Large-Treewidth Graph Decompositions and Applications*

Chandra Chekuri† Julia Chuzhoy ‡

April 16, 2013

Abstract

Treewidth is a graph parameter that plays a fundamental role in several structural and algorithmic results. We study the problem of decomposing a given graph G into node-disjoint subgraphs, where each subgraph has sufficiently large treewidth. We prove two theorems on the tradeoff between the number of the desired subgraphs h, and the desired lower bound r on the treewidth of each subgraph. The theorems assert that, given a graph G with treewidth k, a decomposition with parameters h, r is feasible whenever $hr^2 \leq k/\text{poly}\log(k)$, or $h^3 r \leq k/\text{poly}\log(k)$ holds. We then show a framework for using these theorems to bypass the well-known Grid-Minor Theorem of Robertson and Seymour in some applications. In particular, this leads to substantially improved parameters in some Erdos-Pósa-type results, and faster algorithms for a class of fixed-parameter tractable problems.

*An extended abstract of this paper is to appear in *Proc. of ACM STOC*, 2013.

†Dept. of Computer Science, University of Illinois, Urbana, IL 61801. chekuri@illinois.edu. Supported in part by NSF grant CCF-1016684 and by TTI Chicago on a sabbatical visit.

‡Toyota Technological Institute, Chicago, IL 60637. Email: cjulia@ttic.edu. Supported in part by NSF CAREER grant CCF-0844872 and Sloan Research Fellowship.

1 Introduction

Let $G = (V, E)$ be an undirected graph. We assume that the reader is familiar with the notion of treewidth of a graph G, denoted by $\mathrm{tw}(G)$. The main question considered in this paper is the following. Given an undirected graph G, and integer parameters $h, r < \mathrm{tw}(G)$, can G be partitioned into h node-disjoint subgraphs $G_1, \ldots, G_h$ such that for each i, $\mathrm{tw}(G_i) \geq r$? It is easy to see that for this to be possible, $hr \leq \mathrm{tw}(G)$ must hold. Moreover, it is not hard to show examples of graphs G, where even for $r = 2$, the largest number of node-disjoint subgraphs of G with treewidth at least $r = 2$ is bounded by $h = O\left(\frac{\mathrm{tw}(G)}{\log(\mathrm{tw}(G))}\right)$.[1] In this paper we prove the following two theorems, that provide sufficient conditions for the existence of a decomposition with parameters h, r.

Theorem 1.1 *Let G be any graph with $\mathrm{tw}(G) = k$, and let h, r be any integers with $hr^2 \leq k/\operatorname{poly}\log k$. Then there is an efficient[2] algorithm to partition G into h node-disjoint subgraphs $G_1, \ldots, G_h$ such that $\mathrm{tw}(G_i) \geq r$ for each i.*

Theorem 1.2 *Let G be any graph with $\mathrm{tw}(G) = k$, and let h, r be any integers with $h^3 r \leq k/\operatorname{poly}\log k$. Then there is an efficient algorithm to partition G into h node-disjoint subgraphs $G_1, \ldots, G_h$ such that $\mathrm{tw}(G_i) \geq r$ for each i.*

We observe that the two theorems give different tradeoffs, depending on whether r is small or large. It is particularly useful in applications that the dependence is linear in one of the parameters. We make the following conjecture, that would strengthen and unify the preceding theorems.

Conjecture 1 *Let G be any graph with $\mathrm{tw}(G) = k$, and let h, r be any integers with $hr \leq k/\operatorname{poly}\log k$. Then G can be partitioned into h node-disjoint subgraphs $G_1, \ldots, G_h$ such that $\mathrm{tw}(G_i) \geq r$ for each i.*

Motivation and applications. The starting point for this work is the observation that a special case of Theorem 1.2, with $h = \Omega(\log^2 k)$, is a critical ingredient in recent work on poly-logarithmic approximation algorithms for routing in undirected graphs with constant congestion [Chu12, CL12, CE13]. In particular, [Chu12] developed such a decomposition for edge-disjoint routing, and subsequently [CE13] extended it to the node-disjoint case. However, in this paper, we are motivated by a different set of applications, for which Theorem 1.1 is more suitable. These applications rely on the seminal work of Robertson and Seymour [RS86], who showed that there is a large grid minor in any graph with sufficiently large treewidth. The theorem below, due to Robertson, Seymour and Thomas [RST94], gives an improved quantitative bound relating the size of the grid minor and the treewidth.

Theorem 1.3 (Grid-Minor Theorem [RST94]) *Let G be any graph, and g any integer, such that $\mathrm{tw}(G) \geq 20^{2g^5}$. Then G contains a $g \times g$ grid as a minor. Moreover, if G is planar, then $\mathrm{tw}(G) \geq 6g - 4$ suffices.*

Kawarabayashi and Kobayashi [KK12] obtained an improved bound of $2^{O(g^2 \log g)}$ on the treewidth required to ensure a $g \times g$ grid minor, and a further improvement to a bound of $2^{O(g \log g)}$ was recently claimed by Seymour [Sey12].

[1] Consider a constant degree n-node expander G with girth $\Omega(\log n)$; the existence of such graphs can be shown by the probabilistic method. Let $G_1, \ldots, G_h$ be any collection of node-disjoint subgraphs of G of treewidth at least 2 each. Then each graph G_i must contain a cycle, and by the lower bound on the girth of G, $|V(G_i)| = \Omega(\log n)$, implying that $h = O(n/\log n)$. On the other hand $\mathrm{tw}(G) = \Omega(n)$.

[2] In this paper we use the term efficient algorithm to refer to a randomized algorithm that runs in time polynomial in $|V(G)|$ and k.

Notice that Theorem 1.3 guarantees a grid minor of size sub-logarithmic in the treewidth k in general graphs, and of size $\Omega(k)$ in planar graphs. Demaine and Hajiaghayi [DH08] extended the linear relationship between the grid minor size and the treewidth to graphs that exclude a fixed graph H as a minor (the constant depends on the size of H, see [KK12] for an explicit dependence). A $g \times g$ grid has treewidth g, and it can be partitioned into h node-disjoint grids of size $r \times r$ each, as long as $r\sqrt{h} = O(g)$. Thus, in a general graph G of treewidth k, the Grid-Minor Theorem currently only guarantees that for any integers h, r with $hr^2 = O(\log^{2/5} k)$, there is a partition of G into h node-disjoint subgraphs of treewidth at least r each. Robertson et al. [RST94] observed that, in order for G to contain a $g \times g$ grid as a minor, its treewidth may need to be as large as $\Omega(g^2 \log g)$, and they suggest that this may be sufficient. Demaine et al. [DHK09] conjecture that the treewidth of $\Theta(g^3)$ is both necessary and sufficient.

The existence of a polynomial relationship between the grid-minor size and the graph treewidth is a fundamental open question, that appears to be technically very challenging to resolve. Our work is motivated by the observation that the Grid-Minor Theorem can be bypassed in various applications by using Theorems 1.1 and 1.2. We describe two general classes of such applications below.

Bounds for Erdos-Pósa type results. The duality between packing and covering plays a central role in graph theory and combinatorial optimization. One central result of this nature is Menger's theorem, which asserts that for any graph G, subsets S, T of its vertices, and an integer k, either G contains k node-disjoint paths connecting the vertices of S to the vertices of T, or there is a set X of at most $k-1$ vertices, whose removal disconnects all such paths. Erdos and Pósa [EP65] proved that for every graph G, either G contains k node-disjoint cycles, or there is a set X of $O(k \log k)$ nodes, whose removal from G makes the graph acyclic. More generally, a family $\mathcal{F}$ of graphs is said to satisfy the Erdos-Pósa property, iff there is an integer-valued function f, such that for every graph G, either G contains k disjoint subgraphs isomorphic to members of $\mathcal{F}$, or there is a set S of $f(k)$ nodes, such that $G - S$ contains no subgraph isomorphic to a member of $\mathcal{F}$. In other words, S is a cover, or a hitting set, for $\mathcal{F}$ in G. Erdos-Pósa-type results provide relationships between integral covering and packing problems, and are closely related to fractional covering problems and the integrality gaps of the corresponding LP relaxations.

As an illustrative example for the Erdos-Pósa-type results, let $\mathcal{F}_m$ denote the family of all cycles of length 0 modulo m. Thomassen [Tho88] has proved an Erdos-Pósa-type result for $\mathcal{F}_m$, by showing that for each graph G, either G contains k disjoint copies of cycles from $\mathcal{F}_m$, or there is a subset S of $f(k)$ vertices, whose removal disconnects all such cycles in G (here, $f(k) = 2^{2^{O(k)}}$, and m is considered to be a constant). The proof consists of two steps. In the first step, a simple inductive argument is used to show that for any graph G of treewidth at most w, either G contains k disjoint copies of cycles from $\mathcal{F}_m$, or there is a subset S of $O(kw)$ vertices, whose removal from G disconnects all such cycles. The second step is to show that if G has treewidth at least some value $g(k)$, then it must contain k disjoint copies of cycles from $\mathcal{F}_m$. Combining these two steps together, we obtain that $f(k) = O(k \cdot g(k))$. The second step uses Theorem 1.3 to show that, if $\text{tw}(G) \geq g(k) = 2^{m^{O(k)}}$, then G contains a grid minor of size $k(2m)^{2k-1} \times k(2m)^{2k-1}$. This grid minor is then in turn used to find k disjoint copies of cycles from $\mathcal{F}_m$ in G, giving $f(k) = 2^{m^{O(k)}}$.

Using Theorem 1.1, we can significantly strengthen this result, and obtain $f(k) = \tilde{O}(k)$, as follows.[3] Assume first that we are given any graph G, with $\text{tw}(G) \geq f'(m)k \operatorname{poly}\log k$, where $f'(m)$ is some function of m. Then, using Theorem 1.1, we can partition G into k vertex-disjoint subgraphs of treewidth at least $f'(m)$ each. Using known techniques (such as, e.g., Theorem 1.3), we can then show that each such subgraph must contain a copy of a cycle from $\mathcal{F}_m$. Therefore, if $\text{tw}(G) \geq f'(m)k \operatorname{poly}\log k$, then G contains k disjoint copies of cycles from $\mathcal{F}_m$. Combining this with Step 1 of

[3]Throughout the paper we use $\tilde{O}$ notation to suppress polylogarithmic factors.

the algorithm of Thomassen, we conclude that every graph G either contains k copies of cycles from $\mathcal{F}_m$, or there is a subset S of $f(k) = \tilde{O}(k^2)$ vertices, whose removal from G disconnects all such cycles; a stronger bound of $f(k) = \tilde{O}(k)$ can be obtained by refining the Step 1 argument using a divide and conquer analysis [FST11] (see Lemma 5.4 in Section 5).

There is a large body of work in graph theory and combinatorics on Erdos-Pósa-type results. Several of these rely on the Grid-Minor Theorem, and consequently the function $f(k)$ is shown to be exponential (or even worse) in k. Some fundamental results in this area can be improved to obtain a bound polynomial in k, using Theorem 1.1 and the general framework outlined above. For example, Robertson and Seymour [RS86] derived the following as an important consequence of the Grid-Minor Theorem. Given any fixed graph H, let $\mathcal{F}(H)$ be the family of all graphs that contain H as a minor. Then $\mathcal{F}(H)$ has the Erdos-Pósa property iff H is planar. However, the bound they obtained for $f(k)$ is exponential in k. Using the above general framework, we can show that $f(k) = O(k \cdot \operatorname{poly}\log(k))$ for any fixed H.

