



Deposited via The University of Sheffield.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/id/eprint/147336/>

Version: Published Version

Proceedings Paper:

Goranci, G., Henzinger, M. and Peng, P. (2017) Improved guarantees for Vertex Sparsification in planar graphs. In: 25th Annual European Symposium on Algorithms (ESA 2017). 25th Annual European Symposium on Algorithms (ESA 2017), 04-08 Sep 2017, Vienna, Austria. Leibniz International Proceedings in Informatics, 87. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 44:1-44:14. ISBN: 978-3-95977-049-1. ISSN: 1868-8969.

<https://doi.org/10.4230/LIPIcs.ESA.2017.44>

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:

<https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

Improved Guarantees for Vertex Sparsification in Planar Graphs^{*†}

Gramoz Goranci^{‡1}, Monika Henzinger², and Pan Peng³

1 University of Vienna, Faculty of Computer Science, Vienna, Austria
gramoz.goranci@univie.ac.at

2 University of Vienna, Faculty of Computer Science, Vienna, Austria
monika.henzinger@univie.ac.at

3 University of Vienna, Faculty of Computer Science, Vienna, Austria
pan.peng@univie.ac.at

Abstract

Graph Sparsification aims at compressing large graphs into smaller ones while (approximately) preserving important characteristics of the input graph. In this work we study Vertex Sparsifiers, i.e., sparsifiers whose goal is to reduce the number of vertices. Given a weighted graph $G = (V, E)$, and a terminal set K with $|K| = k$, a quality- q vertex cut sparsifier of G is a graph H with $K \subset V_H$ that preserves the value of minimum cuts separating any bipartition of K , up to a factor of q . We show that planar graphs with all the k terminals lying on the same face admit quality-1 vertex cut sparsifier of size $O(k^2)$ that are also planar. Our result extends to vertex flow and distance sparsifiers. It improves the previous best known bound of $O(k^2 2^{2k})$ for cut and flow sparsifiers by an exponential factor, and matches an $\Omega(k^2)$ lower-bound for this class of graphs.

We also study vertex reachability sparsifiers for directed graphs. Given a digraph $G = (V, E)$ and a terminal set K , a vertex reachability sparsifier of G is a digraph $H = (V_H, E_H)$, $K \subset V_H$ that preserves all reachability information among terminal pairs. We introduce the notion of reachability-preserving minors, i.e., we require H to be a minor of G . Among others, for general planar digraphs, we construct reachability-preserving minors of size $O(k^2 \log^2 k)$. We complement our upper-bound by showing that there exists an infinite family of acyclic planar digraphs such that any reachability-preserving minor must have $\Omega(k^2)$ vertices.

1998 ACM Subject Classification G.2.2 Graph Theory, F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Vertex Sparsification, Graph Sparsification, Planar Graphs, Metric Embedding, Reachability

Digital Object Identifier 10.4230/LIPIcs.ESA.2017.44

1 Introduction

Very large graphs or networks are ubiquitous nowadays, from social networks to information networks. One natural and effective way of processing and analyzing such graphs is to compress or sparsify the graph into a smaller one that well preserves certain properties of the original graph. Such a sparsification can be obtained by reducing the number of

* A full version of the paper is available at <https://arxiv.org/abs/1702.01136>.

† The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement no. 340506.

‡ Partially supported by the Doctoral Programme "Vienna Graduate School on Computational Optimization" which is funded by Austrian Science Fund (FWF, project no. W1260-N35)



edges. Typical examples include cut sparsifiers [3], spectral sparsifiers [36], spanners [40] and transitive reductions [1], which are subgraphs defined on the same vertex set of the original graph G while having much smaller number of edges and still well preserving the cut structure, spectral properties, pairwise distances, transitive closure of G , respectively. Another way of performing sparsification is by reducing the number of *vertices*, which is most appealing when only the properties among a subset of vertices (which are called *terminals*) are of interest (see e.g., [34, 2, 26]). We call such small graphs *vertex sparsifiers* of the original graph. In this version of the paper, we will focus on vertex cut and reachability sparsifiers.

More specifically, given a capacitated undirected graph $G = (V, E, c)$, and a set of terminals K , we are looking for a graph $H = (V_H, E_H, c_H)$ with as few vertices as possible and $K \subseteq V_H$ such that the properties (e.g., reachability) or quantities (e.g., cut value, multi-commodity flow, distance) among vertices in K in H are the same as or close to the corresponding properties or quantities in G . If $|K| = k$, we call the graph G a *k-terminal graph*. We say H is a *quality- q (vertex) cut sparsifier* of G , if for every bipartition $(U, K \setminus U)$ of the terminal set K , the value of the minimum cut separating U from $K \setminus U$ in G is within a factor of q of the value of minimum cut separating U from $K \setminus U$ in H . If H is a quality-1 cut sparsifier, then it will be also called a *mimicking network* [22]. Similarly, we define vertex flow and distance sparsifiers that (approximately) preserve multicommodity flows and distances among terminal pairs, respectively (formal definitions are deferred to the full version). These type of sparsifiers have proven useful in approximation algorithms [34] and also find applications in network routing [12].

Vertex reachability sparsifiers in *directed* graphs is another important and fundamental notion in Graph Sparsification, which has been implicitly studied in the dynamic graph algorithms community [37, 15], and explicitly in [24]. Specifically, given a digraph $G = (V, E)$, $K \subset V$, a digraph $H = (V_H, E_H)$, $K \subset V_H$ is a *vertex reachability sparsifier* of G if for any $x, x' \in K$, there is a directed path from x to x' in H iff there is a directed path from x to x' in G . Note that any k -terminal digraph G always admits a trivial vertex reachability sparsifier H , which corresponds to the transitive closure restricted to the terminals. In this work, we initiate the study of *reachability-preserving minors*, i.e., vertex reachability sparsifiers with H required to be a minor of G . The restriction on H being a minor of G is desirable as it makes sure that H is structurally similar to G , e.g., any minor of a planar graph remains planar. We ask the question whether general graphs admit reachability-preserving minors whose size can be bounded independently of the input graph G , and study it from both the lower- and upper-bound perspective.

Our Results. We provide new constructions for quality-1 (exact) cut, flow and distance sparsifiers for k -terminal planar graphs, where all the terminals are assumed to lie on the same face. We call such k -terminal planar graphs *Okamura-Seymour (OS) instances*. They are of particular interest in the algorithm design and optimization community, due to the classical Okamura-Seymour theorem that characterizes the existence of feasible concurrent flows in such graphs (see e.g., [35, 8, 9, 30]).

We show that the size of quality-1 sparsifiers can be as small as $O(k^2)$ for such instances, for which only exponential (in k) size of cut and flow sparsifiers were known before [27, 2]. Formally, we have the following theorem.

