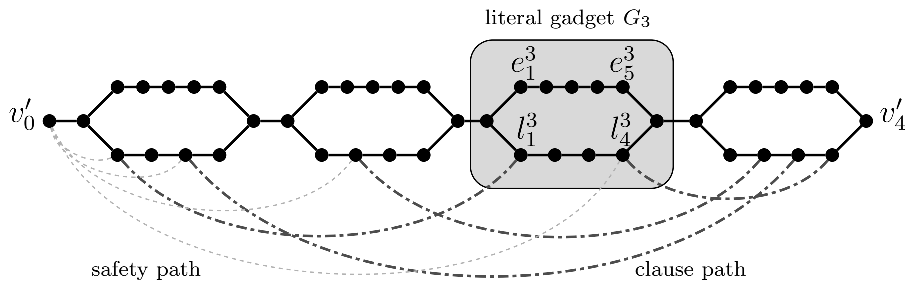

Figure 1: Graph representation $G(\mathcal{C}, \mathcal{S})$ of a 2HS instance.

Proof. First, we observe that any path from v_0' to v_n' using a clause or safety path has length at least $2n(m+2)$ whereas the shortest path between v_0' and v_n' via literal paths has length $n(m+2)$. Thus, it holds that $d_{G(\mathcal{C},\mathcal{S})}(v_0', v_n') = n(m+2)$. For proving the second statement, we consider both directions separately.

($\Longrightarrow$) Let $\mathcal{S}'$ be an admissible solution to the 2HS instance $(\mathcal{C}, \mathcal{S}, k)$. We construct a spanning tree of the graph representation $G(\mathcal{C}, \mathcal{S})$ as follows:

1. For each $s_\mu \in \mathcal{S}'$, we remove the edge $\{l_m^\mu, v_\mu'\}$ which is the last edge on the literal path of the literal gadget G_μ.

2. For each $s_\mu \notin \mathcal{S}'$, we remove the edge $\{v_\mu, e_1^\mu\}$ which is the first edge on the elongation path of the literal gadget G_μ.

3. For each $C_\mu = \{s_\nu, s_\kappa\} \in \mathcal{C}$ do the following: if $s_\nu \in \mathcal{S}'$ then remove the edge $\{l_{\mu-1}^\nu, l_\mu^\nu\}$. If $s_\kappa \in \mathcal{S}'$ then remove the edge $\{l_{\mu-1}^\kappa, l_\mu^\kappa\}$. Here, we denoted $v_0^\nu = v_\nu$ and $v_0^\kappa = v_\kappa$. If not both s_ν and s_κ are elements of $\mathcal{S}'$ then remove an arbitrary edge from the safety path between v_0' and v_μ^ν.

Note that no edge from a clause path was removed during this construction. Now, we have to prove that each cycle in $G(\mathcal{C}, \mathcal{S})$ is broken when applying the three construction rules. By the first and second construction rule, at least one edge of each cycle induced by the literal and elongation paths is removed. The cycles induced by the clause and safety paths are broken by the first and third construction rule: For each clause $C_\mu = \{s_\nu, s_\kappa\} \in \mathcal{C}$, at least one of the sets $\{\{l_{\mu-1}^\nu, l_\mu^\nu\}, \{l_m^\nu, v_\nu'\}\}$ and $\{\{l_{\mu-1}^\kappa, l_\mu^\kappa\}, \{l_m^\kappa, v_\kappa'\}\}$ is removed because $\mathcal{S}'$ is an admissible solution. Thus, either l_μ^ν or l_μ^κ is not reachable via the clause path from v_ν or v_ν' (v_κ or v_κ', respectively). An edge from the safety path is removed, except if both $\{\{l_{\mu-1}^\nu, l_\mu^\nu\}, \{l_m^\nu, v_\nu'\}\}$ and $\{\{l_{\mu-1}^\kappa, l_\mu^\kappa\}, \{l_m^\kappa, v_\kappa'\}\}$ are removed, in which case neither l_μ^ν nor l_μ^κ is reachable via the clause path from any vertex $v_\nu, v_\nu', v_\kappa, v_\kappa'$.

