

Two Constant Approximation Algorithms for Node-Weighted Steiner Tree in Unit Disk Graphs

Feng Zou¹, Xianyue Li², Donghyun Kim¹ and Weili Wu^{1*}

¹ Department of Computer Science, University of Texas at Dallas,
Richardson, TX, 75080

{phenix.zou,donghyunkim}@student.utdallas.edu, weiliwu@utdallas.edu

² School of Mathematics and Statistics, Lanzhou University,
Lanzhou, Gansu, P.R.China, 730000

lixianyue@lzu.edu.cn

Abstract. Given a graph $G = (V, E)$ with node weight $w : V \rightarrow R^+$ and a subset $S \subseteq V$, find a minimum total weight tree interconnecting all nodes in S . This is the node-weighted Steiner tree problem which will be studied in this paper. In general, this problem is NP-hard and cannot be approximated by a polynomial time algorithm with performance ratio $a \ln n$ for any $0 < a < 1$ unless $NP \subseteq DTIME(n^{O(\log n)})$, where n is the number of nodes in s . In this paper, we show that for unit disk graph, the problem is still NP-hard, however it has polynomial time constant approximation. We will present a 4-approximation and a 2.5ρ -approximation where ρ is the best known performance ratio for polynomial time approximation of classical Steiner minimum tree problem in graphs. As a corollary, we obtain that there is polynomial time $(9.875+\varepsilon)$ -approximation algorithm for minimum weight connected dominating set in unit disk graphs.

1 Introduction

Given a graph $G = (V, E)$ with weight function w on E and a subset S , *Steiner tree problem* (STP) is to find a minimum subgraph of G interconnecting all nodes in S . We call the set S as *terminal set*. For any Steiner tree T for S

* Support in part by National Science Foundation under grant CCF-0514796 and CCF-0750992.

and node $u \in V(T)$, we call u as a *terminal node* if $u \in S$, otherwise, we call it as a *Steiner node*. The Steiner tree problem is a classical problem in networks, which is very interesting nowadays. The problem is known to be NP-hard, as well as in most metrics[6]. Lots of effort have been devoted to study the approximation algorithms for this problem[3, 8, 11, 14, 17] and have successfully achieved constant ratios for this problem.

Node-Weighted Steiner Tree problem (NWST) is a variation of the classical STP. Given a graph $G = (V, E)$ with node weight $w : V \rightarrow R^+$ and a subset S of V , the node-weighted Steiner tree problem is to find a Steiner tree for the set S such that its total weighted is minimum. Since all of nodes in terminal set will be contained in any Steiner tree constructed, for convenience, we usually set $w(u) = 0$ for all nodes $u \in S$.

In this paper, We study NWST problem in a special type of graphs called unit disk graph, which has a wide application in networks. A *unit disk graph* is associated with a set of unit disks in the Euclidean plane. Each node is the center of a unit disk. An edge exists between two nodes u and v if and only if $|uv| \leq 1$, where $|uv|$ is the Euclidean distance between u and v . For this special type of graphs, we propose two constant approximation algorithms for NWST problem from different perspectives with approximation ratio 4 and 2.5ρ respectively, where $\rho = 1 + \frac{\ln 3}{2} \approx 1.55$ is the best known approximation for the classical Steiner tree problem[14]. As an application on Minimum Weighted Connected Dominating Set (MSCDS) problem, we obtain a $(9.875+\varepsilon)$ -approximation algorithm for this problem, which improves the best previous approximation ratio $10+\varepsilon$ [9].

The rest of this paper is organized as follows. In section 2, we introduce the related work for NWST problem. In section 3, we first present the main idea of our algorithms. Then, we give some useful definitions and denotations. In subsection 3.2 and 3.3, we introduce the algorithms and prove their approximation ratios, respectively. As a corollary, we obtain a $(9.875+\varepsilon)$ -approximation algorithm for MSCDS problem in unit disk graphs. Final, we conclude our results.

2 Related Work

Quite a few work has been done for NWST problem starting from early 80s. Most of the early work[1, 2, 13, 15, 16] focus on solving the problem and its variations

using linear programming. For instance, Aneja [1] used a specialized integer programming (set covering) formulation to represent the STP and used the row generation scheme to solve the exponential increase of the number of constraints in the formulation related to the size of the problem. Segev [15] proposed an integer programming solution for an extension of the standard Steiner Tree problem using lagrangian relaxation and subgradient optimization. Though these work presented solutions for Steiner tree problem from the perspective of linear programming, they did not provide solid theoretical proof for the approximation ratios of their algorithms at all.