► **Theorem 1.** *For any OS instance G , i.e., a k -terminal planar graph in which all terminals lie on the same face, there exist quality-1 vertex cut, flow and distance sparsifiers of size $O(k^2)$. Furthermore, the resulting sparsifiers are also planar.*

We remark that all the above sparsifiers can be constructed in polynomial time (in n and k), but we will not optimize the running time here. As we mentioned above, previously the only known upper bound on the size of quality-1 cut and flow sparsifiers for OS instance was $O(k^2 2^{2k})$, given by [27, 2]. Our upper bound for cut sparsifier also matches the lower bound of $\Omega(k^2)$ for OS instance given by [27]. More specifically, in [27], an OS instance (that is a grid in which all terminals lie on the boundary) is constructed, and used to show that any mimicking network for this instance needs $\Omega(k^2)$ edges, which is thus a lower bound for planar graphs (see the table below for an overview). Note that that even though our distance sparsifier is not necessarily a minor of the original graph G , it still shares the nice property of being planar as G . It is worth mentioning that in [29], it is proven that there exists a k -terminal planar graph G (not necessarily an OS instance), such that any quality-1 distance sparsifier of G that is planar requires at least $\Omega(k^2)$ vertices.

| Type of sparsifier | Graph family | Upper Bound | Lower Bound |
|--------------------|--------------|----------------------|---------------------------------|
| Cut | Planar | $O(k^2 2^{2k})$ [27] | |
| Cut | Planar OS | $O(k^2)$ [new] | $ E(G') \geq \Omega(k^2)$ [27] |
| Flow | Planar OS | $O(k^2 2^{2k})$ [2] | follows from cut |
| Flow | Planar OS | $O(k^2)$ [new] | follows from cut |
| Distance (minor) | Planar OS | $O(k^4)$ [26] | $\Omega(k^2)$ [26] |
| Distance (planar) | Planar OS | $O(k^2)$ [new] | |

Our second main contribution is the study of reachability-preserving minors. Although reachability is a weaker requirement in comparison to shortest path distances, directed graphs are usually much more cumbersome to deal with from the perspective of graph sparsification. Surprisingly, we show that general digraphs admit reachability-preserving minors with $O(k^4)$ vertices (see Corollary 20), thus matching the bound of Krauthgamer et al. [26] for distances in undirected graphs. A tight integration of our techniques with the compact distance oracles for planar graphs by Thorup [39] yields the following theorem.

► **Theorem 2.** *Given a k -terminal planar digraph, there exists a reachability-preserving minor H of G with size $O(k^2 \log^2 k)$.*

We complement the above result by showing that there exist instances where the above upper-bound is tight up to a $O(\log^2 k)$ factor. The proof is deferred to the full version.

► **Theorem 3.** *For infinitely many $k \in \mathbb{N}$ there exists a k -terminal acyclic directed grid G such that any reachability-preserving minor of G must use $\Omega(k^2)$ non-terminals.*

Our third contribution is a lower bound on the size of any *data structure* (not necessarily a graph) that approximately preserves pairwise terminal distances of *general* k -terminal graphs, which provides a trade-off between the distance stretch and the space complexity. The proof is deferred to the full version.

► **Theorem 4.** *For any $\varepsilon > 0$ and $t \geq 2$, there exists a (sparse) k -terminal n -vertex graph such that $k = o(n)$, and any data structure that approximates pairwise terminal distances within a multiplicative factor of $t - \varepsilon$ or an additive error $2t - 3$ must use $\Omega(k^{1+1/(t-1)})$ bits.*

► **Remark.** Recently and independently of our work, Krauthgamer and Rika [28] constructed quality-1 cut sparsifiers of size $O(\gamma^5 2^{2\gamma} k^4)$ for planar graphs whose terminals are incident to at most $\gamma = \gamma(G)$ faces. In comparison with our upper-bound which only considers the case $\gamma = 1$, the size of our sparsifiers from Theorem 1 is better by a $\Omega(k^2)$ factor.

Our Techniques. We construct our quality-1 cut and distance sparsifiers by repeatedly performing *Wye-Delta transformations*, which are local operations that preserve cut values and distances and have proven very powerful in analyzing electrical networks and in the theory of circular planar graphs (see e.g., [14, 18]). Khan and Raghavendra [25] used Wye-Delta transformations to construct quality-1 cut sparsifiers of size $O(k)$ for trees, while our case (i.e., the planar OS instances) is more general and complicated and previously it was not clear at all how to apply such transformations to a more broad class of graphs.

Our approach is as follows. Given a k -terminal planar graph with terminals lying on the same face, we first embed it into some large grid with terminals lying on the boundary of the grid. Next, we show how to embed this grid into a “more suitable” graph, which we will refer to as “half-grid”. Finally, using the Wye-Delta operations, we reduce the “half-grid” into another graph whose number of vertices can be bounded by $O(k^2)$. Since we argue that the above graph reductions preserve exactly all terminal minimum cuts, our result follows. Gitler [19] proposed a similar approach for studying the reducibility of multi-terminal graphs with the goal to classify all Wye-Delta reducible graphs, which is very different from our motivation of constructing small vertex sparsifiers with good quality.

The distance sparsifiers can be constructed similarly by slightly modifying the Wye-Delta operation. Our flow sparsifiers follow from the construction of cut sparsifiers and the flow/cut gaps for OS instances (which has also been observed by Andoni et al. [2]).

The results for reachability-preserving minors are obtained by exploiting the technique of Coppersmith and Elkin [13] on counting “branching” events between shortest paths in the directed setting (this technique has also been recently leveraged by Bodwin [4]). We then combine our construction with the compact reachability oracle for planar graphs by Thorup [39], to show our upper-bound for planar graphs. The lower-bound follows by adapting the ideas of Krauthgamer et al. [26] from their lower-bound proof on distance-preserving minors for undirected graphs.

Our lower bound of the space complexity of any compression function approximately preserving terminal pairwise distance is derived by combining extremal combinatorics construction of Steiner Triple System that was used to prove lower bounds on the size of distance approximating minors (see [10]) and the incompressibility technique from [33].

Related Work. There has been a long line of work on investigating the tradeoff between the quality of the vertex sparsifier and its size (see e.g., [17, 27, 2]). (Throughout, cut, flow and distance sparsifiers will refer to their vertex versions.) Quality-1 *cut sparsifiers* (or equivalently, mimicking networks) were first introduced by Hagerup et al. [22], who proved that for any graph G , there always exists a mimicking network of size $O(2^{2^k})$. Krauthgamer and Rika [27] showed how to build a mimicking network of size $O(k^2 2^{2^k})$ for any planar graph G that is minor of the input graph. They also proved a lower bound of $\Omega(k^2)$ on the number of edges of the mimicking network of planar graphs, and a lower bound of $2^{\Omega(k)}$ on the number of vertices of the mimicking network for general graphs.