A cycle induced by multiple clause paths not going through any connection vertices cannot occur since the connection is broken at one of the literals in $\mathcal{S}'$. As a result, the path between v_0' and v_n' in T is leading through elongation ($s_\mu \in \mathcal{S}'$) or literal ($s_\mu \notin \mathcal{S}'$) paths only, and does not contain any safety or clause path.

By construction of the graph representation $G(\mathcal{C}, \mathcal{S})$, the distance of v_μ and v_μ' via a literal path is shorter by 1 compared to the distance via an elongation path. Thus, it holds that

$$d_T(v_0', v_n') = (n - |\mathcal{S}'|)(m+2) + |\mathcal{S}'|(m+3) = n(m+2) + |\mathcal{S}'| \leq d_{G(\mathcal{C},\mathcal{S})}(v_0', v_n') + k \ .$$

($\Longleftarrow$) Let T be a spanning tree of $G(\mathcal{C}, \mathcal{S})$ containing all clause path edges and satisfying $d_T(v_0', v_n') \leq d_{G(\mathcal{C},\mathcal{S})}(v_0', v_n') + k$. Since $d_{G(\mathcal{C},\mathcal{S})}(v_0', v_n') + k \leq n(m+3) < 2n(m+2)$, the path between v'_0 and v'_n in T cannot lead through clause or safety paths. Hence, it only goes through literal and/or elongation paths.

By construction, the length of any (intact) elongation path is $m+2$, the length of any (intact) literal path is $m+1$. Therefore, the path from v'_0 to v'_n contains exactly k elongation paths. Let $\mathcal{S}'$ be the set of literals s_μ for which this path leads from v_μ to v'_μ via an elongation path. Here, the literal path must be broken since otherwise T is not a spanning tree. Conversely, for every $s_\mu \notin \mathcal{S}'$, the literal path is not broken, i.e., $(v_\mu, l_1^\mu, \ldots, l_m^\mu, v'_\mu)$ is a path in T. Assume, there is a clause $C_\mu = \{s_\nu, s_\kappa\} \in \mathcal{C}$ (where $\nu < \kappa$) such that $C_\mu \cap \mathcal{S}' = \emptyset$. The clause path corresponding to the clause C_μ connects l_μ^ν with l_μ^κ. Since $s_\nu, s_\kappa \notin \mathcal{S}'$, the vertex l_μ^ν is connected to v'_ν which is connected to v^κ which is connected to l_μ^κ. This is a contradiction to T being a spanning tree. □

4 Graph-Approximating Spanning Trees

In this section, we consider the problem of computing a spanning tree of a graph G that minimizes the distances of pairs of vertices which are connected in the original graph. We study this problem under certain matrix norms. First, we give a formal definition of the problem:

Problem:	GAST (with respect to $\\|\cdot\\|_{L_p}$).
Input:	A connected graph G and an algebraic number γ.
Question:	Is there a spanning tree T of G with $\\|(D_T - D_G) \circ A_G\\|_{L_p} \leq \gamma$?

Remember that D_G and D_T are the distance matrices of graph G and tree T, while A_G denotes the adjacency matrix of G.

In the following, we show that computing such a tree is hard under the L_p-norm for all $p \in \mathbb{N}$. Note that $p = 1$ is a special case of the NP-complete Minimum Fundamental Cycle Basis Problem (Min-FCB) [3].

Theorem 2. GAST *with respect to* $\|\cdot\|_{L_p}$ *is* NP*-complete for all* $p \in \mathbb{N}$.

Proof. The containment in NP is obvious. We prove the hardness by reduction from 2HS using the graph representation $G(\mathcal{C}, \mathcal{S})$. The idea of the reduction is simple: We join the end vertices of $G(\mathcal{C}, \mathcal{S})$, v'_0 and v'_n, by a connection consisting of a couple of paths. Moreover, the same technique is used to force the clause path edges of $G(\mathcal{C}, \mathcal{S})$ into any optimal spanning tree. The number of additional paths and their length depends on the given 2HS instance and is polynomial in the number of literals in $\mathcal{S}$ and clauses in $\mathcal{C}$.

