Constant Approximation for Steiner Forest Activation Problems in Planar Graphs

Yuhao Hu, Jiangwei Pan and Xiangyu Wang

Abstract

In this project, we study the network activation problem on planar graphs. In particular, we obtain a polynomial-time constant-approximation primal-dual algorithm for the Steiner forest activation problem.

1 Introduction

The network activation problem is first introduced in [Pan11] as a generalization of the traditional network design problem. In contrast to the conventional setting that either nodes or edges are assigned constant weights and will be selected independently to fulfill the connectivity requirement, nodes in the network activation problem can take multiple values (weights), and each edge can be “activated” only when the weights on the two endpoints satisfy certain conditions. More specifically, for a graph \( G = (V, E) \), each node \( v \in V \) can take weights from a predetermined set \( D_v \). In addition, an activation function \( f_{uv} : D_u \times D_v \to \{0, 1\} \) is assigned to each edge \( (u, v) \in E \), which takes values of \( x_v \) and \( x_u \) as input and returns 1 if edge \( (u, v) \) can be activated by this pair of weights. Given pairs of vertices \((s_1, t_1), \ldots, (s_k, t_k)\), the Steiner forest activation problem is to assign weights to nodes of \( G \) such that the activated edges of \( G \) connect \((s_i, t_i)\) for all \( i \) and the total weight of nodes is minimized. Note that when \( D_u = \{0, w_u\} \) and \( f_{uv} = x_u \land x_v \), the problem becomes the node-weighted Steiner forest problem.

Planar graphs are important topological structures in practice. Both edge-weighted and node-weighted Steiner forest problems have been shown to have constant approximation algorithms on planar graphs [GW95, AKR95, KR95, DHK09]. For the Steiner forest activation problem, however, the best known result is the greedy algorithm introduced in [Pan11] that achieves a \( O(\log n) \)–approximation on general graphs. We give the first constant-approximation algorithm for the Steiner forest activation problem for planar graphs.

2 Steiner forest activation on planar graphs

In this section, we introduce an algorithm which gives a constant approximation to the optimal solution for the Steiner forest activation problem in planar graphs. We reduce the Steiner forest activation problem to node-weighted Steiner forest problem and apply the primal-dual algorithm in [DHK09] to the reduced problem.

As we will see later, although the original graph is planar, the planarity need not be preserved in the reduced problem. However, it turns out that this is not much trouble for the analysis. In fact, we are able to observe that at each intermediate stage of the algorithm, the components under augmentation can be viewed as a graph which is near-planar, that is, it may not be planar, but shares with planar graphs the nice property that the average degree of the vertices is bounded by some (universal) constant. The constant approximation statement then follows from this observation and some results from the paper [DHK09].
2.1 Reduction to node-weighted Steiner forest

**Definition 1.** Let \( f : D \times D \to R \) be some function, where \( D, R \subset \mathbb{N} \). We define \( f \) to be monotone if \( i \leq i', j \leq j' \) implies \( f(i, j) \leq f(i', j') \) for all \( i, j, i', j' \).

To set up, let \( G = (V, E) \) be an undirected planar graph and \( D = \{1, 2, \ldots, d\} \) the set of possible values to be assigned to each vertex. Let \( f_{uv} : D \times D \to \{0, 1\} \) be a monotone activation function for each \( uv \in E \). Recall the connectivity requirement in the Steiner forest activation problem and let the terminal pairs be \((s_1, t_1), (s_2, t_2), \ldots, (s_k, t_k)\). A feasible solution to the Steiner forest activation problem is an assignment \( x : V \to D \) such that the set of all the edges \( uv \) with \( f_{uv}(x_u, x_v) = 1 \) connects each pair of the terminals.

