

Truthful Mechanisms for Steiner Tree Problems

Jinshan Zhang¹, Zhengyang Liu^{2*}, Xiaotie Deng³, Jianwei Yin¹

¹ Zhejiang University

² Beijing Institute of Technology

³ Peking University

zhangjinshan@zju.edu.cn, zhengyang@bit.edu.cn, xiaotie@pku.edu.cn, zjuyjw@cs.zju.edu.cn

Abstract

Consider an undirected graph $G = (V, E)$ model for a communication network, where each edge is owned by a selfish agent, who reports the cost for offering the use of her edge. Note that each edge agent may misreport her own cost for the use of the edge for her own benefit. In such a non-cooperative setting, we aim at designing an approximately truthful mechanism for establishing a Steiner tree, a minimum cost tree spanning over all the terminals. We present a truthful-in-expectation mechanism that achieves the approximation ratio $\ln 4 + \epsilon \approx 1.39$, which matches the current best algorithmic ratio for STP.

1 Introduction

Given a graph $G = (V, E)$ and a set of terminal vertices $R \subseteq V$, each edge $e \in E$ is associated by a cost $c(e)$. The Steiner Tree Problem (STP) is to find a minimum cost tree over G that spans all the terminal vertices in R . One can also include extra intermediate vertices and edges to reduce the cost of the induced spanning tree. These extra vertices introduced to decrease the total cost of connection are known as Steiner points or Steiner vertices. The STP is one of the most important and fundamental problems widely studied in the fields of computer science and operations research, with wide practical applications such as the design of VLSI (Very Large Scale Integration), optical and wireless communication systems, as well as transportation and distribution networks (Hwang and Richards 1992).

However, finding the optimal solution in the STP is known to be one of Karp's original 21 NP-complete problems (Karp 1972). That means we cannot hope to solve this problem exactly. For the approximate version, it is APX-hard and can not be approximated within $\frac{96}{95}$ unless $P = NP$ (Bern and Plassmann 1989; Chlebík and Chlebíková 2008). During last decades, a sequence of improved approximation ratios for the problem (from $\frac{11}{6}$ to $1 + \frac{\ln 3}{2} + \epsilon \approx 1.55$) is obtained (Zelikovsky 1993; Karpinski and Zelikovsky 1997; Prömel and Steger 2000; Robins and Zelikovsky 2005). Until recently, by leveraging the LP iterative rounding approach, a $\ln 4 + \epsilon \approx 1.39$ -approximate algorithm

is proposed in (Byrka et al. 2013) and later a deterministic more efficient algorithm with the same approximation ratio is given by (Goemans et al. 2012). Very recently, the same approximation ratio has been achieved by Traub and Zenklusen (2021), from a purely combinatorial point of view.

For any large network, such as a network for the STP, there exists heterogeneous resources like vertices and edges, of which may be held by different owners. These owners intend to perform certain tasks to earn their rewards. We take the message broadcast as an example. From the perspective of network management, it seems that the rewards of the owners can be seen as the price of the service that forwards the messages. Therefore, economically speaking, each owner declares that its true price is desirable. It turns out that in several web applications or internet services, one needs to efficiently calculate the solution of the given optimization problem, where additional restrictions to active agents (through appropriate payments) can be proposed to cooperate within the algorithm. This combined output of the calculation and the payment is often referred to as a *mechanism*.

A realistic applicable scenario based on the above motivation can be seen as a two-player game between the Internet Service Provider (ISP) and Content Provider (CP) (Saltzer, Reed, and Clark 1984; Armstrong 2006; Deng, Feng, and Papadimitriou 2016), where CP would like to build a connection among different sites to users (e.g., establishing a Steiner tree among terminal vertices) through the internet service provided by different ISPs (e.g., different edges belongs to different ISPs). Each ISP charges CP in order to maximize her own utility.

From the above consideration and wide applications of the STP, we consider this problem in a strategic and non-cooperative way. More precisely, each edge in the graph is held by a selfish agent who reports her private cost $c(e)$ to the mechanism designer. The edge holder reports the cost of its edge to maximize his utility (i.e., valuation minus the payment, where the valuation is $-c(e)$). It turns out that in many applications the mechanism designer should provide a mechanism that incentivizes each agent to report his true cost. Such a mechanism usually refers to a *truthful* mechanism. Previously, as far as we know, only a 2-approximate truthful mechanism (Wang, Li, and Wang 2004) for the STP with k terminal vertices was known and this ratio was im-

*Zhengyang Liu is the corresponding author.

proved to $(2 - \frac{1}{k})$ (Gualà and Proietti 2005) while preserving truthfulness. (Christodoulou and Sgouritsa 2019) considers the problem of designing network cost-sharing protocols with good equilibria under uncertainty, where only terminals are the agents who need to select a path to the same root and the objective of the mechanism designer is to construct a Steiner tree.

In this paper, we propose a truthful (in expectation) mechanism for STPs on general graphs with arbitrary terminal vertices, of which the ratio matches the current best algorithmic ratio $\ln 4 + \epsilon \approx 1.39$. Precisely, a mechanism is *truthful in expectation* if no agent can obtain more expected utility by deviating from his truthful cost in the mechanism. We employ the decomposition technique due to (Lavi and Swamy 2011) for the design of truthful mechanisms to find a decomposition according to the optimal fractional solution of k Directed Component-based Relaxation (k -DCR), which is an LP relaxation of the STP. A detailed description of such a decomposition is shown in Section 3.

This decomposition (which is a convex combination of a polynomial number of integral solutions) is based on the deterministic algorithm for k -DCR in (Traub and Zenklusen 2021) and achieves the same approximation ratio as that in deterministic algorithms. However, we cannot imply a truthful mechanism by directly employing the technique, since the solution of the k -DCR only considers on the components. We present a specified payment rule by transforming the payment on the component by that technique into the payment on the edge. We then show that our mechanism is truthful in expectation. Finally, by the well-known relation among k -DCR, STP and the maximal-in-range methodology, the approximation ratio $\ln 4 + \epsilon$ of our mechanism is proved by choosing $k = \lceil 2^{\frac{1}{\epsilon}} \rceil$ for any given $\epsilon > 0$. It is worth to mention that a direct application of the decomposition technique (Theorem 1) is the survivable Network Design (Orlowski et al. 2010; Lau et al. 2009).