Let $(\mathcal{C}, \mathcal{S}, k)$ be an instance of 2HS and let N be the number of vertices in its graph representation $G(\mathcal{C}, \mathcal{S})$. We define the graph $G = (V, E)$ such that V consists of the vertices in the graph $G(\mathcal{C}, \mathcal{S})$ and

- N_A vertex sets $\{a_{\mu,1}, a_{\mu,2}, \ldots, a_{\mu,L_A}\}$ for each $\mu \in \{1, \ldots, N_A\}$ and
- $N_B \cdot K$ vertex sets $\{b_{\nu,1}^\mu, b_{\nu,2}^\mu, \ldots, b_{\nu,L_B}^\mu\}$ for each $\mu \in \{1, \ldots, K\}$ and each $\nu \in \{1, \ldots, N_B\}$

where K is the number of clause path edges in $G(\mathcal{C}, \mathcal{S})$ and $N_A, N_B, L_A, L_B \in \mathbb{N}$ are four parameters which will be chosen later.

The edge set E consists of all edges in $G(\mathcal{C}, \mathcal{S})$ and

- N_A paths $A_\mu = (v'_0, a_{\mu,1}, a_{\mu,2}, \ldots, a_{\mu,L_A}, v'_n)$ for each $\mu \in \{1, \ldots, N_A\}$ and
- $N_B \cdot K$ paths $B^\mu_\nu = (u, b^\mu_{\nu,1}, b^\mu_{\nu,2}, \ldots, b^\mu_{\nu,L_B}, v)$ for each $\mu \in \{1, \ldots, K\}$ and for each $\nu \in \{1, \ldots, N_B\}$ where $\{u, v\}$ is a clause path edge.

The gadget is illustrated in Figures 2 and 3 where the latter figure is a detailed view onto the extension of a clause path edge which is not displayed in Figure 2. Obviously, the number of vertices and edges in G are polynomial in n and m if N_A, N_B, L_A, and L_B are polynomial in n and m.

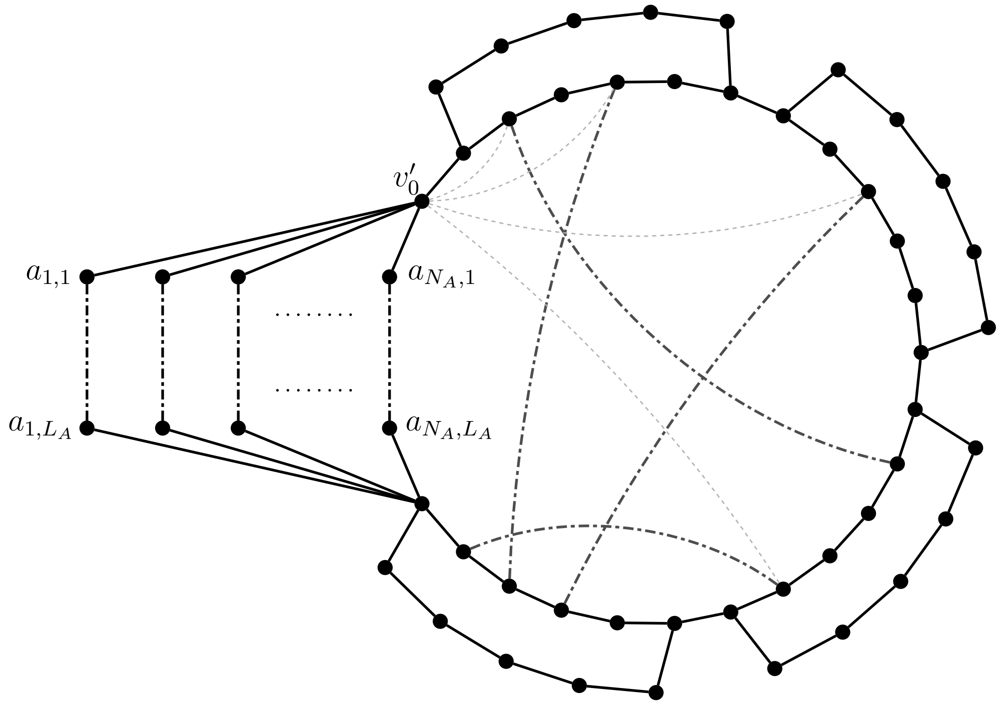

Figure 2: Extended graph representation of a 2HS instance.