In 1991, Berman (see ref. in [10]) proved that NWST problem can not be approximated within a factor of $o(\ln k)$ by giving an approximation-preserving reduction from Set Cover[5] to NWST problem, where k is the size of the terminal set. Later, Klein and Ravi [10] presented the first asymptotically optimal solution of approximation ratio $2 \ln k$, by constructing the Steiner tree with spiders, a tree with at most one node of degree greater than two. Later, this ratio is improved to be $1.35 \ln k$ by Guha and Khuller [7] by introducing a new concept called branch-spider. This is the best known ratio up till now.

As a special case, when the weights of all nodes are same, NWST problem is equivalent to Steiner Tree with Minimum Number of Steiner Nodes (ST-MSN) problem. Given a set of n terminals S in the Euclidean plane and a positive constant c , the ST-MSN problem is to find a Steiner tree for S with minimum number of Steiner nodes such that each edge in the tree has a length no more than c . In [4], they showed that there is a 3-approximation algorithm for ST-MSN. In 2006, M.Min et al.[12] presented this problem in unit disk graph and gave a 3-approximation algorithm for a special terminal set.

3 Node-Weighted Steiner Tree Problem (NWST)

In this section, we study the NWST problem in unit disk graphs and propose two approximation algorithms for solving this problem. The main idea of the first one is to construct the Minimum Spanning Tree (MST) to solve the NWST. In this algorithm, we first construct an edge-weighted complete graph G' on terminal set such that the weight of every edge in G' is equal to the length of the shortest path between its endpoints in the original graph G . Then, compute a MST of the new graph G' . By replacing each edge in the MST with the corresponding shortest

path in the original graph, we obtain a connected subgraph of G containing all nodes in the terminal set. We prove that this yields a valid NWST which has a total weight of no more than 4 times the weight of the optimal Node-Weight Steiner Tree of G .

The second algorithm has a better approximation ratio than the first one, while rooted from a totally different methodology. The main idea is based on the classical Steiner Tree problem. Firstly, we construct an edge-weighted graph G'' with the same node-set and edge-set as the original graph G . Then, we define the weight of every edge in G'' as the half of the sum of its endpoints' weights in G . Finally, we use ρ -approximation algorithm to obtain a Steiner tree of G'' . From the Steiner tree of G'' , we can get a Steiner Tree of G with approximation ratio $2.5\rho \approx 3.875$. As a corollary, we obtain a polynomial time algorithm for minimum weight connected dominating set in unit disk graphs with approximation ratio $9.875+\varepsilon$.

3.1 Preliminaries

In this section, we give some useful definitions and denotations. For a node-weighted (or edge-weighted) graph G with weight function w and a subgraph H of G , denote $w(H)$ be the the weight sum of all nodes (or edges) in H . For any two nodes u and v of G , denote $dist_G(u, v)$ as the weight of the shortest path between u and v , which is calculated as $\min \sum_{v_k \in p} w(v_k)$ among all the possible paths between u and v . Here v_k is the internal node on every possible path p .

Given a edge-weighted graph G and a node subset S , we denote the ρ -approximation algorithm for Steiner Minimum Tree(SMT) as **SMT(G,S)** and the algorithm for finding Minimum Spanning Tree(MST) as **MST(G)**.

3.2 4-Approximation Algorithm

In this algorithm, we first construct a complete graph G' on terminal set S with a weight function c on its edges such that $c(u, v) = dist_G(u, v)$ for any two nodes u and v in S . Denote the optimal Node-Weighted Steiner Tree of the graph G on S as T_{OPT_NWS} and the minimum spanning tree of G' as T_{MST} . We can obtain the following theorem:

Theorem 1. *Suppose that for any terminal set S in G , there always exists a minimum spanning tree T_{MST} for the newly constructed complete graph G' such*

that the maximum degree of it is at most Δ . Then, the weight of T_{MST} is no more than $(\Delta - 1)$ -times of the weight of T_{OPT_NWS} .

