Available online at www.sciencedirect.com

ScienceDirect

Information Processing Letters 104 (2007) 113–116

Information Processing Letters

www.elsevier.com/locate/ipl

Spanning trees with minimum weighted degrees

Mohammad Ghodsi [a,b,*,1], Hamid Mahini [a,b,1], Kian Mirjalali [a], Shayan Oveis Gharan [a], Amin S. Sayedi R. [a], Morteza Zadimoghaddam [a]

[a] *Department of Computer Engineering, Sharif University of Technology, Tehran, Iran*
[b] *School of Computer Science, Institute for Studies in Theoretical Physics and Mathematics (IPM), Tehran, Iran*

Received 4 January 2007; received in revised form 1 May 2007

Available online 22 June 2007

Communicated by F.Y.L. Chin

Abstract

Given a metric graph G, we are concerned with finding a spanning tree of G where the maximum weighted degree of its vertices is minimum. In a metric graph (or its spanning tree), the weighted degree of a vertex is defined as the sum of the weights of its incident edges. In this paper, we propose a 4.5-approximation algorithm for this problem. We also prove it is NP-hard to approximate this problem within a $2-\varepsilon$ factor.

Keywords: Approximation algorithms; Graph algorithms; Spanning trees

1. Introduction

In this paper, we study the problem of finding the Minimum Weighted Degree Spanning Trees (MWDST) in metric graphs. In a weighted undirected graph G, the *weighted degree* of a vertex v, is defined as the sum of the weights of the edges incident to v in G. The *wd-cost* of a tree T (or $wd(T)$) is also defined as the maximum weighted degree of its vertices. We are interested in finding a minimum wd-cost spanning tree in a metric graph. A graph is said to be *metric* iff the weights of its edges hold the triangle inequality. Note that a metric graph is complete. We propose a 4.5-approximation algorithm for MWDST and prove that this problem cannot be approximated within a $2-\varepsilon$ factor in polynomial time unless $\mathcal{NP}=\mathcal{P}$.

This theoretical result can be used in several applications: In communication networks, for example, the weights of edges can represent the link bandwidths. It would be desirable to construct a broadcast subnetwork whose maximum amount of its nodes' bandwidth is minimized. This is an extension of a similar application discussed in [11] which minimizes the maximum degree of a network. Similarly, in sensor networks where nodes have limited powers, such spanning trees can be used to save energy in aggregate operations [13,10]. Similar results have been obtained for minimum degree Steiner trees in graphs [9].

* Corresponding author.
E-mail addresses: ghodsi@sharif.edu (M. Ghodsi), mahini@sharif.edu (H. Mahini), mirjalali@ce.sharif.edu (K. Mirjalali), oveisgharan@ce.sharif.edu (S. Oveis Gharan), sayedi@ce.sharif.edu (A.S. Sayedi R.), zadimoghaddam@ce.sharif.edu (M. Zadimoghaddam).
1 Supported in part by Institute for Theoretical Physics and Mathematics (IPM) under grant number CS1385-2-01.

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2007.06.011

Various problems of computing spanning trees which satisfy given constraints have been studied before [3, 4,6,14]. One of these problems which is to find the minimum cost spanning trees with bounded maximum degree has been studied in [1,2,5,7,12,16]. In these solutions, the weighted degrees are not considered. The problem of finding the minimum degree spanning tree has also been studied before in simple and weighted graphs. For example, it is shown in [9] that there is a polynomial-time algorithm that approximates this problem within one from the optimal solution.

Our problem has been studied in the general weighted graphs in [15] where the author has designed an $O(\log n)$-approximation algorithm for finding MWDST. In [8], the authors propose a polynomial $O(\log n)$-approximation algorithm which finds the minimum degree spanning tree in directed non-weighted graphs.

In this paper, we consider weighted metric graphs and, in Section 2, propose a 4.5-approximation algorithm for finding MWDST. In Section 3, we will prove that it is NP-hard to approximate this problem within a $2 - \varepsilon$ factor.

2. 4.5-Approximation for MWDST

In this section, we develop a 4.5-approximation algorithm for MWDST problem. Initially, we propose a 5-approximation algorithm which specially creates a Hamiltonian path which is then used in the 4.5-approximation algorithm.

