Vertex Sparsification of Cuts, Flows, and Distances

Robert Krauthgamer, Weizmann Institute of Science

WorKer 2015, Nordfjordeid
Graph Sparsification

- Vast literature on "compression" (succinct representation) of graphs
  - We focus on preserving specific features – distances, cuts, etc.

- Edge sparsification:
  - Cut and spectral sparsifiers [Benczur-Karger, …, Batson-Spielman-Srivastava]
  - Spanners and distance oracles [Peleg-Schaffer, …, Thorup-Zwick,…]

- Vertex sparsification (keep only the “terminal” vertices)
  - Cut/multicommodity-flow sparsifier [Moitra,….Chuzhoy]
  - Distances [Gupta, Coppersmith-Elkin]

Graphical representation

Fast query time

or mostly

exactly/approximately
Terminal Cuts

- Network $G$ with edge capacities $c: E(G) \rightarrow \mathbb{R}_+$. ("huge" network)
- $k$ terminals $K^{1/2}V(G)$ ("important" vertices)

We care about terminal cuts:
- $\text{mincut}_G(S) =$ minimum-capacity cut separating $S^{1/2}K$ and $\overline{S} = K \setminus S$.
- (Equivalent to the maximum flow between $S$ and $\overline{S}$.)
Mimicking Networks

A mimicking network of \((G, c)\) is a network \((G', c')\) with same terminals and \(8S \mu K, \text{mincut}_G(S) = \text{mincut}_{G'}(S)\).

**Theorem** [Hagerup-Katajainen-Nishimura-Ragde’95]. Every \(k\)-terminal network has a mimicking network of \(\cdot 2^{2k}\) vertices.

- **Pro:** independent of \(n=|V(G)|\)
- **Con:** more wasteful than listing the \(2^k\) cut values
- (Originally proved for directed networks)

**Intuition:** There are \(\cdot 2^k\) relevant cuts (choices for \(S\)), which jointly partition the vertices to \(\cdot 2^{2k}\) “buckets”; merge each bucket …
Natural Questions

- Narrow the **doubly-exponential gap**?
- Better bounds for specific **graph families**?
- Represent these cut values **more succinctly**?
  - Anything better than a list of $2^k$ values?
  - Remark: function $\text{mincut}_G(.)$ is submodular
## Our Results [K.-Rika’13]

<table>
<thead>
<tr>
<th>Graph Family</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>General graphs</td>
<td>≥ $2^k$</td>
<td>$2^{2k}$ [HKNR]</td>
</tr>
<tr>
<td>Star graph</td>
<td>$k+1$ [CSWZ]</td>
<td></td>
</tr>
</tbody>
</table>

### Theorem [No succinct representation]: Any storage of the terminal-cut values requires $2^{\Omega(k)}$ machine words. (word = $\log n$ bits)

[Chaudhuri-Subrahmanyam-Wagner-Zaroliagis’98]  
[Hagerup-Katajainen-Nishimura-Ragde’95]  
[Khan-Raghavendra’14]
Upper Bound for Planar Graphs

Theorem 1. Every planar $k$-terminal network $G$ admits a mimicking network of size $\cdot O(k^2 2^{2k})$; furthermore, this $G'$ is a minor of $G$.

- Algorithm: merge vertices whenever possible, similarly to [HKNR]
  - Precisely, remove cuts then contract every CC (yields a planar graph)
  - Let $E_S \mu E$ be the cutset realizing $\text{mincut}_G(S)$ [wlog it’s unique]

- Lemma 1a. Removing $E_S$ breaks $G$ into $\cdot k$ CCs
  - That is, for all $S \subseteq K$, $|CC(G \setminus E_S)| \cdot k$.

V:

![Diagram of a planar graph showing cutsets and terminals]

$S = \{t_2, t_3, t_5\}$

Idea: a CC containing no terminals can be merged with another CC.
Two Cutsets Together

- **Lemma 1b.** For all $S, T^{1/2} K$,
  
  $$|CC(G^n(E_S[E_T]))| \cdot |CC(G^nE_S)| + |CC(G^nE_T)| + k \cdot 3k.$$  
  (even without planarity)

**Idea:** Every CC must either contain a terminal or be (exactly) a CC of $G^nE_S$ or of $G^nE_T$. Otherwise, we can “improve” one of the cuts.

\[ T = \{t_2, t_4\} \]

\[ S = \{t_3\} \]
Leveraging Planarity

- Let $E_S^*$ be the dual edges to $E_S$.
  - It is a union of cycles (called circuit), at most $k$ of them by Lemma 1a.