Improved running times for Fixed-Parameter Tractability. The theory of bidimensionality [DH07a] is a powerful methodology in the design of fixed-parameter tractable (FPT) algorithms. It led to sub-exponential (in the parameter k) time FPT algorithms for bidimensional parameters (formally defined in Section 5) in planar graphs, and more generally graphs that exclude a fixed graph H as a minor. The theory is based on the Grid-Minor Theorem. However, in general graphs, the weak bounds of the Grid-Minor Theorem meant that one could only derive FPT algorithms with running time of the form $2^{2^{O(k^{2.5})}} n^{O(1)}$, as shown by Demaine and Hajiaghayi [DH07b]. Our results lead to algorithms with running times of the form $2^{k\ \operatorname{poly}\log(k)} n^{O(1)}$ for the same class of problems as in [DH07b]. Thus, one can obtain FPT algorithms for a large class of problems in general graphs via a generic methodology, where the running time has a singly-exponential dependence on the parameter k.

The thrust of this paper is to prove Theorems 1.1 and 1.2, and to highlight their applicability as general tools. The applications described in Section 5 are of that flavor; we have not attempted to examine specific problems in depth. We believe that the theorems, and the technical ideas in their proofs, will have further applications.

Overview of techniques and discussion. A significant contribution of this paper is the formulation of the decomposition theorems for treewidth, and identifying their applications. The main new and non-trivial technical contribution is the proof of Theorem 1.1. The proof of Theorem 1.2 is similar in spirit to the recent work of [Chu12] and [CE13], who obtained a special case of Theorem 1.2 with $h = \operatorname{poly}\log k$, and used it to design algorithms for low-congestion routing in undirected graphs. We note that Theorem 1.1 gives a substantially different tradeoff between the parameters h, r and k, when compared to Theorem 1.2, and leads to the improved results for the two applications we mentioned earlier. Its proof uses new ingredients with a connection to decomposing expanders as explained below.

Contracted graph, well-linked decomposition, and expanders: The three key technical ingredients in the proof of Theorem 1.1 are in the title of the paragraph. To illustrate some key ideas we first consider how one may prove Theorem 1.1 if G is an n-vertex constant-degree expander, which has treewidth $\Omega(n)$. At a high level, one can achieve this as follows. We can take h disjoint copies of an expander with $\Omega(r)$ nodes each (the expansion certifies that treewidth of each copy is r), and "embed" them into G in a vertex-disjoint fashion. This is roughly possible, modulo various non-trivial technical issues, using short-path vertex-disjoint routing in expanders [LR99]. Now consider a general graph G. For instance it can be a planar graph on n nodes with treewidth $O(\sqrt{n})$; note that the ratio of treewidth to the number of nodes is very different than that in an expander. Here we employ a different strategy, where we cut along a small separator and retain large treewidth on both sides and apply this iteratively until we obtain the desired number of subgraphs. The non-trivial part of the proof is to be able to handle these different scenarios. Another technical difficulty is the following.

Treewidth of a graph is a global parameter and there can be portions of the graph that can be removed without changing the treewidth. It is not easy to cleanly characterize the minimal subgraph of G that has roughly the same treewidth as that of G. A key technical ingredient here is borrowed from previous work on graph decompositions [Räc02, Chu12], namely, the notion of a contracted graph. The contracted graph tries to achieve this minimality, by contracting portions of the graph that satisfy the following technical condition: they have a small boundary and the boundary is well-linked with respect to the contracted portion. Finally, a recurring technical ingredient is the notion of a well-linked decomposition. This allows us to remove a small number of edges while ensuring that the remaining pieces have good conductance. This high-level clustering idea has been crucial in many applications.

Related work on grid-like minors and (perfect) brambles: An important ingredient in the decomposition results is a need to certify that the treewidth of a given graph is large, say at least r. Interestingly, despite being NP-Hard to compute, the treewidth of a graph G has an exact min-max formula via the bramble number [ST93] (see Section 2). However, Grohe and Marx [GM09] have shown that there are graphs G (in fact expanders) for which a polynomial-sized bramble can only certify that treewidth of G is $\Omega(\sqrt{k})$ where $k = \mathrm{tw}(G)$; certifying that G has larger treewidth would require super-polynomial sized brambles. Kreutzer and Tamari [KT10], building on [GM09], gave efficient algorithms to construct brambles of order $\tilde{\Omega}(\sqrt{k})$. They also gave efficient algorithms to compute "grid-like" minors introduced by Reed and Wood [RW12] where it is shown that G has a grid-like minor of size ℓ as long as $\mathrm{tw}(G) = \Omega(\ell^4\sqrt{\log \ell})$. In some applications it is feasible to use a grid-like minor in place of a grid and obtain improved results. Kreutzer and Tamari [KT10] used them to define perfect brambles and gave a meta-theorem to obtain FPT algorithms, for a subclass of problems considered in [DH07b], with a single-exponential dependence on the parameter k. Our approach in this paper is different, and in a sense orthogonal, as we explain below.

First, a grid-like minor is a single connected structure that does not allow for a decomposition into disjoint grid-like minors. This limitation means one needs a global argument to show that a grid-like minor of a certain size implies a lower bound on some parameter of interest. In contrast, our theorems are specifically tailored to decompose the graph and then apply a local argument in each subgraph, typically to prove that the parameter is at least one in each subgraph. The advantage of our approach is that it is agnostic to how one proves a lower bound in each subgraph; we could use the Grid-Minor Theorem or the more efficient grid-like minor theorem in each subgraph. Kreutzer and Tazari [KT10] derive efficient FPT algorithms for a subclass of problems considered in [DH07b] where the class is essentially defined as those problems for which one can use a grid-like minor in place of a grid in the global sense described above. In contrast, we can generically handle all the problems considered in [DH07b] as explained in Section 5.

Second, we discuss the efficiency gains possible via our approach. It is well-known that an α-approximation for sparse vertex separators gives an $O(\alpha)$-approximation for treewidth. Feige et al. [FHL08] obtain an $O(\sqrt{\log \mathrm{tw}(G)})$-approximation for treewidth. Thus we can efficiently certify treewidth to within a much better factor via separators than with brambles. More explicitly, well-linked sets provide a compact certificate for treewidth; informally, a set of vertices X is well-linked in G if there are no small separators for X — see Section 2 for formal definitions. The tradeoffs we obtain through well-linked sets are stronger than via brambles. In particular, the FPT algorithms that we obtain have a running time $2^{k \text{ poly}\log(k)} n^{O(1)}$ where k is the parameter of interest. In contrast the algorithms obtained via perfect brambles in [KT10] have running times of the form $2^{\mathrm{poly}(k)} n^{O(1)}$ where the polynomial is incurred due to the inefficiency in the relationship between treewidth and the size of a grid-like minor. Although the precise dependence on k depends on the parameter of interest, the current bounds require at least a quadratic dependence on k.

Organization: Sections 3 and 4 contain the proofs of Theorem 1.1 and Theorem 1.2 respectively. Section 5 describes the applications; it relies only on the statement of Theorem 1.1 and can be read independently.

2 Preliminaries and Notation

Given a graph G and a set of vertices A, we denote by $\text{out}_G(A)$ the set of edges with exactly one end point in A and by $E_G(A)$ the set of edges with both end points in A. For disjoint sets of vertices A, B the set of edges with one end point in A and the other in B is denoted by $E_G(A, B)$. When clear from context, we omit the subscript G. All logarithms are to the base of 2. We use the following simple claim several times, and its proof appears in the Appendix.

Claim 2.1 *Let $\{x_1, \ldots, x_n\}$ be a set of non-negative integers, with $\sum_i x_i = N$, and $x_i \leq 2N/3$ for all i. Then we can efficiently compute a partition (A, B) of $\{1, \ldots, n\}$, such that $\sum_{i \in A} x_i, \sum_{i \in B} x_i \geq N/3$.*

Graph partitioning. Suppose we are given any graph $G = (V, E)$ with a set T of vertices called terminals. Given any partition $(S, \overline{S})$ of $V(G)$, the *sparsity* of the cut $(S, \overline{S})$ with respect to T is $\Phi(S, \overline{S}) = \frac{|E(S,\overline{S})|}{\min\{|T \cap S|, |T \cap \overline{S}|\}}$. The *conductance* of the cut $(S, \overline{S})$ is $\Psi(S, \overline{S}) = \frac{|E(S,\overline{S})|}{\min\{|E(S)|, |E(\overline{S})|\}}$. We then denote: $\Phi(G) = \min_{S \subset V}\{\Phi(S, \overline{S})\}$, and $\Psi(G) = \min_{S \subset V}\{\Psi(S, \overline{S})\}$. Arora, Rao and Vazirani [ARV09] showed an efficient algorithm that, given a graph G with a set T of k terminals, produces a cut $(S, \overline{S})$ with $\Phi(S, \overline{S}) \leq \alpha_{\text{ARV}}(k) \cdot \Phi(G)$, where $\alpha_{\text{ARV}}(k) = O(\sqrt{\log k})$. Their algorithm can also be used to find a cut $(S, \overline{S})$ with $\Psi(S, \overline{S}) \leq \alpha_{\text{ARV}}(m) \cdot \Psi(G)$, where $m = |E(G)|$. We denote this algorithm by $\mathcal{A}_{\text{ARV}}$, and its approximation factor by α_{ARV} from now on.

Well-linkedness, bramble number and treewidth. The treewidth of a graph $G = (V, E)$ is typically defined via tree decompositions. A tree-decomposition for G consists of a tree $T = (V(T), E(T))$ and a collection of sets $\{X_v \subseteq V\}_{v \in V(T)}$ called bags, such that the following two properties are satisfied: (i) for each edge $ab \in E$, there is some node $v \in V(T)$ with both $a, b \in X_v$ and (ii) for each vertex $a \in V$, the set of all nodes of T whose bags contain a form a non-empty (connected) subtree of T. The *width* of a given tree decomposition is $\max_{v \in V(T)} |X_v| - 1$, and the treewidth of a graph G, denoted by $\text{tw}(G)$, is the width of a minimum-width tree decomposition for G.

It is convenient to work with well-linked sets instead of treewidth. We describe the relationship between them after formally defining the notion of well-linkedness that we require.

Definition 2.1 *We say that a set T of vertices is α-well-linked*[4] *in G, iff for any partition (A, B) of the vertices of G into two subsets, $|E(A, B)| \geq \alpha \cdot \min\{|A \cap T|, |B \cap T|\}$.*

Definition 2.2 *We say that a set T of vertices is* node-well-linked *in G, iff for any pair (T_1, T_2) of equal-sized subsets of T, there is a collection $\mathcal{P}$ of $|T_1|$* **node-disjoint** *paths, connecting the vertices of T_1 to the vertices of T_2. (Note that T_1, T_2 are not necessarily disjoint, and we allow empty paths).*

Lemma 2.1 (Reed [Ree97]) *Let k be the size of the largest node-well-linked set in G. Then $k \leq \text{tw}(G) \leq 4k$.*

We also use the notion of brambles, defined below. Brambles will help us relate the different notions of well-linkedness, and treewidth to each other.

[4]This notion of well-linkedness is based on edge-cuts and we distinguish it from node-well-linkedness that is directly related to treewidth. For technical reasons it is easier to work with edge-cuts and hence we use the term well-linked to mean edge-well-linkedness, and explicitly use the term node-well-linkedness when necessary.

