Mimicking Networks and Succinct Representations of Terminal Cuts*

Robert Krauthgamer Inbal Rika
Weizmann Institute of Science
{robert.krauthgamer,inbal.rika}@weizmann.ac.il

Abstract

Given a large edge-weighted network G with k terminal vertices, we wish to compress it and store, using little memory, the value of the minimum cut (or equivalently, maximum flow) between every bipartition of terminals. One appealing methodology to implement a compression of G is to construct a *mimicking network*: a small network G' with the same k terminals, in which the minimum cut value between every bipartition of terminals is the same as in G. This notion was introduced by Hagerup, Katajainen, Nishimura, and Ragde [JCSS '98], who proved that such G' of size at most 2^{2^k} always exists. Obviously, by having access to the smaller network G', certain computations involving cuts can be carried out much more efficiently.

We provide several new bounds, which together narrow the previously known gap from doubly-exponential to only singly-exponential, both for planar and for general graphs. Our first and main result is that every k-terminal planar network admits a mimicking network G' of size $O(k^2 2^{2k})$, which is moreover a minor of G. On the other hand, some planar networks G require $|E(G')| \geq \Omega(k^2)$. For general networks, we show that certain bipartite graphs only admit mimicking networks of size $|V(G')| \geq 2^{\Omega(k)}$, and moreover, every data structure that stores the minimum cut value between all bipartitions of the terminals must use $2^{\Omega(k)}$ machine words.

1 Introduction

These days, more than ever, we deal with huge graphs such as social networks, communication networks, roadmaps and so forth. But even when our main interest is only in a small portion of the input graph G, we still need to process all or most of it in order to answer our query, since the runtime and memory requirements of many common graph algorithms depend on the input (graph) size. Therefore, a natural question is whether we can find a smaller graph G' that exactly (or approximately) preserves some property of the original graph such as distances, cuts and connectivity. This basic concept is known as a *graph compression* and was first introduced by Feder and Motwani [FM95], although their definition was slightly different technically. They require that the compressed graph has fewer edges than the original graph, and that each graph can be quickly computed from the other one. They have demonstrated how this paradigm leads to significantly improved running time by implementing it for several graph problems.

Yet another significant advantage of the compressed graph G' is that it requires far less memory than storing the original graph G, which could be critical for machines with limited resources such as smartphones, assuming that the preprocessing can be executed in advance on much more powerful machines. This paradigm becomes indispensable when computations on the compressed graph are to be preformed repeatedly (after a one-time preprocessing).

We focus on cuts and flows, which are of fundamental importance in computer science, engineering, and operations research, because of their frequent usage in many application areas. Specifically, we study the compression of a large graph G containing k "important" vertices (called terminals), into a smaller graph G' containing the same terminals, while maintaining the following condition: the minimum cut between every bipartition of the terminals has exactly the same value in G and in G'. The above cut condition can be also stated in terms of maximum flow, because it effectively deals with the single-source single-sink case, for which we have the max-flow min-cut theorem. We now turn to define this problem more formally, restricting our attention (throughout) to undirected graphs.

A *network* (G, c) is a graph G with an edge-costs function $c : E(G) \to \mathbb{R}^+$. The *size* of a network is

*A full version [KR12] of this extended abstract appears at http://arxiv.org/abs/1006.3970. This work was supported in part by The Israel Science Foundation (grant #452/08), by a US-Israel BSF grant #2010418, and by the Citi Foundation.

its number of vertices of G. The network is called a *k-terminal network* if the graph G has k distinguished vertices called *terminals*, denoted $Q = \{q_1, \ldots, q_k\} \subseteq V(G)$. In such a network, a cut $(W, V(G) \setminus W)$ is said to be *S-separating* if it separates the terminals subset $S \subset Q$ from the remaining terminals $\bar{S} := Q \setminus S$, i.e. if $W \cap Q$ is either S or $\bar{S}$. When clear from the context, $(W, V(G) \setminus W)$ may refer not only to a bipartition of the vertices, but also to its corresponding *cutset* (set of edges crossing the cut). The cost of a cut $(W, V(G) \setminus W)$ is the sum of costs of all the edges in the cutset. We let $\mathrm{mincut}_{G,c}(S, \bar{S})$ denote an S-separating cut in the network (G, c) of minimum cost (breaking ties arbitrarily). With a slight abuse of notation, we use the same notation to denote also the cost of the that cut. We also omit the subscript c when clear from the context.

Definition 1. (Mimicking Network [HKNR98]) *Let (G, c) be a k-terminal network. A* mimicking network *of (G, c) is a k-terminal network (G', c') with the same set of terminals Q, such that for all $S \subset Q$,*[1]

$$\mathrm{mincut}_{G',c'}(S, \bar{S}) = \mathrm{mincut}_{G,c}(S, \bar{S}).$$

The above definition (albeit for directed networks) was introduced by Hagerup, Katajainen, Nishimura, and Ragde [HKNR98], who proved that every k-terminal network (G, c) admits a mimicking network of size at most 2^{2^k}. Subsequently, Chaudhuri, Subrahmanyam, Wagner, and Zaroliagis [CSWZ00] studied specific graph families, showing an improved upper bound of $O(k)$ for graphs G that have bounded treewidth. For the special case of outerplanar graphs G, the mimicking network G' they construct is furthermore outerplanar. Some of these previous results hold also for directed networks.

The only lower bound we are aware of on the size of mimicking networks is $k + 1$ for every $k > 3$, even for a star graph, due to [CSWZ00]. For $k = 4, 5$ they further show a matching upper bound. These results are summarized in Table 1. We mention that several other variants of the problem were studied in the literature, in particular when cut values are preserved approximately, see Section 1.2 for details.

1.1 Our Results

(a) Upper bounds. We first prove (in Section 2) a new upper bound for planar graphs, which significantly improves over the bound that follows from previous work (namely, 2^{2^k} known for general graphs [HKNR98]). See also Table 1 for the known bounds.

[1]Throughout, we omit the trivial exclusion $S \neq \emptyset, Q$.

Theorem 1.1. *Every planar k-terminal network (G, c) admits a mimicking network of size at most $O(k^2 2^{2k})$, which is furthermore a minor of G.*

Notice that our theorem constructs for an input graph G a mimicking network that is actually a minor of it, and thus preserves additional properties of G such as planarity.

(b) Lower bounds. We further provide (in Section 3) two nontrivial lower bounds. See Table 1 for comparison with the known bounds. The following theorem addresses general graphs, and narrows the previous doubly-exponential gap (between $k + 1$ and 2^{2^k}) to be only singly-exponential.

Theorem 1.2. *For every $k > 5$ there exists a k-terminal network such that every mimicking network of it has size $2^{\Omega(k)}$. This holds even for bipartite networks with all the terminals on one side and all the non-terminals on the other side.*

The next theorem is for mimicking networks of planar graphs, proving a lower bound on the number of *edges*. If the mimicking network is guaranteed to be sparse (say planar, as is the case in our bound in Theorem 1.1) then we get a similar bound for the number of vertices. But if the mimicking network could be arbitrary (e.g., a complete graph) we do not know how to prove it cannot have $O(k)$ vertices.