Related work. Besides the current best algorithmic bounds for general graph proposed in (Byrka et al. 2013), there is other progress in the design of efficient algorithms for the STP. Chen and Hsieh (2020) present an efficient two-phase heuristic strategy that achieves an approximation ratio of 1.4295 in the greedy regime. Chen and Hsieh (2022) study the STP with bounded edge-length d in which d is the ratio of the maximum edge cost to the minimum edge cost. This work analyzes the algorithm of Byrka et al. (2013) and shows that the approximation ratio of $\frac{d \ln 4}{d + \ln 4 - 1} + \epsilon$ for general graphs and approximation ratio of $\frac{73d}{60d + 13} + \epsilon$ for quasi-bipartite graphs. Siebert et al. (2020) present a set of integer programmings for the Steiner tree problem with the property that the best solution is obtained by solving all, and provide an optimal Steiner tree. Saikia and Karmakar (2021) present two deterministic distributed algorithms for the Steiner tree problem in the congest model, both of which improve the previous complexity of distributed algorithms for the STP.

Based on the decomposition technique in Section 3 originated from (Lavi and Swamy 2011), the hypergraphic relaxation and the integrality gap for the Steiner tree problem are closely related to the mechanism design. In (2013),

Byrka et al. proved an integrality gap 1.55 for k directed component relaxation of the Steiner tree. Chakrabarty et al. (Chakrabarty, Könemann, and Pritchard 2010) describe a shorter proof of the same integrality gap bound, by applying some of their techniques to a randomized loss-contracting algorithm. The same authors (Chakrabarty, Könemann, and Pritchard 2013) prove new partition-based relaxation of hypergraphic linear programming relaxations for the Steiner tree problem is equivalent to many existing hypergraphic relaxation for the Steiner tree problem. In (Feldmann et al. 2016), Feldmann et al. give an efficient constructive proof that bi-directed component relaxation and hypergraphic linear programming relaxation are polyhedrally equivalent in instances that do not have an (edge-induced) claw on Steiner vertices. Goemans et al. (Goemans et al. 2012) first show that the component-based LP relaxation of the Steiner tree problem achieves an integrality of $\ln 4$ and present a deterministic $\ln 4 + \epsilon$ algorithm to find an integral solution with this gap. Very recently, Traub and Zenklusen (Traub and Zenklusen 2021) show the same integrality gap holds for k -DCR of the Steiner tree problem by deterministically finding an integral solution with such a gap through somehow greedy procedures. Our truthful mechanism with the ratio $\ln 4 + \epsilon$ was based on this algorithm, with other fundamental insights.

Another related work is the directed Steiner tree problem, where the given input graph is a directed one. Grandoni et al. (Grandoni, Laekhanukit, and Li 2019) present a tight $O(\frac{\log^2 k}{\log \log k})$ approximation algorithm for directed Steiner tree problem running in quasi-polynomial time e.g., in time $n^{\text{poly}(\log k)}$. Li and Laekhanukit (2021) give the first known lower bound on the integrality gap of the standard flow LP relaxation for the directed Steiner tree problem that is polynomial in n , the number of vertices. Chitnis et al. (2021) systematically study several special cases of directed Steiner Network and determine their parameterized approximability for the parameter k , the demand pairs. By primal-dual scheme, Friggstad and Mousavi (2021) give the first constant-factor approximation algorithm for quasi-bipartite instances of directed Steiner tree problem on graphs that exclude fixed minors.

Before our work, the best known truthful approximation for the Steiner tree problem with the polynomial-time is shown by (Gualà and Proietti 2005) with the ratio $2 - \frac{1}{k}$, where k is the number of terminals. This paper improves this ratio to the current best algorithmic ratio by truthful in expectation mechanisms.

Organizations. The rest of this paper is organized as follows. We will show the necessary preliminaries in Section 2. In Section 3, we then describe the detailed decomposition technique for a related covering problem. Section 4 is devoted to the truthful mechanism for our problem. Finally, we conclude in Section 5.

2 Preliminaries

We are given an undirected graph $G = (V, E)$, a non-negative cost function $c : E \rightarrow \mathbb{R}_{\geq 0}$ for each edge $e \in E$

and a set of terminals $R \subseteq V$. The Steiner Tree Problem (STP) is to find a minimum cost tree on G spanning over the terminal set R where extra intermediate vertices and edges can be added to the graph in order to reduce the cost of the spanning tree. Note that any tree on G spanning on R is a feasible integral solution for STP. We call a vertex $v \in V$ a *Steiner vertex* if $v \in V \setminus R$. We also use $I = (V, E, R, c)$ to denote an instance of the STP and \mathcal{I} to denote the set of all the instances of the STP. Suppose each edge $e \in E$ is held by a rational agent and then her valuation to edge e is $-c(e)$. A randomized mechanism \mathcal{M} is a map $\mathcal{M} : I \rightarrow (X(I), P(I))$, for any instance $I \in \mathcal{I}$, where $X(I) = (X_e(I))_{e \in E}$ is a randomized solution for the STP (i.e., $X_e(I)$ is the allocation probability of edge e) and $P(I) = (P_e(I))_{e \in E} \in R_+^{|E|}$ is the (expected) payment vector for each edge $e \in E$. We use $c'(e)$ to denote the bidding cost of edge e . We write $X_e(I)$ as $X_e(c')$, for $e \in E$, where $c' = (c'(e))_e = (c'(e), c'(-e))$, where $c'(-e)$ denotes the bidding cost vector of all the edges except edge e . Similarly, we have $P_e(c')$, $e \in E$. The (expected) utility $u_e(c'(e), c'(-e))$ of agent e under the bidding vector $c' = (c'(e), c'(-e))$ is

$$u_e(c'(e), c'(-e)) = \mathbb{E}[-c(X_e(c')) - P_e(c')],$$

where $c(X_e(c'))$ is the random cost of edge e under bidding vector $c' = (c'(e), c'(-e))$, and the expectation is taken over all the random bits of the mechanism. Since in our setting it is a reverse auction or a procurement auction where sellers bid, we have $P_e(I) \leq 0$, for all $e \in E$ and $I \in \mathcal{I}$, which means sellers will receive money from buyers.