Definition 2.3 *A bramble in a graph G is a collection $\mathcal{B} = \{G_1, \ldots, G_r\}$ of connected sub-graphs of G, where for every pair G_i, G_j of the subgraphs, either G_i and G_j share at least one vertex, or there is an edge $e = (u, v)$ with $u \in G_i$, $v \in G_j$. We say that a set S of vertices is a hitting set for the bramble $\mathcal{B}$, iff for each $G_i \in \mathcal{B}$, $S \cap V(G_i) \neq \emptyset$. The order of the bramble $\mathcal{B}$ is the minimum size of any hitting set S for $\mathcal{B}$. The bramble number of G, $\mathrm{BN}(G)$, is the maximum order of any bramble in G.*

Theorem 2.1 *[ST93] For every graph G, $\mathrm{tw}(G) = \mathrm{BN}(G) - 1$.*

We then obtain the following simple corollary, whose proof appears in the Appendix.

Corollary 2.1 *Let G be any graph with maximum vertex degree at most Δ, and let T be any subset of vertices, such that T is α-well-linked in G, for some $0 < \alpha < 1$. Then $\mathrm{tw}(G) \geq \frac{\alpha \cdot |T|}{3\Delta} - 1$.*

Lemma 2.1 guarantees that any graph G of treewidth k contains a set X of $\Omega(k)$ vertices, that is node-well-linked in G. Kreutzer and Tazari [KT10] give a constructive version of this lemma, obtaining a set X with slightly weaker properties. Lemma 2.2 below rephrases, in terms convenient to us, Lemma 3.7 in [KT10] .

Lemma 2.2 *There is an efficient algorithm, that, given a graph G of treewidth k, finds a set X of $\Omega(k)$ vertices, such that X is $\alpha^* = \Omega(1/\log k)$-well-linked in G. Moreover, for any partition (X_1, X_2) of X into two equal-sized subsets, there is a collection $\mathcal{P}$ of paths connecting every vertex of X_1 to a distinct vertex of X_2, such that every vertex of G participates in at most $1/\alpha^*$ paths in $\mathcal{P}$.*

Well-linked decompositions. Let S be any subset of vertices in G. We say that S is α-good[5], iff for any partition (A, B) of S, $|E(A, B)| \geq \alpha \cdot \min\{|\mathrm{out}(A) \cap \mathrm{out}(S)|, |\mathrm{out}(B) \cap \mathrm{out}(S)|\}$. An equivalent definition is as follows. Start with graph G and subdivide each edge $e \in \mathrm{out}_G(S)$ by a vertex t_e. Let $\mathcal{T}_S = \{t_e \mid e \in \mathrm{out}_G(S)\}$ be the set of these new vertices, and let H be the sub-graph of the resulting graph, induced by $S \cup \mathcal{T}_S$. Then S is α-good in G iff $\mathcal{T}_S$ is α-well-linked in H.

A set $D : \mathrm{out}(S) \times \mathrm{out}(S) \to \mathbb{R}^+$ of demands defines, for every pair $e, e' \in \mathrm{out}(S)$, a demand $D(e, e')$. We say that D is a c-restricted set of demands, iff for every $e \in \mathrm{out}(S)$, $\sum_{e' \in \mathrm{out}(S)} D(e, e') \leq c$. Assume that S is an α-good subset of vertices in G. From the duality of cuts and flows, and from the known bounds on the flow-cut gap in undirected graphs [LLR95], if D is any set of c-restricted demands over $\mathrm{out}(S)$, then it can be fractionally routed inside $G[S]$ with edge-congestion at most $O(c \log k'/\alpha)$, where $k' = |\mathrm{out}(S)|$.

The following theorem, in its many variations, (sometimes under the name of "well-linked decomposition") has been used extensively in routing and graph decomposition (see e.g. [Räc02, CKS04, CKS05, RZ10, And10, Chu12, CL12, CE13]). For completeness, the proof appears in Appendix.

Theorem 2.2 *Let S be any subset of vertices of G, with $|\mathrm{out}(S)| = k'$, and let $0 < \alpha < \frac{1}{8\alpha_{\mathrm{ARV}}(k') \cdot \log k'}$ be a parameter. Then there is an efficient algorithm to compute a partition $\mathcal{W}$ of S, such that for each $W \in \mathcal{W}$, $|\mathrm{out}(W)| \leq k'$ and W is α-good. Moreover, $\sum_{W \in \mathcal{W}} |\mathrm{out}(W)| \leq k'(1 + 16\alpha \cdot \alpha_{\mathrm{ARV}}(k') \cdot \log k') = k'(1 + O(\alpha \log^{3/2} k'))$. The parameter $\alpha_{\mathrm{ARV}}(k')$ can be set to 1 if the efficiency of the algorithm is not relevant.*

Pre-processing to reduce maximum degree. Let G be any graph with $\mathrm{tw}(G) = k$. The proofs of Theorems 1.1 and 1.2 work with edge-well-linked sets instead of the node-well-linked ones. In order to be able to translate between both types of well-linkedness and the treewidth, we need to reduce

[5]The same property was called "bandwidth property" in [Räc02], and in [Chu12, CL12], set S with this property was called α-well-linked. We choose this notation to avoid confusion with other notions of well-linkedness used in this paper.

the maximum vertex degree of the input graph G. Using the cut-matching game of Khandekar, Rao and Vazirani [KRV09], one can reduce the maximum vertex degree to $O(\log^3 k)$, while only losing a $\text{poly} \log k$ factor in the treewidth, as was noted in [CE13] (see Remark 2.2). We state the theorem formally below. A brief proof sketch appears in the Appendix for completeness.

Theorem 2.3 *Let G be any graph with treewidth k. Then there is an efficient randomized algorithm to compute a subgraph G' of G, with maximum vertex degree at most $O(\log^3 k)$ such that $\text{tw}(G') = \Omega(k/\log^6 k)$.*

Remark 2.3 *In fact a stronger result, giving a constant bound on the maximum degree follows from the expander embedding result in [CE13]. However, the bound on the treewidth guaranteed is worse than in the preceding theorem by a (large) polylogarithmic factor. For our algorithms, the polylogarithmic bound on the degree guaranteed by Theorem 2.3 is sufficient.*

3 Proof of Theorem 1.1

We start with a graph G whose treewidth is at least k. For our algorithm, we need to know the value of the treewidth of G, instead of the lower bound on it. We can compute the treewidth of G approximately, to within an $O(\log(\text{tw}(G)))$-factor, using the algorithm of Amir [Ami10]. Therefore, we assume that we are given a value $k' \geq k$, such that $\Omega(k'/\log k') \leq \text{tw}(G) \leq k'$. We then apply Theorem 2.3, to obtain a subgraph G' of G of maximum vertex degree $\Delta = O(\log^3 k')$ and treewidth $\Omega(k'/\log^7 k')$. Using Lemma 2.2, we compute a subset T of $\Omega(k'/\log^7 k')$ vertices, such that T is $\Omega(1/\log k')$-well-linked in G'.[6]

In order to simplify the notation, we denote G' by G and $|T|$ by k from now on. From the above discussion, $\text{tw}(G) \leq ck \log^7 k$ for some constant c, T is $\Omega(1/\log k)$-well-linked in G, and the maximum vertex degree in G is $\Delta = O(\log^3 k)$; we define the parameter α^* to be $\Omega(1/\log k)$ which is the well-linkedness guarantee given by Lemma 2.2. It is now enough to find a collection $G_1, \ldots, G_h$ of vertex-disjoint subgraphs of G, such that $\text{tw}(G_i) \geq r$ for each i. We use the parameter $r' = c'\Delta^2 r \log^{11} k$, where c' is a sufficiently large constant. We assume without loss of generality that k is large enough, so, for example, $k \geq c'' r \log^{30} k$, where c'' is a large enough constant. We also assume without loss of generality that G is connected.

Definition 3.1 *We say that a subset S of vertices in G is an acceptable cluster, iff $|\text{out}(S)| \leq r'$, $|S \cap T| \leq |T|/2$, and S is α_G-good, for $\alpha_G = \frac{1}{256\alpha_{\text{ARV}}(k)\log k} = \Theta\left(\frac{1}{\log^{1.5} k}\right)$.*

Notice that since the maximum vertex degree in G is bounded by $\Delta < r'$, if S consists of a single vertex, then it is an acceptable cluster. Given any partition $\mathcal{C}$ of the vertices of G into acceptable clusters, we let $H_{\mathcal{C}}$ be the *contracted graph* associated with $\mathcal{C}$. Graph $H_{\mathcal{C}}$ is obtained from G by contracting every cluster $C \in \mathcal{C}$ into a single vertex v_C, that we refer to as a super-node. We delete self-loops, but leave parallel edges. Notice that the maximum vertex degree in $H_{\mathcal{C}}$ is bounded by r'. We denote by $\varphi(\mathcal{C})$ the total number of edges in $H_{\mathcal{C}}$. Below is a simple observation that follows from the α^*-well-linkedness of T in G.

Observation 3.1 *Let $\mathcal{C}$ be any partition of the vertices of G into acceptable clusters. Then $\varphi(\mathcal{C}) \geq \alpha^* k/3$.*

[6] The bounds we claim here are somewhat loose although they do not qualitatively affect our theorems. For instance [FHL08] gives an $O(\sqrt{\log(\text{tw}(G))})$-approximation for treewidth which improves the bound in [Ami10].

Proof: We use Claim 2.1 to find a partition $(\mathcal{A},\mathcal{B})$ of $\mathcal{C}$, such that $\sum_{C\in\mathcal{A}}|T\cap C|, \sum_{C\in\mathcal{B}}|T\cap C| \geq |T|/3$. We then set $A=\bigcup_{C\in\mathcal{A}} C$, $B=\bigcup_{C\in\mathcal{B}} C$. Since T is α^*-well-linked in G, $|E(A,B)|\geq \alpha^*|T|/3=\alpha^*k/3$. Since the edges of $E(A,B)$ also belong to $H_{\mathcal{C}}$, the claim follows. □

Throughout the algorithm, we maintain a partition $\mathcal{C}$ of $V(G)$ into acceptable clusters. At the beginning, $\mathcal{C}=\{\{v\}\mid v\in V(G)\}$. We then perform a number of iterations. In each iteration, we either compute a partition of G into h disjoint sub-graphs, of treewidth at least r each, or find a new partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters, such that $\varphi(\mathcal{C}')\leq\varphi(\mathcal{C})-1$. The execution of each iteration is summarized in the following theorem.

Theorem 3.1 *There is an efficient algorithm, that, given a partition $\mathcal{C}$ of $V(G)$ into acceptable clusters, either computes a partition of G into h disjoint subgraphs of treewidth at least r each, or returns a new partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters, such that $\varphi(\mathcal{C}')\leq\varphi(\mathcal{C})-1$.*

Clearly, after applying Theorem 3.1 at most $|E(G)|$ times, we obtain a partition of G into h disjoint subgraphs of treewidth at least r each. From now on we focus on proving Theorem 3.1. Given a current partition $\mathcal{C}$ of $V(G)$ into acceptable clusters, let H denote the corresponding contracted graph. We denote $n=|V(H)|$, $m=|E(H)|$. Notice that from Observation 3.1, $m\geq\alpha^*k/3$. We now consider two cases, and prove Theorem 1.1 separately for each of them. The first case is when $n\geq k^5$.

3.1 Case 1: $n\geq k^5$

We note that n is large when compared to the treewidth and hence we expect the graph H should have low expansion. Otherwise, we get a contradiction by showing that $\mathrm{tw}(G)>ck\log^7 k$. The proof strategy in the low-expansion regime is to repeatedly decompose along balanced partitions to obtain h subgraphs with treewidth at least r each.

Let $z=k^5$. The algorithm first chooses an arbitrary subset Z of z vertices from H that remains fixed throughout the algorithm. Suppose we are given any subset S of vertices of H. We say that a partition (A,B) of S is γ*-balanced* (with respect to Z), iff $\min\{|A\cap Z|,|B\cap Z|\}\geq\gamma|S\cap Z|$. We say that it is *balanced* iff it is γ-balanced for $\gamma=\frac{1}{4}$. The following claim is central to the proof of the theorem in Case 1.