Theorem 1.3. *For every $k > 5$ there exists a planar k-terminal network such that every mimicking network of it has at least $\Omega(k^2)$ edges.*

Remark. Very recently, we were informed of new results, obtained independently of ours, by Khan, Raghavendra, Tetali and Végh [KRTV12]. Their results include improved upper bounds for general graphs (albeit still doubly-exponential in k), for trees, and for bounded treewidth graphs, as well as lower bounds that are comparable to ours.

(c) Succinct data structures. Our final result is an alternative formulation of graph compression as the problem of storing succinctly (i.e., summarizing or sketching) all the 2^k terminal cuts in a k-terminal network.

Definition 2. *A* terminal-cuts (TC) scheme *is a data structure that uses storage (memory) M to support the following two operations on a k-terminal network (G, c), where $n = |V(G)|$ and $c : E(G) \to \{1, \ldots, n^{O(1)}\}$.*

1. *Preprocessing P, which gets as input the network and builds M.*

Graph family	Lower bounds		Upper bounds	
General graphs	$2^{\Omega(k)}$	Theorem 1.2	2^{2^k}	[HKNR98]
Planar graphs	$\lvert E(G')\rvert \geq \Omega(k^2)$	Theorem 1.3	$O(k^2 2^{2k})$	Theorem 1.1
Bounded treewidth			$O(k)$	[CSWZ00]
Star graphs	$k+1$	[CSWZ00]		

Table 1: Known bounds for the size of mimicking networks

2. *Query Q, which gets as input a subset of terminals S, and uses M (without access to (G,c)) to output* $\text{mincut}_{G,c}(S,\bar{S})$.

Observe that putting together the two conditions above gives $Q(S;P(G)) = \text{mincut}_{G,c}(S,\bar{S})$ for all $S \subset Q$. The *storage requirement* (or *space complexity*) of the TC scheme is the (maximum) number of machine words used by M. Since the value of every cut in (G,c) is at most $n^{O(1)}$, and since we need to be able to represent every vertex in G, we shall count the size of the TC scheme in terms of machine words of $O(\log n)$ bits. An obvious upper bound is 2^k machine words, by explictly storing a list of all the cut values. Perhaps surprisingly, we can show a matching lower bound for any data structure using the technology developed to prove Theorem 1.2. We prove the following Theorem 1.4, including an extension of it to randomized schemes, in Section 4.

THEOREM 1.4. *For every $k > 5$, a terminal-cuts scheme for k-terminal networks requires storage of $2^{\Omega(k)}$ machine words.*

This theorem is related to, but different from, Theorem 1.2. A TC scheme can possibly use its memory M to store an entire mimicking network; a more naive approach would be to store all the terminal-cut values, using at most 2^k machine words. Indeed, our theorem shows that the worst-case memory usage of this naive approach is essentialy optimal.

1.2 Related Work Graph compression can be interpreted quite broadly, and indeed it was studied extensively in the past, with many results known for different graphical features (the properties we wish to preserve). For instance, in the context of preserving the graph distances, concepts such as spanners [PS89] and probabilistic embedding into trees [AKPW95, Bar96], have developed into a rich area with productive area, and variations of it that involve a subset of terminal vertices were studied more recently, see e.g. [CE06, KZ12].

In the context of preserving cuts (and flows), which is also our theme, the problem of graph sparsification [BK96] has recently seen an immense progress, see [BSS09] and references therein. Even closer to our own work are analogous questions that involve a subset of terminals, and the goal is to find a small network that preserves (the cost of) all minimum terminal cuts *approximately*. In particular, Chuzhoy [Chu12] recently showed a constant factor approximation using a network whose size depends on (certain) edge-costs is in the original graph. Another variation of our problem is that of a cut (and flow) sparsifier, in which the compressed network should contain only k vertices (the terminals) and the goal is to minimize the approximation factor (called congestion), see [CLLM10, EGK+10, MM10] for the latest results.

2 Upper Bound for Planar Graphs

In this section we prove Theorem 1.1, showing that every planar k-terminal network (G,c) admits a mimicking network of size $O(k^2 2^{2k})$, which is in fact a minor of G.

2.1 Technical Outline Let G be a planar k-terminal network, and assume it is connected. Let $E_S = \text{mincut}_{G,c}(S,\bar{S})$ be the cutset of a minimum S-separating cut in (G,c), and let $\hat{E}$ be the union of E_S over all subsets $S \subset Q$. Removing the edges $\hat{E}$ from the graph G disconnects it to some number of connected components, and we construct our mimicking network G' by contracting every such connected component into a single vertex. It is easy to verify that these contractions maintain the minimum terminal cuts. This method of constructing G' resembles the one in [HKNR98], except that the sets of vertices that we unite are always connected, hence our G' is a minor of G. We proceed to bound the number of connected components one gets in this way, as this will clearly be the size of our mimicking network G'.

We first consider removing from G a single cutset $\text{mincut}_G(S,\bar{S})$ (for arbitrary $S \subset Q$), and show (in Lemma 2.1) that it can disconnect the graph into at most k connected components. We then extend this result to removing from G two cutsets, namely $\text{mincut}_G(S,\bar{S})$ and $\text{mincut}_G(T,\bar{T})$ (for arbitrary $S,T \subset$

Q), and show (in Lemma 2.2) such a removal can disconnect the graph into at most $3k$ connected components. Next, we consider removing all the $m = 2^{k-1} - 1$ cutsets of the minimum terminal cuts from G (i.e., $\hat{E}$) from G. However, naive counting of the number of resulting connected components, which argues that every additional cutset splits each existing component into at most $O(k)$ components, would give us in total a poor bound of roughly k^m.

The crucial step here is to use the planarity of G to improve the dependence on m significantly, and we indeed obtain a bound that is quadratic in m by employing the dual graph of G denoted by G^*. Loosely speaking, the cutsets in G correspond to (multiple) cycles in G^*, and thus we consider the dual edges of $\hat{E}$, which may be viewed as a subgraph of G^* comprising of (many) cycles. We now use Euler's formula and the special structure of this subgraph of cycles; more specifically, we count its vertices of degree > 2, which turns out to require the aforementioned bound of $3k$ for two sets of terminals S, T. This gives us a bound on the number of faces in this subgraph (in Lemma 2.4), which in turn is exactly the number of connected components in the primal graph (Corollary 2.2).

2.2 Preliminaries Recall that a graph is called a *multi-graph* if we allow it to have parallel edges and loops. A *cycle* in a multi-graph G is a sequence of edges $(u_0, v_0), \ldots, (u_{l-1}, v_{l-1})$ such that $v_i = u_{(i+1) \mod l}$ for all $i = 0, \ldots, l-1$. The cycle is *simple* if it contains l distinct vertices and l distinct edges. Note that two parallel edges define a simple cycle of length 2, and that a loop is a cycle of length 1 that contributes 2 to the degree of its vertex. A *circuit* is a collection of cycles (not necessarily disjoint) $\mathcal{C} = \{C_1, \ldots, C_l\}$. Let $\mathcal{E}(\mathcal{C}) = \bigcup_{i=1}^{l} C_i$ be the set of edges that participate in one or more cycles in the collection (note it is not a multiset, so we discard multiplicities). The cost of a circuit $\mathcal{C}$ is defined as $\sum_{e \in \mathcal{E}(\mathcal{C})} c(e)$.

