A Polynomial-Time Algorithm for \((2 - \frac{2}{k})\)-stable Instances of the \(k\)-terminal cut Problem

Mark Velednitsky ∗
UC Berkeley
marvel@berkeley.edu

Dorit Hochbaum
UC Berkeley
hochbaum@ieor.berkeley.edu

October 19, 2018

Abstract

The \(k\)-terminal cut problem is defined on an edge-weighted graph with \(k\) distinct vertices called “terminals.” The goal is to remove a minimum weight collection of edges from the graph such that there is no path between any pair of terminals. The \(k\)-terminal cut problem is known to be NP-hard. There has been interest in determining special classes of graphs for which \(k\)-terminal cut can be solved in polynomial time.

One special class of graphs is the class of \(\gamma\)-stable graphs, a notion introduced by Bilu and Linial. An instance of \(k\)-terminal cut is said to be \(\gamma\)-stable if edges in the cut can be multiplied by up to \(\gamma\) without changing the unique optimal solution. For several years, the best-known result for \(\gamma\)-stable instances of \(k\)-terminal cut stated that the problem can be solved in polynomial time for \(\gamma \geq 4\) by solving a certain linear program. This result was recently improved to \(\gamma \geq 2 - \frac{2}{k}\) using the same linear program. In this paper, we match the result with a completely different approach, showing that \(\gamma\)-stable instances of \(k\)-terminal cut can be solved in polynomial time for \(\gamma \geq 2 - \frac{2}{k}\), with a faster algorithm. The result is surprising: we show that a known \((2 - \frac{2}{k})\)-approximation algorithm for the problem actually delivers the unique optimal solution for \((2 - \frac{2}{k})\)-stable graphs. For all graphs, including those which are not \((2 - \frac{2}{k})\)-stable, there is an easy-to-check certificate to determine if the output of the algorithm is optimal. The algorithm utilizes only minimum cut procedures, obviating the use of linear programming.

1 Introduction

The \(k\)-terminal cut problem is defined on an edge-weighted graph with \(k\) distinct vertices called “terminals.” The goal is to remove a minimum weight collection of edges from the graph such that there is no path between any pair of terminals. The \(k\)-terminal cut problem is known to be NP-hard [5].

There has been interest in determining special classes of graphs for which \(k\)-terminal cut can be solved in polynomial time. It is known, for example, that the \(k\)-terminal cut problem can be solved in polynomial time on planar graphs when \(k\) is fixed [5]. This is not true for general graphs, since \(k\)-terminal cut is NP-hard even when \(k = 3\).

One class of graphs where \(k\)-terminal cut can be solved in polynomial time is known under the term of \textit{parametrized complexity}. A problem is said to be Fixed-Parameter Tractable (FPT) with respect to a parameter if there is an algorithm for it with running time of the form \(O(f(p)n^{O(1)})\), where \(f\) is any function, \(n\) is the size of the instance, and \(p\) is the value of the parameter. It was first proven in 2004 that \(k\)-terminal cut is fixed-parameter tractable with respect to the weight of the optimal solution \(w(E_{OPT})\) [9]. An improved algorithm was found in 2009 with running time \(O(w(E_{OPT})4^{w(E_{OPT})}n^3)\) [4]. A simpler algorithm generating the same result as in [4] was derived by [11] as a by-product of a branch-and-bound procedure.

∗Fellow, NPSC
Another type of instances for which a polynomial time algorithm for \textit{k-terminal cut} exists is \( \gamma \)-stable graphs, for sufficiently large \( \gamma \). The intuition behind the notion of a \textit{stable} solution is that an optimal solution is stable if some parameters of the problem instance can be perturbed yet the solution remains uniquely optimal. For example, in linear programming, a solution may be considered stable if the coefficients of the variables in the objective function are changed by a small multiplicative factor while maintaining the optimal basic solution. Some studies of the concept of robustness in linear programming may be viewed also through the lens of “stability” [10, 2].