Proof. Suppose that T_{OPT_NWS} contains k Steiner points $\{s_1, s_2, \dots, s_k\}$ in the ordering of the breadth-first search starting from a vertex of S . Meanwhile, we define $N(S)$ as the weight of the minimum spanning tree of the new complete graph for the terminal set S . We claim that

$$N(S \cup \{s_1, s_2, \dots, s_i\}) \leq N(S \cup \{s_1, s_2, \dots, s_{i+1}\}) + (\Delta - 1) * w(s_{i+1}).$$

To show this claim, let G_i be the newly constructed complete graph for the terminal set $S \cup \{s_1, s_2, \dots, s_i\}$. Consider a minimum spanning tree T for G_{i+1} with degree at most Δ . Suppose s_{i+1} has adjacent vertices $v_1, v_2, \dots, v_d (d \leq \Delta)$. Then at least one of the edges $(s_{i+1}, v_j) (j = 1, \dots, d)$ has weight 0. Otherwise, by BFS, there exists a vertex $u \in S \cup \{s_1, s_2, \dots, s_i\}$ such that (s_{i+1}, u) is an edge of $E(G)$. This implies that the weight of this edge in the graph G_{i+1} is 0. If all weight of edges $(s_{i+1}, v_j) (j = 1, \dots, d)$ are more than 0, adding edge (s_{i+1}, u) and deleting one of them, we can get a new tree with weight less than the current one, which is a contradiction. So without loss of generality, assume that the weight of (s_{i+1}, v_1) is 0. Delete all edges $(s_{i+1}, v_j) (j = 2, \dots, d)$, and add $(d - 1)$ edges $(v_1, v_j) (j = 2, \dots, d)$. We obtain a spanning tree T' for G_i . Note that according to the triangle inequality of the distance between two vertices, $c_{G_i}(v_1, v_j) \leq c_{G_{i+1}}(s_{i+1}, v_j) + w(s_{i+1})$. Hence, the claim holds from $N(S \cup \{s_1, s_2, \dots, s_i\}) \leq c(T') \leq N(S \cup \{s_1, s_2, \dots, s_{i+1}\}) + (\Delta - 1) * w(s_{i+1})$. Since $N(S \cup \{s_1, s_2, \dots, s_k\}) = 0$ and by the claim and recurrence, we have

$$\begin{aligned} c(T_{MST}) &= N(S) \leq N(S \cup \{s_1\}) + (\Delta - 1) * w(s_1) \\ &\leq N(S \cup \{s_1, s_2\}) + (\Delta - 1)(w(s_1) + w(s_2)) \\ &\quad \vdots \\ &\leq N(S \cup \{s_1, s_2, \dots, s_k\}) + (\Delta - 1) \left(\sum_{i=1}^k w(s_i) \right) \\ &= (\Delta - 1)w(T_{OPT_NWS}). \end{aligned}$$

□

For every edge of T_{MST} , replacing it with the corresponding shortest path between its endpoints in G , we obtain a connected subgraph H of G containing

Algorithm 1 NWST1 ($G=(V,E,w,S)$)

```

1: Initialize a complete graph  $G' = (V', E', c)$  of  $G$  by setting  $V' = S$ 
2: for each pair of vertex  $(v_i, v_j)$  in graph  $G'$  do
3:   Set the  $c(v_i, v_j) = dis_G(v_i, v_j)$ , the shortest path between  $v_i$  and  $v_j$ 
4: end for
5:  $H = MST(G')$ 
6: for each edge  $(v_i, v_j)$  in graph  $T$  do
7:   Replace it with the corresponding shortest path in graph  $G$ 
8: end for
9: Output  $H$ 

```

all nodes of S . Obviously, the weight sum of all nodes in H is no more than the weight of T_{MST} .

The detailed algorithm is presented in algorithm 1. Since in any unit disk graph, there always exists a spanning tree with maximum degree no more than 5[12], we can obtain the following corollary:

Corollary 1. *The connected subgraph we obtained at the end of the algorithm NWST1 is 4-approximation for T_{OPT_NWS} in unit disk graph.*

3.3 2.5 ρ -Approximation Algorithm

The idea of this algorithm is to convert the node-weighted Steiner tree problem to the classical Steiner tree problem. Firstly, we construct an edge-weighted graph G'' from G as follows by initializing G'' with the same node-set and edge-set as G and an edge weight function w' . For every edge $e = (u, v)$ in G'' , let the edge weight $w'(u, v) = \frac{1}{2}(w(u) + w(v))$. The second step of this algorithm is to compute a Steiner tree T of G'' on S through the ρ -approximation algorithm. Final, view T as the node-weighted Steiner tree of G on S and output it. The pseudo-code of this algorithm is presented in algorithm 2.

