Approximating Steiner Trees and Forests with Minimum Number of Steiner Points

Nachshon Cohen and Zeev Nutov

The Open University of Israel,
nachshonc@gmail.com, nutov@openu.ac.il

Abstract. Let R be a finite set of terminals in a metric space (M, d). We consider finding a minimum size set $S \subseteq M$ of additional points such that the unit-disc graph $G[R \cup S]$ of $R \cup S$ satisfies some connectivity properties. In the Steiner Tree with Minimum Number of Steiner Points (ST-MSP) problem $G[R \cup S]$ should be connected. In the more general Steiner Forest with Minimum Number of Steiner Points (SF-MSP) problem we are given a set $D \subseteq R \times R$ of demand pairs and $G[R \cup S]$ should contains a uv-path for every $uv \in D$. Let Δ be the maximum number of points in a unit ball such that the distance between any two of them is larger than 1. It is known that $\Delta = 5$ in $\mathbb{R}^2$. The previous known approximation ratio for ST-MSP was $\lfloor(\Delta+1)/2\rfloor + 1 + \epsilon$ in an arbitrary normed space [15], and $2.5 + \epsilon$ in the Euclidean space $\mathbb{R}^2$ [5]. Our approximation ratio for ST-MSP is $1 + \ln(\Delta - 1) + \epsilon$ in an arbitrary normed space, which in $\mathbb{R}^2$ reduces to $1 + \ln 4 + \epsilon < 2.3863 + \epsilon$. For SF-MSP we give a simple Δ-approximation algorithm, improving the folklore ratio $2(\Delta - 1)$. Finally, we generalize and simplify the Δ-approximation of Calinescu [3] for the 2-Connectivity-MSP problem, where $G[R \cup S]$ should be 2-connected.

1 Introduction

In the Survivable Network problem we are given a graph $G = (V, E)$ with edge-costs (or node-costs) and a set R of terminals, and seek a minimum-cost subgraph H of G that satisfies some prescribed connectivity requirements between the terminals. A fundamental problem of this type is the Steiner Tree problem, where every pair of terminals should be connected in H. In the Steiner Forest problem, we are given a set D of *demand pairs* from R, and H should contains a uv-path for every $uv \in D$. Steiner Tree is a particular case of Steiner Forest, when the graph (D, R) formed by the demand pairs is connected; if also $R = V$, then we get the Minimum Spanning Tree problem. In the k-Connectivity problem, R should be k-connected in H, namely, H should contain k internally-disjoint paths between any pair of nodes in R; if also $R = V$, then we get the k-Connected Subgraph problem. Note that for $k = 1, 2$, whenever only inclusion minimal solution graphs H that contain R are considered, the condition "R is k-connected in H" is equivalent to the condition "H is k-connected"; this is not so for $k \geq 3$.

In wireless networks, the range and the location of the transmitters determines the resulting communication network. We consider adding a minimum number of transmitters such that the communication network satisfies some connectivity properties. If the range of the transmitters is fixed, our goal is to add a minimum number of transmitters, and we get the following type of problems.

Definition 1. *Let (M, d) be a metric space and let $V \subseteq M$. The* unit-disk graph *of V, denoted by $G[V]$, has node set V and edge set $\{uv : u, v \in V, d(u, v) \leq 1\}$.*

In Survivable Network with Minimum Number of Steiner Points (SN-MSP) problems we are given a set R of terminals in a metric space (M, d), and seek a minimum size set $S \subseteq M$ of additional points such that $G[R \cup S]$ satisfies some prescribed connectivity requirements between the terminals. In this setting, Steiner Tree is transformed into the following problem.

Steiner Tree with Minimum Number of Steiner Points (ST-MSP)
Instance: A finite set $R \subseteq M$ of terminals in a metric space (M, d).
Objective: Find a minimum size set $S \subseteq M$ of additional points such that $G[R \cup S]$ is connected.

In the Steiner Forest with Minimum Number of Steiner Points (SF-MSP) problem, we are given a set D of demand pairs from R, and $G[R \cup S]$ should contains a uv-path for every $uv \in D$. In the k-Connectivity-MSP $G[R \cup S]$ should contain k internally-disjoint uv-paths for any $u, v \in R$. Note that 1-Connectivity-MSP is the ST-MSP problem, while 2-Connectivity-MSP is equivalent to the problem of finding a minimum size $S \subseteq M$ such that $G[R \cup S]$ is 2-connected.

As in previous work, we will assume that our metric space is induced by some normed space, allow to place several points at the same location, and assume that the maximum distance between terminals is polynomial in their number.

The Steiner Tree problem was studied extensively (c.f. [17, 18, 2] and the references therein) and the currently best approximation ratio for it is $\ln 4 + \epsilon$ [2]. Steiner Forest and 2-Connectivity admit ratio 2, c.f. [8] and [12].

We survey some literature on SN-MSP problems. ST-MSP is NP-hard even in $\mathbb{R}^2$, and arises in various wireless network design problems, c.f. [1, 3–5, 10, 11, 14, 15] for only a sample of papers in the area, where it is studied both in $\mathbb{R}^2$ and in general metric spaces. In the latter case, the approximation ratio is usually expressed in terms of the following parameter. Let Δ be the maximum number of "independent" points in the unit ball, such that the distance between any two of them is larger than 1. It is known [16] that Δ equals the maximum degree of a minimum-degree Minimum Spanning Tree in the normed space. For Euclidean distances we have $\Delta = 5$ in $\mathbb{R}^2$ and $\Delta = 11$ in $\mathbb{R}^3$, and in $\mathbb{R}^\ell$ Δ is at most the Hadwiger number [16]; hence $\Delta \leq 2^{0.401\ell(1+o(1))}$, by [9].