Definition 1. A mechanism $\mathcal{M} = (X, P)$ is *truthful in expectation* if for any $I \in \mathcal{I}$, any edge $e \in E$,

$$u_e(c(e), c'(-e)) \geq u_e(c'(e), c'(-e)),$$

where $c'(e)_e = (c'(e), c'(-e))$ are their biddings and $c(e)$ is the truthful cost of the edge e .

For each subgraph $C \subseteq G$, we use $R(C)$ to denote the set of terminal vertices contained in the subgraph C . We consider two different LP relaxations of STP. One is called Bi-directed Cut Relaxation (BCR) and the other is called hypergraphic linear programming relaxation (HYP) (or called directed component relaxation in another form of HYP). We first present the BCR formula. We replace each edge $(u, v) \in E$ by two directed edges (u, v) and (v, u) . For a given cut $U \subseteq V$, we denote by $\delta^+(U) = \{(u, v) \in E \mid u \in U, v \notin U\}$ the set of edges leaving U . We choose any terminal vertex $r \in R$ as a root, then the objective of the BCR is

$$\begin{aligned} \min \quad & \sum_{e \in E} c(e)z_e && \text{(BCR)} \\ \text{s.t.} \quad & \sum_{e \in \delta^+(U)} z_e \geq 1 && U \subseteq V \setminus \{r\} : U \cap R \neq \emptyset \\ & z_e \in \{0, 1\} && e \in E \end{aligned}$$

The original relaxation of the BCR is to relax the constraint $z_e \in \{0, 1\}$, $e \in E$ above to $z_e \geq 0$, $e \in E$. In the following, when we refer to the relaxation form, we mean that the

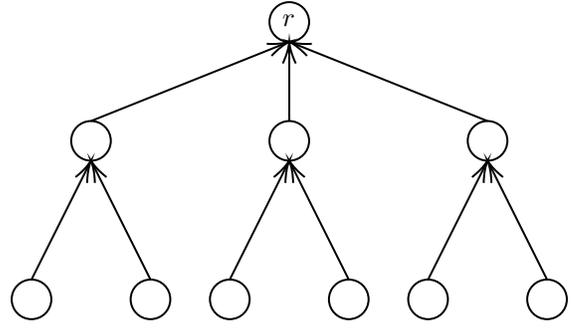


Figure 1: An illustration of the in-arborescence

variables satisfy $z_e \in [0, +\infty)$. Note that the optimal integral solution of the BCR is also the optimal solution of the STP.

We next introduce the formulas of HYPs. The form of the HYP we will show is the Directed Component-based Relaxation (DCR) (Chakrabarty, Könemann, and Pritchard 2010). A component C is simply a subgraph of G with the condition that it is a tree spanning over $V(C)$, where all leaves of C are terminals, while all internal ones are not. We consider a directed version of full components. An in-arborescence is a directed rooted tree with each edge pointing to the root r (see Figure 1).

The directed full-component is an in-arborescence to one of its terminals as the root, called its head. We denote by $H(C)$ the head of directed full-component C . We call the set of all directed full-components \mathcal{D} . By $\Delta^+(S)$, we denote all full-components $C \in \mathcal{D}$ for which the head lies outside S , while some other terminal of C lies inside. We also let $x(\Delta^+(S)) = \sum_{C \in \Delta^+(S)} x_C$. Similar with the BCR, we choose any $r \in R$ as a root. The objective of the directed component relaxation is given by

$$\begin{aligned} \min \quad & \sum_{C \in \mathcal{D}} c(C)x_C && \text{(DCR)} \\ \text{s.t.} \quad & x(\Delta^+(U)) \geq 1 && U \subseteq V \setminus \{r\} : U \cap R \neq \emptyset \\ & x_C \in \{0, 1\} && C \in \mathcal{D} \end{aligned}$$

The central observation is that a collection of full components forms a Steiner tree if and only if its corresponding terminal sets form a hypergraph in which each pair of terminals are joined by a unique hyper-path, a so-called hyper-spanning tree. Therefore, the union of every feasible integral solution of DCR is a feasible solution of Steiner Tree Problems due to the connected constraint that $x(\Delta^+(U)) \geq 1$, where $U \subseteq V \setminus \{r\} : U \cap R \neq \emptyset$ and the minimum objective requirement.

We use \mathcal{D}_k to denote the set of all the directed full components with at most k terminals. For each subset $R' \subseteq R$ with $|R'| = r \leq k$, one can select a component C with $R(C) = R'$ as the component in \mathcal{D}_k , where

$$\begin{aligned} C = \arg \min_{C'} \left\{ c(C') = \sum_{e \in C'} c(e) \mid \right. \\ \left. R(C') = R', C' \text{ is a full directed component} \right\}. \end{aligned} \quad (1)$$

Therefore, $|\mathcal{D}_k| = \sum_{r \in [k]} \binom{|R|}{r}$. There is an one-to-one correspondence among the elements in \mathcal{D}_k and the subset of R with size $r \leq k$, such that for each $R' \subseteq R$ with $|R'| = r \leq k$, there is one $C \in \mathcal{D}_k$ defined above, and vice versa. Hence, if $k = \lceil 2^{\frac{1}{\epsilon}} \rceil$ for any given ϵ , we have that $|\mathcal{D}_k| = \text{poly}(|R|)$. Furthermore, in this case, each component in \mathcal{D}_k can be found in polynomial time (i.e., $O(k|V|(|E| + |V| \ln(|V|)))$ time) (Wang, Li, and Wang 2004). Then we have the restricted k -DCR

$$\begin{aligned} \min \quad & \sum_{C \in \mathcal{D}_k} c(C)x_C \quad (k\text{-DCR}) \\ \text{s.t.} \quad & \sum_{C \in \Delta^+(U)} x_C \geq 1 \quad U \subseteq V \setminus \{r\} : U \cap R \neq \emptyset \\ & x_C \in \{0, 1\} \quad C \in \mathcal{D}_k \end{aligned}$$