Bilu and Linial, in [3], introduced the concept of stability for graph cut problems, where a graph instance is said to be \( \gamma \)-stable if the optimal cut solution remains uniquely optimal when every edge in the cut is multiplied by a factor up to \( \gamma \). Specifically, they showed that the MaxCut problem on \( \gamma \)-stable instances is solved in polynomial time for \( \gamma \geq \sqrt{n} \), where \( n \) is the number of vertices in the graph and \( \Delta \) is the maximum degree. Analogously, an instance of \textit{k-terminal cut} is said to be \( \gamma \)-stable if the optimal solution remains uniquely optimal when every edge in it is multiplied by a factor up to \( \gamma \). Makarychev, Makarychev, and Vijayaraghavan [8] showed that for 4-stable instances of \textit{k-terminal cut}, the solution to a certain linear programming relaxation of the problem will necessarily be integer. They concluded that for 4-stable instances (and, consequentially, for all \( \gamma \)-stable instances with \( \gamma \geq 4 \)), an efficient algorithm exists for solving \textit{k-terminal cut}. This result stood as the best-known for several years. Recently, Angelidakis, Makarychev, and Makarychev [1] improved the result to \( \gamma \geq 2 - 2/k \) using the same linear programming technique. Here, we arrive at the same result from a completely different approach. That is, we show that the \textit{k-terminal cut} is solved in polynomial time for \( \gamma \)-stable graphs for \( \gamma \geq 2 - 2/k \). However, we do so without the use of linear programming and use only a minimum \((s, t)\)-cut procedure.

Our optimization algorithm for \textit{k-terminal cut} on \((2 - 2/k)\)-stable graphs is effectively the \((2 - 2/k)\)-approximation for the \textit{k-terminal cut} problem introduced by Dahlhaus et al. in [5]. We refer to their algorithm as ISO (Algorithm 1, Section 2). ISO works by computing \( k \) minimum cuts which separate one terminal from the rest and taking their union. The union of all but the largest such cut is known to be a feasible \( k \)-terminal cut with weight at most \((2 - 2/k)\) times that of the optimal \( k \)-terminal cut. A useful property of ISO is that for all graphs, including those which are not \( \gamma \)-stable, there is a sufficient criterion for the output of the algorithm to be optimal that is also easy to check.

Our main contribution here is to show that, for \((2 - 2/k)\)-stable graphs, the \((2 - 2/k)\)-approximation algorithm ISO delivers the optimal solution to the \textit{k-terminal cut} problem.

In Section 2 we introduce definitions and notation. In Section 3 we prove some properties of \( \gamma \)-stable graphs. In Section 4 we show the main result: that \((2 - 2/k)\)-stable instances of \textit{k-terminal cut} can be efficiently solved by ISO. We also show that the converse is not necessarily true: instances which can be efficiently solved with ISO are not necessarily \((2 - 2/k)\)-stable.

## 2 Preliminaries

The notation \( \{G = (V, E), w, T\} \) refers to an instance of the \textit{k-terminal cut} problem, where \( G = (V, E) \) is a graph with vertices \( V \) and edges \( E \). \( T = \{t_1, \ldots, t_k\} \) is a set of \( k \) terminals. The weight function \( w \) is a function from \( E \) to \( \mathbb{R}^+ \). For a subset of edges \( E' \subseteq E \), the notation \( w(E') \) is the total sum of weights of edges in \( E' \). \( \sum_{e \in E'} w(e) \). For an instance \( \{G = (V, E), w, T\} \) of the \textit{k-terminal cut} problem, \( E_{\text{OPT}} \) will refer to an optimal solution: a set of edges of minimum total weight whose removal ensures that there is no path between any pair of terminals.

We first define a notion of \( \gamma \)-perturbation for weighted graphs.

**Definition 1 (\( \gamma \)-Perturbation).** Let \( G = (V, E) \) be a weighted graph with edge weights \( w \). Let \( G' = (V, E) \) be a weighted graph with the same set of vertices \( V \) and edges \( E \) and a new set of edge weights \( w' \) such that, for every \( e \in E \) and some \( \gamma > 1 \),

\[ w(e) \leq w'(e) \leq \gamma w(e). \]

Then \( G' \) is a \( \gamma \)-perturbation of \( G \).
We define stable instances as instances where the optimal solution remains uniquely optimal for any $\gamma$-perturbation of the weighted graph.

**Definition 2** ($\gamma$-Stability). Let $\gamma > 1$. An instance $\{G = (V,E), w, T\}$ of $k$-terminal cut is $\gamma$-stable if there is an optimal solution $E_{OPT}$ which is uniquely optimal for $k$-terminal cut for every $\gamma$-perturbation of $G$.