Now, we define

$$\gamma = N^{p+1} + N_A \cdot (L_A + n(m+2) + k)^p + K \cdot N_B \cdot L_B^p$$

and we claim that G has a spanning tree T such that $\|(D_T - D_G) \circ A_G\|^p_{L_p} \leq \gamma$ if and only if $(\mathcal{C}, \mathcal{S}, k)$ has an admissible 2HS-solution $\mathcal{S}'$ of size $|\mathcal{S}'| \leq k$.

Claim 3. *Let $(\mathcal{C}, \mathcal{S}, k)$ be an instance of* 2HS*. Then $G = (V, E)$ has a spanning tree T such that $\|(D_T - D_G) \circ A_G\|^p_{L_p} \leq \gamma$ if $(\mathcal{C}, \mathcal{S}, k)$ has an admissible solution $\mathcal{S}'$ of size $|\mathcal{S}'| \leq k$.*

Proof. Let $\mathcal{S}'$ be an admissible solution of $(\mathcal{C}, \mathcal{S}, k)$ such that $|\mathcal{S}'| \leq k$. We construct the spanning tree T of G as follows: For the part of G which corresponds to $G(\mathcal{C}, \mathcal{S})$, we use the spanning tree $T_{G(\mathcal{C},\mathcal{S})}$ according to Lemma 1. In that construction, we remove two edges for

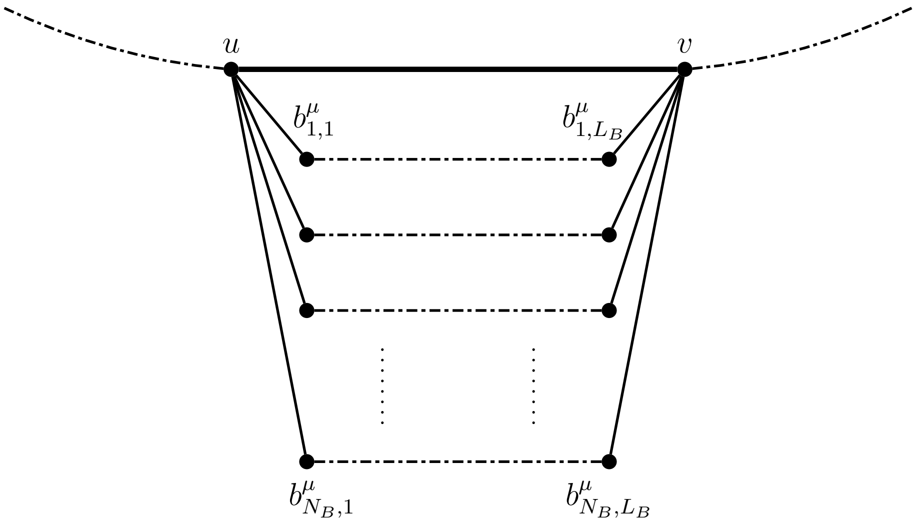

Figure 3: Extension paths of a clause path edge $\{u, v\}$.

each clause in $\mathcal{C}$ as well as one edge for each literal in $\mathcal{S}$ of the given 2HS instance $(\mathcal{C}, \mathcal{S}, k)$. Thus, if N is the number of vertices in $G(\mathcal{C}, \mathcal{S})$, it holds that

$$\sum_{\{u,v\} \in G(\mathcal{C},\mathcal{S})} (d_T(u, v) - 1)^p \le N \cdot N^p = N^{p+1} \ . \tag{1}$$

Note that $\{u, v\} \in G(\mathcal{C}, \mathcal{S})$ denotes all edges in $G(\mathcal{C}, \mathcal{S})$ (it does *not* mean all pairs of vertices). For the sake of readability, we assume this meaning of the notation unless stated otherwise.