We now describe the reduction of the Steiner forest activation problem to the node-weighted Steiner forest problem. We create a graph \( \tilde{G} \) from \( G \) as follows. First, maintain the vertex set \( V \) and create \( d \) additional vertices \((u, 1), (u, 2), \ldots, (u, d)\) for each vertex \( u \in V \). Then connect each of these new vertices to \( u \) by an edge. Finally, for an edge \( uv \in E \), let \((u, i) \) and \((v, j)\) be connected by an edge if and only if \( f_{uv}(x_u, x_v) = 1 \). It is then obvious that the edge \( uv \) is activated by values \( x_u = i \) and \( x_v = j \) if \((u, i) \) and \((v, j)\) are picked in the graph \( \tilde{G} \). If we further set the weights \( w(u) = 0 \) and \( w(u, i) = i \) in \( \tilde{G} \), then it is evident that an optimal solution to the activation problem on \( G \) corresponds to a choice of vertices \((u, i)\) in \( \tilde{G} \) which minimizes the total weight. This completes the reduction step. The construction of \( \tilde{G} \) is illustrated below.

![Figure 1: The Reduction Step: Construction of \( \tilde{G} \).](image)

2.2 Algorithm and analysis

For a general graph \( G = (V, E) \), we describe the LP corresponding to the node-weighted Steiner forest problem. First, for each cut \((S, \bar{S})\), define

\[
  f(S) = \begin{cases} 
    1, & \text{if } (S, \bar{S}) \text{ separates some } (s_i, t_i) \text{ pair,} \\
    0, & \text{otherwise,} 
  \end{cases}
\]

and let

\[
  \Gamma(S) = \{u : u \notin S, \exists v \in S \text{ s.t. } uv \in E\}
\]
be the one-step neighbors of $S$. Now the primal and dual LP can be stated:

**Primal LP**

minimize : $\sum_{v \in V} w(v)x_v$.  
subject to : $\sum_{v \in \Gamma(S)} x_v \geq f(S)$, for all $S \subseteq V$,  
$x_v \geq 0$, for all $v \in V$.

**Dual LP**

maximize : $\sum_{S \subseteq V} f(S)y_S$.  
subject to : $\sum_{S \subseteq V : v \in \Gamma(S)} y_S \leq w(v)$, for all $v \in V$,  
y_S \geq 0$, for all $S \subseteq V$.

**Algorithm**

- $y_S \leftarrow 0$, for all $S$.
- $X \leftarrow \{v : f(\{v\}) = 1\}$.
- While there is some active $S$, increase $y_S$ until the dual constraint becomes tight for some $v$. Let $X \leftarrow X \cup \{v\}$.
- Delete $x \in X$ in the reversed order of its being added during the while loop, preserving the feasibility of $X$.

The algorithm above is identical to the algorithm in the paper [DHK09]. Simply put, it maintains a set $X$ of selected vertices, which, initially, is just the set of terminals. Furthermore, a cut $(S, \bar{S})$ is said to be active if $S$ is a connected component in $X$. As in [DHK09], we define $\text{Viol}(X)$ to be the set of active connected components in $X$. [DHK09] has proven the following theorem.

**Theorem 1 ([DHK09]).** If $\gamma$ is a constant such that for any partial solution $X \subseteq V$ and any minimal feasible augmentation $F$ of $X$ we have

$$\sum \{|F \cap \Gamma(S)| : S \in \text{Viol}(X)\} \leq \gamma|\text{Viol}(X)|,$$

then the algorithm above returns a feasible solution which is a $\gamma$-approximation of the LP-OPT. In particular, if $G$ is planar, $\gamma$ can be chosen to be 6.

In the following, we show that $\gamma = O(1)$ for the reduced problem on graph $\tilde{G}$. Let $X$ be the set of vertices selected at some time $t$ and let $X^*$ be the solution returned by our algorithm. Let $F = X \cup X^*$. Because of the reverse delete step, $F$ is a minimal augmentation of $X$. Moreover, denote the one-step neighbors of $\text{Viol}(X)$ in $F - X$ as $F'$. Let $R$ be the set of vertices obtained by contracting each connected component in $\text{Viol}(X)$.