Claim 3.1 *Let S be any subset of vertices in H with $|S\cap Z|>100$, and let (A,B) a balanced partition of S (with respect to Z), minimizing $|E_H(A,B)|$. Then $|E_H(A,B)|\leq k^2$.*

Proof: To simplify notation, we denote $H[S]$ by H'. Assume that the claim is not true, and assume without loss of generality that $|A\cap Z|\geq|B\cap Z|$. We claim that the set A of vertices is 1-good in H'. Indeed, assume otherwise. Then there is a partition (X,Y) of A, with

$$|E_{H'}(X,Y)|<\min\{|\mathrm{out}_{H'}(X)\cap\mathrm{out}_{H'}(A)|,|\mathrm{out}_{H'}(Y)\cap\mathrm{out}_{H'}(A)|\}.$$

Assume without loss of generality that $|X\cap Z|\geq|Y\cap Z|$. We claim that $(X,B\cup Y)$ is a balanced partition for S, with $|E_H(X,B\cup Y)|<|E_H(A,B)|$, contradicting the minimality of $|E(A,B)|$. To see that $(X,B\cup Y)$ is a balanced partition, observe that $|(B\cup Y)\cap Z|\geq\frac{|S\cap Z|}{4}$ since (A,B) is a balanced partition, and $|X\cap Z|\geq\frac{1}{2}|A\cap Z|\geq\frac{1}{4}|A\cap Z|$ from our assumptions about X and A. Finally, observe that

$$\begin{aligned}|E_H(X, B\cup Y)| &= |E_{H'}(X, B\cup Y)|\\ &= |E_{H'}(X,Y)| + |E_{H'}(X,B)|\\ &< |E_{H'}(Y,B)| + |E_{H'}(X,B)|\\ &= |E_{H'}(A,B)| = |E_H(A,B)|,\end{aligned}$$

contradicting the minimality of $|E_H(A,B)|$. We conclude that the set A of vertices is 1-good in H'. Let Γ be the subset of vertices of A that serve as endpoints to edges in $\text{out}_{H'}(A)$, that is, $\Gamma = \{v \in A \mid \exists e = (u,v) \in \text{out}_{H'}(A)\}$. Then $|\Gamma| \geq \frac{|E_{H'}(A,B)|}{r'} \geq \frac{k^2}{r'}$, since the degrees of vertices in H are bounded by r'. It is also easy to see that Γ is 1-well-linked in the graph $H[A]$, since A is a 1-good set in H'.

Finally, let Γ' be a subset of $|\Gamma|$ vertices in the original graph G, obtained as follows. For each super-node $v_C \in \Gamma$, we select an arbitrary vertex u on the boundary of C (that is, $u \in C$, and it is an endpoint of some edge in $\text{out}(C)$), and add it to Γ'. We claim that the set Γ' of vertices is $\Omega\left(\frac{\alpha_G}{\log k}\right)$-well-linked in G. Indeed, let (X,Y) be any partition of the vertices of G, with $\Gamma'_X = \Gamma' \cap X, \Gamma'_Y = \Gamma' \cap Y$, and assume without loss of generality that $|\Gamma'_X| \leq |\Gamma'_Y|$. We need to show that $|E(X,Y)| \geq \Omega\left(\frac{\alpha_G |\Gamma'_X|}{\log k}\right)$. In order to show this, it is enough to show that there is a flow F, where the vertices in Γ'_X send one flow unit each to the vertices of Γ'_Y, and the total edge-congestion caused by this flow is at most $O(\log k/\alpha_G)$. Let (Γ_X, Γ_Y) be the partition of Γ induced by the partition (Γ'_X, Γ'_Y) of Γ'. Since the set Γ of vertices is 1-well-linked in H, there is a flow F' in H, where every vertex in Γ_X sends 1 flow unit towards the vertices in Γ_Y, every vertex in Γ_Y receives at most one flow unit, and the edge-congestion is at most 1. We now extend this flow F' to obtain the desired flow F in the graph G. In order to do so, we need to specify how the flow is routed across each cluster $C \in \mathcal{C}$. For each such cluster C, flow F defines a set D_C of 2-restricted demands over $\text{out}(C)$ (the factor 2 comes from both the flow routed across the cluster, and the flow that originates or terminates in it). Since C is α_G-well-linked, this set D_C of demands can be routed inside C with congestion at most $O(\log r'/\alpha_G) \leq O(\log k/\alpha_G)$. Concatenating the flow F' with the resulting flows inside each cluster $C \in \mathcal{C}$ gives the desired flow F. We conclude that we have obtained a set Γ of at least $\frac{k^2}{r'}$ vertices in G, such that Γ is $\Omega(\alpha_G/\log k)$-well-linked. From Corollary 2.1, it follows that $\text{tw}(G) \geq \Omega\left(\frac{\alpha_G(k)\cdot k^2}{r'\cdot\Delta\log k}\right) = \Omega\left(\frac{k^2}{r\cdot\log^{22.5}k}\right) > ck\log^7 k$, since we have assumed that $k \geq c''r\log^{30}k$ for a large enough constant c''. This contradicts the fact that $\text{tw}(G) \leq ck\log^7 k$. □

We now show an algorithm to find the desired collection $G_1, \ldots, G_h$ of subgraphs of G. We use the algorithm $\mathcal{A}_{\text{ARV}}$ of Arora, Rao and Vazirani [ARV09] to find a balanced partition of a given set S of vertices of H; the algorithm is applied to H with $S \cap Z$ as the terminals. Given any such set S of vertices, the algorithm $\mathcal{A}_{\text{ARV}}$ returns a γ_{ARV}-balanced partition (A,B) of S, with $|E_H(A,B)| \leq \alpha_{\text{ARV}}(z)\cdot \mathsf{OPT}$, where OPT is the smallest number of edges in any balanced partition, and γ_{ARV} is some constant. In particular, from Claim 3.1, $|E(A,B)| \leq \alpha_{\text{ARV}}(z)\cdot k^2$, if $|S\cap Z| \geq 100$.

We start with $\mathcal{S} = \{V(H)\}$, and perform h iterations. At the beginning of iteration i, set $\mathcal{S}$ will contain i disjoint subsets of vertices of H. An iteration is executed as follows. We select a set $S \in \mathcal{S}$, maximizing $|Z \cap S|$, and compute a γ_{ARV}-balanced partition (A,B) of S, using the algorithm $\mathcal{A}_{\text{ARV}}$. We then remove S from $\mathcal{S}$, and add A and B to it instead. Let $\mathcal{S} = \{X_1, \ldots, X_{h+1}\}$ be the final collection of sets after h iterations. From Claim 3.1, the increase in $\sum_{X\in\mathcal{S}} |\text{out}_H(X)|$ is bounded by $k^2\alpha_{\text{ARV}}(z)$ in each iteration. Therefore, throughout the algorithm, $\sum_{X\in\mathcal{S}} |\text{out}_H(X)| \leq k^2\alpha_{\text{ARV}}(z)h$ holds. In the following observation, we show that for each $X_i \in \mathcal{S}$, $|X_i \cap Z| \geq \frac{\gamma_{\text{ARV}}\cdot z}{2h}$.

Observation 3.2 *Consider some iteration i of the algorithm. Let $\mathcal{S}_i$ be the collection of vertex subsets at the beginning of iteration i, let $S \in \mathcal{S}_i$ be the set that was selected in this iteration, and let $\mathcal{S}_{i+1}$ be the set obtained after replacing S with A and B. Then $|A \cap Z|, |B \cap Z| \geq \gamma_{\text{ARV}} \cdot |S \cap Z|$, and for each $S' \in \mathcal{S}_{i+1}$, $|S' \cap Z| \geq \frac{\gamma_{\text{ARV}} \cdot z}{2h}$.*

Proof: The proof is by induction on the number of iterations. At the beginning of iteration 1, $\mathcal{S}_1 = \{V(H)\}$, so the claim clearly holds for $\mathcal{S}_1$. Assume now that the claim holds for iterations $1, \ldots, i-1$, and consider iteration i, where some set $S \in \mathcal{S}_i$ is replaced by A and B. Since (A, B) is a γ_{ARV}-balanced cut of S, and $|S| \geq \frac{\gamma_{\text{ARV}} \cdot z}{2h} > 100$, it follows that $|A \cap Z|, |B \cap Z| \geq \gamma_{\text{ARV}} \cdot |S \cap Z|$.

Since we have assumed that the claim holds for iterations $1, \ldots, i-1$, and we have shown that $|A \cap Z|, |B \cap Z| \geq \gamma_{\text{ARV}} \cdot |S \cap Z|$, it follows that the ratio $\max_{S' \in \mathcal{S}_{i+1}}\{|S' \cap Z|\}/\min_{S' \in \mathcal{S}_{i+1}}\{|S' \cap Z|\} \leq 1/\gamma_{\text{ARV}}$. Therefore, for each $S' \in \mathcal{S}_{i+1}$, $|S' \cap Z| \geq \frac{\gamma_{\text{ARV}} \cdot z}{2h}$ □

Among the sets $X_1, \ldots, X_{h+1}$, there can be at most one set X_i, with $|T \cap \left(\bigcup_{v_C \in X_i} C\right)| > |T|/2$. We assume without loss of generality that this set is X_{h+1}, and we will ignore it from now on. Consider now some set X_i, for $1 \leq i \leq h$. Since graph H is connected, and X_i contains at least $\frac{\gamma_{\text{ARV}} \cdot z}{2h}$ vertices (the vertices of $X_i \cap Z$), while $|\text{out}_H(X_i)| \leq k^2 h \alpha_{\text{ARV}}(z)$, it follows that $|E_H(X_i)| \geq \frac{1}{2}\left(\frac{\gamma_{\text{ARV}} \cdot z}{2h} - k^2 h \alpha_{\text{ARV}}(z)\right) \geq \frac{\gamma_{\text{ARV}} \cdot z}{8h} > 64|\text{out}_H(X_i)|$, as $z = k^5$, and k is large enough.

Let X'_i be the subset of vertices obtained from X_i, by un-contracting all super-nodes of X_i. Then $|E_G(X'_i)| \geq |E_H(X_i)| \geq 64|\text{out}_H(X_i)| = 64|\text{out}_G(X'_i)|$.

Our next step is to compute a decomposition $\mathcal{W}_i$ of X'_i into α_G-good clusters, using Theorem 2.2. Notice that $k' = |\text{out}_G(X'_i)| \leq k^2 h \alpha_{\text{ARV}}(z) \leq 5k^2 h \alpha_{\text{ARV}}(k) < k^4$ since $z = k^5$; therefore the choice of $\alpha_G = \frac{1}{256\alpha_{\text{ARV}}(k)\log k} < \frac{1}{8\alpha_{\text{ARV}}(k')\log k'}$ satisfies the conditions of the theorem.

Fix some $1 \leq i \leq h$. Assume first that for every cluster $C_i \in \mathcal{W}_i$, $|\text{out}_G(C_i)| \leq r'$. Then we can obtain a new partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters with $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$, as follows. We add to $\mathcal{C}'$ all clusters $C \in \mathcal{C}$ that are disjoint from X'_i, and we add all clusters in $\mathcal{W}_i$ to it as well. Clearly, the resulting partition $\mathcal{C}'$ consists of acceptable clusters only. We now show that $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$. Indeed,

$$\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - |\text{out}_H(X_i)| - |E_H(X_i)| + \sum_{R \in \mathcal{W}_i} |\text{out}_G(R)|$$

From the choice of α_G, $\sum_{R \in \mathcal{W}_i} |\text{out}_G(R)| < 3|\text{out}_G(X'_i)| = 3|\text{out}_H(X_i)|$ holds, while $|E_H(X_i)| \geq 64|\text{out}_H(X_i)|$. Therefore, $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$.