Note that the optimal solution need not be $\gamma$ times as good as any other solution, since two solutions may share many edges. Given an alternative feasible solution, $E_{ALT}$, to the optimal cut, $E_{OPT}$, in a $\gamma$-stable instance, we can make a statement about the relative weights of the edges where the cuts differ:

**Lemma 1** ($\gamma$-Stability). Consider an instance $\{G = (V,E), w, T\}$ of $k$-terminal cut with optimal cut $E_{OPT}$. Let $\gamma > 1$. $G$ is $\gamma$-stable iff for every alternative feasible $k$-terminal cut $E_{ALT} \neq E_{OPT}$, we have

$$w(E_{ALT} \setminus E_{OPT}) > \gamma w(E_{OPT} \setminus E_{ALT}).$$

**Proof.** First, note that $E_{ALT}$ cannot be a strict subset of $E_{OPT}$ (since $E_{OPT}$ is optimal) and that the claim is trivial if $E_{OPT}$ is a strict subset of $E_{ALT}$. Thus, for the rest of the proof we can assume that both $E_{ALT} \setminus E_{OPT}$ and $E_{OPT} \setminus E_{ALT}$ are non-empty.

For the “if” direction, consider an arbitrary $\gamma$-perturbation of $G$ in which the edge $e$ is multiplied by $\gamma_e$. We first derive the following two inequalities,

$$\sum_{e \in E_{OPT}} \gamma_e w(e) = \sum_{e \in E_{OPT} \cap E_{ALT}} \gamma_e w(e) + \sum_{e \in E_{OPT} \setminus E_{ALT}} \gamma_e w(e) \leq \sum_{e \in E_{OPT} \cap E_{ALT}} \gamma_e w(e) + \gamma w(E_{OPT} \setminus E_{ALT}),$$

and

$$\sum_{e \in E_{ALT}} \gamma_e w(e) = \sum_{e \in E_{OPT} \cap E_{ALT}} \gamma_e w(e) + \sum_{e \in E_{ALT} \setminus E_{OPT}} \gamma_e w(e) \geq \sum_{e \in E_{OPT} \cap E_{ALT}} \gamma_e w(e) + w(E_{ALT} \setminus E_{OPT}).$$

Since we have the inequality

$$w(E_{ALT} \setminus E_{OPT}) > \gamma w(E_{OPT} \setminus E_{ALT}),$$

we conclude that

$$\sum_{E_{OPT}} \gamma_e w(e) < \sum_{E_{ALT}} \gamma_e w(e).$$

Hence, $E_{OPT}$ remains uniquely optimal in any $\gamma$-perturbation.

For the “only if” direction, if $G$ is $\gamma$-stable, then we can multiply each edge in $E_{OPT}$ by $\gamma$ and $E_{OPT}$ will still be uniquely optimal:

$$w(E_{ALT} \setminus E_{OPT}) + \gamma w(E_{ALT} \cap E_{OPT}) > \gamma w(E_{OPT} \setminus E_{ALT}) + \gamma w(E_{ALT} \cap E_{OPT}).$$

Thus,

$$w(E_{ALT} \setminus E_{OPT}) > \gamma w(E_{OPT} \setminus E_{ALT}).$$

We next make a few observations about $\gamma$-stability:
**Fact 1.** Any $k$-terminal cut instance that is stable with $\gamma > 1$ must have a unique optimal solution.

*Proof.* By Definition 1 any graph is a $\gamma$-perturbation of itself. Thus, by Definition 2 the optimal solution must be unique. \qed

**Fact 2.** Any $k$-terminal cut instance that is $\gamma_2$-stable is also $\gamma_1$-stable for any $1 < \gamma_1 < \gamma_2$.

*Proof.* The set of $\gamma_1$-perturbations is a subset of the set of $\gamma_2$-perturbations, since

$$w(e) \leq w'(e) \leq \gamma_1w(e) \implies w(e) \leq w'(e) \leq \gamma_2w(e).$$

Thus, for example, every instance which is 4-stable is necessarily 2-stable, but not the other way around. We capture this relation in Figure 1. \qed

The next two facts are not used directly in our proof, but they give intuition for how $\gamma$-stable instances can be constructed and manipulated.