For a graph G, let $CC(G)$ denote the set of connected components in the graph. In particular, if $CC(G) = \{P_1, \ldots, P_h\}$ then $V(G) = P_1 \cup \cdots \cup P_h$ as a disjoint union. For a subset of the vertices $W \subset V(G)$, let $\delta(W)$ denote the set of edges with exactly one endpoint in W, i.e. $\delta(W) = \{(u, v) \in E(G) : u \in W, v \notin W\}$. A vertex in G with degree more than 2 will be called a *meeting* vertex of G. We introduce special notation for two (disjoint) sets of vertices:

$$V_2(G) = \{v \in V : \deg(v) = 2\};$$
$$V_m(G) = \{v \in V : \deg(v) > 2\};$$

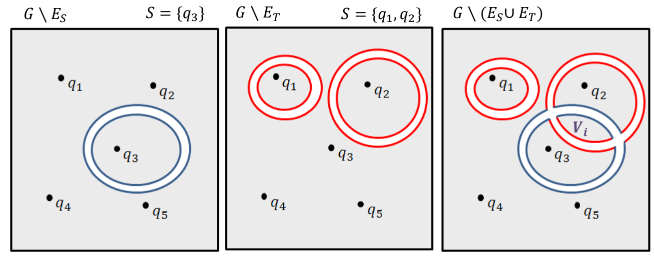

Figure 1: As depicted in gray, $G \setminus E_S$ has two connected components, $G \setminus E_T$ has three, and $G \setminus (E_S \cup E_T)$ has five. Notice the connected component V_i of $G \setminus (E_S \cup E_T)$ contains no terminals.

and for two (disjoint) sets of edges:

$$E_2(G) := \{(u, v) \in E(G) : u, v \in V_2(G)\};$$
$$E_m(G) := \{(u, v) \in E(G) : u \in V_m(G) \text{ or } v \in V_m(G)\}.$$

2.3 Proof of Theorem 1.1 Let (G, c) be a k-terminal network with terminals $Q = \{q_1, \ldots, q_k\}$, where G is a connected plane graph with faces F (if G is not connected we can apply the proof for every connected component separately). We may assume, using small perturbation on the edges cost, that every two different subsets of edges in G have different total cost. In the proof we will use the notations E_S and $\hat{E}$ defined in Section 2.1.

Lemma 2.1. (One cutset) *For every subset of terminals S, the graph $G \setminus E_S$ has at most k connected components.*

Proof. If there are more than k connected components then there is at least one connected component without any terminal vertex. Since G is connected, we can unite it to any other connected component by removing some edge from E_S. We get a new cutset that separates S from $\bar{S}$ with smaller total cost than E_S in contradiction to the minimality. ■

Lemma 2.2. (Two cutsets) *For every two subsets of terminals S and T, the graph $G \setminus (E_S \cup E_T)$ has at most $|CC(G \setminus E_S)| + |CC(G \setminus E_T)| + k$ connected components.*

We illustrate this lemma in Figure 1. The idea is that if $G \setminus (E_S \cup E_T)$ has too many connected components, then we can find one that contains no terminals, and that moving it to the other side of (say) $G \setminus E_S$ contradicts the minimality of E_S.

Proof. [of Lemma 2.2] Let $CC(G \setminus (E_S \cup E_T)) = \{P_0, \ldots, P_h\}$. For every P_i, we let $W_S(P_i) := \delta(P_i) \cap E_S$

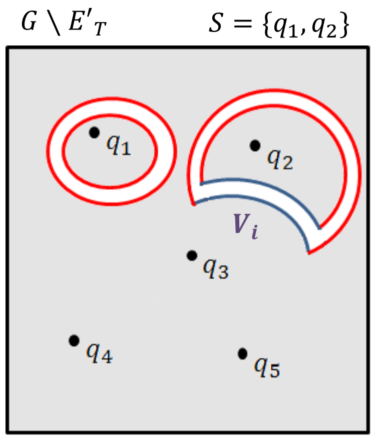

Figure 2: $E'_T = (E_T \cup W_S) \setminus W_T$, where the red edges we removed are W_T, and the blue edges we added are W_S.

be the set of edges in E_S that have exactly one of their endpoints in P_i, and similarly $W_T(P_i) := \delta(P_i) \cap E_T$. We can use the above notation to associate every connected components P_i of $G \setminus (E_S \cup E_T)$, to one of the following four sets:

1. $W_S(P_i) = \emptyset$; in particular, $P_i \in CC(G \setminus E_T)$.
2. $W_T(P_i) = \emptyset$; in particular, $P_i \in CC(G \setminus E_S)$.
3. $W_S(P_i) = W_T(P_i)$; in particular $P_i \in CC(G \setminus E_S) \cap CC(G \setminus E_T)$.
4. $W_S(P_i) \neq \emptyset$, $W_T(P_i) \neq \emptyset$ and $W_S(P_i) \neq W_T(P_i)$; in particular $P_i \notin CC(G \setminus E_S) \cup CC(G \setminus E_T)$.

Every connected component that belongs to the last set (i.e. there are at least two different edges in $\delta(P_i)$, one from E_T and one from E_S) will be called a *mixed* connected component of $G \setminus (E_S \cup E_T)$. Thus, the number of connected components in $G \setminus (E_S \cup E_T)$ is bounded by $|CC(G \setminus E_S)| + |CC(G \setminus E_S)|$ plus the number of mixed connected components of $G \setminus (E_S \cup E_T)$.

Assume towards contradiction that there are more than k mixed connected components in $G \setminus (E_S \cup E_T)$. Therefore, there exists at least one mixed connected component, say with out loss of generality P_0, without any terminal in it. Since P_0 is a mixed connected component in $G \setminus (E_S \cup E_T)$ we know that $W_S(P_0) \neq \emptyset$, $W_T(P_0) \neq \emptyset$ and $W_S(P_0) \neq W_T(P_0)$. For simplicity from now on we will drop the p_0 and refer W_S and W_T to $W_S(P_0)$ and $W_T(P_0)$ correspondingly. By the perturbation on the edges cost the total cost of these two subsets must be different. Assume without loss of generality that $c(W_S) < c(W_T)$. We will replace the edges W_T by the edges W_S in the cutset of T and call this new set of edges E'_T, i.e. $E'_T = (E_T \cup W_S) \setminus W_T$. It is clear that $c(E'_T) < c(E_T)$. We will prove that E'_T is also a cutset that separate T from $\bar{T}$ in the graph G, contradicting the definition of E_T. See Figures 1 and 2.

Denote $CC(G \setminus E_T) = \{P'_0, \ldots, P'_{h'}\}$ and assume without loss of generality that the set of edges W_S connects the connected component P_0 and the t connected components $P_1, \ldots P_t$ of $G \setminus (E_S \cup E_T)$ into one connected component P'_0 in $G \setminus E_T$. Therefore, by adding the edges $E_S \setminus (W_S \cup W_T)$ to the graph $G \setminus (E_S \cup E_T)$ We will get the graph $G' = G \setminus (E_T \cup W_S)$ and its connected components will be $P_0, P_1, \ldots, P_t, P'_1, \ldots, P'_{h'}$. Since the graph G' do not contains any edge from E_T, the sets T and $\bar{T}$ are still separated.