Assume now that for each $1 \leq i \leq h$, there is at least one cluster $C_i \in \mathcal{W}_i$ with $|\text{out}_G(C_i)| \geq r'$. Let $\{C_1, \ldots, C_h\}$ be the resulting collection of clusters, where for each i, $C_i \in \mathcal{W}_i$. For $1 \leq i \leq h$, we now let $G_i = G[C_i]$. It is easy to see that the graphs $G_1, \ldots, G_h$ are vertex-disjoint. It now only remains to show that each graph G_i has treewidth at least r. Fix some $1 \leq i \leq h$, and let $\Gamma_i \subseteq C_i$ contain the endpoints of edges in $\text{out}_G(C_i)$, that is, $\Gamma_i = \{v \in C_i \mid \exists e = (u, v) \in \text{out}_G(C_i)\}$. Then, since C_i is an α_G-good set of vertices, Γ_i is α_G-well-linked in the graph G_i. Moreover, $|\Gamma_i| \geq |\text{out}(C_i)|/\Delta \geq r'/\Delta$. From Corollary 2.1, $\text{tw}(G_i) \geq \frac{\alpha_G r'}{3\Delta^2} - 1 \geq r$.

3.2 Case 2: $n < k^5$

Since vertex degrees in H are bounded by r', $m = O(k^5 r') = O(k^6)$. The algorithm for Case 2 consists of two phases. In the first phase, we partition $V(H)$ into a number of disjoint subsets $X_1, \ldots, X_\ell$, where, on the one hand, for each X_i, the conductance of $H[X_i]$ is large, while, on the other hand,

$\sum_{i=1}^{\ell} |\text{out}(X_i)| \leq |E(H)|/10$. We discard all clusters X_i with $|\text{out}(X_i)| \geq |E(X_i)|/2$, denoting by $\mathcal{X}$ the collection of the remaining clusters, and show that $\sum_{X_i \in \mathcal{X}} |E(X_i)| = \Omega(\alpha^* k)$. If any cluster $X \in \mathcal{X}$ has $|E(X)| \leq 2r'$, then we find a new partition $\mathcal{C}'$ of the vertices of G into acceptable clusters, with $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$. Therefore, we can assume that for every cluster $X \in \mathcal{X}$, $|E(X)| > 2r'$. We then proceed to the second phase. Here, we take advantage of the high conductance of each $X_i \in \mathcal{X}$ to show that X_i can be partitioned into h_i vertex-disjoint sub-graphs, such that we can embed a large enough expander into each such subgraph. The value h_i is proportional to $|E(X_i)|$, and we ensure that $\sum_{X_i \in \mathcal{X}} h_i \geq h$ to get the desired number of subgraphs. The embedding of the expander into each sub-graph is then used as a certificate that this sub-graph (or more precisely, a sub-graph of G obtained after un-contracting the super-nodes) has large treewidth.

Phase 1 We use the following theorem, that allows us to decompose any graph into a collection of high-conductance connected components, by only removing a small fraction of the edges. A similar procedure has been used in previous work, and can be proved using standard graph decomposition techniques. The proof is deferred to the Appendix.

Theorem 3.2 *Let H be any connected n-vertex graph containing m edges. Then there is an efficient algorithm to compute a partition $X_1, \ldots, X_\ell$ of the vertices of H, such that: (i) for each $1 \leq i \leq \ell$, the conductance of graph $H[X_i]$, $\Psi(H[X_i]) \geq \frac{1}{160\alpha_{\text{ARV}}(m)\log m}$; and (ii) $\sum_{i=1}^{\ell} |\text{out}(X_i)| \leq m/10$.*

The algorithm in phase 1 uses Theorem 3.2 to partition the contracted graph H into a collection $\{X_1, \ldots, X_\ell\}$ of clusters. Recall that $m = |E(H)|$, and $n = |V(H)|$. We are guaranteed that $\sum_{i=1}^{\ell} |E(X_i)| \geq 0.9m$ and $\sum_{i=1}^{\ell} |\text{out}(X_i)| \leq 0.1m$ from Theorem 3.2.

Let $\mathcal{X}'$ contain all clusters X_i with $|\text{out}(X_i)| \geq \frac{1}{2}|E(X_i)|$, and let $\mathcal{X}$ contain all remaining clusters. Notice that $\sum_{X_i \in \mathcal{X}'} |E(X_i)| \leq 2\sum_{X_i \in \mathcal{X}'} |\text{out}(X_i)| \leq 2\sum_{i=1}^{\ell} |\text{out}(X_i)| \leq 0.2m$. Therefore, $\sum_{X_i \in \mathcal{X}} |E(X_i)| \geq \frac{1}{2}m \geq \frac{\alpha^* k}{6}$ from Observation 3.1. From now on we only focus on clusters in $\mathcal{X}$.

Assume first that there is some cluster $X_i \in \mathcal{X}$, with $|E(X_i)| \leq 2r'$. We claim that in this case, we can find a new partition $\mathcal{C}'$ of the vertices of G into acceptable clusters, with $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$. We first need the following simple observation.

Observation 3.3 *Assume that for some $X_i \in \mathcal{X}$, $|E(X_i)| \leq 2r'$ holds. Let $X' = \bigcup_{v_C \in X_i} C$. Then $|X' \cap T| < |T|/2$.*

Proof: Assume otherwise. Observe that since $X_i \in \mathcal{X}$, $|\text{out}(X_i)| \leq |E(X_i)|/2 \leq r'$ must hold. Let $\mathcal{C}' \subseteq \mathcal{C}$ be the collection of all acceptable clusters, whose corresponding super-nodes belong to X_i. Let $C^* = \bigcup_{v_C \in V(H) \setminus X_i} C$, and let $\mathcal{C}'' = \mathcal{C}' \cup \{C^*\}$. From our assumption, $|C^* \cap T| \leq |T|/2$, so we can use Claim 2.1 to partition the clusters in $\mathcal{C}''$ into two subsets, $\mathcal{A}$ and $\mathcal{B}$, such that $\sum_{C \in \mathcal{A}} |T \cap C|, \sum_{C \in \mathcal{B}} |T \cap C| \geq |T|/3$. Let $A = \bigcup_{C \in \mathcal{A}} C$ and let $B = \bigcup_{C \in \mathcal{B}} C$. Then $|E_G(A, B)| \geq \alpha^* |T|/3$, from the α^*-well-linkedness of the set T of terminals. Let $E' = E_H(X_i) \cup \text{out}_H(X_i)$. Observe that E' is also a subset of edges of G, and it disconnects A from B in G. However, $|E'| \leq 3r' < \alpha^* k/3$, a contradiction. □

Let X_i' be the set of vertices of G, obtained from X_i by un-contracting the super-nodes of X_i. We apply Theorem 2.2 to the set X_i' of vertices, to obtain a partition $\mathcal{W}_i$ of X_i' into α_G-good clusters. It is easy to see that all clusters in $\mathcal{W}_i$ are acceptable, since we are guaranteed that for each $R \in \mathcal{W}_i$, $|\text{out}(R)| \leq |\text{out}(X_i')| \leq r'$, and $|R \cap T| < |T|/2$. The new partition $\mathcal{C}'$ of the vertices of G into acceptable clusters is obtained as follows. We include all clusters of $\mathcal{C}$ that are disjoint from X_i', and we additionally include all clusters in $\mathcal{W}_i$. From the above discussion, all clusters in $\mathcal{C}'$ are acceptable. It now only remains to bound $\varphi(\mathcal{C}')$. It is easy to see that $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - |E_H(X_i)| - |\mathrm{out}_H(X_i)| + \sum_{R \in \mathcal{W}_i} |\mathrm{out}(R)|$. The choice of α_G ensures that $\sum_{R \in \mathcal{W}_i} |\mathrm{out}(R)| \leq 1.25|\mathrm{out}_G(X'_i)| = 1.25|\mathrm{out}_H(X_i)|$. Since $|E_H(X_i)| > 2|\mathrm{out}_H(X_i)|$, we get that $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$.

From now on, we assume that for every cluster $X_i \in \mathcal{X}$, $|E(X_i)| \geq 2r'$.

Phase 2 For convenience, we assume without loss of generality, that $\mathcal{X} = \{X_1, \ldots, X_z\}$. For $1 \leq i \leq z$, let $m_i = |E(X_i)|$. Recall that from the above discussion, $m_i \geq r'$, and $\sum_{i=1}^{z} m_i \geq \alpha^* k/6$. We set $h_i = \lceil \frac{6 m_i h}{\alpha^* k} \rceil$. Let $X'_i = \bigcup_{v_C \in X_i} C$. In the remainder of this section, we will partition, for each $1 \leq i \leq z$, the graph $G[X'_i]$ into h_i vertex-disjoint subgraphs, of treewidth at least r each. Since $\sum_{i=1}^{z} h_i \geq \sum_{i=1}^{z} \frac{6 m_i h}{\alpha^* k} \geq h$, this will complete the proof of Theorem 1.1.

From now on, we focus on a specific graph $H[X_i]$, and its corresponding un-contracted graph $G[X'_i]$. Our algorithm performs h_i iterations. In the first iteration, we embed an expander over $r'' = r \operatorname{poly}\log k$ vertices into $H[X_i]$. We then partition $H[X_i]$ into two sub-graphs: H_1, containing all vertices that participate in this embedding, and H'_1 containing all remaining vertices. Our embedding will ensure that $\sum_{v \in V(H_1)} d_H(v) \leq r^2 \operatorname{poly}\log k$, or in other words, we can obtain H'_1 from $H[X_i]$ by removing only $r^2 \operatorname{poly}\log k$ edges from it, and deleting isolated vertices. We then proceed to the second iteration, and embed another expander over r'' vertices into H'_1. This in turn partitions H'_1 into H_2, that contains the embedding of the expander, and H'_2, containing the remaining edges. In general, in iteration j, we start with a sub-graph H'_{j-1} of H, and partition it into two subgraphs: H_j containing the embedding of an expander, and H'_j that becomes an input to the next iteration. Since we ensure that for each graph H_j, the total out-degree of its vertices is bounded by $r^2 \operatorname{poly}\log k$, each residual graph H'_j is guaranteed to contain a large fraction of the edges of the original graph $H[X_i]$. We show that this in turn guarantees that H'_j contains a large sub-graph with a large conductance, which will in turn allow us to embed an expander over a subset of r'' vertices into H'_j in the following iteration.

We start with the following theorem that forms the technical basis for iteratively embedding multiple expanders of certain size into a larger expander.

Theorem 3.3 *Let* $\mathbf{G}$ *be any graph with* $|E(\mathbf{G})| = m$ *and* $\Psi(\mathbf{G}) \geq \gamma$, *where* $\gamma \leq 0.1$. *Let* $\mathbf{H}$ *be a sub-graph obtained from* $\mathbf{G}$ *by removing some subset* S_0 *of vertices and all their adjacent edges, so* $\mathbf{H} = \mathbf{G} - S_0$. *Assume further that* $|E(\mathbf{G}) \setminus E(\mathbf{H})| \leq \gamma m/8$. *Then we can efficiently compute a subset* S *of vertices in* $\mathbf{H}$, *such that* $\mathbf{H}[S]$ *contains at least* $m/2$ *edges and has conductance at least* $\frac{\gamma}{4\alpha_{\mathrm{ARV}}(m)}$.