**Fact 3.** To construct a $\gamma$-stable instance, take any instance where the unique optimal cut $E_{OPT}$ is known and multiply all the edges which are not in the optimal cut ($E \setminus E_{OPT}$) by a factor of $\gamma$.

*Proof.* Here, we use Lemma 1. Let $G'$ be the new graph with weight function $w'$. If $E_{OPT}$ was the unique optimal in $G$, then for any alternate cut $E_{ALT}$ we have

$$w(E_{ALT} \setminus E_{OPT}) > w(E_{OPT} \setminus E_{ALT}).$$

By construction,

$$w'(E_{ALT} \setminus E_{OPT}) = \gamma w(E_{ALT} \setminus E_{OPT})$$
$$w'(E_{OPT} \setminus E_{ALT}) = w(E_{OPT} \setminus E_{ALT}),$$

so

$$w'(E_{ALT} \setminus E_{OPT}) > \gamma w'(E_{OPT} \setminus E_{ALT}).$$ \qed

**Fact 4.** Stability is scale-invariant: that is, if $\{G, w, T\}$ is a $\gamma$-stable instance then, for $\alpha > 0$, the instance $\{G, \alpha w, T\}$, in which every edge weight is multiplied by $\alpha$, is also $\gamma$-stable.
Proof. Here, we use Lemma 1. For any alternative cut $E_{\text{ALT}}$, we have

$$w(E_{\text{ALT}} \setminus E_{\text{OPT}}) > \gamma w(E_{\text{OPT}} \setminus E_{\text{ALT}}) \iff \alpha w(E_{\text{ALT}} \setminus E_{\text{OPT}}) > \gamma \alpha w(E_{\text{OPT}} \setminus E_{\text{ALT}}).$$

Now we introduce the algorithm ISO, which finds a feasible $k$-terminal cut in the graph of at most $(2 - 2/k)$ times the weight of the optimal $k$-terminal cut. The algorithm ISO uses minimum $(t_i, T \setminus t_i)$-cuts. These are minimum $(s, t)$-cuts where one terminal $t_i$ is the source node $s$ and the rest of the terminals $T \setminus t_i$ are shrunk into a sink node $t$. We refer to these cuts as a minimum $t_i$-isolating cuts.

**Definition 3 (Minimum $t_i$-Isolating Cut).** The minimum $t_i$-isolating cut is a minimum $(s, t)$-cut which separates source terminal $s = t_i$ from all the other terminals shrunk into a single sink terminal $t = T \setminus \{t_i\}$.

See Algorithm 1 for a description of ISO, which takes the union of all the $t_i$-isolating cuts except the largest. We use $E_i$ to denote the set of edges in the minimum $t_i$-isolating cut and we use $E_{\text{ISO}}$ to denote the output of the algorithm.

Algorithm 1 ISO

\begin{verbatim}
for $i = 1, \ldots, k$ do
    $E_i \leftarrow$ the set of edges in a $t_i$-isolating cut.
end for
$I \leftarrow \text{argmax}_i w(E_i)$
$E_{\text{ISO}} = \bigcup_{i \neq I} E_i$
\end{verbatim}

The following lemma is due to [5]:

**Lemma 2.** Algorithm ISO returns a 2-approximation for the optimal $k$-terminal cut:

$$w(E_{\text{OPT}}) \leq w(E_{\text{ISO}}) \leq (2 - 2/k)w(E_{\text{OPT}}).$$

Proof. Consider the set of edges in $E_{\text{OPT}}$ with an endpoint reachable from $t_i$. Call this set of edges $E_{i, \text{OPT}}$. These edges are a $t_i$-isolating cut, so they must be at least as large as the minimum $t_i$-isolating cut:

$$w(E_i) \leq w(E_{i, \text{OPT}}).$$

Summing these inequalities for all the $i$, we have

$$\sum_i w(E_i) \leq \sum_i w(E_{i, \text{OPT}}).$$

An edge can be at most double-counted in the sum over $E_{i, \text{OPT}}$. It will be counted once for each terminal $t_i$ that is reachable from one of its endpoints (and there is at most one terminal reachable from each endpoint). Thus,