Lemma 1. *Given a metric graph G and a spanning tree T of G rooted at r, there is a polynomial-time algorithm for finding a Hamiltonian path $h(T,r)$ in G with wd-cost at most $5M$, where*

$$M = \max_{e \in E(T)} \{w(e)\}.$$

$h(T,r)$ is of the form $v_1, v_2, \ldots, v_n$ and its edge sequence is $e_1, e_2, \ldots, e_{n-1}$. Hamiltonian path $h(T,r)$ has also the following properties:

- *$v_1 = r$ and v_n is one of r's children in T,*
- *$w(e_1)$ and $w(e_{n-1})$ do not exceed $2M$, and*
- *for each $i \in \{1, 2, \ldots, n-2\}$,*

$$\min\{w(e_i), w(e_{i+1})\} \leqslant 2M, \quad \text{and}$$
$$\max\{w(e_i), w(e_{i+1})\} \leqslant 3M.$$

Proof. We propose a recursive algorithm and prove it by induction on n. The solution is trivial for $n \leqslant 2$.

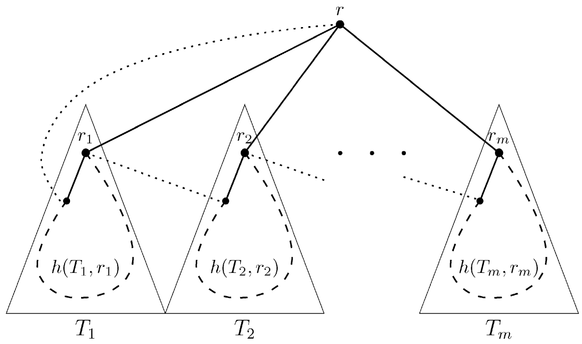

Fig. 1. Construction of $h(T,r)$ from the spanning tree T.

Let $r_1, r_2, \ldots, r_m$ be r's children in T, and $T_1, T_2, \ldots, T_m$ be subtrees of T rooted at r_i's, respectively. Solutions $h(T_i, r_i)$ can be found recursively in polynomial time. The solution is created as:

$$h(T,r) = r, h(T_1,r_1)^R, h(T_2,r_2)^R, \ldots, h(T_m,r_m)^R, \tag{1}$$

where X^R is the reverse of sequence X. This is shown in Fig. 1.

The first property of the $h(T,r)$ is obvious from its construction. Clearly, $w(e_1) \leqslant 2M$ is true due to the triangle inequality, and $w(e_{n-1}) \leqslant 2M$ comes from the induction hypothesis as it was already in $h(T_m, r_m)$.

It is now sufficient to prove the last property in order to show that the wd-cost of $h(T,r)$ is at most $5M$. The condition holds true for the internal nodes of $h(T_i, r_i)$ (where $1 \leqslant i \leqslant m$) by the induction hypothesis. The first node of $h(T_i, r_i)$, r_i ($i \in \{1, 2, \ldots, m-1\}$),[2] is adjacent to two other vertices in $h(T,r)$ through edges e_j and e_{j+1} (for some j). The induction hypothesis gives us the result $w(e_j) \leqslant 2M$ and we have $w(e_{j+1}) \leqslant 3M$ as a consequence of the triangle inequality. Now, consider the last node of $h(T_i, r_i)$ ($i \in \{1, 2, \ldots, m\}$) and its two incident edges e_j and e_{j+1} in $h(T,r)$. Again, $w(e_{j+1}) \leqslant 2M$ is a result of the induction hypothesis, and $w(e_j) \leqslant 3M$ comes from the triangle inequality. Note that in case of $i = 1$ the stronger result $w(e_j) \leqslant 2M$ holds. □

Lemma 2. *Assume that the metric graph G has a spanning tree of wd-cost at most R. We can find a spanning tree T of G with wd-cost at most $4.5R$ in polynomial time.*

[2] Case $i = m$ should not be considered, because it is the last vertex of $h(T,r)$.

Proof. Construct the graph G' by deleting each edge of G that is heavier than $\frac{R}{2}$ in weight. Assume that G' has connected components $C_1, C_2, \ldots, C_k$, and G_i is the induced subgraph of G by vertex set C_i. We know that each G_i has a spanning tree with total edge weights no more than $\frac{R}{2}$. According to Lemma 1, G_i has a Hamiltonian path P_i with wd-cost at most $\frac{5R}{2}$.

Let E_1 be the set of G edges that go from G_i to G_j (for each $i \neq j$) each having weight of at most R. It is clear that the weights in E_1 are greater than $\frac{R}{2}$. Edge set E_2 is defined as the union of E_1 and edges of Hamiltonian paths P_i, $1 \leqslant i \leqslant k$.