Note that every feasible integral solution of k -DCR is also a feasible solution of Steiner tree problems. We call an integral solution of the BCR a feasible integral solution for the BCR, i.e., $z_e \in \{0, 1\}, \forall e \in E$. Recall that the optimal integral solution of the BCR is the optimal solution of the STP. Similarly, we can say integral solutions for the DCR or k -DCR. We use $\text{opt}(\text{BCR})$ and $\text{opt}_f(\text{BCR})$ to denote the value of optimal integral solution (which is the optimal value of STP) and optimal fractional solution for BCR. Similarly, we have the integral solution $\text{opt}(\text{DCR})$ and the fractional solution $\text{opt}_f(\text{DCR})$ for the DCR and we write $\text{opt}(k\text{-DCR})$ and $\text{opt}_f(k\text{-DCR})$ as $\text{opt}_k(\text{DCR})$ and $\text{opt}_{f,k}(\text{DCR})$, respectively.

Definition 2. *The approximation ratio of a mechanism \mathcal{M} for Steiner tree problem concerning the optimal integral solution (resp. optimal fractional solution) is defined as*

$$\alpha(\mathcal{M}) = \max_{I \in \mathcal{I}} \frac{c(\mathcal{M}(I))}{\text{opt}(I)} \quad \left(\text{resp. } \alpha(\mathcal{M}) = \max_{I \in \mathcal{I}} \frac{c(\mathcal{M}(I))}{\text{opt}_f(I)} \right),$$

where $c(\mathcal{M}(I))$ is the cost output by the algorithm \mathcal{M} on instance I and $\text{opt}(I)$ is the optimal integral solution (resp. optimal fractional solution) on instance I .

It is well known that

Lemma 1 ((Borchers and Du 1997)). $\frac{\text{opt}_k(\text{DCR})}{\text{opt}(\text{BCR})} \leq \rho_k = 1 + \frac{1}{\lceil \log k \rceil}$.

Hence, choose $k = 2^{\frac{1}{\epsilon}}$, we know that the optimal integral solution of k -DCR is an $(1 + \epsilon)$ -approximate optimal Steiner tree problem. In addition, we also have

Lemma 2 ((Chakrabarty, Könemann, and Pritchard 2010)). $\text{opt}_{f,k}(\text{DCR}) \leq \rho_k \text{opt}_f(\text{BCR})$.

We list some known results from literature including most recent work.

Proposition 1 ((Traub and Zenklusen 2021)). *There is a deterministic polynomial time algorithm for finding an integral solution of k -DCR with the approximation ratio $\ln 4 \approx 1.39$ concerning $\text{opt}_{f,k}(\text{DCR})$.*

Definition 3. *We call a Steiner tree instance Steiner claw-free if the graph G has no Steiner vertex with at least three Steiner neighbors.*

We call a Steiner tree instance quasi-bipartite graphs if no Steiner vertices are adjacent (which form an independent set). Note that Steiner claw-free graph includes quasi-bipartite graph as a special case.

Proposition 2 ((Goemans et al. 2012; Feldmann et al. 2016)). *There is a deterministic polynomial-time algorithm for finding an integral solution of BCR of Steiner claw-free STP with approximation ratio $\ln 4 \approx 1.39$ concerning $\text{opt}_f(\text{BCR})$*

3 Truthful Mechanism for One Dimensional Covering Problem (OCP)

In (Lavi and Swamy 2011), the authors provide a general framework to transfer an α -approximate algorithm (with respect to the optimal fractional solution) for linear packing problem into a truthful in expectation mechanism. We recast their approach for one-dimensional covering problem, which provides a foundation for the design of the truthful in expectation mechanisms for the STP. Although compared to their method, we both use dual programming and the ellipsoid method to reduce the problems into linear programming with a polynomial-size of constraints, the design of detailed estimation including the objective value of the dual and the key point of the cutting ways of the hyperplane by the ellipsoid method are different.

One-dimensional Covering Problem (OCP) is defined as follows:

$$\begin{aligned} \min \quad & \sum_{i \in M} c_i x_i \quad (\text{OCP}) \\ \text{s.t.} \quad & \sum_{i \in M_j} a_i x_i \geq b_j \quad j \in J, M_j \subseteq M \\ & x_i \in \{0, 1\} \quad i \in M \end{aligned}$$

Here $c_i \geq 0$ and $a_i \geq 0, i \in M$, and M is the index set and $|J|$ is the number of covering constraints. The procedure of decomposition of any optimal fractional solution $\vec{x}^* = (x_i^*)$, $i \in M$ of the one-dimensional covering problem (OCP) is given as follows. Let $\{\vec{x}^\ell\}_{\ell \in L}$ be the set of all the feasible integral solutions of the OCP, where L is the index set and $|L|$ is the number of all the feasible integral solutions of the OCP. In the following part of this section, we suppose there exists a polynomial time algorithm \mathcal{A} for OCP that achieves the approximation ratio α with respect to the optimal fractional solution of OCP. We write the Primal of the decomposition LP as follows.

$$\begin{aligned} \max \quad & \sum_{\ell \in L} \lambda_\ell \quad (\text{Primal}) \\ \text{s.t.} \quad & \sum_{\ell \in L} \lambda_\ell x_i^\ell = \alpha x_i^* \quad \forall i \in M \\ & \sum_{\ell \in L} \lambda_\ell \leq 1 \\ & \lambda_\ell \geq 0 \quad \ell \in L \end{aligned}$$

The Dual of the above decomposition LP is as follows.

$$\begin{aligned}
\min \quad & \sum_{i \in M} \alpha w_i x_i^* + z && \text{(Dual)} \\
\text{s.t.} \quad & \sum_{i \in M} w_i x_i^\ell + z \geq 1 && \forall \ell \in L \\
& z \geq 0, w_i \text{ is arbitrary} && i \in M
\end{aligned}$$

Let \mathcal{P} denote the set all the feasible fractional solutions of OCP. Let $\mathbb{Z}(\mathcal{P})$ denote the set of all the feasible integral solutions of OCP. Note that the parameters $\{c_i\}_{i \in M}$ in OCP are non-negative. Let $\bar{w}^+ = (w_i^+)_{i \in M} = (\max\{w_i, 0\})_{i \in M}$. We have the following lemma.