Quality-1 vertex flow sparsifiers have been studied in [2, 20], albeit only for restricted families of graphs like quasi-bipartite, series-parallel, etc. It is not known if any general undirected graph G admits a constant quality flow sparsifier with size independent of $|V(G)|$ and the edge capacities. For the quality 1 distance sparsifiers, Krauthgamer, Nguyen and Zondiner [26] introduced the notion of *distance-preserving minors*, and showed an upper-bound of size $O(k^4)$ for general undirected graphs. They also gave a lower bound of $\Omega(k^2)$ on the size of such a minor for planar graphs. Over the last two decades, there has been a considerable amount of work on understanding the tradeoff between the sparsifier’s quality q and its size for $q > 1$, i.e., when the sparsifiers only *approximately* preserve the corresponding properties [11, 2, 34, 31, 6, 17, 32, 21, 7, 5, 17, 23, 10, 16].

2 Preliminaries

Let $G = (V, E, c)$ be an undirected graph with terminal set $K \subset V$ of cardinality k , where $c : E \rightarrow \mathbb{R}_{\geq 0}$ assigns a non-negative capacity to each edge. We will refer to such a graph as a k -terminal graph. Throughout the paper we will be dealing with two special types of graphs.

A *grid* graph is a graph with $n \times n$ vertices $\{(u, v) : u, v = 1, \dots, n\}$, where (u, v) and (u', v') are adjacent if $|u' - u| + |v' - v| = 1$. For $k < n$, a *half-grid* graph with k terminals is a graph $T_k^n = (V, E)$ with $K \subset V$ and $n(n+1)/2$ vertices $\{(i, j) : i \leq j \text{ and } i, j = 1, \dots, n\}$, where (i, j) and (i', j') are connected by an edge if $|i' - i| + |j' - j| = 1$, and additional diagonal edges between (i, i) and $(i+1, i+1)$ for $i = 1, \dots, n-1$. Moreover, each terminal vertex in T_k^n must be one of its diagonal vertices, i.e., every $x \in K$ is of the form (m, m) for some $m \in \{1, \dots, n\}$. Let \hat{T}_k^n be the same graph as T_k^n but excluding the diagonal edges.

Let $U \subset V$ and $S \subset K$. We say that a cut $(U, V \setminus U)$ is S -separating if it separates the terminal subset S from its complement $K \setminus S$, i.e., $U \cap K$ is either S or $K \setminus S$. We will refer to such cut as a *terminal cut*. The cutset $\delta(U)$ of a cut $(U, V \setminus U)$ represents the edges that have one endpoint in U and the other one in $V \setminus U$. The cost $\text{cap}_G(\delta(U))$ of a cut $(U, V \setminus U)$ is the sum over all capacities of the edges belonging to the cutset. We let $\text{mincut}_G(S, K \setminus S)$ denote the minimum cost of any S -separating cut of G . A graph $H = (V_H, E_H, c_H)$, $K \subset V_H$ is a *vertex cut sparsifier* of G with *quality* $q \geq 1$ if for any $S \subset K$, $\text{mincut}_G(S, K \setminus S) \leq \text{mincut}_H(S, K \setminus S) \leq q \cdot \text{mincut}_G(S, K \setminus S)$.

Let $G = (V, E)$ be a directed graph with terminal set $K \subset V$, $|K| = k$, which we will refer to as a k -terminal digraph. We say G is a k -terminal DAG if G has no directed cycles. The *in-degree* of a vertex v , denoted by $\deg_G^-(v)$, is the number of edges directed towards v in G . A digraph $H = (V_H, E_H)$, $K \subset V_H$ is a *vertex reachability sparsifier* of G if for any $x, x' \in K$, there is a directed path from x to x' in H iff there is a directed path from x to x' in G . If H is obtained by performing minor operations in G , then we say that H is a *reachability-preserving minor* of G . We define the *size* of H to be the number of non-terminals in H , i.e. $|V_H \setminus K|$.

Wye-Delta Transformations. In this section we investigate the applicability of some graph reduction techniques that aim at reducing the number of non-terminals in a k -terminal graph. We start by reviewing the so-called *Wye-Delta* operations in graph reductions. These operations consist of five basic rules, which we describe below.

1. *Degree-one reduction:* Delete a degree-one non-terminal and its incident edge.
2. *Series reduction:* Delete a degree-two non-terminal y and its incident edges (x, y) and (y, z) , and add a new edge (x, z) of capacity $\min\{c(x, y), c(y, z)\}$.
3. *Parallel reduction:* Replace all parallel edges by a single edge whose capacity is the sum over all capacities of parallel edges.
4. *Wye-Delta transformation:* Let x be a degree-three non-terminal with neighbours $\delta(x) = \{u, v, w\}$. Assume w.l.o.g.¹ that for any pair $(u, v) \in \delta(x)$, $c(u, x) + c(v, x) \geq c(w, x)$, where $w \in \delta(v) \setminus \{u, v\}$. Then we can delete x (along with all its incident edges) and add edges (u, v) , (v, w) and (w, u) with capacities $(c(u, x) + c(v, x) - c(w, x))/2$, $(c(v, x) + c(w, x) - c(u, x))/2$ and $(c(u, x) + c(w, x) - c(v, x))/2$, respectively.

¹ Suppose there exist a pair $(u, v) \in \delta(x)$ with $c(u, x) + c(v, x) < c(w, x)$, where $w \in \delta(v) \setminus \{u, v\}$. Then we can simply set $c(w, x) = c(u, x) + c(v, x)$, since any terminal minimum cut would cut the edges (u, x) and (v, x) instead of the edge (w, x) .

5. *Delta-Wye transformation*: Delete the edges of a triangle connecting x , y and z , introduce a new non-terminal vertex w and add new edges (w, x) , (w, y) and (w, z) with edge capacities $c(x, y) + c(x, z)$, $c(x, y) + c(y, z)$ and $c(x, z) + c(y, z)$ respectively.

The following lemma (which follows from the above definitions) shows that the above rules preserve exactly all terminal minimum cuts.

► **Lemma 5.** *Let G be a k -terminal graph and G' be a k -terminal graph obtained from G by applying one of the rules 1 – 5. Then G' is a quality 1-vertex cut sparsifier of G .*

For our application, it will be useful to enrich the set of rules by introducing two new operations. These operations can be realized as series of the operations 1-5.