- Lemma 1c. The union $\bigcup S(E_S^*)$ partitions the plane into $\cdot O(k^2 2^{2k})$ “connected regions”.

- Idea: By Euler’s formula, it suffices to sum up all vertex degrees $>2$. These are “attributed” to some intersection $E_S^* \& E_T^*$. Every pair $S,T$ “contributes” $\cdot O(k^2)$ by Lemma 1b and Euler’s formula.
**Theorem 2.** For every $k > 5$ there is a $k$-terminal network, whose mimicking networks must have size $\Omega(\log k)$.

- Proved independently by [Khan-Raghavendra’14]

**Proof Sketch:** consider a bipartite graph

- Lemma 2a. Each $\text{mincut}_G(S)$ is obtained uniquely ($u_S$ vs. the rest)
- Thus, each green edge belongs to only one cut.

**Intuition for next step:** Graphs $G'$ with few edges have insufficient “degrees of freedom” to create these $2^{\Omega(k)}$ cuts. Use linear algebra...
Lower Bounds - Techniques

- **Lemma 2b.** The cutset-edge incidence matrix $A_G$ has $\text{rank}(A_G) \geq 2^{\Omega(k)}$.

$$
\begin{align*}
S \subset K & \rightarrow \\
& \left( \begin{array}{c}
e \in E \\
1_{\{e \in \text{mincut}(S)\}}
\end{array} \right) \cdot \left( \begin{array}{c}
c(e) \\
\vdots
\end{array} \right) = \left( \begin{array}{c}
\text{mincut}(S) \\
\vdots
\end{array} \right)
\end{align*}
$$

- **Lemma 2c.** WHP, after perturbing capacities to $c^\wedge$ (add noise $2[0,2]$), every mimicking network $(G',c')$ satisfies $|E(G')| \geq \text{rank}(A_G)$.

- **Difficulty:** Infinitely-many possible $c'$, cannot take union bound...

- **Workaround:**
  - Fix $G'$ (without capacities $c'$) and a matrix $A_{G'}$.
  - $\Pr[9c' \text{ such that } (G',c') \text{ mimicks } (G,c^\wedge)] = 0$.
  - Union bound over finitely many $G'$ and $A_{G'}$. 
Succinct Representation

- **Theorem 3.** Every (randomized) data structure that stores the terminal-cut values of a network requires $2^{\Omega(k)}$ memory words.

  - Thus, naively listing all $2^k$ cut values achieves optimal storage.

**Proof Sketch:**

- Use the same bipartite graph.
- “Plant” $r$ arbitrary bits by perturbing $r$ edge capacities.
- Since $\text{rank}(A_G) \geq r$, the bits can be recovered from the mincut values.
- Hence, data structure must have $\Omega(r)$ bits.
Further Questions About Cuts

- Close the (still) exponential gap?
  - Perhaps show the directed case is significantly different?

- Extend the planar upper bound
  - To excluded-minor graphs?
  - To vertex-cuts or directed networks?

- Extend to multi-commodity flows?
  - A stronger requirement than cuts

- Smaller network size by allowing approximation of cuts?
  - We already know size is some function $s(k)$, independent of $n$
  - Our lower bound is not “robust”
Approximate Vertex-Sparsifiers

Definition: Quality = approximation-factor guarantee for all cuts

- **Extreme case:** retain only terminals, i.e. $s(k) = k$ [Moitra’09]
  - Quality $O(\log k / \log \log k)$ is possible [Charikar-Leighton-Li-Moitra’10, Makarychev-Makarychev’10, Englert-Gupta-K.-Raecke-TalgamCohen-Talwar’10]
  - And $\Omega((\log k)^{1/2} / \log \log k)$ is required [Makarychev-Makarychev’10]

- **Goal:** constant-factor quality using network size $<< 2^{2^k}$?
  - Maybe even $(1 + ^2)$-quality using size $s'(k, 2)$

- **Theorem [Chuzhoy’12].** $O(1)$-quality using network size $C^{O(\log \log C)}$
  - where $C$ is total capacity of edges incident to terminals
  - Note: $C$ might grow with $n = |V|
Our Results [Andoni-Gupta-K.’14]

- **Theorem 4.** Bipartite* networks admit \((1+\sqrt{2})\)-quality sparsifiers of size \(\text{poly}(k/2)\)
  - Bipartite* = the non-terminals form an independent set
  - Bypasses \(2^{\Omega(k)}\) bound we saw for exact sparsifiers (even in bipartite)

- **Theorem 5.** Networks of treewidth \(w\) admit \(O(\log w / \log \log w)\)-quality (flow) sparsifiers of size \(O(w \cdot \text{poly}(k))\)