In finite metric spaces, ST-MSP is equivalent to the variant of the Node-Weighted Steiner Tree problem when terminals have costs 0 and other nodes have cost 1. Klein and Ravi [13] proved that this variant is Set-Cover hard to approximate, and gave an $O(\ln |R|)$-approximation algorithm for general weights.

Hence up to constants, even for finite metric spaces, the ratio $O(\ln |R|)$ of [13] is the best possible unless P=NP. Note however, that this does not exclude constant ratios for metric/normed spaces with small Δ, e.g., $\Delta = 5$ in $\mathbb{R}^2$.

Most algorithms for SN-MSP problems applied the following reduction method, by solving the corresponding Survivable Network instance obtained as follows.

Definition 2. *Given a finite set R of points in a metric space (M, d) and an integer $k \geq 1$, the (multi)graph G_R has node set R and k parallel edges between every pair of nodes. The costs of the k edges between u, v are defined as follows. Let $\hat{d}_{uv} = \max\{\lceil d(u, v) \rceil - 1, 0\}$. If $\hat{d}_{uv} > 0$, then all the k edges have cost $\hat{d}_{uv}$. If $\hat{d}_{uv} = 0$, then one edge has cost 0 and the others have cost 1.*

It is easy to see that any solution of cost C to the corresponding Survivable Network instance defines a solution S of size C to the original SN-MSP instance, where every node in S has degree exactly 2; such a solution is called a *bead solution*. Conversely, any bead solution S can be converted into a solution to the Survivable Network instance (in a normed space) of cost at most $|S|$ (c.f. [10, 3]). Due to this bijective correspondence, we simply define a bead solution as a solution to the corresponding Survivable Network instance, and denote the optimal value of a bead solution to an instance I by $\tau = \tau(I)$. If the Survivable Network instance admits a ρ-approximation algorithm, and if for the given SN-MSP instance there exists a bead solution S of size $\leq \alpha \mathsf{opt}$, then we get a $\rho\alpha$-approximation algorithm for the SN-MSP instance. Equivalently, for a class $\mathcal{I}$ of SN-MSP instances, define a parameter α by $\alpha = \alpha(\mathcal{I}) = \sup_{I \in \mathcal{I}} \frac{\tau(I)}{\mathsf{opt}(I)}$. Then approximation ratio ρ for Survivable Network instances that correspond to class $\mathcal{I}$ implies approximation ratio $\alpha\rho$ for SN-MSP instances in class $\mathcal{I}$.

Măndoiu and Zelikovsky [14] showed that $\alpha = \Delta - 1$ for ST-MSP. Since the SN instance that corresponds to ST-MSP is the MST problem that can be solved in polynomial time, this gives a $(\Delta - 1)$-approximation algorithm for ST-MSP.

A common method to attack various Steiner Tree problems is by a reduction to the Minimum Connected Spanning Subhypergraph problem. This method was initiated by Zelikovsky [17], and improved in a long series of papers culminating in the paper of Byrka et al. [2]. For ST-MSP in $\mathbb{R}^2$, Chen and Du [5] applied it to get the currently best known ratio $2.5 + \epsilon$. In arbitrary normed spaces, the ratio $\Delta - 1$ of [14] was improved to $\lfloor (\Delta + 1)/2 \rfloor + 1 + \epsilon$ in [15] also using the same method. In this work we use the so called "Relative Greedy Heuristic" due to Zelikovsky [18] , and obtain the following result.

Theorem 1. ST-MSP *with constant Δ admits an approximation scheme with ratio $1 + \ln(\Delta - 1) + \epsilon$. In particular, in $\mathbb{R}^2$ the ratio is $1 + \ln 4 + \epsilon < 2.3863 + \epsilon$.*

We note that previous works, as well as Theorem 1, assume that ST-MSP instances with a constant number of terminals can be solved in polynomial time. This condition holds in $\mathbb{R}^2$ if the maximum distance between terminals is polynomial in the number of terminals, see [4, Lemma 11] and the discussion there. If such instances can be approximated within a factor of β, then an additional factor of β is invoked in the ratio of Theorem 1.

In the proof of Theorem 1, most of our effort is spent on bounding the parameter α_k (so called "Steiner ratio") for ST-MSP (see Theorem 3), which is the loss in the approximation ratio as a result of reducing the problem to the Minimum Connected Spanning Subhypergraph problem in a hypergraph of rank k (see Theorem 3). Bounds on α_k have also been computed for several other "Steiner problems", including ST-MSP. However, none of the previous papers succeeded to reveal the somewhat irregular behavior of α_k in the case of ST-MSP. The key difficulty lies in the fact that α_k has a "good" behavior only when $k = \Omega(\Delta)$. This is the reason why in Theorem 1 we require that Δ is a constant, to ensure that our algorithm can be implemented in polynomial time.

We now consider SF-MSP and k-Connectivity-MSP problems. The result in [14] implies that $\alpha = \Delta - 1$ for SF-MSP. Combined with the known 2-approximation for Steiner Forest, this gives ratio $2(\Delta - 1)$ for SF-MSP. Kashyap, Khuller, and Shayman [11] considered 2-Connectivity-MSP. Their algorithm constructs a 2-Connectivity instance as in Definition 2 and then converts its solution into a bead solution to the 2-Connectivity-MSP instance. Although they analyzed a performance of specific 2-approximation algorithm – the algorithm of Khuller and Raghavachari [12] for 2-Connectivity, they essentially proved that $\alpha = \Delta$ in this case, which implies ratio 2Δ. The analysis of this specific algorithm was recently improved by Calinescu [3], showing that its tight performance is Δ.