Lemma 3. *For any vector \bar{w} , we can find (in polynomial time) an integral solution $x^\ell \in \mathbb{Z}(\mathcal{P})$ of OCP with the objective coefficient \bar{w} such that*

$$\sum_{i \in M} w_i x_i^\ell \leq \alpha \min_{x \in \mathcal{P}} \sum_{i \in M} w_i x_i$$

Proof. We can use \mathcal{A} to find such an integral solution \bar{x}' with $\bar{w}^+ = \bar{c}$ as coefficients in objective function in OCP. Let $x_i^\ell = x_i'$, if $w_i \geq 0$ and $x_i' = 1$, otherwise. Clearly, x^ℓ is a feasible integral solution for OCP, i.e., $\bar{x}^\ell \in \mathcal{P}$. Let \bar{x}'' be the solution of $\min_{\bar{x} \in \mathcal{P}} \sum_{i \in M} w_i x_i$, i.e., $\sum_{i \in M} w_i x_i'' = \min_{\bar{x} \in \mathcal{P}} \sum_{i \in M} w_i x_i$ and $\bar{x}'' \in \mathcal{P}$. By definition of \bar{x}' , $\sum_{i \in M} w_i^+ x_i' \leq \alpha \sum_{i \in M} w_i^+ x_i''$. By definition of \bar{x}'' , $\sum_{i \in M} w_i^+ x_i'' = \sum_{i \in M} w_i^+ x_i' \leq \alpha \sum_{i \in M} w_i^+ x_i'$. Since $x_i^\ell = 1$ for all $w_i < 0$, $i \in M$, we then have $\sum_{i \in M} w_i x_i^\ell \leq \alpha \sum_{i \in M} w_i x_i'' = \alpha \min_{\bar{x} \in \mathcal{P}} \sum_{i \in M} w_i x_i$. \square

Lemma 4. *We can in polynomial time find a decomposition $\sum_{\ell \in L} \lambda_\ell x_i^\ell = \alpha x_i^*$ with $\sum_{\ell \in L} \lambda_\ell = 1$ and $\lambda_\ell > 0$, for all $\ell \in L$.*

Proof. First, note that if $\sum_{i \in M} \alpha w_i x_i^* + z < 1$, by Lemma 3, we can find an integral solution \bar{x}^ℓ in polynomial time such that $\sum_{i \in M} w_i x_i^\ell + z \leq \alpha \min_{x \in \mathcal{P}} \sum_{i \in M} w_i x_i + z \leq \sum_{i \in M} \alpha w_i x_i^* + z < 1$. Hence, the optimal value of Dual is at least 1, by strong duality, the optimal value of Primal is at least 1. As $\sum_{\ell \in L} \lambda_\ell \leq 1$, we know the optimal value of Dual and Primal are both 1.

Secondly, we can add the constraint $\sum_{i \in M} \alpha w_i x_i^* + z \leq 1$ into the Dual. The ellipsoid algorithm (Grötschel, Lovász, and Schrijver 1993) can be used to solve the Dual as follows. We observe that the following separation oracle can be used to solve the Dual (or identify polynomial-size equivalent constraints of the Dual). These constraints will be the violated inequalities returned by the separation oracle during the execution of the ellipsoid method, that are used to cut the ellipsoid. If $\sum_{i \in M} \alpha w_i x_i^* + z < 1$, by Lemma 3, we can find a violation constraint of the Dual as the cutting plane. Otherwise, we can use the half hyperplane $\sum_{i \in M} \alpha w_i x_i^* + z \leq 1$ as the cutting plane.

Therefore, we can find an equivalent Dual with a set of constraints of size polynomial to solve the Dual, thus the Primal, which completes the proof. \square

As a direct consequence of Lemma 4, a truthful in expectation mechanism can be achieved by VCG payment as that in (Lavi and Swamy 2011).

Theorem 1. *Suppose there is a polynomial-time algorithm for OCP that achieves the approximation ratio α with respect to the optimal fractional solution of OCP. Then, we can in polynomial time construct a truthful in expectation mechanism for OCP with the same approximation ratio α .*

Proof. Let L_{poly} be the index set of the polynomial-size decomposed solution in Lemma 4 i.e., $\sum_{\ell \in L_{\text{poly}}} \lambda_\ell x_i^\ell = \alpha x_i^*$, $\forall i \in M$. Let opt_{-i} denote the optimal objective value of the OCP by removing agent i . Let $S_i = \alpha(\sum_{j \neq i} c_j x_j^* - \text{opt}_{-i})$. We will show that the mechanism with the following allocation rule and payment rule is truthful and (ex-post) individual rational and with the approximation ratio α :

1. The allocation rule: select the solution x^ℓ with probability λ_ℓ , $\ell \in L_{\text{poly}}$;
2. The payment rule: for each random selection, charge agent i with the payment $\frac{c_i x_i^\ell}{\sum_{\ell \in L_{\text{poly}}} \lambda_\ell c_i x_i^\ell} \cdot S_i$.