- **Theorem 6.** Series-parallel networks admit exact (quality 1) (flow) sparsifiers of size \(O(k)\)
Main Idea: Structure Sampling

- Edge sampling useful for *edge*-sparsifiers [BK’96, SS’11]
- But does not work here, need to sample entire *sub-structures*
Sampling in Bipartite Graphs

- Sample non-terminal vertices, together with incident edges
  - reweight edges accordingly

Vertex Sparsification of Cuts, Flows, and Distances

17
Sampling in Bipartite Graphs

- Sample **non-terminal vertices**, together with incident edges
  - reweight edges accordingly
- Uniform sampling does not work
Non-uniform Sampling

- Non-terminal $v$ has sampling probability $p_v$
- If sampled, reweight edges by factor $1/p_v$
- Expectation is right:
  - Consider a partition $K = S \cup T$
  - $\text{mincut}(S, T) = \sum_v \min\{c(v, S), c(v, T)\}$
  - $\text{mincut}'(S, T) = \sum_v \frac{I_v}{p_v} \cdot \min\{c(v, S), c(v, T)\}$

Vertex Sparsification of Cuts, Flows, and Distances 19
How to choose $p_v$?

- **Want**
  1) concentrates
  \[ \text{mincut}(S,T) = \sum_v \frac{l_v}{p_v} \cdot \min\{c(v,S), c(v,T)\} \]
  concentrates
  2) small i.e.
  \[ \sum_v p_v \text{ small i.e. } \text{poly}\left(\frac{k}{\epsilon}\right) \]
- **Issue**: contribution can come from just a few terms
- **Issue**: contribution can come from just a few terms
**Importance sampling**

- \( \text{mincut}'(S, T) = \sum_v \frac{l_v}{p_v} \cdot \min\{c(v, S), c(v, T)\} \)

**Idea 1:** Choose proportional to contribution

- **Issue:** contribution depends on partition, but \( p_v \) cannot

**Idea 2:** for any \( K = S \cup T \), large contribution comes from one pair of terminals

- **Issue:** contribution depends on partition \( S \cup T \), but \( p_v \) cannot

**Idea 2:** for any \( K = S \cup T \), large contribution comes from one pair of terminals \( s \in S, t \in T \)

- (up to factor )
- enough to “take care” of all pairs
- enough to “take care” of all pairs \( s, t \)
Actual Sampling

\[
p_v = P \max_{s,t} \frac{\min\{c_{v,s},c_{v,t}\}}{\sum_u \min\{c_{u,s},c_{u,t}\}}
\]

(proof idea)
1) over-estimates the contribution \( \rightarrow \) concentration
2) Apply union bound over all choices of cuts
3) \( \sum_v p_v \leq F k^2 \)

oversampling (if there were only two terminals, how important would be?)

factor \((k/\varepsilon)\) how important would be?
Open Questions

- Extend to **general networks**?
  - Want to beat size $2^k$ (exact sparsification)
  - Need to sample other structures (flow paths??)

- What about **flow-sparsifiers**?
  - In bipartite networks: ✓ (our technique extends)
  - In general networks: no bound $s'(k,^2)$ is known
  - **A positive indication**: can build there is a data structure of size $(1/^2)^k^2$
    (“big table” with all values)
Generalizing Gomory-Hu Trees?

Theorem [Gomory-Hu'61]. In every network $G$, all the minimum $st$-cuts can be represented by a tree (on the same vertex set).
- Surprising redundancy! size $O(n)$ vs. the original graph’s $O(n^2)$

Desirable extensions:
- $3$-way: represent all minimum $\{s,t,u\}$-cuts
- $p$-sets: represent all minimum $\{s_1,\ldots,s_p\} - \{t_1,\ldots,t_p\}$ cuts
- Any redundancy at all?

Theorem 7 [Chitnis-Kamma-K]: The number of distinct
- $3$-way cuts is $\Theta(n^2)$
- $p$-set cuts is $\Theta(n^{2p-1})$
- These bounds are tight
- But non-constructive and provide no compression
Gomory-Hu Tree for Terminals

Corollary of [Gomory-Hu'61]. Can represent all terminal cuts (i.e., only ), using size $O(k)$

Question: Does it extend to all 3-way cuts? All $p$-set cuts?