We now discuss k-Connectivity-MSP with $k \geq 3$. Bredin et al. [1] considered in $\mathbb{R}^2$ a related problem of adding a minimum size S such that $G[R \cup S]$ is k-connected (note that in k-Connectivity-MSP we require k-connectivity only between terminals). For this problem in $\mathbb{R}^2$, they gave an $O(k^5)$-approximation algorithm, but essentially they implicitly proved that for this class of problems $\alpha = O(\Delta k^3)$. Recently, it was shown in [10] that $\alpha = \Theta(\Delta k^2)$ for a much more general class of Survivable Network problems in any normed space.

Let $\tau^* = \tau^*(I)$ denote the optimal value of a *fractional* bead solution of an SN-MSP instance I, namely, τ^* is the optimum value of a standard cut-LP relaxation for the corresponding Survivable Network instance (see Section 4). We observe that if the algorithm we use for the Survivable Network instance computes a solution of cost at most $\rho\tau^*$, then the relevant parameter is the following.

Definition 3. *For a class $\mathcal{I}$ of SN-MSP instances, let $\alpha^* = \alpha^*(\mathcal{I}) = \sup_{I \in \mathcal{I}} \frac{\tau^*(I)}{\mathsf{opt}(I)}$.*

Theorem 2. *For any feasible solution S to SF-MSP there exists a half-integral bead solution of value at most $\Delta|S|/2$; thus $\alpha^* = \Delta/2$ for SF-MSP. Consequently, if Steiner Forest admits a polynomial time algorithm that computes a solution of cost at most $\rho\tau^*$, then SF-MSP admits approximation ratio $\rho \cdot \Delta/2$; thus SF-MSP admits a Δ-approximation algorithm. The same holds for 2-Connectivity-MSP.*

The idea behind Theorem 2 is as follows. From previous work [16, 3] we get that for any solution S, $G[R \cup S]$ contains a solution G in which the nodes in S have degree at most Δ. Our main innovation is comparing the optimal solution with a *fractional* (in fact, half-integral) bead solution, rather than an actual bead solution. For 2-Connectivity-MSP this idea appeared implicitly in the paper of Calinescu [3], but our explicit approach is much simpler and more general.

2 Proof of Theorem 1

We consider a generic problem defined in [15], that includes both ST-MSP and the classic Steiner Tree problem.

Generic Steiner Tree
Instance: A (possibly infinite) graph $G = (V, E)$, a finite set $R \subseteq V$ of terminals, and a monotone subadditive cost function c on subgraphs of G.
Objective: Find a minimum-cost connected finite subtree T of G containing R.

Definition 4. *For an instance of* Generic Steiner Tree *and* $2 \leq k \leq |R|$, *the hypergraph* $\mathcal{H}_k = (R, \mathcal{E}_k)$ *has hyperedge set* $\mathcal{E}_k = \{A \subseteq R : 2 \leq |A| \leq k\}$. *The cost* $c^*(A)$ *of* $A \in \mathcal{E}_k$ *is the cost of an optimal solution* T_A *to the* Generic Steiner Tree *instance with terminal set* A.

The construction in Definition 4 converts a Generic Steiner Tree instance into a Minimum Connected Spanning Subhypergraph instance in a hypergraph $\mathcal{H}_k$ of rank k. Any solution of cost C to this instance correspond to a solution of value at most C to Generic Steiner Tree instance, by the subadditivity and monotonicity of the cost function in the Generic Steiner Tree problem. This reduction invokes a fee in the approximation ratio, given in the following definition.

Definition 5. *Given an instance* I *of* Generic Steiner Tree *let* $\tau_k(I)$ *denote the minimum cost of a connected spanning sub-hypergraph of* $\mathcal{H}_k$. *The* k-*ratio for a class* $\mathcal{I}$ *of* Generic Steiner Tree *instances is defined by* $\alpha_k = \sup_{I \in \mathcal{I}} \frac{\tau_k(I)}{\mathsf{opt}(I)}$.

Note that for $\mathcal{I}$ being the class of ST-MSP instances, α_2 is the parameter α defined in the introduction, and that by [14] we have $\alpha_2 = \alpha = \Delta - 1$. We have $\alpha_k = 1$ for instances with $|R| = k$, and in general α_k is monotone decreasing and approaching 1 when k becomes larger.

Particular cases of the following statement can be found in several papers. We failed to find the general version in the literature, and thus provide a proof for completeness of exposition.

Lemma 1. *There exists a polynomial time algorithm that given a hypergraph* $\mathcal{H} = (R, \mathcal{E})$ *with hyper-edge cost* $\{c(A) : A \in \mathcal{E}\}$ *and a spanning tree* T^* *of (edges of size 2 of)* $\mathcal{H}$ *computes a spanning connected sub-hypergraph* $\mathcal{T}$ *of* $\mathcal{H}$ *of cost at most* $\tau \left(1 + \ln \frac{c(T^*)}{\tau}\right)$, *where* τ *is the minimum-cost of a connected spanning sub-hypergraph of* $\mathcal{H}$.

Proof. Given a tree $T = (R, F)$ let us say that $A \subseteq R$ *overlaps* $F' \subseteq F$ if the graph obtained from $T \setminus F'$ by shrinking A into a single node is a tree. Given edge cost $\{c(e) : e \in F\}$ let $F(A)$ be a maximum cost edge set overlapped by A.