Now it remain to add the edges W_T to the graph G' in order to get the desirable graph $G \setminus E'_T$. Assume without loss of generality that P'_0 contains terminals from T. Then, by the minimality of E_T, if edges from W_T connect between P'_0 and P'_i, then the terminals of P'_i are from $\bar{T}$. In particular, adding the edges W_T to G' will connect P_0 to some connected components P'_i that contains only terminals from $\bar{T}$. Since P_0 does not contains any terminals, the connected component that was combined by the edges W_T contains only terminals from $\bar{T}$, and so E'_T separate between T and $\bar{T}$. ■

Planar duality. Recall that every planar graph G has a dual graph G^*, whose vertices correspond to the faces of G, and whose faces correspond to the vertices of G, i.e., $V(G^*) = \{v^*_f : f \in F(G)\}$ and $F(G^*) = \{f^*_v : v \in V(G)\}$. Every edge $e = (v, u) \in E(G)$ with cost $c(e)$ that lies on the boundary of two faces $f_1, f_2 \in F(G)$ has a dual edge $e^* = (v^*_{f_1}, v^*_{f_2}) \in E(G^*)$ with the same cost $c(e^*) = c(e)$ that lies on the boundary of the faces f^*_v and f^*_u. For every subset of edges $H \subset E(G)$, let $H^* := \{e^* : e \in H\}$ denote the subset of the corresponding dual edges in G^*.

The following theorem describes the duality between two different kinds of edge sets – minimum cuts and minimum circuits – in a plane multi-graph. It is a straightforward generalization of the case of st-cuts (whose duals are cycles) to three or more terminals. We are not aware of a reference for this precise statement, although it is similar to [HS85, Rao87]. See also Figure 3 for illustration.

Theorem 2.1. (Duality of cutsets and circuits) *Let G be a connected plane multi-graph, let G^* be its dual graph, and fix a subset of the vertices $W \subseteq V(G)$. Then, $H \subset E(G)$ is a cutset in G that has minimum cost among those separating W from $V(G) \setminus W$ if and only if the dual set of edges $H^* \subseteq E(G^*)$ is actually $\mathcal{E}(\mathcal{C})$ for a circuit $\mathcal{C}$ in G^* that has minimum cost among those separating the corresponding faces*

$\{f_v^* : v \in W\}$ *from* $\{f_v^* : v \in V(G) \setminus W\}$.

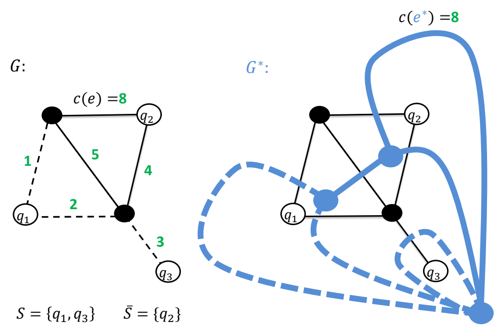

Figure 3: A planar 3-terminal network G (in black), with E_S depicted as dashed edges. The dual graph G^* is shown in blue, with E_S^* depicted as dashed edges.

Recall that removing edges from a graph G disconnects it into (one or more) connected components. The next lemma characterizes this behavior in terms of the dual graph G^*. See Figure 4 for illustration. Recall that $G[H]$ is a standard notation for the subgraph of G induced by the subset (of edges or vertices) H.

Lemma 2.3. (Dual of a connected component) *Let G be a connected plane multi-graph, let G^* be its dual, and fix a subset of edges $H \subset E(G)$. Then P is a connected component in $G \setminus H$ if and only if its dual set of faces $\{f_v^* : v \in P\}$ is a face of $G^*[H^*]$.*

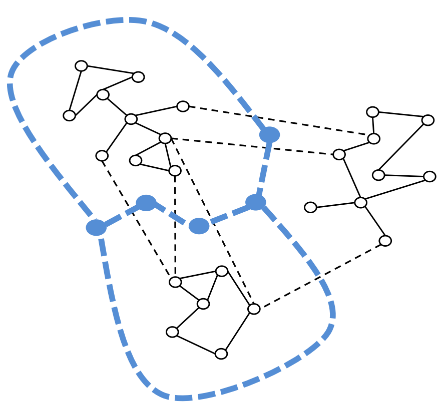

Figure 4: The graph G is in black. Removing the black dashed edges H disconnects the graph G into three connected components. The blue bold dashed edges are the dual edges H^*, that form the dual subgraph $G^*[H^*]$.

Leveraging the planarity. We proceed with the proof of Theorem 1.1, and now use the duality of planar graphs. In the following corollary we will deal with the dual graph $G^*[E_S^* \cup E_T^*]$ for two arbitrary subsets of terminals S and T.

Corollary 2.1. *For all $S, T \subset Q$, the graph $G^*[E_S^* \cup E_T^*]$ has at most $6k$ meeting vertices.*

Proof. [of Corollary 2.1] According Lemmas 2.1 and 2.2, the graph $G \setminus (E_S \cup E_T)$ has at most $|CC(G \setminus E_S)| + |CC(G \setminus E_T)| + k \leq 3k$ connected components. By Lemma 2.3 every connected component in $G \setminus (E_S \cup E_T)$ corresponds to a face in $G^*[E_S^* \cup E_T^*]$. Therefore, $G^*[E_S^* \cup E_T^*]$ has at most $3k$ faces.

By the duality of cuts and circuits, every set of edges E_S^* is a circuit. Therefore, every vertex v appearing in these edges E_S^*, has degree at least 2. $E_S^* \cup E_T^*$ is circuit as well, and all its vertices have degree at least 2, i.e. $V(G^*[E_S^* \cup E_T^*]) = V_2(G^*[E_S^* \cup E_T^*]) \cup V_m(G^*[E_S^* \cup E_T^*])$. To simplify the notation we denote $G_{ST}^* = G^*[E_S^* \cup E_T^*]$. By Handshaking lemma,

$$
\begin{aligned}
2|E(G_{ST}^*)| &= \sum_{v \in V(G_{ST}^*)} \deg(v) \\
&\geq 3|V_m(G_{ST}^*)| + 2|V_2(G_{ST}^*)| \\
&= 2|V(G_{ST}^*)| + |V_m(G_{ST}^*)|.
\end{aligned}
$$

By Euler's formula

$$
\begin{aligned}
3k &\geq |F(G_{ST}^*)| \\
&= |E(G_{ST}^*)| - |V(G_{ST}^*)| + |CC(G_{ST}^*)| + 1 \\
&\geq \frac{1}{2}|V_m(G_{ST}^*)|,
\end{aligned}
$$

and the corollary follows. ■

Recall that in Section 2.1 we defined $\hat{E} := \bigcup_{S \subset Q} E_S$, and denote its set of dual edges by $\hat{E}^* := \{e^* : e \in \hat{E}\} = \bigcup_{S \subset Q} E_S^*$.