We also know that G has a spanning tree T_R with wd-cost no more than R. As we know that the weight of each edge in E_1 is at least $\frac{R}{2}$. Therefore, each vertex in T_R is incident to at most one edge of E_1. Define G'' as a simple non-weighted graph with edge set E_2. Consider spanning subgraph of G'' whose edge set is the union of $E_1 \cap E(T_R)$ and edges of Hamiltonian paths P_i where $E(T_R)$ is the edge set of T_R and $1 \leqslant i \leqslant k$. We assert that this subgraph is connected. This is because the vertices inside each G_i are connected through P_i and two different components G_i and G_j are connected via edge set $E_1 \cap E(T_R)$. Therefore, G'' has a spanning tree with maximum degree at most 3, since each of its vertices uses at most one edge of $E_1 \cap E(T_R)$ and two edges of P_i's. We conclude that a spanning tree T of G'' with maximum degree of at most 4 can be found in polynomial time using the algorithm of [9]. Considering T as a spanning tree of G, we now prove that the maximum weighted degree of each vertex in T is at most $4.5R$. Each vertex v in T is incident to at most 4 edges and there are three possible cases:

- Two edges are from E_1 and two from P_i (for some $1 \leqslant i \leqslant k$). In this case, the weighted degree of v in $T \cap P_i$ is at most $5R/2$. So its weighted degree in T is at most $2 \times R + 5R/2 = 4.5R$.
- Three edges are from E_1 and one from P_i (for some $1 \leqslant i \leqslant k$). In this case, the weight of any edge from P_i is at most $3R/2$ using Lemma 1. So the weighted degree of v in T is at most $3 \times R + 3R/2 = 4.5R$.
- All edges are from E_1. So, the weighted degree of v in T is at most $4R$.

Therefore, wd-cost(T) is at most $4.5R$. □

Theorem 1. *MWDST problem in metric graphs can be approximated within a* 4.5 *factor in polynomial time.*

Proof. Let G be a metric graph with n vertices. Assume wd-cost of MWDST in G is R. We know that R is at least 0 and at most $(n-1)W$, where W is the maximum weight of the edges in G. We use a binary search to find an spanning tree T with wd-cost of at most $4.5R$ by the following simple algorithm:

1. Set $L = 0$ and $U = (n-1)W$ which are the lower and upper bounds for wd-cost.
2. Set $M = (L+U)/2$ and use Lemma 2 by assuming that G has a spanning tree of wd-cost at most M. We know that if $M \geqslant R$, then the algorithm in this lemma will find an spanning tree with wd-cost at most $4.5M$. Therefore, get the result of Lemma 2 and check whether its wd-cost is more than $4.5M$. If we could not find such spanning tree, it means than M is less than R. So set $L = M$ and go to step 3. If we find an spanning tree of wd-cost at most $4.5M$, save the result as T_M, set $U = M$, and go to step 3.
3. If $U = L$, return the best solution among saved T_M's. If $U \neq L$ go again to step 2.

We know if the algorithm saves an spanning tree T_M whose wd-cost is at most $4.5M$. On the other hand, it is clear that the algorithm will save a T_M with $M \leqslant R$. Therefore, the algorithm will find an spanning tree with the desired properties. □

3. $(2-\varepsilon)$ Inapproximability for MWDST

In this section we prove that MWDST is hard to approximate.

Theorem 2. *For every constant ε $(0 < \varepsilon \leqslant 1)$, it is $\mathcal{NP}$-hard to approximate MWDST in metric graphs within a $2-\varepsilon$ factor in polynomial time.*

Proof. We prove that, if MWDST can be approximated within a $2-\varepsilon$ factor, then the Hamiltonian path problem can be solved in polynomial time. Let $G(V, E)$ be an instance of Hamiltonian path problem, where $V = v_1, \ldots, v_n$ is the set of its vertices and E is its edges.

We construct a graph H from G as follows. For each i $(1 \leqslant i \leqslant n)$, we put two vertices u_i and u'_i in H and connect them with an edge of weight 0. For each edge $e(v_i, v_j) \in E$, we add 4 edges (u_i, u_j), (u_i, u'_j), (u'_i, u_j), and (u'_i, u'_j) each with weight 1. Now, we put an edge of weight 2 between any two vertices, if there is no edge between them.

It is clear that H is a metric graph. We prove that there is a Hamiltonian path in G if and only if there is a spanning tree in H with wd-cost of 1. Consider a Hamiltonian path $P = v_{\pi_1}, v_{\pi_2}, \ldots, v_{\pi_n}$ in G. The Hamiltonian path $T = u_{\pi_1}, u'_{\pi_1}, u_{\pi_2}, u'_{\pi_2}, u_{\pi_3}, u'_{\pi_3}, \ldots, u_{\pi_n}, u'_{\pi_n}$, is a spanning tree of H with wd-cost of 1.