Note that $F \setminus F(A)$ is an edge set of a minimum cost spanning tree in the graph obtained from T by shrinking A into a single node; hence for given A, $F(A)$ can be computed in polynomial time. Consider the following algorithm.

Algorithm 1: LOCAL REPLACEMENT ALGORITHM

---

1 **Input:** A hypergraph $\mathcal{H} = (R, \mathcal{E})$ with hyper-edge cost $\{c(A) : A \in \mathcal{E}\}$,
   and a spanning tree $T^* = (R, F^*)$ of (edges of size 2 of) $\mathcal{H}$.

2 **Initialization:** $\mathcal{J} \leftarrow \emptyset$, $F \leftarrow F^*$, $T \leftarrow (R, F)$.

3 **while** $c(F) > 0$ **do**

4   Find $A \in \mathcal{E}$ with $\frac{c(F(A))}{c(A)}$ maximum.

5   **if** $c(F(A)) > c(A)$ **then**

6   Update $T, \mathcal{H}$: remove $F(A)$ and shrink A into a single node.
            $F \leftarrow F \setminus F(A)$ and $\mathcal{J} \leftarrow \mathcal{J} \cup \{A\}$.

7   **Else STOP** and **return** $\mathcal{T} = (R, F \cup \mathcal{J})$.

8 **return** $\mathcal{T} = (R, F \cup \mathcal{J})$

The following statement appeared in [17] (see also [2]); we provide a proof for completeness of exposition.

Claim. Let $T = (R, F)$ be a tree with edge costs $\{c(e) : e \in F\}$ and let $(R, \mathcal{E})$ be a connected hypergraph. Then $\sum_{A \in \mathcal{E}} c(F(A)) \geq c(F)$. Thus there exists $A \in \mathcal{E}$ such that

$$\frac{c(F(A))}{c(A)} \geq \frac{c(F)}{c(\mathcal{E})} \ .$$

Proof. For a node $v \in A$, let C_v be the connected component in $T \setminus F(A)$ that contains v. For an edge $e \in F(A)$ that connects two components C_u, C_v, let $y(e) = uv$ be the replacement edge of e, of cost $c(y(e)) = c(e)$. The graph $T \cup \{y(e)\}$ contains a single cycle and $y(e)$ is the heaviest edge in this cycle, since otherwise $F(A)$ is not minimal. For a hyperedge $A \in \mathcal{E}$ let $y(A) = \cup_{e \in F(A)} y(e)$ be the replacement set of A, and let $y(\mathcal{E}) = \cup_{A \in \mathcal{E}} y(A)$. It is easy to see that $y(A)$ spans A, and $y(\mathcal{E})$ spans R. Consider a MST on $T \cup y(\mathcal{E})$. By the cycle property of a MST, no edge from $y(\mathcal{E})$ would participate in that MST, so $c(T) \leq c(y(\mathcal{E}))$. Finally, $c(y(\mathcal{E})) = \sum_{A \in \mathcal{E}} y(A) = \sum_{A \in \mathcal{E}} c(F(A))$, and the claim follows. $\qquad\square$

At every iteration $|F|$ decreases by at least 1, hence the algorithm runs in polynomial time, and clearly it computes a feasible solution. We prove the approximation ratio. Let F_i and $\mathcal{J}_i$ be the set stored in F and $\mathcal{J}$, respectively, at the beginning of iteration $i+1$, and let A_i be the hyperedge picked at iteration i. Denote $f_i = c(F_i)$ and $s_i = c(A_i)$, and recall that τ denotes the minimum cost of a connected spanning sub-hypergraph of $\mathcal{H}$. At iteration i we remove $F_{i-1}(A_i)$ from F_{i-1} after verifying that $c(F_{i-1}(A_i)) > c(A_i) = s_i$. Hence

$$f_i \leq f_{i-1} - \max\{c(F_{i-1}(A_i)), c(A_i)\} = f_{i-1} - s_i \cdot \max\left\{\frac{c(F_{i-1}(A_i))}{c(A_i)}, 1\right\}$$

By the claim above, $\frac{c(F_{i-1}(A_i))}{c(A_i)} \geq \frac{f_{i-1}}{\tau}$. Thus we have

$$f_i \leq f_{i-1} - s_i \cdot \max\{f_{i-1}/\tau, 1\} \ . \tag{1}$$

The algorithm stops if either $c(F_q) = 0$ or $c(F(A)) \leq c(A)$ at iteration $q + 1$. In the latter case, $1 \geq c(F_q)/\tau$ follows by the claim above. In both cases, we have that there exists an index q such that $f_{q-1} > \tau \geq f_q$ holds. Now we use the following statement from [6].

Claim. Let $\tau > 0$ and $f_0, \ldots, f_q$ and $s_1, \ldots, s_q$ be sequences of positive reals satisfying $f_0 > \tau \geq f_q$, such that (1) holds. Then $f_q + \sum_{i=1}^{q} s_i \leq \tau(1 + \ln(f_0/\tau))$.