Additionally, we break the paths A_μ and B^ν_κ as follows:

1. For each path A_μ, $\mu \in \{1, \dots N_A\}$, we remove the edge $\{a_{\mu,1}, a_{\mu,2}\}$.
2. For each path B^ν_κ, $\nu \in \{1, \dots, K\}$ and $\kappa \in \{1, \dots, N_B\}$, we remove the edge $\{b^\nu_{\kappa,1}, b^\nu_{\kappa,2}\}$.

For the first construction rule, the contribution to $\|(D_T - D_G) \circ A_G\|^p_{L_p}$ is bounded by

$$\sum_{\mu=1}^{N_A} \sum_{\{u,v\} \in A_\mu} (d_T(u, v) - 1)^p \le N_A \cdot (L_A + n(m + 2) + k)^p \tag{2}$$

and for the second rule, we obtain

$$\sum_{\nu=1}^{K} \sum_{\kappa=1}^{N_B} \sum_{\{u,v\} \in B^\nu_\kappa} (d_T(u, v) - 1)^p \le K \cdot N_B \cdot L^p_B \ . \tag{3}$$

Combining (1), (2), and (3), we get the required quality of T

$$\sum_{\{u,v\} \in E} (d_T(u, v) - 1)^p \le N^{p+1} + N_A \cdot (L_A + n(m + 2) + k)^p + K \cdot N_B \cdot L^p_B = \gamma \ .$$

□

Claim 4. *Let $(\mathcal{C}, \mathcal{S}, k)$ be an instance of* 2HS. *Then $(\mathcal{C}, \mathcal{S}, k)$ has an admissible solution $\mathcal{S}'$ of size $|\mathcal{S}'| \leq k$ if the graph G has a spanning tree T such that $\|(D_T - D_G) \circ A_G\|_{L_p}^p \leq \gamma$ with $N_A > N^{p+1}$, $N_B > N_A \cdot (2N^2)^p$, $L_A = N^2 \cdot (n(m+2)+k)$, and $L_B > N^2 + 1$.*

Proof. Let T be a spanning tree of $G = (V, E)$ such that

$$\|(D_T - D_G) \circ A_G\|_{L_p}^p \leq \gamma = N^{p+1} + N_A \cdot (L_A + n(m+2) + k)^p + K \cdot N_B \cdot L_B^p \,. \tag{4}$$

Assume, there is a clause path edge which does not belong to T. We consider the extension paths of this edge (see Figure 3) and distinguish two different cases. Either there is exactly one path B_κ^ν ($\kappa \in \{1, \ldots, N_B\}$ and ν depends on the removed clause path edge), which is intact, or each of these paths is broken. Note that two or more intact extension paths of the same clause path edge would imply the existence of a cycle. For the first case, we can lower bound the contribution to $\|(D_G - D_T) \circ A_G\|_{L_p}^p$ for removing an edge from A_μ, $\mu \in \{1, \ldots, N_A\}$, by

$$\sum_{\mu=1}^{N_A} \sum_{\{u,v\} \in A_\mu} (d_T(u,v) - 1)^p \geq N_A \cdot (L_A + n(m+2))^p \tag{5}$$

and for all deleted edges of paths B_κ^ν, $\nu \in \{1, \ldots, K\}$ and $\kappa \in \{1, \ldots, N_B\}$, we obtain

$$\begin{aligned} \sum_{\nu=1}^{K} \sum_{\kappa=1}^{N_B} \sum_{\{u,v\} \in B_\kappa^\nu} (d_T(u,v) - 1)^p &\geq (K-1) \cdot N_B \cdot L_B^p + (N_B - 1) \cdot (2L_B)^p + L_B^p \\ &= K \cdot N_B \cdot L_B^p + (2^p - 1) \cdot (N_B - 1) \cdot L_B^p \,. \end{aligned} \tag{6}$$