Let T be a spanning tree in H with wd-cost of 1. It is clear that T does not have any edge of weight 2. Also, no vertex in T is incident to two edges of weight 1. From H, we construct a spanning tree P in G as follows. Connect v_i and v_j in P if and only if T has at least one of the edges (u_i, u_j), (u_i, u'_j), (u'_i, u_j), and (u'_i, u'_j). If a vertex v_i is incident to 3 edges in P, then either u_i or u'_i is incident to the two edges of weight 1 in T. Connectivity of T in H implies the connectivity of P in G. So P is a Hamiltonian path or a Hamiltonian cycle.

If MWDST has a polynomial-time α-approximation algorithm with $\alpha < 2$, we can run it on H and this will determine whether or not H has a spanning tree with wd-cost of 1. Therefore, we can determine in polynomial time whether G has a Hamiltonian path. □

By setting $\varepsilon = 1$ in Theorem 2, we can conclude finding MWDST in metric graphs is $\mathcal{NP}$-hard.

4. Conclusion

We considered the problem of finding the minimum weighted degree spanning tree in metric graphs. In such a graph G, we look for a spanning tree whose maximum weighted degree is minimized. The weighted degree of a vertex is defined as the sum of the weights of its incident edges. In this paper, we proposed a 4.5-approximation algorithm for this problem. We also proved that this problem cannot be approximated within a $2 - \varepsilon$ factor. So, the problem of finding a close-to-2-approximation for this problem remains open and seems to be challenging.

References

[1] M.X. Goemans, Bounded degree minimum spanning trees, in: Proc. of the 47th Annual IEEE Symposium on Foundations of Computer Science, 2006, pp. 273–282.

[2] R. Ravi, M. Singh, Delegate and conquer: An LP-based approximation algorithms for minimum degree MSTs, in: Proc. of the 33rd International Colloquium on Automata, Languages and Programming, 2006, pp. 169–180.

[3] P.M. Camerini, G. Galbiati, The bounded path tree problem, SIAM Journal on Algebraic and Discrete Methods 3 (4) (1982) 474–484.

[4] P.M. Camerini, G. Galbiati, F. Maffioli, Complexity of spanning tree problems, Part I, European Journal of Operations Research 5 (1980) 346–352.

[5] K. Chaudhuri, S. Rao, S. Riesenfeld, K. Talwar, Push relabel and an improved approximation algorithm for the bounded-degree MST problem, in: Proc. of the 33rd International Colloquium on Automata, Languages and Programming, 2006, pp. 191–201.

[6] P.M. Camerini, G. Galbiati, F. Maffioli, On the complexity of finding multi-constrained spanning trees, Discrete Applied Mathematics 5 (1983) 39–50.

[7] K. Chaudhuri, S. Rao, S. Riesenfeld, K. Talwar, What would Edmonds do? Augmenting paths, witnesses and improved approximations for bounded-degree MSTs, in: Proc. of the 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, 2005.

[8] M. Furer, B. Raghavachari, An NC approximation algorithm for the minimum degree spanning tree problem, in: Proc. of the 28th Ann Allerton Conference on Communication, Control and Computing, 1990, pp. 274–281.

[9] M. Furer, B. Raghavachari, Approximating the minimum degree spanning tree to within one from the optima degree, in: Proc. of the Third Annual ACM–SIAM Symposium on Discrete Algorithms, 1992, pp. 317–324.

[10] K. Kalpakis, K. Dasgupta, P. Namjoshi, Maximum lifetime data gathering and aggregation in wireless sensor networks, in: Proc. of IEEE International Conference on Networking, 2002.

[11] P.N. Klein, R. Krishnan, B. Raghavachari, R. Ravi, Approximation algorithms for finding low-degree subgraphs, Networks 44 (3) (2004) 203–215.

[12] J. Konemann, R. Ravi, A matter of degree: Improved approximation algorithms for degree-bounded minimum divining trees, in: Proc. of the 32nd Annual ACM Symposium on Theory of Computing, 2000.

[13] B. Krishnamachari, D. Estrin, S. Wicker, Modelling data-centric routing in wireless sensor networks, in: Proc. of IEEE INFOCOM, 2002.

[14] C.H. Papadimitriou, M. Yannakakis, The complexity of restricted spanning tree problems, Journal of the ACM 29 (1982) 285–309.

[15] R. Ravi, Steiner trees and beyond: Approximation algorithms for network design, PhD thesis, available as Technical Report TR-CS- 93-41, Brown University, September 1993.

[16] R. Ravi, M.V. Marathe, S.S. Ravi, D.J. Rosenkrantz, H.B. Hunt, Many birds with one stone: multi-objective approximation algorithms, in: Proc. of the 25th Annual ACM Symposium on Theory of Computing, 1993.