6. *Edge deletion (with vertex x)*: For a degree-three non-terminal with neighbours u, v , the edge (u, v) can be deleted, if it exists. To achieve this, we use a Delta-Wye transformation followed by a series reduction.
7. *Edge replacement*: For a degree-four non-terminal vertex with neighbours x, u, v, w , if the edge (x, u) exists, then it can be replaced by the edge (v, w) . To achieve this, we use a Delta-Wye transformation followed by a Wye-Delta transformation.

A k -terminal graph G is *Wye-Delta* reducible to another k -terminal graph H , if G is reduced to H by repeatedly applying one of the operations 1-7. We obtain the following lemma, whose proof we defer to the full version.

► **Lemma 6.** *Let G and H be k -terminal graphs. Moreover, let G be Wye-Delta reducible to H . Then H is a quality 1-vertex cut sparsifier of G .*

Graph Embeddings. Throughout this paper, we will be dealing with the embedding of a planar graph into a square *grid* graph. One way of drawing graphs in the plane are *orthogonal grid-embeddings* [41]. In such a setting, the vertices correspond to distinct points and edges consist of alternating sequences of vertical and horizontal segments. Equivalently, one can view this as drawing our input graph as a subgraph of some grid. Formally, a *node-embedding* ρ of $G_1 = (V_1, E_1)$ into $G_2 = (V_2, E_2)$ is an injective mapping that maps V_1 into V_2 , and E_1 into paths in G_2 , i.e., (u, v) maps to a path from $\rho(u)$ to $\rho(v)$, such that every pair of paths that correspond to two different edges in G_1 is vertex-disjoint (except possibly at the endpoints). If G_2 is a planar graph, then $\rho(G_1)$ and G_1 are also planar. Thus, if G_1 and G_2 are planar we then refer to ρ as an *orthogonal embedding*. Moreover, given a planar graph G_1 drawn in the plane, the embedding ρ is called *region-preserving* if $\rho(G_1)$ and G_1 have the same planar topological embedding.

Let G_1 be a k -terminal graph. Since the embedding does not affect the vertices of G_1 , the terminals of G_1 are also terminals in $\rho(G_1)$. Although the embedding does not consider capacity of the edges in G_1 , we can still guarantee that such an embedding preserves all terminal minimum cuts, for which we make use of the following operation:

1. *Edge subdivision*: Let (u, v) be an edge of capacity $c(u, v)$. Delete (u, v) , introduce a new vertex w and add edges (u, w) and (w, v) , each of capacity $c(u, v)$.

The proof of the following Lemma is deferred to the full version.

► **Lemma 7.** *Let ρ be a node-embedding and let G_1 and $\rho(G_1)$ be k -terminal graphs defined as above. Then $\rho(G_1)$ preserves exactly all terminal minimum cuts of G .*

3 An Exact Vertex Cut Sparsifier of Size $O(k^2)$

In this section we show that given a k -terminal planar graph, where all terminals lie on the same face, one can construct a quality-1 vertex cut sparsifier of size $O(k^2)$. Note that it suffices to consider the case when all terminals lie on the *outer* face.

Embedding into Grids. It is well-known that one can obtain an orthogonal embedding of a planar graph with maximum-degree at most three into a grid (see Valiant [41]). However, our input planar graph can have arbitrarily large maximum degree. In order to be able to make use of such an embedding, we need to first reduce our input graph to a bounded-degree graph while preserving planarity and all terminal minimum cuts. We achieve this by making use of a *vertex splitting* technique, which we describe below.

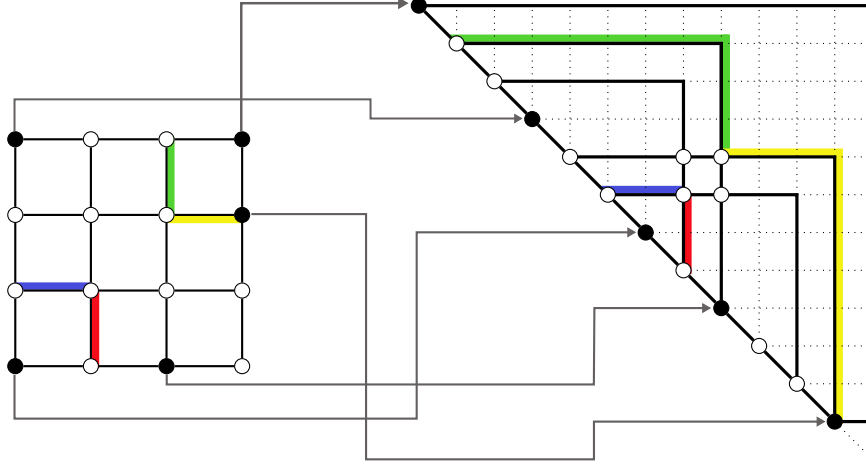
Given a k -terminal planar graph $G' = (V', E', c')$ with $K \subset V'$ lying on the outer face, vertex splitting produces a k -terminal planar graph $G = (V, E, c)$ with $K \subset V$ such that the maximum degree of G is at most three. Specifically, for each vertex v of degree $d > 3$ with neighboring vertices u_1, \dots, u_d , we delete v and introduce new vertices v_1, \dots, v_d along with edges $\{(v_i, v_{i+1}) : i = 1, \dots, d-1\}$, each of capacity $C + 1$, where $C = \sum_{e \in E'} c'(e)$. Further, we replace the edges $\{(u_i, v) : i = 1, \dots, d\}$ with $\{(u_i, v_i) : i = 1, \dots, d\}$, each of corresponding capacity. If v is a terminal vertex, we set one of the v_i 's to be a terminal vertex. It follows that the resulting graph G is planar and terminals can be still embedded on the outer face. Note that while the degree of every vertex v_i is at most 3, the degree of any other vertex is not affected. The proof of the claim below is deferred to the full version.

► **Claim 8.** *Let G' and G be k -terminal graphs defined as above. Then G preserves exactly all minimum terminal cuts of G' , i.e., G is a quality-1 cut sparsifier of G' .*

Let $G = (V, E)$ be a k -terminal graph obtained by vertex splitting of all vertices of degree larger than 3 of $G' = (V', E')$. Further, let $n' = |V'|$, $m' = |E'|$, $n = |V|$ and $m = |E|$. Then it is easy to show that $n \leq 2m'$ and $m \leq m' + n \leq 3m'$. Since G' is planar, we have that $n = O(n')$ and $m = O(n')$. Thus, by just a linear blow-up on the size of vertex and edge sets, we may assume w.l.o.g. that our input graph is a planar graph of degree at most three.