Now calculating the utility of agent i when he bids c'_i while his true cost is c_i .

$$\begin{aligned}
U_i(c'_i) &= - \sum_{\ell \in L_{\text{poly}}} \lambda_\ell c_i x_i^\ell(c'_i) \\
&\quad - \sum_{\ell \in L_{\text{poly}}} \lambda_\ell \frac{c_i x_i^\ell(c'_i)}{\sum_{\ell \in L_{\text{poly}}} \lambda_\ell c_i x_i^\ell(c'_i)} \cdot S_i(c'_i) \\
&= - \sum_{\ell \in L_{\text{poly}}} \lambda_\ell c_i x_i^\ell(c'_i) - \frac{\sum_{\ell \in L_{\text{poly}}} \lambda_\ell c_i x_i^\ell(c'_i)}{\sum_{\ell \in L_{\text{poly}}} \lambda_\ell c_i x_i^\ell(c'_i)} \cdot S_i(c'_i) \\
&= - \sum_{\ell \in L_{\text{poly}}} \lambda_\ell c_i x_i^\ell(c'_i) - S_i(c'_i) = -\alpha c_i x_i^*(c'_i) - S_i(c'_i) \\
&= \alpha(-c_i x_i^*(c'_i) - \sum_{j \neq i} c_j x_j^*(c'_i) + \text{opt}_{-i}) \\
&= -\alpha(\sum_{j \in M} c_j x_j^*(c'_i) + \text{opt}_{-i}) \leq -\alpha(\sum_{j \in M} c_j x_j^*(c_i) + \text{opt}_{-i}) \\
&= U_i(c_i).
\end{aligned}$$

Therefore, the mechanism with the above allocation and payment rules is truthful in expectation. Thus, Theorem 1 follows. \square

4 Truthful Mechanisms for Steiner Tree Problems

The mechanism is based on the decomposition techniques in Section 3. A straightforward application of this technique gives a truthful in expectation mechanism for Steiner claw-free STP.

Theorem 2. *There is a truthful in expectation mechanism with approximation ratio $\ln 4 \approx 1.39$ for Steiner claw-free STP, which runs in polynomial time.*

Proof. The proof is a direct consequence of Proposition 2 and Theorem 1 by noting that BCR is a special OCP. \square

It is interesting to observe that in Theorem 2, the allocation rule and payment rule are obtained by the BCR, but not based on the linear programming of the STP and hence the mechanism is truthful based on Theorem 1. As the allocation is a feasible solution to the STP, which induces a $\ln 4 \approx 1.39$ approximation mechanism.

It is worth to be aware that a direct application of Theorem 1 and Proposition 1 does not imply a truthful in expectation mechanism for STP in general graphs. Employing Proposition 1 and Theorem 1, a truthful in expectation mechanism is given for k -DCR with the ratio $\ln 4$ by allowing each directed component to be held by a single agent. This is not a truthful mechanism for STP since in that problem each agent only controls a single edge. We will use Proposition 1 and Lemma 4 to design such a truthful mechanism with the ratio $\ln 4 + \epsilon$. Our approach consists of two steps. First, we will use the allocation rule given by Lemma 4. Second, using Maximal in Range (MIR), we will provide a sophisticated payment rule based on the above allocation rule. In intuition behind this payment rule comes from the payment rule of VCG payment rule over the directed components. Finally, this mechanism will be shown as the desired mechanism for the STP with the ratio $\ln 4 + \epsilon$ by choosing $k = 2^{\frac{1}{\epsilon}}$. Recall that when $k = 2^{\frac{1}{\epsilon}}$, we have that $|\mathcal{D}_k| = \text{poly}(|R|)$.

Note that k -DCR is an OCP. Hence, by Lemma 4, we can in polynomial time find a decomposition $\sum_{\ell \in L} \lambda_\ell x_C^\ell = \alpha x_C^*$ with $\sum_{\ell \in L} \lambda_\ell = 1$ and $\lambda_\ell \geq 0$, for all $\ell \in L$ and $C \in \mathcal{D}_k$, where $\{x_C^*\}_{C \in \mathcal{D}_k}$ is the optimal fractional solution of k -DCR, and L has the polynomial size of the input of STP. Recall that $\text{opt}_{f,k}(\text{DCR})$ denotes the optimal fractional value of k -DCR. For each edge $e \in E$, let DCR_{-e} (resp. k -DCR $_{-e}$) denotes the DCR form (resp. k -DCR) of STP by removing the edge e from the graph. Similarly, we have $\text{opt}_{f,k}(\text{DCR}_{-e})$. The mechanism \mathcal{M} for k -DCR works as follows.

1. The allocation rule X : select the solution \bar{x}^ℓ with the probability λ_ℓ .
2. The payment rule P : for each edge e , the payment for e is

$$P_e = \alpha \cdot \text{opt}_{f,k}(\text{DCR}) - \sum_{\ell \in I} \sum_{C: e \in C} \lambda_\ell c(e) x_C^\ell - \alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e}).$$

Theorem 3. *The mechanism \mathcal{M} is a truthful in expectation mechanism for Steiner tree problems in general graphs that achieves the approximation ratio $\ln 4 + \epsilon \approx 1.39 + \epsilon$, running in polynomial time for any given $\epsilon > 0$.*

Proof. We observe the following fact of the mechanism \mathcal{M} .

By the decomposition and the allocation rule X of \mathcal{M} , the approximation ratio of \mathcal{M} for k -DCR is $\ln 4$ with respect to $\text{opt}_{f,k}(\text{DCR})$. By the fact that $\text{opt}_{f,k}(\text{DCR}) \leq \rho_k \cdot \text{opt}_f(\text{BCR})$, where $\rho_k = 1 + \frac{1}{\lceil \log k \rceil}$. \mathcal{M} approximates $\text{opt}_f(\text{BCR})$ with the ratio $\ln 4 + \epsilon$ when $k = 2^{\frac{1}{\epsilon}}$. This also shows that the running time of \mathcal{M} is polynomial time for

any given $\epsilon > 0$ provided $k = 2^{\frac{1}{\epsilon}}$. Therefore, it is sufficient to show that \mathcal{M} is a truthful in expectation mechanism.