Let q be an index such that $f_{q-1} > \tau \geq f_q$ holds. We may assume that $f_0 = c(F^*) > \tau > 0$. Note that $c(\mathcal{J}_q) = \sum_{i=1}^{q} s_i$ and that $c(F_i) + c(\mathcal{J}_i) \leq c(F_{i-1}) + c(\mathcal{J}_{i-1})$ for any i. Hence from the claim above we conclude that

$$c(\mathcal{T}) \leq c(F_q) + c(\mathcal{J}_q) = f_q + \sum_{i=1}^{q} s_i \leq \tau(1 + \ln(f_0/\tau)) = \tau \left(1 + \ln \frac{c(T^*)}{\tau} \right) .$$

$\square$

Corollary 1. *For any constant k,* Generic Steiner Tree *admits an approximation ratio $\alpha_k (1 + \ln \alpha_2)$, provided that for any $A \in \mathcal{E}_k$, the instance with the terminal set A can be solved in polynomial time.*

Proof. By the assumptions, the hypergraph $\mathcal{H}_k$, and the costs $c^*(A)$ with the corresponding trees T_A for $A \in \mathcal{E}_k$, can be computed in polynomial time. We can also compute in polynomial time an optimal spanning tree T^* in $\mathcal{H}_2$; note that $c(T^*) \leq \alpha_2 \cdot \mathsf{opt}$. Then we apply the algorithm in Lemma 1 to compute a sub-hypergraph $\mathcal{T}$ of $\mathcal{H}_k$ of c^*-cost at most $\tau \left(1 + \ln \frac{c(T^*)}{\tau} \right)$. Let opt denote the optimal solution value for the Generic Steiner Tree instance. Note that $\mathsf{opt} \leq \tau \leq \alpha_k \mathsf{opt}$. Let $T = \cup_{A \in \mathcal{T}} T_A$. Since $\mathcal{T}$ is a connected hypergraph, T is a feasible solution to the Generic Steiner Tree instance. We have $c(T) \leq \sum_{A \in \mathcal{T}} c(T_A) = c^*(\mathcal{T})$, by the monotonicity and the subadditivity of the c-costs. Thus we have:

$$c(T) \leq c^*(\mathcal{T}) \leq \tau \left(1 + \ln \frac{c(T^*)}{\tau} \right) = \tau \left(1 + \ln \frac{c(T^*)/\mathsf{opt}}{\tau/\mathsf{opt}} \right) \leq \alpha_k \mathsf{opt} \left(1 + \ln \alpha_2 \right) .$$

$\square$

Du and Zhang [7] showed that for the classic Steiner Tree problem, $\alpha_k \leq 1 + 1/\lfloor \lg k \rfloor$, where $\lg k = \log_2 k$. In Section 3 we prove the following.

Theorem 3. *For* ST-MSP*, $\alpha_k \leq 1 + \frac{2}{\lfloor \lg \lfloor k/(\Delta-1) \rfloor \rfloor}$ for any integer $k \geq 2\Delta - 2$.*

Note that for an instance I of ST-MSP with Δ independent points on the unit ball we have $\tau_k(I) = \frac{\Delta}{k}$ and $\mathsf{opt}(I) = 1$, which implies $\alpha_k \geq \frac{\tau_k(I)}{\mathsf{opt}(I)} = \frac{\Delta}{k}$. Hence $k > \Delta/2$ is necessary if we want $\alpha_k < 2$.

From Corollary 1 and Theorem 3 we conclude that for any constant $k \geq 2\Delta - 2$, it is possible to compute in polynomial time a solution to an ST-MSP instance of size at most $\alpha_k (1 + \ln(\Delta - 1)) \mathsf{opt}$, where α_k is as in Theorem 3. For the metric space $\mathbb{R}^2$, and given a constant $\epsilon > 0$ let $k = 2^{\Theta(1/\epsilon)}$. Then by Theorem 3, $\alpha_k \leq 1 + \epsilon/(1 + \ln 4)$, and the approximation ratio of our algorithm is $1 + \ln 4 + \epsilon$. This completes the proof of Theorem 1.

3 Proof of Theorem 3

For a tree $T = (V, F)$ and $A \subseteq V$ let $T_A = (V_A, F_A)$ be the inclusion minimal subtree of T that contains A. To prove Theorem 3 we prove the following.

Lemma 2. *Let $T = (V, F)$ be a tree of maximum degree $\Delta \geq 2$, let $R \subseteq V$, and let $S = V \setminus R$. Then for any integer $k \geq 2\Delta - 2$ there exists a connected hypergraph $\mathcal{H} = (R, \mathcal{E})$ of rank $\leq k$ such that $\sum_{A \in \mathcal{E}} |V_A \cap S| \leq \left(1 + \frac{2}{\lfloor \lg \lfloor k/(\Delta - 1) \rfloor \rfloor}\right) |S|$.*

To prove Lemma 2 we prove the following.

Lemma 3. *Let $T = (V, F)$ be a tree with edge costs $\{c(e) \geq 1 : e \in F\}$ and let $R \subseteq V$. Then for any integer $p \geq 2$ there exists a connected hypergraph $\mathcal{H} = (R, \mathcal{E})$ of rank $\leq p$ such that $\sum_{A \in \mathcal{E}} c(F_A) + |\mathcal{E}| - 1 \leq \left(1 + \frac{2}{\lfloor \lg p \rfloor}\right) c(T)$.*

Lemma 3 will be proved later. Now we show that it implies Lemma 2. An *R-component* of T is a maximal inclusion subtree of T such that all its leaves are in R but none of its internal nodes is in R. It is easy to see that it is sufficient to prove Lemma 2 for each R-component separately, hence we may assume that R is the set of leaves of T.

If T is a star, then since $k \geq 2\Delta - 2 \geq \Delta$, we let $\mathcal{E}$ to consist of a single hyperedge $A = R$. Then $|V_A \cap S| = 1 = |S|$, and Lemma 2 holds in this case.