Valiant [41] and Tamassia et al. [38] showed that a k -terminal planar graph G with n vertices and degree at most three admits an orthogonal region-preserving embedding into some square grid of size $O(n) \times O(n)$. By Lemma 7, we know that the resulting graph exactly preserves all terminal minimum cuts of G . We remark that since the embedding is region-preserving, the outer face of the input graph is embedded to the outer face of the grid. Therefore, all terminals in the embedded graph lie on the outer face of the grid. Performing appropriate edge subdivisions, we can make all the terminals lie on the boundary of some possibly larger grid. Further, we can add dummy non-terminals and zero edge capacities to transform our graph into a full-grid H . We observe that the latter does not affect any terminal min-cut. The above leads to the following:

► **Lemma 9.** *Given a k -terminal planar graph G , where all terminals lie on the outer face, there exists a k -terminal grid graph H , where all terminals lie on the boundary such that H preserves exactly all terminal minimum cuts of G . The resulting graph has $O(n^2)$ vertices and edges.*



■ **Figure 1** Embedding grid into half-grid. Black vertices represent terminals while white vertices represent non-terminals. The counter-clockwise ordering starts at the top right terminal. Coloured edges and paths correspond to the mapping of the respective edges: blue for edges $((i, 1), (i, 2))$, red for edges $((n-1, j), (n, j))$, green for edges $((1, j), (2, j))$ and yellow for edges $((i, n-1), (i, n))$, where $i, j = 2, \dots, n-1$.

Embedding Grids into Half-Grids. Next, we show how to embed square grids into half-grid graphs (see Section 2), which will facilitate the application of Wye-Delta transformations. The existence of such an embedding was claimed in the thesis of Gitler [19], but no details on its construction were given.

Let G be a k -terminal square grid on $n \times n$ vertices where terminals lie on the boundary of the grid. We obtain the following:

► **Lemma 10.** *There exists a node embedding of the grid G into T_k^ℓ , where $\ell = 4n - 3$.*

Proof. Our construction works as follows (See Fig. 1 for an example). We first fix an ordering on the vertices lying on the boundary of the grid in the order induced by the grid. Then we embed each vertex according to that order into the diagonal vertices of the half-grid, along with the edges that form the boundary of the grid. The sub-grid obtained by removing all boundary vertices is embedded appropriately into the upper-part of the half-grid. Finally, we show how to embed edges between the boundary and the sub-grid vertices and argue that such an embedding is indeed vertex-disjoint for any pair of paths.

We start with the embedding of the vertices of G . Let us first consider the boundary vertices. The ordering imposed on these vertices can be viewed as starting with the upper-right vertex $(1, n)$ and visiting the rest of vertices in a counter-clockwise direction until reaching the vertex $(2, n)$. We map the vertices on the boundary as follows.

1. The vertex $(1, j)$ is mapped to the vertex $(n - j + 1, n - j + 1)$ for $j = 2, \dots, n$,
2. The vertex $(i, 1)$ is mapped to the vertex $(n + i - 1, n + i - 1)$ for $i = 1, \dots, n - 1$,
3. The vertex (n, j) is mapped to the vertex $(2n + j - 2, 2n + j - 2)$ for $j = 1, \dots, n - 1$,
4. The vertex (i, n) is mapped to the vertex $(4n - i - 2, 4n - i - 2)$ for $i = 2, \dots, n$.

Now we consider the vertices that belong to the induced sub-grid S of G of size $(n-2)^2$ when removing the boundary vertices of our input grid. We map the vertex (i, j) to the vertex $(n + i - 1, 2n + j - 2)$ for $i, j = 2, \dots, n - 1$. In other words, for every vertex of S we make a vertical shift by $n - 1$ units and an horizontal shift by $2n - 2$ units. By construction, it is not hard to check that every vertex of G is mapped to a different vertex of T_k^ℓ and all terminal vertices lie on the diagonal of T_k^ℓ .

We continue with the embedding of the edges of G . First, every edge between two boundary vertices in G is embedded to the edge between the corresponding mapped diagonal vertices of T_k^ℓ , except the edge between $(1, n)$ and $(2, n)$. For this edge, we define an edge embedding between the corresponding vertices $(1, 1)$ and $(4n - 4, 4n - 4)$ of T_k^ℓ by using the path:

$$(1, 1) \rightarrow (1, 2) \rightarrow \dots \rightarrow (1, 4n - 3) \rightarrow (2, 4n - 3) \rightarrow \dots \rightarrow (4n - 4, 4n - 3) \rightarrow (4n - 4, 4n - 4).$$

Next, every edge of the sub-grid S is embedded in to the edge connecting the mapped endpoints of that edge in T_k^ℓ . In other words, if (i, j) and (i', j') were connected by an edge e in S , then $(n + i - 1, 2n + j - 2)$ and $(n + i' - 1, 2n + j' - 2)$ are connected by an edge e' in T_k^ℓ and e is mapped to e' . Finally, the only edges that remain are those connecting a boundary vertex of G with a boundary vertex of S . We distinguish four cases depending on the edge position.

1. The edge $((i, 2), (i, 1))$ is mapped to the horizontal path given by:

$$(n + i - 1, 2n) \rightarrow (n + i - 1, 2n - 1) \rightarrow \dots \rightarrow (n + i - 1, n + i - 1) \text{ for } i = 2, \dots, n - 1.$$

2. The edge $((n - 1, j), (n, j))$ is mapped to the vertical path given by:

$$(2n - 2, 2n + j - 2) \rightarrow (2n - 1, 2n + j - 2) \rightarrow \dots \rightarrow (2n + j - 2, 2n + j - 2) \text{ for } j = 2, \dots, n - 1.$$

3. The edge $((2, j), (1, j))$ is mapped to the L -shaped path:

$$\begin{aligned} (n + 1, 2n + j - 2) &\rightarrow (n, 2n + j - 2) \rightarrow \dots \rightarrow (n - j + 1, 2n + j - 2) \\ &\rightarrow (n - j + 1, 2n + j - 3) \rightarrow \dots \rightarrow (n - j + 1, n - j + 1) \text{ for } j = 2, \dots, n - 1. \end{aligned}$$

4. The edge $((i, n - 1), (i, n))$ is mapped to the L -shaped path:

$$\begin{aligned} (n + i - 1, 3n - 3) &\rightarrow (n + i - 1, 3n - 2) \rightarrow \dots \rightarrow (n + i - 1, 4n - i - 2) \\ &\rightarrow (n + i, 4n - i - 2) \rightarrow \dots \rightarrow (4n - i - 2, 4n - i - 2) \text{ for } i = 2, \dots, n - 1. \end{aligned}$$

By construction, it follows that the paths in our edge embedding are vertex disjoint. \blacktriangleleft

Reducing Half-Grids and Bringing the Piece Together. We now review the construction of Gitler [19], which shows how to reduce half-grids to much smaller half-grids (excluding diagonal edges) whose size depends only on k . Recall that \hat{T}_k^n is the graph T_k^n without the diagonal edges. The proof of the lemma below is deferred to the full version.