Observe that $\alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e}) \geq \alpha \cdot \text{opt}_{f,k}(\text{DCR})$ since any feasible solution of k -DCR $_{-e}$ is a feasible solution for k -DCR. Therefore, $P_e \leq 0$, which shows that the payment is no negative money transfer. Given $c(-e)$, when consider the agent e , we write $c = (c(e), c(-e))$ and $c' = (c'(e), c(-e))$. By the allocation rule, the expected allocation X_e of edge e is $X_e(c(e)) = \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell x_C^\ell(c(e))$. Observe that

$$\begin{aligned} & \alpha \cdot \text{opt}_{f,k}(\text{DCR}(c')) - \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell c'(e) x_C^\ell(c') \\ &= \sum_{e \in E} \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell c'(e) x_C^\ell(c') - \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell c'(e) x_C^\ell(c') \\ &= \sum_{e' \neq e} \sum_{\ell \in L} \sum_{C: e' \in C} \lambda_\ell c(e') x_C^\ell(c'). \end{aligned}$$

Therefore, the expected utility u_e of edge e is given by

$$\begin{aligned} u_e(c') &= -c(e)X_e(c') - P_e(c') \\ &= -c(e) \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell x_C^\ell(c') - [\alpha \cdot \text{opt}_{f,k}(\text{DCR}(c')) \\ &\quad - \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell c'(e) x_C^\ell(c') - \alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e})] \\ &= -c(e) \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell x_C^\ell(c') \\ &\quad - \left[\sum_{e' \neq e} \sum_{\ell \in L} \sum_{C: e' \in C} \lambda_\ell c(e') x_C^\ell(c') - \alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e}) \right] \\ &= \alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e}) \\ &\quad - \left[\sum_{e' \neq e} \sum_{\ell \in L} \sum_{C: e' \in C} \lambda_\ell c(e') x_C^\ell(c') + c(e) \sum_{\ell \in L} \sum_{C: e \in C} \lambda_\ell x_C^\ell(c') \right] \\ &= \alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e}) - \sum_{e' \in E} \sum_{C: e' \in C} c(e') \sum_{\ell \in L} \lambda_\ell x_C^\ell(c') \\ &= \alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e}) - \alpha \sum_{e' \in E} \sum_{C: e' \in C} c(e') x_C^*(c') \\ &= \alpha \cdot \text{opt}_{f,k}(\text{DCR}_{-e}) - \alpha \sum_{C' \in \mathcal{D}_k(c')} c(C') x_{C'}^*(c'), \end{aligned}$$

where $c(e)$ and $c'(e)$ denote the true cost and the bid of edge e respectively and $x^*(c')$ to denote the optimal fractional solution for k -DCR when e bids $c'(e)$ given the bids $c(-e)$ of all other edges, and $\mathcal{D}_k(c'(e))$ is the set of k -directed component when edge e bids $c'(e)$. Note that in the above calculation, we use C to denote the component containing $R(C)$, which is one-to-one correspondence. The optimal solution $\{x_{C'}^*(c')\}_{C' \in \mathcal{D}_k(c')}$ is a feasible solution for k -DCR and the fact that the true cost of a component containing $R(C')$ under the bid $c = (c(e), c(-e))$ is no more than that under the bid $c' = (c'(e), c(-e))$, i.e., for any $R' \subseteq R$, let $C' \in \mathcal{D}_k(c')$ containing R' (with $R(C') = R'$)

) and $C \in \mathcal{D}_k(c)$ containing R' (with $R(C) = R'$), we have $c(C') \leq c(C)$, where $c(C') = \sum_{e \in C'} c(e)$ using the true cost by recalling the formula (1). Note that $C' \in \mathcal{D}_k(c')$ has an one-to-one correspondence to $C \in \mathcal{D}_k(c)$ with $R(C) = R(C') = R' \subseteq R$. Therefore, we have

$$\begin{aligned} \sum_{C' \in \mathcal{D}_k(c')} c(C') x_{C'}^*(c') &\geq \sum_{C \in \mathcal{D}_k(c)} c(C) x_{C'}^*(c') \\ &\geq \sum_{C \in \mathcal{D}_k(c)} c(C) x_C^*(c), \end{aligned}$$

where the first inequality comes from $c(C') \leq c(C)$ and the second comes from that $\{x_{C'}^*(c')\}_{C' \in \mathcal{D}_k(c')}$ is a feasible solution for k -DCR. Thus, $u_e(c'(e)) \leq u_e(c(e))$. This shows that the mechanism \mathcal{M} is truthful in expectation. \square

Remark: Note that each feasible integral solution x^ℓ corresponds to a union set of directed components in \mathcal{D}_k , which contains a tree spanning on all the terminals R used as the final solution of the STP. However, the payment rule we use is not based on this tree but the union set of directed components.

5 Conclusion

We present a truthful-in-expectation mechanism for the STP on general graphs with the ratio $\ln 4 + \epsilon \approx 1.39$, matching the current best algorithmic ratio for the STP. It is interesting to see to what extent the decomposition approach can be applied to other covering combinatorial problems for truthful approximate mechanisms. Besides, we are also interested to know whether there exists a deterministic truthful or universally truthful mechanism for the STP with the ratio ≈ 1.39 .

Acknowledgments

This work was supported by the National Key Research and Development Program of China (2022YFF0902005). Jianwei Yin was supported by the National Science Fund for Distinguished Young Scholars (No. 61825205). Jinshan Zhang was supported by the Key Research and Development Jianbing Program of Zhejiang Province (2023C01002), Hangzhou Major Project and Development Program (2022AIZD0140) and Yongjiang Talent Introduction Programm (2022A-236-G). Xiaotie Deng was supported by the National Natural Science Foundation of China (No. 62172012). Zhengyang Liu was supported by the National Natural Science Foundation of China (No. 62002017) and Beijing Institute of Technology Research Fund Program for Young Scholars. We are also grateful for the valuable comments from the anonymous reviewers.

References

Armstrong, M. 2006. Competition in two-sided markets. *The RAND journal of economics*, 37(3): 668–691.