$$\sum_i w(E_i) \leq \sum_i w(E_{i, \text{OPT}}) \leq 2w(E_{\text{OPT}}).$$

Finally, we know that $E_{\text{ISO}}$ was created by combining all the $t_i$-isolating cuts except the largest. The largest $t_i$-isolating cut has at least $\frac{1}{k}$ of the total weight of all the $t_i$-isolating cuts, so we have:

$$w(E_{\text{ISO}}) \leq \sum_{i \neq I} w(E_i) \leq (1 - 1/k) \sum_i w(E_i).$$

Combining all these inequalities yields

$$w(E_{\text{ISO}}) \leq (2 - 2/k)w(E_{\text{OPT}}).$$

$\square$
Certificate of Optimality  Lemma 4.2 of \cite{5} states that the set of nodes which remain connected to \( t_i \) after the edges in \( E_i \) are removed (the source set of the \( t_i \)-isolating cut) will necessarily remain connected to \( t_i \) when the edges of \( E_{\text{OPT}} \) are removed. It follows that, for any graph (not necessarily stable), the output of ISO is optimal if every node appears in exactly one of these \( k \) source sets. Or, equivalently, if every edge that appears in \( \cup_i E_i \) appears in exactly two of the sets \( E_1, E_2, \ldots, E_k \).

3 Properties of Instances with Edges Removed

In this section, we introduce properties of instances of \( k \)-terminal cut that are hereditary in some sense. We show that when an edge from the optimal cut solution is removed from a \( \gamma \)-stable instance, the resulting instance is still \( \gamma \)-stable (Lemma 3). We also show that when an edge from a minimum \( t_i \)-isolating cut is removed from a graph, the remaining edges are a minimum \( t_i \)-isolating cut in the resulting graph (Lemma 4).

Lemma 3. Let \( \{G = (V, E), w, T\} \) be a \( \gamma \)-stable instance of the \( k \)-terminal cut problem (\( \gamma > 1 \)). Let \( E_{\text{OPT}} \) be the unique optimal solution. Let \( \{G' = (V, E \setminus \{e\}), w, T\} \) be the instance created by removing an arbitrary edge \( e \in E_{\text{OPT}} \) from \( G \). The resulting instance is a \( \gamma \)-stable instance of the \( k \)-terminal cut problem. Furthermore, the unique optimal cut in \( G' \) is exactly \( E_{\text{OPT}} \setminus e \).

Proof. We prove the latter half of the claim first.

Assume that there is some new optimal cut \( E_{\text{OPT}}' \) in \( G' \), so \( w(E_{\text{OPT}}') \leq w(E_{\text{OPT}} \setminus e) \). If \( E_{\text{OPT}}' \) is a \( k \)-terminal cut in \( G' \), then \( E_{\text{OPT}}' \cup \{e\} \) is a \( k \)-terminal cut in \( G \), so

\[
w(E_{\text{OPT}}' \cup \{e\}) = w(E_{\text{OPT}}') + w(e) \leq w(E_{\text{OPT}} \setminus e) + w(e) = w(E_{\text{OPT}}).
\]

Since \( E_{\text{OPT}} \) is the unique optimal solution (Fact 1), this is a contradiction. Hence, \( E_{\text{OPT}}' = E_{\text{OPT}} \setminus e \).

To prove that \( \{G', w, T\} \) is also \( \gamma \)-stable, let \( E_{\text{OPT}}' = E_{\text{OPT}} \setminus e \) and let \( E_{\text{ALT}}' \neq E_{\text{OPT}}' \) be an arbitrary \( k \)-terminal cut in \( G' \).

It must be true that \( E_{\text{ALT}}' \cup \{e\} \) is a cut in \( G \), since \( e \) is the only edge added to \( G \) from \( G' \). We have

\[
w(E_{\text{ALT}}' \setminus E_{\text{OPT}}') = w(E_{\text{ALT}}' \cup \{e\} \setminus E_{\text{OPT}}' \cup \{e\}) > \gamma w(E_{\text{OPT}}' \cup \{e\} \setminus E_{\text{ALT}}' \cup \{e\}) = \gamma w(E_{\text{OPT}}' \setminus E_{\text{ALT}}').
\]

The first and last equality follow from the fact that \( e \) is added or removed from both cuts. The middle inequality follows from Lemma 1.