► **Lemma 11** ([19]). *For any positive k, n with $k < n$, T_k^n is Wye-Delta reducible to \hat{T}_k^k .*

Combining the above reductions leads to the following theorem:

► **Theorem 12.** *Let G be a k -terminal planar graph where all terminals lie on the outer face. Then G admits a quality 1-vertex cut sparsifier of size $O(k^2)$, which is also a planar graph.*

Proof. Let n denote the number of vertices in G . First, we apply Lemma 9 on G to obtain a grid graph H with $O(n^2)$ vertices, which preserves exactly all terminal minimum cuts of G . We then apply Lemma 10 on H to obtain a node embedding ρ into the half-grid T_k^ℓ , where $\ell = 4n - 3$. By Lemma 7, $\rho(H)$ preserves exactly all terminal minimum cuts of H . We can further extend $\rho(H)$ to the full half-grid T_k^ℓ , if dummy non-terminals and zero edge capacities are added. Finally, we apply Lemma 11 on T_k^ℓ to obtain a Wye-Delta reduction to the reduced half-grid graph \hat{T}_k^k . It follows by Lemma 6 that \hat{T}_k^k is a quality 1-vertex cut sparsifier of T_k^ℓ , where the size guarantee is immediate from the definition of \hat{T}_k^k . \blacktriangleleft

The results about flow and distance sparsifiers are deferred to the full version.

4 Reachability-Preserving Minors for General Digraphs

In this section we show that any k -terminal digraph admits a reachability-preserving minor of size $O(k^4)$. We accomplish this by first restricting our attention to DAGs, and then showing how to generalize the result to any digraph. Details are deferred to the full version.

We first introduce the following definition. Given a k -terminal digraph G with a terminal pair-set P , we say that H is a *reachability-preserving minor with respect to P* , if H is a minor of G that preserves the reachability information only among the pairs in P . Note that the previous definition of reachability-preserving minor of G corresponds to the special case when the pair-set P is *trivial*, i.e., for any pair $x, x' \in K$, both (x, x') and (x', x) belong to P . Observe that the trivial pair-set contains $k(k - 1)$ terminal-pairs.

We next review a useful scheme for breaking ties between shortest paths connecting some vertex pair from P . This tie-breaking is usually achieved by slightly perturbing the edge lengths of the original graph such that no two paths have the same length (note that in our case, edge lengths are initially one). The perturbation gives a *consistent* scheme in the sense that whenever π is chosen as a shortest path, every sub-path of π is also chosen as a shortest path. Below we formalize these ideas using two definitions and a lemma from [4].

► **Definition 13** (Tie-breaking Scheme). Given a k -terminal G , a *shortest path tie breaking scheme* is a function π that maps every pair of vertices (s, t) to some shortest path between s and t in G . For any pair-set P , we let $\pi(P)$ denote the union over all shortest paths between pairs in P with respect to the scheme π .

► **Definition 14** (Consistency). A tie-breaking scheme is consistent if, for all vertices y, x, x', y' , if $x, x' \in \pi(y, y')$ with $d(y, x) < d(y, x')$, then $\pi(x, x')$ is a sub-path of $\pi(y, y')$.

► **Lemma 15** ([4]). *For any k -terminal G , there is a consistent tie-breaking scheme in G .*

Let G be a k -terminal DAG. Given a tie-breaking scheme π , the first step to construct a reachability-preserving minor is to start with an empty graph H and then for every pair $p \in P$, repeatedly add the shortest-path $\pi(p)$ to H . We can alternatively think of this as deleting vertices and edges that do not participate in any shortest path among terminal-pairs in P with respect to the scheme π . Clearly, the DAG $H = (V_H, E_H)$, $E_H := \pi(P)$, is a minor of G and preserves all reachability information among pairs in P . We next review the notion of a branching event, which will be useful to bound the size of H .

► **Definition 16** (Branching Event). A *branching event* is a set of two distinct directed edges $\{e_1 = (u_1, v), e_2 = (u_2, v)\}$ that enter the same node v .

► **Lemma 17.** *The DAG H has at most $|P|(|P| - 1)/2$ branching events.*

The proof of the above lemma is deferred to the full version. We now have all the tools to present our algorithm for constructing reachability-preserving minors for DAGs.

The proofs of the lemma and the theorem below are deferred to the full version.

► **Lemma 18.** *Given a k -terminal DAG G with a pair-set P , the above algorithm outputs a reachability-preserving minor H of size $O(|P|^2)$ for G with respect to P .*

► **Theorem 19.** *Given a k -terminal digraph G with a pair-set P , there is an algorithm that constructs a reachability-preserving minor H of size $O(|P|^2)$ with respect to P .*

Taking P to be the trivial pair-set we obtain the following corollary.

► **Corollary 20.** *Any k -terminal digraph admits reachability-preserving minor of size $O(k^4)$.*

Algorithm 1 MINORSPARSIFYDAG (k -terminal DAG G , pair-set P)

- 1: Set $H = \emptyset$ and compute a consistent tie-breaking scheme π for shortest paths in G .
 - 2: For each $p \in P$, add the shortest path $\pi(p)$ to H .
 - 3: **while** there is an edge (u, v) directed towards a non-terminal v with $\deg_H^-(v) = 1$ **do**
 - 4: Contract the edge (u, v) .
 - 5: **end while**
 - 6: **return** H
-

5 Reachability-Preserving Minors for Planar Digraphs

In this section we show that any k -terminal planar digraph G admits a reachability-preserving minor of size $O(k^2 \log^2 k)$. This matches the lower-bound of Theorem 3 up to a $O(\log^2 k)$ factor. The main idea is as follows. Given a k -terminal planar digraph G with the trivial pair-set P , $|P| = k(k-1)$, our goal will be to slightly increase the number of terminals while considerably reducing the size of the pair-set P , under the condition that no reachability information is lost among the terminal-pairs in P .

We apply the following Preprocessing Step. Given a k -terminal digraph G , we apply Theorem 19 to get a reachability-preserving minor G' . To simplify the notation, we will use G instead of G' , i.e., throughout we assume that G has at most $O(k^4)$ vertices.

Decomposition into Path-Separable Digraphs and the Algorithm. We say that a graph $G = (V, E)$ admits an α -separator if there exists a set $S \subset V$ whose removal partitions G into connected components, each of size at most $\alpha \cdot |V|$, where $1/2 \leq \alpha < 1$. If the vertices of S consist of the union over r paths of G , for some $r \geq 1$, we say that G is (α, r) -path separable. We now review the following reduction due to Thorup [39].