Henceforth assume that T is not a star. For $v \in S$ let $R(v)$ be the set of neighbors of v in R, and note that $|R(v)| \leq \Delta - 1$. Let $T' = (V', F') = T \setminus R$ and let $R' = \{v \in S : R(v) \neq \emptyset\}$. Applying Lemma 3 on T' with unit edge-costs and R', we obtain that for $p = \lfloor k/(\Delta - 1) \rfloor$ there exists a connected hypergraph $\mathcal{H}' = (R', \mathcal{E}')$ of rank $\leq p$ such that $\sum_{A' \in \mathcal{E}'} |F'_{A'}| + |\mathcal{E}'| - 1 \leq \left(1 + \frac{2}{\lfloor \lg p \rfloor}\right) |F'|$. Note that $|F'| = |V'| - 1$ and that $|V'_{A'}| = |F'_{A'}| - 1$ for every $A' \in \mathcal{E}'$. Hence

$$\sum_{A' \in \mathcal{E}'} |V'_{A'}| - 1 \leq \left(1 + \frac{2}{\lfloor \lg p \rfloor}\right) (|V'| - 1) \leq \left(1 + \frac{2}{\lfloor \lg p \rfloor}\right) |V'| - 1 \ .$$

For $A' \in \mathcal{E}'$ let $A = \cup_{v \in A'} R(v)$; then $|A| \leq p(\Delta - 1)$. Let $\mathcal{E} = \{A : A' \in \mathcal{E}'\}$. Then $\mathcal{H} = (R, \mathcal{E})$ is a connected hypergraph of rank $\leq p(\Delta - 1) \leq k$, and

$$\sum_{A \in \mathcal{E}} |V_A \cap S| = \sum_{A' \in \mathcal{E}'} |V'_{A'}| \leq \left(1 + \frac{2}{\lfloor \lg p \rfloor}\right) |V'| = \left(1 + \frac{2}{\lfloor \lg \lfloor k/(\Delta - 1) \rfloor \rfloor}\right) |S| \ .$$

In the rest of this section we prove Lemma 3, by extending the proof of Du and Zhang [7] of an existence of a connected hypergraph $\mathcal{H} = (R, \mathcal{E})$ of rank $\leq p$ such that $\sum_{A \in \mathcal{E}} c(F_A) \leq \left(1 + \frac{1}{\lfloor \lg p \rfloor}\right) c(T)$. We have an extra term of $|\mathcal{E}| - 1$, and we show that this term can be bounded by $\frac{c(T)}{\lfloor \lg p \rfloor}$.

We start by transforming the tree into a (rooted) binary tree T with edge-costs, which node set is partitioned into a set R of terminals and a set S of non-terminals, such that the following properties hold:

(A) R is the set of leaves of T.

(B) The cost of any edge of T is either 0 or is at least 1, and among the edges that connect a node in $S = V \setminus R$ to its children, at most one has cost 0.

(C) T is a full binary tree, namely, every $v \in S$ has exactly 2 children.

To obtain such a tree, root T at an arbitrary non-leaf node $\hat{s} \in S = V \setminus R$, and apply the following standard reductions.

1. While T has a leaf in S, remove this leaf; hence every leaf of T is in R. Then, for every $v \in R$ that is not a leaf, add to T a new node v' and an edge vv' of cost 0, add v' to R, and move v from R to S. After this step, properties (A) and (B) hold.

2. While there is $v \in S$ that has one child, replace the path P of length 2 that contains v by a single edge of cost $c(P)$, and exclude v from S. After this step, every $v \in S$ has at least 2 children.

3. While there is $v \in S$ that has more than 2 children, do the following. Let u be a child of v such that the cost of the edge vu is at least 1. Add a new node v' and the edge vv' of cost 0, and for every child of u' of v distinct from u replace the edge vu' by the edge vu'. After this step, all the three properties (A), (B), and (C) hold.

Consequently, to prove Lemma 3, it is sufficient to prove the following.

Lemma 4. *Let $T = (V, F)$ be a tree with edge costs $c(e)$ and leaf set R, satisfying (A),(B),(C), Then for any integer $p \geq 2$ there exists a connected hypergraph $\mathcal{H} = (R, \mathcal{E})$ of rank $\leq p$ such that $\sum_{A \in \mathcal{E}} c(F_A) + |\mathcal{E}| - 1 \leq \left(1 + \frac{2}{\lceil \lg p \rceil}\right) c(T)$.*

Let $T = (V, F)$ be a rooted tree with leaf set R and let $S = V \setminus R$. For two nodes u, v of T let $P_T(u, v)$ denote the unique path in T between u and v.

Definition 6. *We say that T is* proper *if every node in S has at least 2 children. We say that a mapping $f : S \to R$ is T-proper if:*

(i) For every $u \in S$, $f(u)$ is a descendant of u.

(ii) The paths $\{P_T(u, f(u)) : u \in S\}$ are edge disjoint.

Given a subtree T' of T with leaf set L' and a proper mapping f, the set of terminal connecting paths of T' is $\{P_T(u, f(u)) : u \in L' \setminus R\}$. Let $\hat{T}'$ denote the tree obtained from T' by adding to T' all the terminal connecting paths.

Du and Zhang [7] proved that any proper tree T admits a proper mapping. We prove the following.