Proof: We show an algorithm to find the required set S of vertices. We start with $S = V(\mathbf{H})$, and a collection $\mathcal{W}$ of disjoint vertex subsets of $\mathbf{H}$, that is initially empty. We then perform a number of iterations, where in each iteration we apply the algorithm $\mathcal{A}_{\mathrm{ARV}}$ for approximating a minimum-conductance cut to graph $\mathbf{H}[S]$. If the algorithm returns a partition (A, B) of S, such that $|E_{\mathbf{G}}(A)| \leq |E_{\mathbf{G}}(B)|$, and $|E_{\mathbf{G}}(A, B)| < \gamma |E_{\mathbf{G}}(A)|/4$, then we delete the vertices of A from S, add A to $\mathcal{W}$, and proceed to the next iteration. Otherwise, if the conductance of the computed cut is at least $\gamma/4$, then we terminate the algorithm, and we let S^* denote the final set S. Then clearly, the conductance $\Psi(\mathbf{H}[S^*]) \geq \frac{\gamma}{4\alpha_{\mathrm{ARV}}(m)}$. It now only remains to prove that $\mathbf{H}[S^*]$ contains at least $m/2$ edges.

Let R_0 denote the set of all edges that belong to $E(\mathbf{G})$ but not to $E(\mathbf{H})$. Notice that each such edge has at most one endpoint in the initial set S. We keep track of an edge set R that is initialized to R_0. In each iteration, we add some edges to R, charging them to the edges that already belong to R. We will ensure that if S is the current set, then all edges in $\mathrm{out}(S)$ belong to R. Set R will also contain all edges in $\bigcup_{A \in \mathcal{W}} \mathrm{out}(A)$.

Consider some iteration where we have computed a partition (A, B) of the current set S, and deleted the vertices of A from S. Notice that this means that $|E_{\mathbf{G}}(A,B)| < \gamma|E_{\mathbf{G}}(A)|/4$. Consider the partition (A,C) of the vertices of $\mathbf{G}$, where $C = B \cup (V(\mathbf{G}) \setminus S)$. Since $\Psi(\mathbf{G}) \geq \gamma$, $|E_{\mathbf{G}}(A,C)| \geq \gamma|E_{\mathbf{G}}(A)|$. Since $E_{\mathbf{G}}(A,C) \subseteq E_{\mathbf{G}}(A,B) \cup (\text{out}_{\mathbf{G}}(S) \cap \text{out}_{\mathbf{G}}(A))$, and $|E_{\mathbf{G}}(A,B)| < \gamma|E_{\mathbf{G}}(A)|/4$, we have that $|\text{out}_{\mathbf{G}}(S) \cap \text{out}_{\mathbf{G}}(A)| \geq 3\gamma|E_{\mathbf{G}}(A)|/4 \geq 3|E_{\mathbf{G}}(A,B)|$.

The edges of $\text{out}_{\mathbf{G}}(S) \cap \text{out}_{\mathbf{G}}(A)$ must already belong to R. We add the edges of $E_{\mathbf{G}}(A,B)$ to R, and we charge their cost to the edges of $\text{out}_{\mathbf{G}}(S) \cap \text{out}_{\mathbf{G}}(A)$. Each edge in $\text{out}_{\mathbf{G}}(S) \cap \text{out}_{\mathbf{G}}(A)$ is then charged at most $1/3$, and each such edge will never be charged again, as none of its endpoints is any longer contained in the new set $S = B$.

Using this charging scheme, the total direct charge to each edge of R is at most $1/3$, and the total amount charged to each edge in R_0 (including direct and indirect charging) is a geometric series whose sum is bounded by 1. Therefore, $|R| \leq 2|R_0| \leq \gamma m/4$.

We now assume for contradiction that $E_{\mathbf{H}}(S^*) = |E_{\mathbf{G}}(S^*)| < m/2$. Recall that $S_0 = V(\mathbf{G}) \setminus V(\mathbf{H})$. Observe that for each cluster $A \in \mathcal{W}$, $|E_{\mathbf{G}}(A)| \leq m/2$, since, when A was added to $\mathcal{W}$, there was another cluster B disjoint from A with $|E_{\mathbf{G}}(A)| \leq |E_{\mathbf{G}}(B)|$. Let $\mathcal{W}' = \mathcal{W} \cup \{S_0, S^*\}$. From the above discussion, for each set $Z \in \mathcal{W}'$, $|E_{\mathbf{G}}(Z)| \leq m/2$, while $\sum_{Z \in \mathcal{W}'} |E_{\mathbf{G}}(Z)| \geq m - |R| \geq 0.9m$. Using Claim 2.1, we can partition the clusters in $\mathcal{W}'$ into two subsets, $\mathcal{A}, \mathcal{B}$, such that $\sum_{Z \in \mathcal{A}} |E_{\mathbf{G}}(Z)|, \sum_{Z \in \mathcal{B}} |E_{\mathbf{G}}(Z)| \geq 0.3m$. Let $X = \bigcup_{Z \in \mathcal{A}} Z$, $Y = \bigcup_{Z \in \mathcal{B}} Z$. Then, since $\mathbf{G}$ has conductance at least γ, $|E_{\mathbf{G}}(X,Y)| \geq 0.3\gamma m$. However, $E_{\mathbf{G}}(X,Y) \subseteq R$, and as we have shown, $|R| \leq \gamma m/4 < 0.3\gamma m$, a contradiction. □

The next theorem is central to the execution of Phase 2. The theorem shows that, if we are given a sub-graph H' of H that has a high enough conductance, and contains at least r' edges, then we can find a subset S of r' vertices of H', such that the following holds: if $S' = \bigcup_{v_C \in S} C$, and $G' = G[S']$, then $\text{tw}(G') \geq r$. In order to show this, we embed an expander over a set of $r'' = r \operatorname{poly}\log k$ of vertices into H', and define S to be the set of all vertices of H' participating in this embedding. The embedding of the expander into $H'[S]$ is then used to certify that the treewidth of the resulting graph G' is at least r. The proof of the following theorem is deferred to the Appendix.

Theorem 3.4 *Let H' be any vertex-induced subgraph of H, such that $|E(H')| \geq r'$, and $\Psi(H') \geq \frac{1}{640\alpha^2_{\text{ARV}}(m)\log m}$. Then there is an efficient algorithm to find a subset S of at most r' vertices of H', such that, if G' is obtained from $H'[S]$ by un-contracting the super-nodes in S, then $\text{tw}(G') \geq r$.*

We are now ready to complete the description of the algorithm for Phase 2. Our algorithm considers each one of the subsets $X_i \in \mathcal{X}$ of vertices separately. Fix some $1 \leq i \leq z$. If $h_i = 1$, then by Theorem 3.4, graph $G[X_i]$ has treewidth at least r. Otherwise, we perform h_i iterations. At the beginning of every iteration j, we are given some vertex-induced subgraph H_j of $H[X_i]$, with $|E(H_j)| \geq m_i/2 \geq r'$ and $\Psi(H_j) \geq \frac{1}{640\alpha^2_{\text{ARV}}(m)\log m}$. At the beginning, $H_1 = H[X_i]$, and as observed before, $\Psi(H_1) \geq \frac{1}{160\alpha_{\text{ARV}}(m)\log m} \geq \frac{1}{640\alpha^2_{\text{ARV}}(m)\log m}$. In order to execute the jth iteration, we apply Theorem 3.4 to graph $H' = H_j$, and compute a subset S of at most r' vertices of H'. We denote this set of vertices by S^i_j, and we let $H^i_j = H[S^i_j]$. We also let G^i_j be the sub-graph of G, obtained by un-contracting the super-nodes of H^i_j. From Theorem 3.4, $\text{tw}(G^i_j) \geq r$. We then apply Theorem 3.3 to graph $\mathbf{G} = H[X_i]$, set $S_0 = \bigcup_{j'=1}^{j} S^i_{j'}$, and $\mathbf{H} = \mathbf{G} \setminus S_0$, to obtain the graph $H_{j+1} = \mathbf{H}[S]$ that becomes an input to the next iteration.

In order to show that we can carry this process out for h_i iterations, it is enough to prove that $\sum_{j=1}^{h_i} \sum_{v \in S^i_j} d_H(v) \leq \gamma m_i/8$, where $\gamma = \frac{1}{160\alpha_{\text{ARV}}(m)\log m}$. Indeed, since the vertex degrees in H are bounded by r',

$$\sum_{j=1}^{h_i} \sum_{v \in S_j^i} d_H(v) \le \sum_{j=1}^{h_i} r' \cdot |S_j^i| \le (r')^2 \cdot h_i \le O\left(\frac{m_i r^2 h \operatorname{poly}\log k}{\alpha^* k}\right),$$

by substituting $h_i = \lceil \frac{6 m_i h}{\alpha^* k} \rceil$. Since we assume that $r^2 h < k/\operatorname{poly}\log k$, and $m = O(k^6)$, it follows that the sum is bounded by $\frac{m_i}{1280 \alpha_{\text{ARV}}(m) \log m}$, as required.

Our final collection of subgraphs is $\Pi = \{G_j^i \mid 1 \le i \le z, 1 \le j \le h_i\}$. From the above discussion, Π contains $\sum_{i=1}^{z} h_i \ge h$ subgraphs of treewidth at least r each.

4 Proof of Theorem 1.2

We again use Theorem 2.3 to obtain a graph G' whose maximum vertex degree is $O(\log^3 k)$, and $\text{tw}(G') = \Omega(k/\operatorname{poly}\log k)$. From Lemma 2.2, there is an efficient algorithm to find a subset T of $\Omega(k/\operatorname{poly}\log k)$ vertices, that we refer to as *terminals* from now on, such that T is α^*-well-linked, for $\alpha^* = \Omega(1/\log k)$. For notational convenience, we denote G' by G and $|T|$ by k from now on. Using this notation, the maximum vertex degree in G is bounded by $\Delta = O(\log^3 k)$. We use a parameter $r' = 2^{20} r \Delta^2 h \alpha_{\text{ARV}}(k)$, and we assume that $k \ge 2^{10} h^2 r' \log k/\alpha^* = 2^{30} h^3 r \Delta^2 \log k \cdot \alpha_{\text{ARV}}(k)/\alpha^* = \Omega(h^3 r \operatorname{poly}\log k)$.

We say that a subset $C \subseteq V(G)$ of vertices is an *acceptable* cluster, iff $|\text{out}(C)| \le r'$, $|C \cap T| \le |T|/2$, and $G[C]$ is connected. As before, given a partition $\mathcal{C}$ of the vertices of G into acceptable clusters, we define a corresponding contracted graph $H_{\mathcal{C}}$, obtained from G by contracting every cluster $C \in \mathcal{C}$ into a super-node v_C and removing self-loops. We again denote by $\varphi(\mathcal{C})$ the number of edges in the graph $H_{\mathcal{C}}$. Observe that if C is a cluster consisting of a single node, then C is acceptable. Therefore, the partition $\{\{v\} \mid v \in V(G)\}$ of the vertices of G, where every vertex belongs to a separate cluster is a partition into acceptable clusters. The following observation is analogous to Observation 3.1. Its proof is identical and is omitted here.

Observation 4.1 *Let $\mathcal{C}$ be any partition of $V(G)$ into acceptable clusters, and let $H_{\mathcal{C}}$ be the corresponding contracted graph. Then $|E(H_{\mathcal{C}})| \ge \alpha^* k/3$.*

As before, our algorithm consists from a number of iterations. At the beginning of each iteration, we are given a partition $\mathcal{C}$ of the vertices of G into acceptable clusters, and the corresponding contracted graph H. The initial partition, that serves as the input to the first iteration, is $\{\{v\} \mid v \in V(G)\}$. The execution of each iteration is summarized in the following theorem.