Lemma 2.4. *The graph $G^*[\hat{E}^*]$ has at most $O(k^2 2^{2k})$ faces.*

Proof. [of Lemma 2.4] Using Theorem 2.1 we get that for every $S \subset Q$, E_S is a minimum cutset in G if and only if E_S^* (the dual set of edges of E_S) is a minimum circuit in G^*. Moreover, as defined in Section 2.3 $\hat{E}^* = \bigcup_{S \subset Q} E_S^*$. Thus, $\hat{E}^*$ is also a circuit, and so

$$|V(G^*[\hat{E}^*])| = |V_2(G^*[\hat{E}^*])| + |V_m(G^*[\hat{E}^*])|, \tag{2.1}$$

$$|E(G^*[\hat{E}^*])| = |E_2(G^*[\hat{E}^*])| + |E_m(G^*[\hat{E}^*])|. \tag{2.2}$$

According to the definitions and the Handshaking lemma we get that

$$(2.3) \qquad |E_2(G^*[\hat{E}^*])| \le |V_2(G^*[\hat{E}^*])|.$$

By a union bound, the two following inequalities holds

$$(2.4) \quad |V_m(G^*[\hat{E}^*])| \le \sum_{S\subset Q} |V(G^*[E_S^*]) \cap V_m(G^*[\hat{E}^*])|$$

$$(2.5) \quad |E_m(G^*[\hat{E}^*])| \le \sum_{S\subset Q} |E(G^*[E_S^*]) \cap E_m(G^*[\hat{E}^*])|$$

Fix a subset S. We will start by bounding the set of vertices $V(G^*[E_S^*]) \cap V_m(G^*[\hat{E}^*])$. For every vertex v in $V(G^*[E_S^*]) \cap V_m(G^*[\hat{E}^*])$ there exists a subset T such that v is also in $V(G^*[E_S^*]) \cap V_m(G^*[E_S^* \cup E_T^*])$. According to Corollary 2.1, $|V_m(G^*[E_S^* \cup E_T^*])| \le 6k$. Therefore $|V(G^*[E_S^*]) \cap V_m(G^*[E_S^* \cup E_T^*])| \le 6k$, and by union bound on all the subsets T we get $|V(G^*[E_S^*]) \cap V_m(G^*[\hat{E}^*])| \le 6k2^k$.

We will now move to bound $E(G^*[E_S^*]) \cap E_m(G^*[\hat{E}^*])$. By Lemma 2.1 there are at most k cycles that cover the graph $G^*[E_S^*]$, so every vertex in $V(G^*[E_S^*]) \cap V_m(G^*[\hat{E}^*])$ can be shared by at most k cycles of $G^*[E_S^*]$, which bound the degree of every vertex in $G^*[E_S^*]$ by $2k$. Thus

$$(2.6) \quad \begin{aligned} &|E(G^*[E_S^*]) \cap E_m(G^*[\hat{E}^*])| \\ &\qquad \le 2k|V(G^*[E_S^*]) \cap V_m(G^*[\hat{E}^*])| = O(k^2 2^k) \end{aligned}$$

We can bound $|CC(G^*[\hat{E}^*])|$ by extending the argument in Lemma 2.1. Assume toward contradiction that $|CC(G^*[\hat{E}^*])| \ge k+1$. Thus, there exists at least one connected component P in $G^*[\hat{E}^*]$ that does not contains any terminal face of G^*. By the construction of $\hat{E}^*$, there exists a subset S such that P contains at least one cycle C of the circuit E_S^*. Since P does not contain any terminal face, we can remove some edge e^* of the cycle C from the circuit E_S^* and get a circuit with smaller cost that separates between f_S^* and $f_{\bar{S}}^*$ in contradiction.

Now by Euler's formula,

$$\begin{aligned} &|F(G^*[\hat{E}^*])| \\ &\quad = |E(G^*[\hat{E}^*])| - |V(G^*[\hat{E}^*])| + 1 + |CC(G^*[\hat{E}^*])| \\ &\quad \le |E_m(G^*[\hat{E}^*])| - |V_m(G^*[\hat{E}^*])| + 1 + k \\ &\quad \le \textstyle\sum_{S\subset Q} O(k^2 2^k) + 1 + k = O(k^2 2^{2k}) \end{aligned}$$

the first inequality is by Equations (2.1), (2.2) and (2.3), the second inequality is by Equations (2.5) and (2.6), and the lemma follows. ■

COROLLARY 2.2. *There are at most $O(k^2 2^{2k})$ connected components in the graph $G \setminus \hat{E}$.*

This corollary follows from Lemma 2.4 by applying Lemma 2.3 with $H = \hat{E}$. We now complete the proof of Theorem 1.1. Merge the vertices in each connected component of $G \setminus \hat{E}$ into a single vertex (formally, contract all the internal edges in each connected component) and call this new multi-graph M. Notice there is at most one terminal vertex in each connected component. So a vertex in M, which corresponds to a connected component (of $G \setminus \hat{E}$) that contains some terminal vertex q, will be identified with that terminal q. To be concrete, the vertices and the terminals of M are the sets

$$V(M) := \{v_i : \ P_i \in CC(G \setminus \hat{E})\}$$

$$Q(M) := \{q = v_i : \ P_i \in CC(G \setminus \hat{E}) \text{ and } q \in P_i\}$$

In addition, (v_i, v_j) is an edge in M if there exist two vertices $u_i, u_j \in E(G)$ such that $u_i \in P_i$, $u_j \in P_j$ and (u_i, u_j) is an edge in G. The cost of every edge $(v_i, v_j) \in E(M)$ is

$$c'(v_i, v_j) := \sum_{u_i \in P_i, u_j \in P_j:\ (u_i,u_j) \in E(G)} c(u_i, u_j).$$

It is easy to verify that M is a minor of G with $O(k^2 2^{2k})$ vertices that includes the same k terminals Q. We now prove that (M, c') is a mimicking network of G using the same argument as in [HKNR98], but applied to the connected components. Fix a subset of terminals S. Since we only contract edges, every cut that separates S and $\bar{S}$ in M has a cut in G that separates S and $\bar{S}$ with the same cost, thus $\text{mincut}_{M,c'}(S, \bar{S}) \ge \text{mincut}_{G,c}(S, \bar{S})$. In the other direction, notice that by the construction of M, all the vertices in each connected component of $G \setminus \hat{E}$ are on the same side of the minimum S-separating cut in G. Thus, there is a cut in M that separates between S and $\bar{S}$ and has cost $\text{mincut}_G(S, \bar{S})$. Combining these together, we get the equality $\text{mincut}_{M,c'}(S, \bar{S}) = \text{mincut}_{G,c}(S, \bar{S})$ for every S, and Theorem 1.1 follows.

3 Lower Bounds

In this section we prove Theorems 1.2. While Theorem 1.3 belongs here as well, its proof is deferred to the full version [KR12].

3.1 Techniques and Proof Outline All our lower bounds are proved using the same technique, which basically counts the number of "degrees of freedom" needed to express all the relevant cut values. Formally, we develop a certain machinery based on linear algebra, which relates the size of any mimicking network to the rank of some matrix.

The lower bound proofs start by describing a k-terminal network (G, c) that seems minimal in the sense that it does not admit a smaller mimicking network. The networks used in Theorems 1.2 and 1.3 are different, see Section 3 for details. We then identify the minimum cost S-separating cuts for all (or some) $S \subset Q$, and capture this information in a matrix.