By assumption, we have $N_A > N^{p+1}$, $L_B > N^2 + 1$, $N_B > N_A \cdot (2N^2)^p$ with $p \in \mathbb{N}$. Thus,

$$N_A \cdot (N^2 \cdot (n(m+2) + k) + n(m+2))^p > N^{p+1}$$

and

$$(2^p - 1) \cdot (N_B - 1) \cdot L_B^p > N_A \cdot ((N^2 + 1) + (n(m+2) + k))^p$$

imply the following contradiction to (4):

$$\begin{aligned} & N_A \cdot (N^2 \cdot (n(m+2) + k) + n(m+2))^p + (2^p - 1) \cdot (N_B - 1) \cdot L_B^p \\ & \qquad > N^{p+1} + N_A \cdot ((N^2 + 1) \cdot (n(m+2) + k))^p \\ \implies\ & N_A \cdot (L_A + n(m+2))^p + K \cdot N_B \cdot L_B^p + (2^p - 1) \cdot (N_B - 1) \cdot L_B^p \\ & \qquad > N^{p+1} + N_A \cdot (L_A + n(m+2) + k)^p + K \cdot N_B \cdot L_B \\ \implies\ & \|(D_T - D_G) \circ A_G\|_{L_p}^p > \gamma \,. \end{aligned}$$

For the second case where each path $B_\kappa^{\{u,v\}}$ with $\kappa \in \{1, \ldots, N_B\}$ is broken, the lower bound in (5) holds and (6) changes to

$$\begin{aligned} \sum_{\nu=1}^{K} \sum_{\kappa=1}^{N_B} \sum_{\{u,v\} \in B_\kappa^\nu} (d_T(u,v) - 1)^p &\geq (K-1) \cdot N_B \cdot L_B + N_B \cdot (L_B + 2n(m+2))^p \\ &= K \cdot N_B \cdot L_B^p + N_B \cdot (2n(m+2))^p \,. \end{aligned}$$

Analogously to the previous case, we get a contradiction to (4). Thus, all K clause path edges are forced into each optimal spanning tree by the extension paths B_ν^μ, $\mu \in \{1, \dots, K\}$ and $\nu \in \{1, \dots, N_B\}$.

Now, we turn our attention to the paths A_μ, $\mu \in \{1, \dots, N_A\}$, and show that all of these paths must be broken. Afterwards, we prove that the distance between v_0' and v_n' is at most $n(m+2)+k$. Since all clause path edges belong to T, we are able to apply Lemma 1 in order to prove the existence of an admissible solution $\mathcal{S}'$ of size $|\mathcal{S}'| \leq k$ for the 2HS instance $(\mathcal{C}, \mathcal{S}, k)$. To this end, we first assume that there is an intact path A_μ for some $\mu \in \{1, \dots, N_A\}$. The contribution of all broken paths A_μ includes the length of the intact path:

$$\sum_{\mu=1}^{N_A} \sum_{\{u,v\} \in A_\mu} (d_T(u,v) - 1)^p \geq (N_A - 1) \cdot (2L_A)^p .$$

Since the direct connection between v_0' and v_n' (inside the original 2HS gadget) is broken, there exists some edge $\{u, v\} \in G(\mathcal{C}, \mathcal{S})$ such that

$$(d_T(u,v) - 1)^p \geq L_A^p .$$

This is a contradiction to (4) since

$$\begin{aligned}
& \|(D_T - D_G) \circ A_G\|_{L_p}^p > \gamma \\
\Longleftarrow \quad & (N_A - 1) \cdot (2L_A)^p + L_A^p + K \cdot N_B \cdot L_B^p \\
& \qquad > N \cdot N^{p+1} + N_A \cdot (L_A + n(m+2) + k)^p + K \cdot N_B \cdot L_B^p \\
\Longleftarrow \quad & (N_A - 1) \cdot (2N^2 \cdot (n(m+2) + k))^p + (N^2 \cdot (n(m+2) + k))^p \\
& \qquad > N^{p+1} + N_A \cdot ((N^2 + 1) \cdot (n(m+2) + k))^p \\
\Longleftarrow \quad & (N_A - 1) \cdot 2^p \cdot N^{2p} > N_A \cdot (N^2 + 1)^p \\
\Longleftarrow \quad & N^2 \cdot 2^p \cdot N^{2p} > (N^2 + 1)^{p+1} \\
\Longleftarrow \quad & N^2 \cdot 2^{\frac{p}{p+1}} > N^2 + 1
\end{aligned}$$