Theorem 4.1 *Given a partition $\mathcal{C}$ of the vertices of G into acceptable clusters, there is an efficient randomized algorithm, that w.h.p. either computes a new partition $\mathcal{C}'$ of the vertices of G into acceptable clusters with $\varphi(\mathcal{C}') \le \varphi(\mathcal{C}) - 1$, or finds a partition of G into h disjoint subgraphs with treewidth at least r each.*

By applying theorem 4.1 to graph G repeatedly, we are guaranteed to find a partition of G into h disjoint subsets of treewidth at least r each after $O(|E(G)|)$ iterations. From now on we focus on proving Theorem 4.1. Let $\mathcal{C}$ be the current partition of $V(G)$ into acceptable clusters, and let H be the corresponding contracted graph. We denote $|E(H)|$ by m. From Observation 4.1, $m \ge \alpha^* k/3$.

Our algorithm consists of two steps. In the first step, we compute a random partition of $V(H)$ into $(h+1)$ disjoint subsets $X_1, \ldots, X_{h+1}$. At most one of these subsets may contain more than half the terminals, and we ignore this subset in the second step. For each one of the remaining subsets X_i, we either extract a subgraph G_i of G whose treewidth is at least r, or find a new partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters with $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$.

We start with the first step. Partition the vertices of $V(H)$ into $(h+1)$ disjoint subsets $X_1, \ldots, X_{h+1}$, as follows. Each vertex $v \in V(H)$ chooses an index $1 \leq j \leq h+1$ independently uniformly at random, and is then added to X_j. The following claim is very similar to a claim that was proved in [Chu12], and we include its proof in the Appendix for completeness, since we have changed the parameters.

Claim 4.1 *With probability at least $\frac{1}{2}$, for each $1 \leq j \leq h+1$, $|\mathrm{out}_H(X_j)| < \frac{16m}{h}$, while $|E_H(X_j)| \geq \frac{m}{8h^2}$.*

Given a partition $X_1, \ldots, X_{h+1}$, we can efficiently check whether the conditions of Claim 4.1 hold for it. If this is not the case, we compute a new random partition, until the conditions of Claim 4.1 hold. Clearly, after a polynomial number of iterations, w.h.p. we obtain a partition with desired properties.

From now on we assume that we are given a partition $X_1, \ldots, X_{h+1}$ of the vertices of H, for which the conditions of Claim 4.1 hold. For each $1 \leq j \leq h+1$, let X'_j be the subset of vertices of G, obtained by un-contracting the super-nodes in X_j, that is, $X'_j = \bigcup_{v_C \in X_j} C$. Notice that at most one subset X'_j may contain more than $|T|/2$ terminals. We assume without loss of generality that this subset is X'_{h+1}, and we ignore it from now on. Observe that for each $1 \leq j \leq h$, $|E_G(X'_j)| \geq |E_H(X_j)| > \frac{|\mathrm{out}_H(X_j)|}{128h} = \frac{|\mathrm{out}_G(X'_j)|}{128h}$. In our next step, we show that for each $1 \leq j \leq h$, we can either find a subset $S_j \subseteq X'_j$ of r'/Δ vertices, that are γ-well-linked in $G[X'_j]$, for $\gamma = \frac{6\Delta^2 r}{r'}$, or we can find a new partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters with $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$.

Claim 4.2 *For each $1 \leq j \leq h$, we can either find a subset $S_j \subseteq X'_j$ of r'/Δ vertices, such that S_j is γ-well-linked in $G[X'_j]$, for $\gamma = \frac{6\Delta^2 r}{r'}$, or we can compute a new partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters with $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$.*

We complete the proof of Claim 4.2 below, and show that the proof of Theorem 4.1 follows from it here. If we find a partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters with $\varphi(\mathcal{C}') \leq \varphi(\mathcal{C}) - 1$, then we return this partition. Otherwise, we let $G_j = G[X'_j]$, for $1 \leq j \leq h$. From Corollary 2.1, using the sets S_j, $\mathrm{tw}(G_j) \geq \frac{|S_j|\gamma}{3\Delta} - 1 = \frac{r'}{\Delta} \cdot \frac{6\Delta^2 r}{r'} \cdot \frac{1}{3\Delta} - 1 \geq r$. In order to complete the proof of Theorem 1.2, it is therefore enough to prove Claim 4.2.

Proof: Fix some $1 \leq j \leq h$. We maintain a partition $\mathcal{W}_j$ of the vertices of X'_j, where at the beginning, $\mathcal{W}_j = X'_j$. We then perform a number of iterations, as follows.

In every iteration, we select any cluster $C \in \mathcal{W}_j$ with $|\mathrm{out}(C)| \geq r'$. Let Γ be any subset of r' edges in $\mathrm{out}(C)$. We set up an instance of the sparsest cut problem, as follows. Subdivide every edge $e \in \Gamma$ by a vertex t_e, and let $T' = \{t_e \mid e \in \Gamma\}$. Consider the sub-graph of the resulting graph induced by $C \cup T'$, where the vertices of T' serve as terminals. We apply the algorithm $\mathcal{A}_{\mathrm{ARV}}$ to the resulting instance of the sparset cut problem. Let (A, B) be the resulting partition of C, and let (Γ_A, Γ_B) be the resulting partition of the edges of Γ, that is, $\Gamma_A = \mathrm{out}(A) \cap \Gamma$, and $\Gamma_B = \mathrm{out}(B) \cap \Gamma$. Assume without loss of generality that $|\mathrm{out}(A)| \leq |\mathrm{out}(B)|$. Two cases are possible. If $|E(A,B)| \geq \gamma \cdot \alpha_{\mathrm{ARV}}(r') \min\{|\Gamma_A|, |\Gamma_B|\}$, then we define the set S_j to contain the endpoints of the edges of Γ that belong to C. Since the degree of every vertex in G is at most Δ, $|S_j| \geq r'/\Delta$, and since the algorithm $\mathcal{A}_{\mathrm{ARV}}$ returned a cut whose sparsity is at least $\gamma \cdot \alpha_{\mathrm{ARV}}(r')$, it follows that set S_j is γ-well-linked in C, and hence in $G[X'_j]$.

Otherwise, $|E(A,B)| < \gamma \cdot \alpha_{\mathrm{ARV}}(r')|\Gamma_A|$. Every edge in $\mathrm{out}(A) \cap \mathrm{out}(C)$ is then charged $|E(A,B)|/|\mathrm{out}(A) \cap \mathrm{out}(C)| \leq r'\gamma\alpha_{\mathrm{ARV}}(r')/|\mathrm{out}(A) \cap \mathrm{out}(C)|$ for the edges of $E(A,B)$. Notice that the total charge is at least $|E(A,B)|$. The charge to every edge of $\mathrm{out}(A)\cap\mathrm{out}(C)$ can also be bounded by $\gamma\cdot\alpha_{\mathrm{ARV}}(r')$, since $|E(A,B)|<\gamma\cdot\alpha_{\mathrm{ARV}}(r')|\mathrm{out}(A)\cap\mathrm{out}(C)|$.

The algorithm terminates when we either find the desired subset S_j of vertices, or when every cluster $C\in\mathcal{W}_j$ has $|\mathrm{out}(C)|<r'$. In the former case, we return the set S_j as the output. In the latter case, we build a partition $\mathcal{C}'$ of $V(G)$ into acceptable clusters, with $\varphi(\mathcal{C}')\le\varphi(\mathcal{C})-1$. The collection $\mathcal{C}'$ of clusters contains all clusters $C\in\mathcal{C}$ with $C\cap X'_j=\emptyset$. Additionally, for each cluster $C\in\mathcal{W}_j$, we add all connected components of $G[C]$ to $\mathcal{C}'$. It is easy to see that $\mathcal{C}'$ is an acceptable clustering. It now only remains to show that $\varphi(\mathcal{C}')\le\varphi(\mathcal{C})-1$. Observe that:

$$\varphi(\mathcal{C}')\le\varphi(\mathcal{C})-|\mathrm{out}_G(X'_j)|-|E_G(X'_j)|+\sum_{C\in\mathcal{W}_j}|\mathrm{out}_G(C)|$$

We show that $\sum_{C\in\mathcal{W}_j}|\mathrm{out}_G(C)|<\left(1+\frac{1}{2^8h}\right)|\mathrm{out}_G(X'_j)|$. Since $|E_G(X'_j)|\ge\frac{|\mathrm{out}_G(X'_j)|}{128h}$, we will obtain that $\varphi(\mathcal{C}')\le\varphi(\mathcal{C})-1$.

In order to bound $\sum_{C\in\mathcal{W}_j}|\mathrm{out}(C)|$, we use the charging scheme defined above. Consider some cluster C that belonged to $\mathcal{W}_j$ at some point of the algorithm execution, and assume that we have replaced C with A and B, where $|\mathrm{out}(A)|\le|\mathrm{out}(B)|$, charging every edge in $\mathrm{out}(A)\cap\mathrm{out}(C)$ at most $r'\gamma\alpha_{\mathrm{ARV}}(r')/|\mathrm{out}(A)\cap\mathrm{out}(C)|$. Recall that $|E(A,B)|\le\gamma\alpha_{\mathrm{ARV}}(r')\cdot|\Gamma_A|\le\gamma\alpha_{\mathrm{ARV}}(r')\cdot|\mathrm{out}(A)\cap\mathrm{out}(C)|<0.1|\mathrm{out}(A)\cap\mathrm{out}(C)|$, since $\gamma=\frac{6\Delta^2r}{r'}$, and $r'=2^{20}r\Delta^2h\alpha_{\mathrm{ARV}}(k)$. Therefore, $|\mathrm{out}(A)|<2|\mathrm{out}(C)|/3$. The charge to the edges of $\mathrm{out}(A)\cap\mathrm{out}(C)$ can be bounded by $\frac{r'\gamma\alpha_{\mathrm{ARV}}(r')}{|\mathrm{out}(C)\cap\mathrm{out}(A)|}\le\frac{2r'\gamma\alpha_{\mathrm{ARV}}(r')}{|\mathrm{out}(A)|}$, since $|\mathrm{out}(A)|=|\mathrm{out}(C)\cap\mathrm{out}(A)|+|E(A,B)|<2|\mathrm{out}(C)\cap\mathrm{out}(A)|$.