Lemma 5. *Let $T = (V, F)$ be a proper tree and let $F_1 \subseteq F$ be such that any $u \in S$ has a child connected to u by an edge in F_1. Then there exists a T-proper mapping f such that for every $u \in S$, the path $P_T(u, f(u))$ contains at least one edge in F_1.*

Proof. The proof is by induction on the height of the tree. Let T be a tree as in the lemma of height h. If $h = 1$, then T has one internal node (the root), say u,

and we set $f(u)$ to be the node that is connected to u by an edge in F_1. Suppose that the statement is true for trees with height $h - 1 \geq 1$, and we prove it for trees of height h. Let T' be obtained from T by removing nodes of distance h from the root. By the induction hypothesis, for T' there exists a mapping f' as in the lemma. Let u be an internal node of T. Consider two cases.

Suppose that u is an internal node of T'. If $f'(u)$ is a leaf of T, then define $f(u) = f'(u)$. If $f'(u)$ is an internal of T, then $f'(u)$ is a leaf of T', and all its children in T are leaves. Then we set $f(u)$ to be a child of $f'(u)$ that is connected to $f'(u)$ by an edge in F_1

Suppose that u is a leaf of T'. Then the children of u in T are leaves, and we set $f(u)$ to be a child of u that is connected to u by an edge in F_1.

It is easy to verify that the obtained mapping f meets the requirements. □

The following statement is implicitly proved by Du and Zhang [7].

Lemma 6 ([7]). *Let T be a proper binary tree with non-negative edge costs and let f be a proper mapping. Then for any integer $p \geq 2$ there exists an edge-disjoint partition $\mathcal{T}$ of T into subtrees such that the following holds:*

(i) *The hypergraph with node set R and hyperedge set $\mathcal{E} = \{\hat{T}' \cap R : T' \in \mathcal{T}\}$ is connected and has rank at most p.*
(ii) *The total number of terminal connecting paths of all subtrees in $\mathcal{T}$ is at least $|\mathcal{T}| - 1$, and their total cost is at most $c(T)/\lfloor \lg p \rfloor$.*

We now finish the proof of Lemma 4, and thus also of Lemma 3. Let $F_1 = \{e \in F : c(e) \geq 1\}$ and let f be a proper mapping as in Lemma 5. Let $\mathcal{T}$ be a partition as in Lemma 6, and let $\mathcal{E}$ be as in Lemma 6(i), so the hypergraph $\mathcal{H} = (R, \mathcal{E})$ is connected and has rank at most p. By Lemma 6(ii), the total number of terminal connecting paths of all subtrees is at least $|\mathcal{T}| - 1 = |\mathcal{E}| - 1$, while their total cost is at most $c(T)/\lfloor \lg p \rfloor$. Every terminal connecting path contains an edge from F_1, by Lemma 5, and thus has cost at least 1. Hence the total cost of all terminal connecting paths is at least $|\mathcal{E}| - 1$. Consequently

$$|\mathcal{E}| - 1 \leq \frac{c(T)}{\lfloor \lg p \rfloor} \ .$$

For $A = \hat{T}' \cap R \in \mathcal{E}$ let $P(T')$ denote the union of the edge sets of the terminal connecting paths of T'. Then $c(F_A) \leq c(\hat{T}') = c(T) + c(P(T'))$, hence

$$\sum_{A \in \mathcal{E}} c(F_A) \leq \sum_{T' \in \mathcal{T}} [c(T') + c(P(T'))] = \sum_{T' \in \mathcal{T}} c(T') + \sum_{T' \in \mathcal{T}} c(P(T')) \leq c(T) + \frac{c(T)}{\lfloor \lg p \rfloor} \ .$$

Summarizing, we have

$$\sum_{A \in \mathcal{E}} c(F_A) + |\mathcal{E}| - 1 \leq c(T) + \frac{c(T)}{\lfloor \lg p \rfloor} + \frac{c(T)}{\lfloor \lg p \rfloor} = \left(1 + \frac{2}{\lfloor \lg p \rfloor}\right) c(T) \ .$$

The proof of Lemma 4, and thus also of Lemma 3 and Theorem 3 is now complete.

4 Proof-sketch of Theorem 2

Definition 7. *For a subset C of nodes of a (multi-)graph $G = (V, E)$ let us use the following notation: $\Gamma_G(C)$ is the set of neighbors of C in G; $\delta_G(C) = \delta_E(C)$ is the set of edges in E with exactly one endnode in C; $E(C)$ is the set of edges in E with both endnodes in C. Given $R \subseteq V$, an R-component of G is a subgraph of G with node set $C \cup \Gamma_G(C)$ and edge set $E(C) \cup \delta_G(C)$, where C is a connected component of $G \setminus R$.*

The following important property of feasible solutions was proved for SF-MSP by Robins and Salowe [16] and for 2-Connectivity-MSP by Calinescu [3].

Lemma 7 ([16],[3]). *Let S be an inclusion minimal feasible solution for an instance of SF-MSP or 2-Connectivity-MSP. Then $G[R \cup S]$ contains a subgraph G that satisfies the requirements such that every R-component of G is a tree and such that $\deg_G(v) \leq \Delta$ for every $v \in S$.*

Lemma 8. *Suppose that $G[R \cup S]$ contains a tree T with leaf set R. Let S' be obtained from S by replacing each $v \in S$ by $\deg_T(v)$ copies of v. Then $G[R \cup S']$ contains a simple cycle on $R \cup S'$, called a DFS cycle of T.*

Proof. Traverse the tree T in a DFS order; each time a node $v \in S$ is visited, choose a different copy of v. □