Thus, \( \{G', w, T\} \) is \( \gamma \)-stable. \( \square \)

Lemma 4. Let \( \{G = (V, E), w, T\} \) be an instance of the \( k \)-terminal cut problem. Let \( E_i \) be a minimum \( t_i \)-isolating cut. Let \( \{G'_i = (V, E \setminus \{e_i\}), w, T\} \) be the instance created by removing an arbitrary edge \( e_i \in E_i \) from \( G \). Then in \( G'_i \), \( E_i \setminus e_i \) is a minimum \( t_i \)-isolating cut.

Proof. Assume that there is some new optimal \( t_i \)-isolating cut \( E'_i \) in \( G'_i \), so \( w(E'_i) < w(E_i \setminus e_i) \). If \( E'_i \) is a \( t_i \)-isolating cut in \( G'_i \), then \( E'_i \cup \{e_i\} \) is a \( t_i \)-isolating cut in \( G \), so

\[
w(E'_i \cup \{e_i\}) = w(E'_i) + w(e_i) < w(E_i \setminus e_i) + w(e_i) = w(E_i).
\]

This is a contradiction, so \( E'_i = E_i \setminus e_i \). \( \square \)

4 ISO is optimal for \((2 - 2/k)\)-Stable Instances

Theorem 1. In a \((2 - 2/k)\)-stable instance of the \( k \)-terminal cut problem, \( \{G = (V, E), w, T\} \), the \((2 - 2/k)\)-approximate solution \( E_{\text{ISO}} \) is also the unique optimal solution \( E_{\text{OPT}} \):

\[
E_{\text{OPT}} = E_{\text{ISO}}.
\]
Figure 2: Counter-example showing that the converse of Theorem 1 is false.

Proof. To start, we prove a slightly different claim. We will prove that \( E_{\text{OPT}} = \cup_i E_i \), the union of \( t_i \)-isolating cuts without the largest one removed. We will primarily consider three sets: \( E_{\text{OPT}} \), the optimal set of edges to cut in \( G \), \( \cup_i E_i \), the union of the minimum \( (t_i, T \setminus t_i) \)-cuts, and \( E_{\text{INT}} = E_{\text{OPT}} \cap (\cup_i E_i) \), the intersection of the previous two sets.

Assume, for contradiction, that \( E_{\text{OPT}} \neq \cup_i E_i \). Consider the instance \( \{G', w, T\} \) is a \((2 - 2/k)\)-stable instance. Let \( E'_{\text{OPT}} \) be the optimal \( k \)-terminal cut in \( G' \) and let \( E'_i \) be a minimum \( t_i \)-isolating cut in \( G' \). By Lemmas 3 and 4, \( E'_i = E_i \setminus E_{\text{INT}} \) and \( E'_{\text{OPT}} = E_{\text{OPT}} \setminus E_{\text{INT}} \). Let \( E'_{\text{ISO}} \) be the union of all the \( E'_i \) except the one with largest weight.

By construction, \( E'_{\text{ISO}} \) and \( E'_{\text{OPT}} \) are disjoint. Applying Lemma 1 in the \((2 - 2/k)\)-stable graph \( G' \), we have \( w(E'_{\text{ISO}} \setminus E'_{\text{OPT}}) > (2 - 2/k)w(E'_{\text{OPT}} \setminus E_{\text{ISO}}) \). Since \( E'_{\text{ISO}} \) and \( E'_{\text{OPT}} \) are disjoint, we can simplify to
\[
w(E'_{\text{ISO}}) > (2 - 2/k)w(E'_{\text{OPT}}) \tag{1}
\]

But, on the other hand, \( E'_{\text{ISO}} \) is the union of all minimum \( t_i \)-isolating cuts except the one with largest weight in \( G' \). From Lemma 2, we have
\[
w(E'_{\text{ISO}}) \leq (2 - 2/k)w(E'_{\text{OPT}}) \tag{2}
\]

The inequalities (1) and (2) contradict. Thus, \( E_{\text{OPT}} = \cup_i E_i \).