**Lemma 1 ([DHK09]).** $|F'| \leq 2|R|$.

**Lemma 2.** For any vertex $u$, there exists at most one $i \in D$ such that $(u, i) \in F'$.
Proof. If \((u, i), (u, j) \in F'\) with \(i < j\), simply discard \((u, i)\). By the monotonicity of the activation function \(f\), the resulting set remains a feasible solution. This contradicts the minimality of \(F\).

Now we proceed to bound \(\sum |F \cap \Gamma(S)|\) by bounding the number of edges between \(R\) and \(F'\). Note that the graph induced by \(R \cap F'\) need not be planar, so the argument in [DHK09] cannot directly apply. We observe that, for any \(S \in \text{Viol}(X)\), each vertex \(u\) in \(S\) must be of one of the following two types:

(a) There exists some \(i \in D\) such that \((u, i) \in F'\).

(b) Such \(i\) does not exist.

Furthermore, any edge between all the type (a) vertices and \(F'\) must be between one of the following pairs:

(a1) \((u, i) \in S \in \text{Viol}(X), (u, j) \in F'\).

(a2) \((u, i) \in S \in \text{Viol}(X), (v, j) \in F', u \neq v\).

Similarly, there is only one case for the edges between type (b) vertices and \(F'\):

(b1) \((u, i) \in S \in \text{Viol}(X), (v, j) \in F', u \neq v\).

In each of these three classes, we consider the contribution of the edges to the number of edges between \(R\) and \(F'\).

We first bound type (a) edges. It is easy to see that the number of edges that start at (a1) is bounded by \(|F'\|\). For (a2), according to our definition, there exists an \((u, i') \in F'\) for \((u, i) \in S \in \text{Viol}(X)\). Therefore, we can charge the edge \((u, i)(v, j)\) to \((u, i')(v, j)\). Note that each edge \((u, i')(v, j)\) in \(F'\) is charged at most twice. Therefore, the total number of edges starting at (a2) is bounded by \(2\alpha|F'|\), where \(\alpha\) is the constant that bounds the average degree of a planar graph.

Next, we bound the edges between (b) and \(F'\). The difficulty rises as \(R \cup F'\) might not necessarily be planar. Our goal is to construct a refined contraction which adds a few more vertices (compare to \(|R|\)) but yield planarity.

First, we have the following observation: for one connected component \(S \in \text{Viol}(X)\), there might exist multiple \((u, i)\) that belong to (b) and connect to the same \((v, j) \in F'\). In such case, we can arbitrarily select one \((u, i)\) and disregard all the other edges. The reasoning is that these edges are considered only as one edge in \(R\).

Our second step is to perform a contraction on the original graph \(G\). We contract each node that is associated with type (b) vertex to one connected node that is associated with type (a) vertex within the same connected component (arbitrarily). If one connected component contains only type (b) vertices, we simply contract the whole as a node. The resulted graph \(G'\) possesses following properties,

i. \(G'\) is planar.

ii. The number of edges between (b) and \(F'\) is at most twice the number of edges in \(G'\).

Notice that for each edge in \(G'\), say \(uv\), there exist at most two edges between type (b) and \(F'\) can be charged to it: either \((u', i)(v, j)\), where \(u'\) belongs to (b) and connects to \(u\) and \((v, j) \in F'\) or vice versa.

The total number of nodes in \(G'\) is bounded by \(|F'| + |R|\). Therefore, the number of edges between type (b) and \(F'\) is bounded by \(2\alpha(|F'| + |R|)\).

Therefore, we have the following theorem...
**Theorem 2.** The algorithm above is a \((8\alpha + 2)\)-approximation to the LP-OPT for the Steiner forest activation problem.

**Proof.** Using Lemma 1, the number of edges between \(R\) and \(F'\) is bounded by \((8\alpha + 2)|R|\). The result follows from Theorem 1.

---

**References**