► **Theorem 21** ([39]). *Given a digraph G , we can construct a series of digraphs G_0, \dots, G_n such that the number of vertices and edges over all G_i 's is linear in the number of vertices and edges in G , and*

1. *Each vertex and edge of G appears in at most two G_i 's.*
2. *For all $u, v \in V$, if there is a dipath R from u to v in G , there is a G_i that contains R .*
3. *Each $G_i = (V_i, E_i)$ is $(1/2, 6)$ -path separable.*
4. *Each G_i is a minor of G . In particular, if G is planar, so is G_i .*

Now we review how directed reachability can be efficiently represented by separator dipaths. Let G be a k -terminal directed graph that contains some directed path Q . Assume that the vertices of Q are ordered in increasing order in the direction of Q . For each terminal $x \in K$, let $\text{to}_x[Q]$ be the first vertex in Q that can be reached by x , and let $\text{from}_x[Q]$ be the last vertex in Q that reaches x . Let (s, t) be a terminal pair and let R be the directed path from s to t in G . We say that R intersects Q iff s can reach $\text{to}_s[Q]$ and t can be reached from $\text{from}_t[Q]$ in Q , and $\text{to}_s[Q]$ precedes $\text{from}_t[Q]$ in Q .

We now are going to combine the above tools to give our labelling algorithm aimed at reducing the size of the trivial pair-set P . By Theorem 21, we restrict our attention only to the digraphs G_i . Let $K_i := V(G_i) \cap K$ be the set of terminals restricted to the graph G_i .

► **Lemma 22.** *Let G be a k -terminal planar digraph. Let $P' := \cup_{i=1}^{t-1} P'_i$ be the union over all pair-sets output by running Algorithm 2 below on each digraph G_i . Then the size of $|P'|$ is at most $O(k \log k)$. Moreover, if H is a reachability-preserving minor of G with respect to P' , then H is a reachability-preserving minor of G with respect to all terminal pairs.*

Algorithm 2 REDUCEPAIRSET (planar digraph G_i , terminals K_i)

```

1: if  $|V(G_i)| \leq 1$  or  $K_i = \emptyset$  then return  $\emptyset$ .
2: Let  $P'_i = \emptyset$  be the new pair-set.
3: Compute a  $1/2$ -separator  $S$  of  $G_i$  consisting of 6 dipaths by Item 4 of Theorem [39].
4: for each dipath  $Q \in S$  do
5:   // Addition of terminal connections with  $Q$ 
6:   Let  $Q'$  be the set of existing terminals of  $Q$ .
7:   for each terminal  $x \in K_i$  do
8:     Compute  $\text{to}_x[Q]$  and  $\text{from}_x[Q]$ , declare them terminals and add them to  $Q'$ .
9:     Add  $(x, \text{to}_x[Q])$  and  $(\text{from}_x[Q], x)$  to  $P'_i$ .
10:  end for
11:  // Sparsification of  $Q$  using  $Q'$ 
12:  Define directed pairs  $(s, t)$ , where  $s$  and  $t$  are consecutive terminals of  $Q'$ ,
    according to the ordering of  $Q$  and add all these pairs to  $P'_i$ .
13: end for
14: Let  $(G_i^{(1)}, K_i^{(1)})$  and  $(G_i^{(2)}, K_i^{(2)})$  be the resulting graphs from  $G \setminus S$ ,
    where  $K_i^{(1)}$  and  $K_i^{(2)}$  are disjoint subsets of the terminals  $K$  separated by  $S$ .
15: // Note that reachability info. about terminals in  $S$  are taken care of.
16: return  $P'_i \cup \bigcup_{j=1}^2 \text{REDUCEPAIRSET}(G_i^{(j)}, K_i^{(j)})$ .

```

Proof. By preprocessing, G has at most $O(k^4)$ vertices. Throughout, it will be useful to think of the above algorithm as simultaneously running it on each digraph G_i . By Item 2 of Theorem 21, each terminal appears in at most two G_i 's. Thus at each recursive level, there will be at most $O(k)$ active G_i 's. Also, note that the separator properties imply that there are $O(\log k)$ recursive calls overall.

We next bound the size of the pair-set P' . Let q denote the total number of newly added terminals in Line 8 per recursive level. Since there are $O(k)$ terminals, each adding at most $O(1)$ new terminals, it follows that $q = O(k)$. First, we argue about the number of pairs added in Line 9. Since this is bounded by $O(q)$, we get that there are $O(k \log k)$ pairs overall. Second, we bound the number of pairs added when sparsifying the separator paths, i.e., pair additions in Line 13. For all the separators in the same recursive level, we can write $q := \sum_i |Q'_i|$, where Q'_i denotes the set newly added terminals for some separator dipath (Line 7). By Line 12, it follows that we need only $(|Q'_i| - 1)$ pairs to represent each such dipath. Thus, per recursive call, the total number of pairs added in Line 13 is $O(q) = O(k)$. Summing these overall $O(\log k)$ levels, and combining this with the previous bound, gives the claimed bound on $|P'|$.

Finally, we argue that P' is a pair-set that can recover reachability information among terminals. Fix any terminal pair (s, t) and let R be a directed path from s to t in G . By Item 3 of Theorem 21, there is some digraph G_i that contains R . Then, R must intersect with some separator dipath Q , at some level of the recursion of the above algorithm on G_i . The above discussion gives that P' contains all the necessary information to give a (possibly) another directed path from s to t in G . ◀

Applying Theorem 19 on the digraph G with pair-set P' , as defined by the above lemma, we get Theorem 2.