Bern, M.; and Plassmann, P. 1989. The Steiner problem with edge lengths 1 and 2. *Information Processing Letters*, 32(4): 171–176.

Borchers, A.; and Du, D.-Z. 1997. The k -Steiner ratio in graphs. *SIAM Journal on Computing*, 26(3): 857–869.

Byrka, J.; Grandoni, F.; Rothvoß, T.; and Sanità, L. 2013. Steiner tree approximation via iterative randomized rounding. *Journal of the ACM (JACM)*, 60(1): 1–33.

Chakrabarty, D.; Könemann, J.; and Pritchard, D. 2010. Integrality gap of the hypergraphic relaxation of Steiner trees: A short proof of a 1.55 upper bound. *Operations Research Letters*, 38(6): 567–570.

Chakrabarty, D.; Könemann, J.; and Pritchard, D. 2013. Hypergraphic LP relaxations for Steiner trees. *SIAM Journal on Discrete Mathematics*, 27(1): 507–533.

Chen, C.-Y.; and Hsieh, S.-Y. 2020. An efficient approximation algorithm for the Steiner tree problem. In *Complexity and Approximation*, 238–251. Springer.

Chen, C.-Y.; and Hsieh, S.-Y. 2022. An improved algorithm for the Steiner tree problem with bounded edge-length. *Journal of Computer and System Sciences*, 123: 20–36.

Chitnis, R.; Feldmann, A. E.; and Manurangsi, P. 2021. Parameterized approximation algorithms for bidirected Steiner network problems. *ACM Transactions on Algorithms (TALG)*, 17(2): 1–68.

Chlebík, M.; and Chlebíková, J. 2008. The Steiner tree problem on graphs: Inapproximability results. *Theoretical Computer Science*, 406(3): 207–214.

Christodoulou, G.; and Sgouritsa, A. 2019. Designing networks with good equilibria under uncertainty. *SIAM Journal on Computing*, 48(4): 1364–1396.

Deng, X.; Feng, Z.; and Papadimitriou, C. H. 2016. Power-law distributions in a two-sided market and net neutrality. In *International Conference on Web and Internet Economics*, 59–72. Springer.

Feldmann, A. E.; Könemann, J.; Olver, N.; and Sanità, L. 2016. On the equivalence of the bidirected and hypergraphic relaxations for Steiner tree. *Mathematical programming*, 160(1): 379–406.

Friggstad, Z.; and Mousavi, R. 2021. A Constant-Factor Approximation for Quasi-bipartite Directed Steiner Tree on Minor-Free Graphs. *arXiv preprint arXiv:2111.02572*.

Goemans, M. X.; Olver, N.; Rothvoß, T.; and Zenklusen, R. 2012. Matroids and integrality gaps for hypergraphic Steiner tree relaxations. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, 1161–1176.

Grandoni, F.; Laekhanukit, B.; and Li, S. 2019. $O(\log^2 k / \log \log k)$ -approximation algorithm for directed Steiner tree: a tight quasi-polynomial-time algorithm. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, 253–264.

Grötschel, M.; Lovász, L.; and Schrijver, A. 1993. The ellipsoid method. In *Geometric Algorithms and Combinatorial Optimization*, 64–101. Springer.

Gualà, L.; and Proietti, G. 2005. A truthful $(2-2/k)$ -approximation mechanism for the Steiner tree problem with k terminals. In *International Computing and Combinatorics Conference*, 390–400. Springer.

Hwang, F. K.; and Richards, D. S. 1992. Steiner tree problems. *Networks*, 22(1): 55–89.

- Karp, R. M. 1972. Reducibility among combinatorial problems. In *Complexity of computer computations*, 85–103. Springer.
- Karpinski, M.; and Zelikovsky, A. 1997. New approximation algorithms for the Steiner tree problems. *Journal of Combinatorial Optimization*, 1(1): 47–65.
- Lau, L. C.; Naor, J.; Salavatipour, M. R.; and Singh, M. 2009. Survivable network design with degree or order constraints. *SIAM Journal on Computing*, 39(3): 1062–1087.
- Lavi, R.; and Swamy, C. 2011. Truthful and near-optimal mechanism design via linear programming. *Journal of the ACM (JACM)*, 58(6): 1–24.
- Li, S.; and Laekhanukit, B. 2021. Polynomial Integrality Gap of Flow LP for Directed Steiner Tree. *arXiv preprint arXiv:2110.13350*.
- Orlowski, S.; Wessäly, R.; Pióro, M.; and Tomaszewski, A. 2010. SNDlib 1.0—Survivable network design library. *Networks: An International Journal*, 55(3): 276–286.
- Prömel, H. J.; and Steger, A. 2000. A new approximation algorithm for the Steiner tree problem with performance ratio $5/3$. *Journal of Algorithms*, 36(1): 89–101.
- Robins, G.; and Zelikovsky, A. 2005. Tighter bounds for graph Steiner tree approximation. *SIAM Journal on Discrete Mathematics*, 19(1): 122–134.
- Saikia, P.; and Karmakar, S. 2021. Improved distributed approximation for Steiner tree in the CONGEST model. *Journal of Parallel and Distributed Computing*, 158: 196–212.
- Saltzer, J. H.; Reed, D. P.; and Clark, D. D. 1984. End-to-end arguments in system design. *ACM Transactions on Computer Systems (TOCS)*, 2(4): 277–288.
- Siebert, M.; Ahmed, S.; and Nemhauser, G. 2020. A linear programming based approach to the Steiner tree problem with a fixed number of terminals. *Networks*, 75(2): 124–136.
- Traub, V.; and Zenklusen, R. 2021. Local search for weighted tree augmentation and Steiner tree. *arXiv preprint arXiv:2107.07403*.
- Wang, W.; Li, X.-Y.; and Wang, Y. 2004. Truthful multicast routing in selfish wireless networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking*, 245–259.
- Zelikovsky, A. Z. 1993. An $11/6$ -approximation algorithm for the network Steiner problem. *Algorithmica*, 9(5): 463–470.