Since \( \cup_i E_i \) is the union of all the isolating cuts, \( E_{\text{ISO}} \subseteq \cup_i E_i \), so \( E_{\text{ISO}} \subseteq E_{\text{OPT}} \). But \( E_{\text{ISO}} \) is a feasible cut, so it cannot be a strict subset of \( E_{\text{OPT}} \). We conclude that \( E_{\text{ISO}} = E_{\text{OPT}} \). \( \square \)

Corollary 1. In a \((2 - 2/k)\)-stable instance of the \textbf{k-terminal cut} problem, the (optimal) solution \( E_{\text{ISO}} \) is unique.

It is worth noting that the converse of Theorem 1 is false. That is, there exist instances of \textbf{k-terminal cut} on which \( E_{\text{ISO}} = E_{\text{OPT}} \) but for which the stability may be arbitrarily small. Consider, for example, the five-node, three-terminal graph in Figure 2. Algorithm ISO returns the optimal solution, but the graph is only \((1 + \epsilon)\)-stable, where \( \epsilon > 0 \) can be made arbitrarily small.

Let \( T(n, m) \) be the running time of a minimum \((s, t)\)-cut algorithm on a graph with \( n \) nodes and \( m \) edges. ISO requires using a minimum \((s, t)\)-cut algorithm \( k \) times, so its running time overall is \( O(kT(n, m)) \). Using, for example, the push-relabel or HPF (pseudoflow) algorithms, \( T(n, m) = O(mn \log \frac{n^2}{m}) \) [6, 7]. This is a considerable improvement over algorithms for solving linear programs (used in 8 and 11).

5 Conclusions

In this paper, we demonstrate that the \textbf{k-terminal cut} problem is polynomial-time solvable on \( \gamma \)-stable graphs for \( \gamma \geq (2 - 2/k) \). This improves a previous result for \( \gamma \)-stable graphs for \( \gamma \geq 4 \) and matches a recent result for \( \gamma \geq (2 - 2/k) \). Our result speeds up the complexity from that required to solve a linear programming problem to the complexity of \( k \) minimum cuts. The optimal solution to \textbf{k-terminal cut} on \( \gamma \)-stable graphs for \( \gamma \geq (2 - 2/k) \) is also shown to be a \((2 - 2/k)\)-approximation for \textbf{k-terminal cut} in
general graphs and includes an easy-to-check certificate to determine if its output is in fact optimal in general graphs. It is interesting to investigate whether one can reduce further the value of $\gamma$ or demonstrate that it is impossible under some conditions.

References

[1] Haris Angelidakis, Konstantin Makarychev, and Yury Makarychev. Algorithms for stable and perturbation-resilient problems. In Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, pages 438–451. ACM, 2017.

[2] Aharon Ben-Tal and Arkadi Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. Mathematical programming, 88(3):411–424, 2000.

[3] Yonatan Bilu and Nathan Linial. Are stable instances easy? Combinatorics, Probability and Computing, 21(5):643–660, 2012.

[4] Jianer Chen, Yang Liu, and Songjian Lu. An improved parameterized algorithm for the minimum node multiway cut problem. Algorithmica, 55(1):1–13, 2009.

[5] Elias Dahlhaus, David S. Johnson, Christos H. Papadimitriou, Paul D. Seymour, and Mihalis Yannakakis. The complexity of multiterminal cuts. SIAM Journal on Computing, 23(4):864–894, 1994.

[6] Andrew V Goldberg and Robert E Tarjan. A new approach to the maximum-flow problem. Journal of the ACM (JACM), 35(4):921–940, 1988.

[7] Dorit S Hochbaum and James B Orlin. Simplifications and speedups of the pseudoflow algorithm. Networks, 61(1):40–57, 2013.

[8] Konstantin Makarychev, Yury Makarychev, and Aravindan Vijayaraghavan. Bilu-linial stable instances of max cut and minimum multiway cut. In Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms, pages 890–906. Society for Industrial and Applied Mathematics, 2014.

[9] Dániel Marx. Parameterized graph separation problems. In International Workshop on Parameterized and Exact Computation, pages 71–82. Springer, 2004.

[10] Stephen M Robinson. A characterization of stability in linear programming. Operations Research, 25(3):435–447, 1977.

[11] Mark Velednitsky and Dorit Hochbaum. Isolation branching: A branch and bound algorithm for the k-terminal cut problem. 2018.