References

- 1 Alfred V. Aho, M. R. Garey, and Jeffrey D. Ullman. The transitive reduction of a directed graph. *SIAM J. Comput.*, 1(2):131–137, 1972.
- 2 Alexandr Andoni, Anupam Gupta, and Robert Krauthgamer. Towards $(1 + \varepsilon)$ -approximate flow sparsifiers. In *Proc. of the 25th SODA*, pages 279–293, 2014.
- 3 András A. Benczúr and David R. Karger. Approximating s - t minimum cuts in $\tilde{O}(n^2)$ time. In *Proc. of the 28th STOC*, pages 47–55, 1996.
- 4 Greg Bodwin. Linear size distance preservers. In *Proc. of the 28th SODA*, pages 600–615, 2017.
- 5 T-H Hubert Chan, Donglin Xia, Goran Konjevod, and Andrea Richa. A tight lower bound for the steiner point removal problem on trees. In *Proc. of the 9th APPROX/RANDOM*, pages 70–81, 2006.
- 6 Moses Charikar, Tom Leighton, Shi Li, and Ankur Moitra. Vertex sparsifiers and abstract rounding algorithms. In *Proc. of the 51th FOCS*, pages 265–274, 2010.
- 7 Chandra Chekuri, Anupam Gupta, Ilan Newman, Yuri Rabinovich, and Alistair Sinclair. Embedding k -outerplanar graphs into l_1 . *SIAM J. Discrete Math.*, 20(1):119–136, 2006.
- 8 Chandra Chekuri, Sanjeev Khanna, and F Bruce Shepherd. Edge-disjoint paths in planar graphs with constant congestion. *SIAM J. Comput.*, 39(1):281–301, 2009.
- 9 Chandra Chekuri, F Bruce Shepherd, and Christophe Weibel. Flow-cut gaps for integer and fractional multiflows. In *Proc. of the 21st SODA*, pages 1198–1208, 2010.
- 10 Yun Kuen Cheung, Gramoz Goranci, and Monika Henzinger. Graph minors for preserving terminal distances approximately - Lower and Upper Bounds. In *Proc. of the 43rd ICALP*, pages 131:1–131:14, 2016.
- 11 Julia Chuzhoy. On vertex sparsifiers with steiner nodes. In *Proc. of the 44th STOC*, pages 673–688, 2012.
- 12 Julia Chuzhoy. Routing in undirected graphs with constant congestion. In *Proc. of the 44th STOC*, pages 855–874, 2012.
- 13 Don Coppersmith and Michael Elkin. Sparse sourcewise and pairwise distance preservers. *SIAM J. Discrete Math.*, 20(2):463–501, 2006.
- 14 Edward B Curtis, David Ingerman, and James A Morrow. Circular planar graphs and resistor networks. *Linear algebra and its applications*, 283(1):115–150, 1998.
- 15 Krzysztof Diks and Piotr Sankowski. Dynamic plane transitive closure. In *Proc. of the 15th ESA*, pages 594–604, 2007.
- 16 Michael Elkin, Arnold Filtser, and Ofer Neiman. Terminal embeddings. In *Proc. of the 18th APPROX/RANDOM*, pages 242–264, 2015.
- 17 Matthias Englert, Anupam Gupta, Robert Krauthgamer, Harald Räcke, Inbal Talgam-Cohen, and Kunal Talwar. Vertex sparsifiers: New results from old techniques. *SIAM J. Comput.*, 43(4):1239–1262, 2014.
- 18 Thomas A Feo and J Scott Provan. Delta-wye transformations and the efficient reduction of two-terminal planar graphs. *Operations Research*, 41(3):572–582, 1993.
- 19 Isidoro Gitler. *Delta-Wye-Delta Transformations: Algorithms and Applications*. PhD thesis, Department of Combinatorics and Optimization, University of Waterloo, 1991.
- 20 Gramoz Goranci and Harald Räcke. Vertex sparsification in trees. In *Proc. of the 14th WAOA*, pages 103–115, 2016.
- 21 Anupam Gupta. Steiner points in tree metrics don't (really) help. In *Proc. of the 12th SODA*, pages 220–227, 2001.
- 22 Torben Hagerup, Jyrki Katajainen, Naomi Nishimura, and Prabhakar Ragde. Characterizing multiterminal flow networks and computing flows in networks of small treewidth. *J. Comput. Syst. Sci.*, 57(3):366–375, 1998.

- 23 Lior Kamma, Robert Krauthgamer, and Huy L Nguyen. Cutting corners cheaply, or how to remove steiner points. *SIAM J. Comput.*, 44(4):975–995, 2015.
- 24 Irit Katriel, Martin Kutz, and Martin Skutella. Reachability substitutes for planar digraphs. In *Technical Report MPI-I-2005-1-002*. Max-Planck-Institut für Informatik, 2005.
- 25 Arindam Khan and Prasad Raghavendra. On mimicking networks representing minimum terminal cuts. *Inf. Process. Lett.*, 114(7):365–371, 2014.
- 26 Robert Krauthgamer, Huy L Nguyen, and Tamar Zondiner. Preserving terminal distances using minors. *SIAM J. Discrete Math.*, 28(1):127–141, 2014.
- 27 Robert Krauthgamer and Inbal Rika. Mimicking networks and succinct representations of terminal cuts. In *Proc. of the 24th SODA*, pages 1789–1799, 2013.
- 28 Robert Krauthgamer and Inbal Rika. Refined vertex sparsifiers of planar graphs. *CoRR*, abs/1702.05951, 2017.
- 29 Robert Krauthgamer and Tamar Zondiner. Preserving terminal distances using minors. In *Proc. of the 39th ICALP*, pages 594–605, 2012.
- 30 James R Lee, Manor Mendel, and Mohammad Moharrami. A node-capacitated okamura-seymour theorem. In *Proc. of the 45th STOC*, pages 495–504, 2013.
- 31 Frank Thomson Leighton and Ankur Moitra. Extensions and limits to vertex sparsification. In *Proc. of the 42nd STOC*, pages 47–56, 2010.
- 32 Konstantin Makarychev and Yury Makarychev. Metric extension operators, vertex sparsifiers and lipschitz extendability. In *Proc. of the 51th FOCS*, pages 255–264, 2010.
- 33 Jiří Matoušek. On the distortion required for embedding finite metric spaces into normed spaces. *Israel Journal of Mathematics*, 93(1):333–344, 1996.
- 34 Ankur Moitra. Approximation algorithms for multicommodity-type problems with guarantees independent of the graph size. In *Proc. of the 50th FOCS*, 2009.
- 35 Haruko Okamura and Paul D. Seymour. Multicommodity flows in planar graphs. *J. Comb. Theory, Ser. B*, 31(1):75–81, 1981.
- 36 Daniel A. Spielman and Shang-Hua Teng. Spectral sparsification of graphs. *SIAM J. Comput.*, 40(4):981–1025, 2011.
- 37 Sairam Subramanian. A fully dynamic data structure for reachability in planar digraphs. In *Proc. of the 1st ESA*, pages 372–383, 1993.
- 38 Roberto Tamassia and Ioannis G Tollis. Planar grid embedding in linear time. *IEEE Trans. Circuits Syst.*, 36(9):1230–1234, 1989.
- 39 Mikkel Thorup. Compact oracles for reachability and approximate distances in planar digraphs. *J. ACM*, 51(6):993–1024, 2004.
- 40 Mikkel Thorup and Uri Zwick. Approximate distance oracles. *J. ACM*, 52(1):1–24, 2005.
- 41 Leslie G. Valiant. Universality considerations in VLSI circuits. *IEEE Trans. Computers*, 30(2):135–140, 1981.