Given a Steiner Forest instance, we say that a set $A \subset V$ is *deficient* if $|A \cap \{u, v\}| = 1$ for some $uv \in D$. It is easy to see that H is a feasible solution to a Steiner Forest instance iff $\delta_H(A) \geq 1$ for every deficient set A. To formulate a similar condition for 2-Connectivity we need a definition of a *biset*, which is an ordered pair of sets $\mathbb{A} = (A, A^+)$ such that $A \subseteq A^+$; $\Gamma(\mathbb{A}) = A^+ \setminus A$ is the *boundary* of $\mathbb{A}$. Let $\delta_E(\mathbb{A})$ denote the set of edges in E with one end in A and the other in $V \setminus A^+$. It is known that H is a feasible solution to 2-Connectivity iff $x(\delta_E(\mathbb{A})) \geq 2 - |\Gamma(\mathbb{A})|$ for every biset $\mathbb{A}$. The cut-LP relaxations for Steiner Forest and 2-Connectivity minimize $\sum_{e \in E} c_e x_e$ over the polytopes Π_{SF} and Π_{2C}, respectively, defined by:

$$\Pi_{\mathsf{SF}} = \{x \in \mathbb{R}^E : x(\delta_E(A)) \geq 1 \ \forall \text{ deficient set } A, x_e \geq 0\}$$
$$\Pi_{\mathsf{2C}} = \{x \in \mathbb{R}^E : x(\delta_E(\mathbb{A})) \geq 2 - |\Gamma(\mathbb{A})| \ \forall \text{ biset } \mathbb{A}, 0 \leq x_e \leq 1\}$$

We will say that a graph with edge capacities x_e is a fractional bead solution for SF-MSP or for 2-Connectivity-MSP, if the characteristic vector of the edge-set of the graph belongs to Π_{SF} or to Π_{2C}, respectively.

Lemma 9. *Let G be as in Lemma 7. Then replacing each connected component C of $G \setminus R$ by a DFS cycle on $\Gamma_G(C)$ of capacity $1/2$ results in a fractional half-integral bead solution for SF-MSP or 2-Connectivity-MSP, respectively.*

Proof. For SF-MSP the statement is obvious, while for 2-Connectivity-MSP, due to space limitation, the proof will be provided in the full version. □

Theorem 2 easily follows from Lemmas 7, 8, and 9.

5 Conclusions

Our main results are a $(1 + \ln(\Delta - 1) + \epsilon)$-approximation scheme for ST-MSP, and a Δ-approximation algorithm for SF-MSP. For ST-MSP in $\mathbb{R}^2$ this improves the ratio $2.5 + \epsilon$ of [5]. For SF-MSP this improves the folklore ratio $2(\Delta - 1)$ that follows from the work of [14]. We believe that the methods presented in this paper will lead to improved approximation algorithms for related problems.

References

1. J. Bredin, E. Demaine, M. Hajiaghayi, and D. Rus. Deploying sensor networks with guaranteed fault tolerance. *IEEE/ACM Trans. Netw.*, 18(1):216–228, 2010.
2. J. Byrka, F. Grandoni, T. Rothvoß, and L. Sanità. Steiner tree approximation via iterative randomized rounding. *J. ACM*, 60(1), 2013.
3. G. Calinescu. Relay placement for two-connectivity. In *Networking*, volume 2, pages 366–377, 2012.
4. D. Chen, D.-Z. Du, X.-D. Hu, G.-H. Lin, L. Wang, and G. Xue. Approximations for Steiner trees with minimum number of steiner points. *Theoretical Comput. Science*, 262(1):83–99, 2001.
5. X. Cheng, D. Du, L. Wang, and B. Xu. Relay sensor placement in wireless sensor networks. *Wireless Networks*, 14(3):347–355, 2008.
6. N. Cohen and Z. Nutov. A $(1 + \ln 2)$-approximation algorithm for minimum-cost 2-edge-connectivity augmentation of trees with constant radius. *Theor. Comput. Sci.*, 67-74:489–490, 2013.
7. D.-Z. Du and Y. Zhang. On better heuristics for steiner minimum trees. *Math. Program.*, 57:193–202, 1992.
8. M. X. Goemans and D. P. Williamson. A general approximation technique for constrained forest problems. *SIAM J. Comput.*, 24(2):296–317, 1995.
9. G. Kabatjansky and V. Levenstein. Bounds for packing of the sphere and in space. *Prob. Information Trans.*, 14:117, 1978.
10. L. Kamma and Z. Nutov. Approximating survivable networks with minimum number of steiner points. *Networks*, 60(4):245–252, 2012.
11. A. Kashyap, S. Khuller, and M. Shayman. Relay placement for fault tolerance in wireless networks in higher dimensions. *Comput. Geometry*, 44:206215, 2011.
12. S. Khuller and B. Raghavachari. Improved approximation algorithms for uniform connectivity problems. *J. Algorithms*, 21(2):434–450, 1996.
13. P. Klein and R. Ravi. A nearly best-possible approximation algorithm for node-weighted steiner trees. *J. Algorithms*, 19(1):104–115, 1995.
14. I. Măndoiu and A. Zelikovsky. A note on the MST heuristic for bounded edge-length Steiner trees with minimum number of Steiner points. *Information Processing Letters*, 75(4):165–167, 2000.
15. Z. Nutov and A. Yaroshevitch. Wireless network design via 3-decompositions. *Information Processing Letters*, 109(19):1136–1140, 2009.
16. G. Robins and J. Salowe. Low-degree minimum spanning trees. *Discrete Comput. Geom.*, 14:151–165, 1995.
17. A. Zelikovsky. An 11/6-approximation algorithm for the network steiner problem. *Algorithmica*, 9:463–470, 1993.
18. A. Zelikovsky. Better approximation bounds for the network and euclidean steiner tree problems. Technical Report CS-96-06, University of Virginia, 1996.