To better illustrate this algorithm, we give an example. Given a node-weighted graph G and a terminal set S as in figure1. a, firstly, we transform it into graph G' 1. b according to the weight assignment. The numbers besides each vertex in G are the weights associated with them and we set the weight of all vertices in the terminal set as 0. Secondly, we calculate the SMT for graph G' , which is the subgraph in figure 1. c. Figure 1. d presents the NWST for original graph G .

Algorithm 2 NWST2 ($G = (V, E, w, S)$)

- 1: Initialize an edge-weighted graph $G' = (V', E', w', S')$ by setting $V' = V, S' = S$ and $E' = E$
 - 2: **for** each edge (v_i, v_j) in graph G' **do**
 - 3: Assign the weight of this edge $w'(v_i, v_j) = (w(v_i) + w(v_j))/2$.
 - 4: **end for**
 - 5: $T = \text{SMT}(G', S)$
 - 6: Output T
-

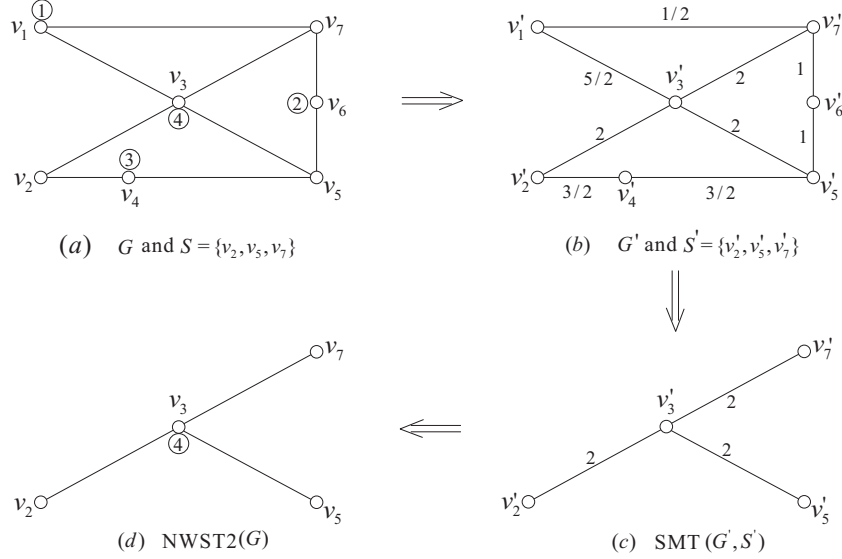


Fig. 1. An example of 2.5ρ algorithm

Since G is a unit disk graph, for any terminal set S , there always exists a Steiner minimum tree with maximum degree no more than 5[12]. The following lemma proves the relationship between the weight of the optimal SMT in graph G'' and the weight of the optimal NWST we want.

Lemma 1. Denote $T_{\text{OPT_SM}}$ as the optimal Steiner Minimum Tree for G'' on the set S and $T_{\text{OPT_NWS}}$ as the optimal Node-weighted Steiner Tree of G on the same terminal set S , respectively. Then $w'(T_{\text{OPT_SM}}) \leq 2.5w(T_{\text{OPT_NWS}})$ when G is a unit disk graph.

Proof. Consider T_{OPT_NWS} as a Steiner tree on S of G'' . For convenience, denote T as T_{OPT_NWS} , V as $V(T_{OPT_NWS})$, E as $E(T_{OPT_NWS})$ and $d_T(u)$ as the degree of node u in tree T , we have

$$\begin{aligned} w'(T_{OPT_SM}) &\leq w'(T_{OPT_NWS}) \\ &= \sum_{e=uv \in E} \left(\frac{1}{2}(w(u) + w(v))\right) \\ &= \sum_{u \in V} \frac{d_T(u)}{2} w(u) \\ &\leq \frac{5}{2} \sum_{u \in V} (w(u)) = 2.5w(T). \end{aligned}$$

Hence, the lemma holds. \square

If we consider the Steiner tree T of G'' as a subgraph of G , we can get the following lemma:

Lemma 2. *For any Steiner tree T of G'' on terminal set S , if we view it as a subgraph G , then T is also a Steiner tree of G on set S and $w(T) \leq w'(T)$.*

Proof. Since for any Steiner tree, the degree of each Steiner node is not less than 2 and all weight of nodes in terminal set is 0, the lemma holds. \square

Theorem 2. *Algorithm NWST2 is a 2.5ρ -approximation for node-weighted Steiner tree problem in unit disk graph.*

Proof. Let T be the output of NWST2. Denote T_{OPT_SM} and T_{OPT_NWS} as Lemma 1. By Lemmas 1 and 2, we have

$$w(T) \leq w'(T) \leq \rho w'(T_{OPT_SM}) \leq 2.5\rho w(T_{OPT_NWS}).$$

\square

Since the node-weighted Steiner tree can be used in the MWCDS problem to interconnect all nodes of the CDS, we can obtain the following corollary.