DEFINITION 3. (CUTSETS-EDGES INCIDENCE MATRIX) *Let (G, c) be a k-terminal network, and fix an enumeration $S_1, \ldots, S_m$ of all $m = 2^{k-1}-1$ distinct and nontrivial bipartitions $Q = S_i \cup \bar{S}_i$. The* cutset-edge incidence matrix *of (G, c) is the matrix $A_{G,c} \in \{0,1\}^{m \times E(G)}$ given by*

$$(A_{G,c})_{i,e} = \begin{cases} 1 & \text{if } e \in \text{mincut}_{(G,c)}(S_i, \bar{S}_i); \\ 0 & \text{otherwise.} \end{cases}$$

We also define the vector of minimum-cut values between every bipartition of terminals

$$\Phi_{G,c} = \begin{pmatrix} \text{mincut}_{G,c}(S_1, \bar{S_1}) \\ \vdots \\ \text{mincut}_{G,c}(S_m, \bar{S_m}) \end{pmatrix} \in \mathbb{R}^m.$$

Here and throughout, we shall omit the subscript c when it is clear from the context. Observe that if we think of the edge costs c as a column vector $\vec{c} \in (\mathbb{R}^+)^{E(G)}$, then $A_G \cdot \vec{c} = \Phi_G$. For a given $S \subset Q$, a minimum S-separating cut $(W, V(G) \setminus W)$ is called *unique* if all other S-separating cuts have a strictly larger cost.

The core of our analysis is the next lemma, as it immediately provides a lower bound on the size of any mimicking network; the theorems would follow by calculating the rank of A_G.

LEMMA 3.1. (MAIN TECHNICAL LEMMA) *Let (G, c) be a k-terminal network. Let A_G be its cutset-edge incidence matrix, and assume that for all $S \subset Q$ the minimum S-separating cut of G is unique. Then there is for G an edge-costs function $\hat{c} : E(G) \to \mathbb{R}^+$, under which every mimicking network (G', c') satisfies $|E(G')| \geq \text{rank}(A_{G,c})$.*

Notice that the bound is proved not for (G, c) but rather for $(G, \hat{c})$; indeed, the edge-costs $\hat{c}$ are a small random perturbation of c. Thus, the proof of this lemma first shows that a small perturbation does not change the cutset-edge incidence matrix, i.e. $A_{G,c} = A_{G,\hat{c}}$. This is where the uniqueness property is used. Next, fix a small graph G' that can potentially be a mimicking network, but without specifying its edge-costs c'; now let $\mathcal{E}_{G'}$ be the event that $(G, \hat{c})$ admits a mimicking network of the form (G', c'). Since G' has too few edges (whose costs are undetermined/free variables), we can use linear algebra to show that $\Pr[\mathcal{E}_{G'}] = 0$. The lemma then follows by a union bound over the finitely many (unweighted) graphs G' of the appropriate size.

3.2 Proof of Lemma 3.1 We turn to proving Lemma 3.1. Recall that this lemma considers a k-terminal network (G, c), and assuming a certain (uniqueness) condition, asserts that there is for G a modified edge-costs function $\hat{c}$, under which every mimicking network must have at least $\text{rank}(A_{G,c})$ edges, where $A_{G,c}$ is a cutset-edge incidence matrix of (G, c).

The proof employs two lemmas and the following notation. For $S \subset Q$, let $\Delta_{G,c}(S) \geq 0$ be the difference between the two smallest costs among all S-separating cuts in G. Observe that if these two are not equal (i.e., $\Delta_{G,c}(S) > 0$) then the minimum S-separating cut is said to be unique in G. We also denote $\Delta_{G,c} := \min_{S \subset Q} \Delta_{G,c}(S)$.

LEMMA 3.2. *For every edge-costs function $w : E(G) \to [0, \frac{1}{\Delta_{G,c}|E(G)|}]$ the cutset-edge incidence matrix of (G, c) is equal to the cutset-edge incidence matrix of $(G, c+w)$, i.e. $A_{G,c} = A_{G,c+w}$, where $c + w : e \to c(e) + w(e)$.*

Proof. Let w be an edge-costs function $w : E(G) \to [0, \frac{1}{\Delta_{G,c}|E(G)|}]$. Since (G, c) and $(G, c+w)$ have the same vertices and edges, every S_i-separating cut in (G, c) is also a S_i-separating cut in $(G, c+w)$ and vice versa. The value of every such cutset in $(G, c+w)$ is ranged from the value of this cutset in G to the value of this cutset in G plus $\frac{1}{\Delta_{G,c}}$. In particular, $\text{mincut}_{G,c}(S_i, \bar{S}_i) \leq \text{mincut}_{G,c+w}(S_i, \bar{S}_i) \leq \text{mincut}_{G,c}(S_i, \bar{S}_i) + \frac{1}{\Delta_{G,c}}$. Thus, $\text{mincut}_{G,c+w}(S_i, \bar{S}_i)$ is smaller (by at least $\frac{\Delta_{G,c}-1}{\Delta_{G,c}}$) than every cut that separates S_i and $\bar{S}_i$ in G. Therefore it must be the case that the cutsets of the minimum S_i-separating cuts in (G, c) and in $(G, c+w)$ are the same. ∎

We proceed with the proof of Lemma 3.1. Sample an edge-costs function $w : E(G) \to [0, \frac{1}{\Delta_{G,c}|E(G)|}]$ by independently choosing each $w(e)$ from that range uniformly at random. By the above lemma, $A_{G,c} = A_{G,c+w}$ so in the rest of the proof we will omit the edge-costs function and denote this matrix by A_G. Now we argue that every mimicking network of $(G, c+w)$ must has at least $r := \text{rank}(A_G)$ edges. Consider some network G' with $|E(G')| < r$, and let's see if it can potentially be a mimicking network of $(G, c+w)$. Notice that every edge-costs function $c' : E(G') \to \mathbb{R}^+$ for this G' yields a cutset-edge incidence matrix $A_{G',c'}$ of size $m \times (r-1)$ (if some graph has less than $r-1$ edges we can pad the irrelevant columns with zeros). Since this matrix has only ones and zeros in its entries, there are only $2^{m(r-1)}$ such matrices. The next lemma proves that for every fixed matrix $A \in \{0,1\}^{m \times (r-1)}$, the probability that there exists an edge-costs function $c' : E(G') \to \mathbb{R}^+$ such that $A \cdot \vec{c'} = A_G \cdot (\vec{c} + \vec{w}) = \Phi_G$ is zero.

Lemma 3.3. *Fix a matrix $A \in \{0,1\}^{m\times(r-1)}$, and let W_{A_G} and W_A be the span of the columns of A_G and A, respectively. If each $w(e)$ is independently sampled uniformly at random from $[0, \frac{1}{\Delta_{G,c}|E(G)|}]$, then*

$$\Pr_w[A_G \cdot (\vec{c} + \vec{w}) \in W_A] = 0.$$

Proof. Without loss of generality let the first r columns of the matrix A_G, $\{\vec{a_1}, \ldots, \vec{a_r}\}$, be the basis for the space W_{A_G}. Since $\text{rank}(A) < r = \text{rank}(A_G)$ we get that $\dim(W_A) < \dim(W_{A_G})$. Thus there must be some basis vector of W_{A_G}, say without loss of generality $\vec{a_1}$, that is not in the subspace W_A and denote by $c(e_1) + w(e_1)$ its corresponding cost.