which is true if $N_A > N^{p+1}$, $L_A = N^2 \cdot (n(m+2) + k)$, $N > 1$, and $p \in \mathbb{N}$.

Closing the proof, we assume that the distance between v_0' and v_n' is greater than $n(m+2)+k$. Choosing $N_A > N^{p+1}$, we obtain a contradiction to (4) since $N_A \cdot (L_A + n(m+2) + k + 1)^p > N^{p+1} + N_A \cdot (L_A + n(m+2) + k)^p$ implies $\|(D_T - D_G) \circ A_G\|_{L_p}^p > \gamma$. □

This proves the theorem. ■

5 Conclusion

We investigated the complexity of the GAST problem which is related to a couple of other network optimization problems. We proved NP-completeness for all L_p-norms ($p \in \mathbb{N}$). A remaining open question is the approximability of GAST for $p > 1$. The case $p = 1$ is a special case of the Minimum Fundamental Cycle Base (Min-FCB) problem and can be approximated within $\mathcal{O}(\log^2 n)$, since the distances are metric [11].

Acknowledgements

We thank Stefan Eckhardt for helpful discussions.

References

[1] P. M. Camerini, G. Galbiati, and F. Maffioli. Complexity of Spanning Tree Problems: Part I. *European Journal of Operational Research*, 5(5):346–352, 1980.

[2] E. Dahlhaus, P. Dankelmann, W. Goddard, and H. C. Swart. MAD Trees and Distance-Hereditary Graphs. *Discrete Applied Mathematics*, 131(1):151–167, 2003.

[3] N. Deo, G. Prabhu, and M. S. Krishnamoorthy. Algorithms for Generating Fundamental Cycles in a Graph. *ACM Transactions on Mathematical Software*, 8(1):26–42, 1982.

[4] S. Eckhardt, S. Kosub, M. G. Maaß, H. Täubig, and S. Wernicke. Combinatorial Network Abstraction by Trees and Distances. In *Proc. of the 16th Int. Symposium on Algorithms and Computation (ISAAC'05)*, LNCS 3827, pages 1100–1109, Springer, 2005.

[5] G. Galbiati and E. Amaldi. On the Approximability of the Minimum Fundamental Cycle Basis Problem. In *Proceedings of the 1st Int. Workshop on Approximation and Online Algorithms (WAOA'03)*, LNCS 2909, pages 151–164, Springer, 2004.

[6] R. Hassin and A. Tamir. On the Minimum Diameter Spanning Tree Problem. *Information Processing Letters*, 53(2):109–111, 1995.

[7] T. C. Hu. Optimum Communication Spanning Trees. *SIAM Journal on Computing*, 3(3):188–195, 1974.

[8] D. S. Johnson, J. K. Lenstra, and A. H. G. Rinnooy Kan. The Complexity of the Network Design Problem. *Networks* 8(4):279–285, 1978.

[9] E. Minieka. A Polynomial Time Algorithm for Finding the Absolute Center of a Network. *Networks*, 11(4):351–355, 1981.

[10] D. Peleg and E. Reshef. Deterministic Polylog Approximation for Minimum Communication Spanning Trees. In *Proc. of the 25th Int. Colloquium on Automata, Languages and Programming (ICALP'98)*, LNCS 1443, pages 670–681, Springer, 1998.

[11] E. Reshef. Approximating Minimum Communication Cost Spanning Trees and Related Problems. Master's Thesis, Weizmann Institute of Science, Rehovot, Israel, 1999.