Corollary 2. *Using node-weighted Steiner tree to interconnect all nodes of the CDS produces $(9,875+\epsilon)$ -approximation for minimum weighted connected dominating set in unit disk graph.*

Proof. For any node-weighted graph G and a terminal set S , denote T_{OPT} and T_{OPT_CDS} be the optimal Steiner tree of G on S and optimal CDS of G , respectively. Since the induced graph $G[S \cup T_{OPT_CDS}]$ is connected, this graph contain a Steiner tree of G on S . Thus, $w(T_{OPT}) \leq w(T_{OPT_CDS})$. By [9], we can get a dominating set C of G with $w(C) \leq (6 + \varepsilon)w(T_{OPT_CDS})$. Then, using algorithm NWST2 for C , we can obtain a Steiner tree T with $w(T) \leq 2.5\rho w(T_{OPT})$. Clearly, $V(T)$ is a connected dominating set of G and

$$\begin{aligned} w(V(T)) &= w(C) + w(T) \\ &\leq (6 + \varepsilon)w(T_{OPT_CDS}) + 2.5\rho w(T_{OPT}) \\ &\leq (9.875 + \varepsilon)w(T_{OPT_CDS}). \end{aligned}$$

□

4 Conclusion

In this paper, we propose two constant approximation algorithms for NWST problem in unit disk graph. The approximation ratios of these two algorithms are 4 and 2.5ρ , respectively. As a corollary, we improve the approximation of MWCDS in unit disk graphs from $10+\varepsilon$ to $9.875+\varepsilon$.

References

1. Y.P.Aneja: An integer linear programming approach to the Steiner problem in graphs. *Networks*, **10** (1980) 167–178
2. J.E.Beasley: An algorithm for the Steiner problem in graphs. *Networks*, **14** (1984) 147–159
3. P.Berman and V.Ramaiyer: Improved approximations for the Steiner tree problem. *Journal of Algorithms*, **17** (1994) 381-408
4. D.Chen, D.Z.Du, X.D.Hu, G.H.Lin, L.Wang and G.Xue: Approximation for Steiner tree with minimum number of Steiner points. *Theoretical Computer Science*, **262** (2001) 83–99
5. U.Feige: A threshold of $\ln n$ for approximating set cover. *J. ACM*, **45** (1998) 634–652
6. M.R.Garey and D.S.Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Fransico, 1978
7. S.Guha and S.Khuller: Improved Methods for Approximating Node Weighted Steiner Trees and Connected Dominating Sets. *Information and Computation*, **150** (1999) 57–74

8. S.Hougardy and H.J.Prömel: A 1.598 Approximation Algorithm for the Steiner Problem in Graphs. SODA, (1998) 448–453
9. Y.Huang, X.Gao, Z.Zhang and W.Wu: A Better Constant-Factor Approximation for Weighted Dominating Set in Unit Disk Graph. Preprint
10. P.Klein and R.Ravi: A nearly best-possible approximation algorithm for node-weighted steiner trees. Journal of Algorithms, **19** (1995) 104–115
11. L.T.Kou, G.Markowsky and L.Berman: A Fast Algorithm for Steiner Trees, (1981) 141–145
12. M.Min, H.Du, X.Jia, C.X.H, S.C.H.Huang and W.Wu: Improving Construction for Connected Dominating Set with Steiner Tree in Wireless Sensor Networks. Journal of Global Optimizatio, **35** (2006) 111–119
13. A.Moss and Y.Rabani: Approximation Algorithms for Constrained Node Weighted Steiner Tree Problems. STOC,(2001)
14. G.Robins and A.Zelikovski: Improved Steiner Tree Approximation in Graphs. Proc. of 11th. ACM-SIAM Symposium on Discrete. Algorithms (2000) 770–779
15. A.Segev: The node-weighted steiner tree problem. Networks, **17** (1987) 1–17
16. M.L.Shore, L.R.Foulds and R.B.Gibbons: An algorithm for the Steiner problem in graphs. Networks, **12** (1982) 323–333
17. A.Zelikovsky: An 11/6 approximation algorithm for the network Steiner problem. Algorithmica, **9** (1993) 463–470