We will calculate the number of vectors in W_A that can be expressed as linear combination with the vector $\vec{a_1}$. Let $f(\alpha) = \alpha\vec{a_1} + \sum_{i=2}^{r}(c(e_i) + w(e_i))\vec{a_i}$. If there are at least two such vectors, $f(\alpha)$ and $f(\alpha')$ (where $\alpha, \alpha' \neq 0$) in W_A, then $\vec{a_1}$ will be in W_A because W_A is a subspace. So there is at most one α such that $f(\alpha) \in W_A$.

Since each $w(e_i)$ is sampled independently from a uniform distribution over $[0, \frac{1}{\Delta_{G,c}|E(G)|}]$, the probability that $c(e_1) + w(e_1) = \alpha$ is 0. By independence of $w(e_i)$ for all $i \in [r]$ we can sample $w(e_1)$ last which completes Lemma 3.3. ■

To complete the proof of Lemma 3.1, we will calculate the probability that there exists a mimicking network (G', c') for the network $(G, c + w)$, such that $|E(G')| < r$.

$$\begin{aligned}
&\Pr_w[\exists \text{ mimicking network } (G', c') \text{ with } |E(G')| < r] \\
&= \Pr_w[\exists \text{ 0-1 matrix } A_{G',c'} \text{ s.t. } A_{G',c'} \cdot \vec{c'} = A_G(\vec{c} + \vec{w})] \\
&\leq \Pr_w[\exists \text{ 0-1 matrix } A_{G',c'} \text{ s.t. } A_G(\vec{c} + \vec{w}) \in W_{A_{G',c'}}] \\
&\leq \sum_{A\in\{0,1\}^{m\times(r-1)}} \Pr_w[A_G \cdot (\vec{c} + \vec{w}) \in W_A] = 0,
\end{aligned}$$

where the first equality is by the definition of a mimicking network, the following inequality is because the condition is necessary (but not sufficient), the second inequality is by a union bound over all possible matrices, and the final equality is by Lemma 3.3. Denoting $\hat{c} = c + w$, we see that every mimicking network (G', c') for the network $(G, \hat{c})$ has at least $\text{rank}(A_G)$ edges. Lemma 3.1 follows.

3.3 Lower bound for general graphs We now prove Theorem 1.2 which asserts that for every k there exists a k-terminal network such that its mimicking network must have $2^{\Omega(k)}$ non-terminals. The proof constructs a bipartite k-terminal network, with all its terminals on one side and all its non-terminals on the other side. As we will show, the rank of its cutset-edge incidence matrix is at least $2^{\Omega(k)}$, and the corresponding cuts are unique, hence applying Lemma 3.1 to this matrix will complete the proof of Theorem 1.2.

Proof. [of Theorem 1.2] Consider a complete bipartite graph $G = (Q, U, E)$, where one side of the graph consists of the k terminals $Q = \{q_1, \ldots, q_k\}$, the other side of the graph consists of $l = \binom{k}{\frac{2}{3}k}$ non-terminals $U = \{u_{S_1}, \ldots, u_{S_l}\}$, with $S_1, \ldots, S_l$ denoting the different subsets of terminals of size $\frac{2}{3}k$. The costs of the edges of G are as follows: every non-terminal u_{S_i} is connected by edges of cost 1 to every terminal in S_i, and by edges of cost $2 + \varepsilon$ to every terminal in $\bar{S_i} = Q \setminus S_i$, for sufficient small $\varepsilon > 0$, in fact $\varepsilon = \frac{1}{k}$ suffices. Let $c(u_{S_i}, q_j)$ denote the cost of edge (u_{S_i}, q_j), and define $c(u_{S_i}, S_j) := \sum_{q\in S_j} c(u_{S_i}, q)$. The proofs of the following two lemmas are deferred to the full version [KR12].

Lemma 3.4. *The minimum S_i-separating cut is obtained uniquely by the cut $(W, V(G) \setminus W)$ where $W = \{u_{S_i}\} \cup \bar{S_i}$ and $V(G) \setminus W = \{u_{S_j} : j \neq i\} \cup S_i$.*

Lemma 3.5. *Let A_G be a cutset-edge incidence matrix of G.Then $\text{rank}(A_G) \geq l$.*

We can now complete the proof of Theorem 1.2. Applying Lemma 3.1 to our bipartite graph G and its cutset-edge incidence matrix A_G, we get that every mimicking network G' of G has at least $l = 2^{\Omega(k)}$ edges. It follows that $|V(G')| \geq \sqrt{|E(G')|} \geq 2^{\Omega(k)}$. ■

4 Lower Bounds for Data Structures

We can extend the definition of a (deterministic) TC scheme to a randomized one by letting the two operations access a common source of random bits. (We do not assume the random bits are stored explicitly in M, even though it might be required in some implementations.) We then change the requirement from the query operation to be

$$\Pr[Q(S; M) = \text{mincut}_{G,c}(S, \bar{S})] \geq 2/3,$$

where the probability is taken over the data structure's random bits. Our lower bound in Theorem 1.4 holds also for randomized schemes, even those with shared randomness (that is not stored explicitly).

We now prove Theorem 1.4, which asserts that a terminal-cuts scheme requires $2^{\Omega(k)}$ words in the worst-case. Fix k and let (G, c) be the k-terminal bipartite graph constructed in Section 3.3. Recall that $l := \binom{k}{2k/3}$

is the number of subsets of terminals of size $2k/3$, each corresponding to a non-terminal in G. The number of vertices in G is $n := k + l = 2^{\Theta(k)}$, and size of a machine word is $O(\log n) = \Theta(k)$ bits. Assume towards contradiction there is a terminal-cuts scheme that can handle every k-terminal network using less than $l/100$ bits. For now, let us assume the scheme is deterministic.

Let $A_{G,c}$ be the cutset-edge incidence matrix of (G, c). By Lemma 3.5, $\mathrm{rank}(A_{G,c}) \geq l$. Let us assume that the first l columns of $A_{G,c}$ are linearly independent (otherwise, we just reorder them), and let e_j denote the edge of G corresponding to the j-th column of $A_{G,c}$.

Let $\mathcal{W}$ denote the collection of 2^l edge-costs functions $w : E(G) \to \{0, \frac{1}{6k^2 l}\}$ satisfying that $w(e_j) = 0$ for all $j > l$. As in Section 3.3, every function $w \in \mathcal{W}$ defines a graph $(G, c + w)$, whose cutset-edge incidence matrix is denoted $A_{G,c+w}$. We can now apply Lemma 3.2, since $6k > \Delta_{G,c}$ and $|E(G)| = kl$, and obtain that for all $w \in \mathcal{W}$ the network $(G, c+w)$ has the same cutset-edge incidence matrix as (G, c), i.e. $A_{G,c} = A_{G,c+w}$. Using the above bound on the rank of $A_{G,c}$ we can deduce that for every two different functions $w \neq w' \in \mathcal{W}$, we have $A_{G,c} \cdot (\vec{c} + \vec{w}) \neq A_{G,c} \cdot (\vec{c} + \vec{w'})$, i.e. there exists $S \subset Q$ such that $\mathrm{mincut}_{G,c+w}(S, \bar{S}) \neq \mathrm{mincut}_{G,c+w'}(S, \bar{S})$.