Want a bound that depends only on $k$ (not on $n$)

Observe: $p$-set cuts is a special case of mimicking networks

Our bounds on number of distinct cuts extend

But they are non-constructive:

Currently looking at (information-theoretic) lower bounds
Terminal Distances

- Graph \( G \) with edge lengths \( l: E(G) \rightarrow \mathbb{R}_+ \). ("huge" network)
- \( k \) terminals \( K^{1/2} V(G) \) ("important" vertices)

We care about terminal distances:
- \( d_G(s,t) = \) shortest-path distance according to \( l \) between \( s,t \in 2K \).
A distance-preserving minor of \((G, l)\) is a minor \(G'\) with edge-lengths \(l'\) that contains the same \(k\) terminals and 
\[d_G(s, t) = d_{G'}(s, t).\]

Why require a minor? To avoid a trivial solution…

Vertex Sparsification of Cuts, Flows, and Distances
Our Results [K.-Nguyen-Zondiner’14]

We ask: What is the smallest $f^*(k)$ such that every $k$-terminal graph $G$ admits a distance-preserving minor $G'$ with $|V(G')| \leq f^*(k)$?

<table>
<thead>
<tr>
<th>Graph Family $\mathcal{F}$</th>
<th>Bounds on $f^*(k, \mathcal{F})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trees</td>
<td>$2k - 2$</td>
</tr>
<tr>
<td>General Graphs</td>
<td>$\Omega(k^2)$, $O(k^4)$</td>
</tr>
<tr>
<td>Planar Graphs</td>
<td>$\Omega(k^2)$, $O(k^4)$</td>
</tr>
<tr>
<td>Treewidth $p$</td>
<td>$\Omega(pk)$, $O(p^3k)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Graph Family $\mathcal{F}$</th>
<th>Bounds on $f^*(k, \mathcal{F})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trees</td>
<td>$2k - 2$</td>
</tr>
<tr>
<td>General Graphs</td>
<td>$\Omega(k^2)$, $O(k^4)$</td>
</tr>
<tr>
<td>Planar Graphs</td>
<td>$\Omega(k^2)$, $O(k^4)$</td>
</tr>
<tr>
<td>Treewidth $p$</td>
<td>$\Omega(pk)$, $O(p^3k)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Graph Family $\mathcal{F}$</th>
<th>Bounds on $f^*(k, \mathcal{F})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trees</td>
<td>$2k - 2$</td>
</tr>
<tr>
<td>General Graphs</td>
<td>$\Omega(k^2)$, $O(k^4)$</td>
</tr>
<tr>
<td>Planar Graphs</td>
<td>$\Omega(k^2)$, $O(k^4)$</td>
</tr>
<tr>
<td>Treewidth $p$</td>
<td>$\Omega(pk)$, $O(p^3k)$</td>
</tr>
</tbody>
</table>
Naive algorithm:
1. Consider the graph induced by shortest paths between the terminals
2. Eliminate all non-terminals with degree 2

Analysis:
- Shortest paths between terminals → pairs of paths
- Each pair incurs at most two vertices of degree ≥ 2 ("intersections")
- Thus, number of non-terminals is at most $O(k^4)$
Outline of our original proof:
- $G$ is a 2-D grid (with specific edge-lengths and terminals)
- Main lemma: Any $G'$ must have a planar separator of size $\Omega(k)$
- Using the planar separator theorem, $|V(G')| \geq \Omega(k^2)$

More elementary proof:
- $G$ is just a $(k/4) \times (k/4)$ grid
- Terminals: the boundary vertices
- In $G'$, “horizontal” shortest-paths (from left to right terminals) do not intersect
- Same for “vertical” shortest-paths
- Every horizontal path must intersect every vertical path
- These $\Theta(k^2)$ intersection points must be distinct

Proof extends to $(1+\varepsilon)$-approximation, proving $|V(G')| \geq \Omega(1/\varepsilon^2)$.
Our Results [Kamma-K.-Nguyen’14]

- **Theorem 8.** Every $k$-terminal graph $G$ with edge-lengths $l$ admits a polylog($k$) distance-approximating minor of size $k$.
  - I.e., a minor $G'$ containing only the terminals with new edge-lengths $l'$ whose terminal-distances approximate $G$ within factor polylog($k$).

- Previously:
  - Approximation factor $k$ is easy
  - Probabilistic approximation factor $O(\log k)$ (i.e., by a convex combination of minors) [Englert-Gupta-K.-Raecke-TalgamCohen-Talwar’10]
Further Questions

About distances:

- Close the gap (for exact version) between $\Omega(k^2)$ and $O(k^4)$
  - For general and for planar graphs
- What about $1+\epsilon$ approximation?
- Other extreme: Best approximation using a minor of size $k$?
  - Prove a lower bound of $\Omega(\log\log k)$?

High-level plan:

- Maintain other combinatorial properties
- Discover redundancies (exploit them by data structures?)
- Matching lower bounds (information-theoretic?)

Thank You!