Now, the assumed terminal-cuts scheme uses less than $l/100$ bits, and thus, by the pigeonhole principle, there must be $w \neq w' \in \mathcal{W}$, whose preprocessing results with the exact same memory image $M = P(G, c + w) = P(G, c + w')$. Consequently, for all queries $S \subset Q$, the scheme will report the same answer under inputs $(G, c + w)$ and $(G, c + w')$, which means that $\mathrm{mincut}_{G,c+w}(S, \bar{S}) = \mathrm{mincut}_{G,c+w'}(S, \bar{S})$ and is a contradiction.

Notice that the edge costs of the graphs $(G, c + w)$ for $w \in \mathcal{W}$ can be easily scaled so that they are all in the range $\{0, 1, \ldots, n^{O(1)}\}$. We conclude that a terminals-cut scheme for k terminals requires, in the worst case, storage of at least $\frac{l/100}{O(\log n)} \geq 2^{\Omega(k)}$ words. This proves Theorem 1.4 for deterministic schemes.

Proof for randomized schemes (sketch). The proof for randomized schemes follows the same outline, the main difference being that we replace the simple collision argument between $w \neq w'$, with well-known entropy (information) bounds. First, the data structure's success probability can be amplified to at least (say) $1 - 2^{2k}$, by straightforward independent repetitions, while increasing the storage requirement by a factor of $O(k)$. So assume henceforth this very high probability is the case.

Now let us choose $w \in \mathcal{W}$ at random, which corresponds to choosing a random string of l bits. Using the data structure, one can retrieve with very high probability the value $\mathrm{mincut}_{G,c+w}(S, \bar{S}) = A_{G,c} \cdot (\vec{c} + \vec{w})$. Applying a union bound over all 2^k subsets $S \subset Q$, with very high probability one would retrieves correctly all these values. In this case, since the first l columns of $A_{G,c}$ yield an invertible matrix, we could actually recover the vector w itself (with high probability). But since w is effectively a random string of l bits, it follows by standard entropy bounds that M must have at least $2^{\Omega(l)}$ bits, and the theorem is completed just like for a deterministic scheme.

5 Concluding Remarks

Define a *generalized mimicking network* of a k-terminal network (G, c) to be a k-terminal network (G', c') with the same set of terminals Q, such that for all disjoint $S, T \subset Q$, the minimum cost of a cut separating S from T is the same, namely $\mathrm{mincut}_{G',c'}(S, T) = \mathrm{mincut}_{G,c}(S, T)$. Although this definition increases the number of cuts that must be preserved, our upper bound for planar graphs extends to this more general definition (but with larger constants in the exponents), and the same is true for the upper bound for general graphs by [HKNR98].

The upper bound of Hagerup, Katajainen, Nishimura, and Ragde [HKNR98] holds for both directed and undirected graphs. Our paper only discusses undirected graphs; our lower bounds actually hold for directed graphs as well. It is an interesting question whether there is a significant difference between the maximum size of a mimicking network in the directed and undirected versions of the problem, either for general graphs or for some natural family of graphs.

References

[AKPW95] N. Alon, R. M. Karp, D. Peleg, and D. West. A graph-theoretic game and its application to the k-server problem. *SIAM J. Comput.*, 24(1):78–100, February 1995.

[Bar96] Y. Bartal. Probabilistic approximation of metric spaces and its algorithmic applications. In *37th Annual Symposium on Foundations of Computer Science*, pages 184–193. IEEE, 1996.

[BK96] A. A. Benczúr and D. R. Karger. Approximating s-t minimum cuts in $\tilde{O}(n^2)$ time. In *28th Annual ACM Symposium on Theory of Computing*, pages 47–55. ACM, 1996.

[BSS09] J. D. Batson, D. A. Spielman, and N. Srivastava. Twice-ramanujan sparsifiers. In *41st Annual ACM symposium on Theory of computing*, pages 255–262. ACM, 2009.

[CE06] D. Coppersmith and M. Elkin. Sparse sourcewise and pairwise distance preservers. *SIAM J. Discrete Math.*, 20:463–501, 2006.

[Chu12] J. Chuzhoy. On vertex sparsifiers with Steiner nodes. In *44th symposium on Theory of Computing*, pages 673–688. ACM, 2012.

[CLLM10] M. Charikar, T. Leighton, S. Li, and A. Moitra. Vertex sparsifiers and abstract rounding algorithms. In *51st Annual Symposium on Foundations of Computer Science*, pages 265–274. IEEE Computer Society, 2010.

[CSWZ00] S. Chaudhuri, K. V. Subrahmanyam, F. Wagner, and C. D. Zaroliagis. Computing mimicking networks. *Algorithmica*, 26:31–49, 2000.

[EGK$^+$10] M. Englert, A. Gupta, R. Krauthgamer, H. Räcke, I. Talgam-Cohen, and K. Talwar. Vertex sparsifiers: New results from old techniques. In *13th International Workshop on Approximation, Randomization, and Combinatorial Optimization*, volume 6302 of *Lecture Notes in Computer Science*, pages 152–165. Springer, 2010.

[FM95] T. Feder and R. Motwani. Clique partitions, graph compression and speeding-up algorithms. *J. Comput. Syst. Sci.*, 51(2):261–272, 1995.

[HKNR98] T. Hagerup, J. Katajainen, N. Nishimura, and P. Ragde. Characterizing multiterminal flow networks and computing flows in networks of small treewidth. *J. Comput. Syst. Sci.*, 57:366–375, 1998.

[HS85] D. S. Hochbaum and D. B. Shmoys. An $O(|V|^2)$ algorithm for the planar 3-cut problem. *SIAM J. Algebraic Discrete Methods*, 6(4):707–712, 1985.

[KR12] R. Krauthgamer and I. Rika. Mimicking networks and succinct representations of terminal cuts. *CoRR*, abs/1207.6246, 2012.

[KRTV12] A. Khan, P. Raghavendra, P. Tetali, and L. A. Végh. On mimicking networks representing minimum terminal cuts. *CoRR*, abs/1207.6371, 2012.

[KZ12] R. Krauthgamer and T. Zondiner. Preserving terminal distances using minors. In *39th International Colloquium on Automata, Languages, and Programming*, volume 7391 of *Lecture Notes in Computer Science*, pages 594–605. Springer, 2012.

[MM10] K. Makarychev and Y. Makarychev. Metric extension operators, vertex sparsifiers and lipschitz extendability. In *51st Annual Symposium on Foundations of Computer Science*, pages 255–264. IEEE, 2010.

[PS89] D. Peleg and A. A. Schäffer. Graph spanners. *J. Graph Theory*, 13(1):99–116, 1989.

[Rao87] S. Rao. Finding near optimal separators in planar graphs. In *28th Annual Symposium on Foundations of Computer Science*, pages 225–237. IEEE, 1987.