Consider now some edge $e=(u,v)$. We bound the charge to the edge e via the vertex u. Let $C_1,C_2,\ldots,C_z$ be the clusters that belonged to $\mathcal{W}_j$ over the course of the algorithm, such that for each $1\le i\le z$, $u\in C_i$, $v\notin C_i$, and e was charged via u when cluster C_i was created. Then for each $2\le i\le z$, $|\mathrm{out}(C_i)|<2|\mathrm{out}(C_{i-1})|/3$ must hold, and edge e was charged at most $\frac{2r'\gamma\alpha_{\mathrm{ARV}}(r')}{|\mathrm{out}(C_i)|}$ for the creation of cluster C_i. Moreover, we are guaranteed that $|\mathrm{out}(C_{z-1})|>r'$. Therefore, the total charge to e via u for creating clusters $C_1,\ldots,C_{z-1}$ is bounded by:

$$\frac{2r'\gamma\alpha_{\mathrm{ARV}}(r')}{|\mathrm{out}(C_1)|}+\frac{2r'\gamma\alpha_{\mathrm{ARV}}(r')}{|\mathrm{out}(C_2)|}+\cdots+\frac{2r'\gamma\alpha_{\mathrm{ARV}}(r')}{|\mathrm{out}(C_{z-1})|}\le\frac{2r'\gamma\alpha_{\mathrm{ARV}}(r')}{r'}\left(1+\frac{2}{3}+\left(\frac{2}{3}\right)^2+\cdots\right)\le6\gamma\alpha_{\mathrm{ARV}}(r').$$

In the last iteration, e is charged at most $\gamma\alpha_{\mathrm{ARV}}(r')$ for creating cluster C_z. Therefore, the total direct charge to e via u is bounded by $7\gamma\alpha_{\mathrm{ARV}}(r')<\frac{1}{2^{11}h}$, since $\gamma=\frac{6\Delta^2r}{r'}$ and $r'=2^{20}r\Delta^2h\alpha_{\mathrm{ARV}}(k)$. The total direct charge to edge e, via both u and v, is then bounded by $\frac{1}{2^{10}h}$, and the total charge to any edge $e\in\mathrm{out}(X'_j)$, including the direct and the indirect charge (that happens when e is charged for some edge e', which is in turn charged for some other edges), is at most $\frac{1}{2^9h}$, since the indirect charge forms a geometrically decreasing sequence. We conclude that $\sum_{C\in\mathcal{W}_j}|\mathrm{out}(C)|<\left(1+\frac{1}{2^8h}\right)|\mathrm{out}_G(X'_j)|$, and $\varphi(\mathcal{C}')\le\varphi(\mathcal{C})-1$, as required. □

5 Applications

We now describe two applications of Theorem 1.1. Consider an integer-valued *parameter* P that associates a number $P(G)$ with each graph G. For instance, $P(G)$ could be the size of the smallest vertex cover of G, or it could be the maximum number of vertex-disjoint cycles in G. We say that P is *minor-closed* if $P(G) \geq P(H)$ for any minor H of G, that is, the value does not increase when deleting edges or contracting edges. A number of interesting parameters are minor-closed. Following [DH07a], we say that P has the *parameter-treewidth bound*, if there is some function $f : \mathbb{Z}_+ \to \mathbb{Z}_+$ such that $P(G) \leq k$ implies that $\text{tw}(G) \leq f(k)$. In other words, if the treewidth of G is large, then $P(G)$ must also be large. A minor-closed property P has the parameter-treewidth bound iff it has the bound on the family of grids. This is because an $r \times r$ grid has treewidth r, and the Grid-Minor Theorem shows that sufficiently large treewidth implies the existence of a large grid minor. This approach also has the advantage that grids are simple and concrete graphs to reason about. However, this approach for proving parameter treewidth bounds suffers from the (current) qunatitative weakness in the Grid-Minor theorem. For a given parameter P, one can of course focus on methods that are tailored to it. Alternatively, good results can be obtained in special classes of graphs such as planar graphs, and graphs that exclude a fixed graph as a minor, due to the linear relationship between the treewidth and the grid-minor size in such graphs. Theorem 1.1 allows for a generic method to change the dependence $f(k)$ from exponential to polynomial, under some mild restrictions. The following subsections describe these applications.

5.1 FPT Algorithms in General Graphs

Let P be any minor-closed graph parameter, and consider the decision problem associated with P: Given a graph G and an integer k, is $P(G) \leq k$? We say that parameter P is *fixed-parameter tractable*, iff there is an algorithm for this decision problem, whose running time is $h(k) \cdot n^{O(1)}$ where n is the size of G and h is a function that depends only on k. There is a vast literature on Fixed-Parameter Tractability (FPT), and we refer the reader to [DF07, Nie06, FG10, BDFM12].

Observe that for any minor-closed parameter P, and any fixed integer k, the family $\mathcal{F} = \{G \mid P(G) \leq k\}$ of graphs is a minor-closed family. That is, if $G \in \mathcal{F}$, and G' is a minor of G, then $G' \in \mathcal{F}$. Therefore, from the work of Robertson and Seymour on graph minors and the proof of Wagner's conjecture, there is a finite family $H_{\mathcal{F}}$ of graphs, such that $\mathcal{F}$ is precisely the set of all graphs that do not contain any graph from $H_{\mathcal{F}}$ as a minor. In particular, in order to test whether $P(G) \leq k$, we only need to check whether G contains a graph from $H_{\mathcal{F}}$ as a minor, and this can be done in time $O(n^3)$ (where we assume that k is a constant), using the work of Robertson and Seymour. However, even though the family $H_{\mathcal{F}}$ of graphs is known to exist, no explicit algorithms for constructing it are known. The family $H_{\mathcal{F}}$ of course depends on P, and moreover, even for a fixed property P, it varies with the parameter k. Therefore, the theory only guarantees the existence of a non-uniform FPT algorithm for every minor-closed parameter P. For this reason, it is natural to consider various restricted classes of minor-closed parameters. Motivated by the existence of sub-exponential time algorithms on planar and H-minor-free graphs, a substantial line of work has focused on *bidimensional* parameters — see Demaine et al. [DFHT05], and the survey in [DH07a]. Demaine and Hajiaghayi [DH07b] proved the following generic theorem on Fixed-Parameter Tractability of minor-closed bidimensional properties that satisfy some mild additional conditions.

Theorem 5.1 ([DH07b]) *Consider a minor-closed parameter P that is positive on some $g \times g$ grid, is at least the sum over the connected components of a disconnected graph, and can be computed in $h(w)n^{O(1)}$ time given a width-w tree decomposition of the graph. Then there is an algorithm that decides whether P is at most k on a graph with n vertices in $[2^{2^{O(g\sqrt{k})^5}} + h(2^{O(g\sqrt{k})^5})]n^{O(1)}$ time.*

The main advantage of the above theorem is its generality. However, its proof uses the Grid-Minor Theorem, and hence the running time of the algorithm is doubly exponential in the parameter k. Demaine and Hajiaghayi also observed, in the following theorem, that the running time can be reduced to singly-exponential in k if the Grid-Minor Theorem can be improved substantially.

Theorem 5.2 ([DH07b]) *Assume that every graph of treewidth greater than $\Theta(g^2 \log g)$ has a $g \times g$ grid as a minor. Then for every minor-closed parameter P satisfying the conditions of Theorem 5.1, there is an algorithm that decides whether $P(G) \leq k$ on any n-vertex graph G in $[2^{O(g^2 k \log(gk))} + h(O(g^2 k \log(gk)))]n^{O(1)}$ time.*

We show below that, via Theorem 1.1, we can bypass the need to improve the Grid-Minor Theorem.

Theorem 5.3 *Consider a minor-closed parameter P that is positive on all graphs with treewidth $\geq p$, is at least the sum over the connected components of a disconnected graph, and can be computed in $h(w)n^{O(1)}$ time given a width-w tree decomposition of the graph. Then there is an algorithm that decides whether P is at most k on a graph with n vertices in $[2^{\tilde{O}(p^2 k)} + h(\tilde{O}(p^2 k))]n^{O(1)}$ time.*

Proof: Let $k' = \tilde{\Theta}(p^2 k)$. If the given graph G has treewidth greater than k', then by Theorem 1.1 it can be partitioned into k node-disjoint subgraphs $G_1, \ldots, G_k$ where $\mathrm{tw}(G_i) \geq p$ for each i. Let G' be obtained by the union of these disconnected graphs (equivalently we remove the edges that do not participate in the graphs G_i from G). From the assumptions on P, $P(G_i) \geq 1$ for each i, and $P(G') \geq \sum_i P(G_i) \geq k$. Moreover, since P is minor-closed, $P(G) \geq P(G')$. Therefore, if $\mathrm{tw}(G) \geq k' = \tilde{\Omega}(p^2 k)$ then $P(G) \geq k$ must hold.

We use known algorithms, for instance [Ami10], that, given a graph G, either produce a tree decomposition of width at most $4w$ or certify that $\mathrm{tw}(G) > w$ in $2^{O(w)}n^{O(1)}$ time. Using such an algorithm we can detect in $2^{O(k')}n^{O(1)}$ time whether G has treewidth at least k', or find a tree decomposition of width at most $4k'$.

If $\mathrm{tw}(G) \geq k'$, then, as we have argued above, $P(G) \geq k$. We then terminate the algorithm with a positive answer. Otherwise, $\mathrm{tw}(G) < 4k'$ and we can use the promised algorithm that runs in time $h(4k') \cdot n^{O(1)}$ to decide whether $P(G) < k$ or not. The overall running time of the algorithm is easily seen to be the claimed bound. □

Remark 5.1 *In the proof of Theorem 5.3 the assumption on P being minor-closed is used only in arguing that $P(G') \geq P(G)$. Thus, it suffices to assume that the parameter P does not increase under edge deletions (in addition to the assumption on P over disconnected components of a graph).*

Note that the running time is singly-exponential in p and k. How does one prove an upper bound on p, the minimum treewidth guaranteed to ensure that the parameter value is positive? For some problems it may be easy to directly obtain a good bound on p. The following corollary shows that one can always use grid minors to obtain a bound on p. The run-time dependence on the grid size g is doubly exponential since we are using the Grid-Minor Theorem, but it is only singly-exponential in the parameter k. Thus, if g is considered to be a fixed constant, we obtain singly-exponential Fixed-Parameter Tractability algorithms in general graphs for all the problems that satisfy the conditions in Theorem 5.1.

Corollary 5.1 *Consider a minor-closed parameter P that is positive on some $g \times g$ grid, is at least the sum over the connected components of a disconnected graph, and can be computed in $h(w)n^{O(1)}$ time given a width-w tree decomposition of the graph. Then there is an algorithm that decides whether $P(G) \leq k$ on a graph G with n vertices in $[2^{\tilde{O}(k \cdot 2^{O(g^5)})} + h(\tilde{O}(2^{O(g^5)}k))]n^{O(1)}$ time.*

Proof: If P is minor-closed and positive on some $g \times g$ grid then it is positive on a graph G with treewidth $p > 20^{2g^5}$ via Theorem 1.3. Plugging this value of p into the bound from Theorem 5.3 gives the desired result. □

Remark 5.2 *The results in [RW12, KT10] can be used to obtain a singly exponential dependence on g, provided P can be shown to be positive on a graph that contains a* grid-like *minor of size g.*

5.2 Bounds for Erdos-Pósa theorems

Let $\mathcal{F}$ be any family of graphs. Following the notation in [Ree97], we say that the $\mathcal{F}$-packing number of G, denoted by $p_{\mathcal{F}}(G)$, is the maximum number of node-disjoint subgraphs of G, each of which is isomorphic to a member of $\mathcal{F}$. An $\mathcal{F}$-cover is a set X of vertices, such that $p_{\mathcal{F}}(G - X) = 0$; that is, removing X ensures that there is no subgraph isomorphic to a member of $\mathcal{F}$ in G. The $\mathcal{F}$-covering number of G, denoted by $c_{\mathcal{F}}(G)$ is the minimum cardinality of an $\mathcal{F}$-cover for G. It is clear that $p_{\mathcal{F}}(G) \leq c_{\mathcal{F}}(G)$ always holds. A family $\mathcal{F}$ is said to satisfy the Erdos-Pósa property if there is function $f : \mathbb{Z}_+ \to \mathbb{Z}_+$ such that $c_{\mathcal{F}}(G) \leq f(p_{\mathcal{F}}(G))$ for all graphs G. In other words, for every integer k, either G has k disjoint copies of graphs from $\mathcal{F}$, or there is a set of $f(k)$ nodes, whose removal from G ensures that no subgraph isomorphic to a graph from $\mathcal{F}$ remains in G. Erdos and Pósa [EP65] showed such a property when $\mathcal{F}$ is the family of cycles, with $f(k) = \Theta(k \log k)$.

There is an important connection between treewidth and Erdos-Pósa property as captured by the following two lemmas. The proofs closely follow the arguments in [Tho88, FST11] and appear in the Appendix.

Lemma 5.3 *Let $\mathcal{F}$ be any family of connected graphs, and let $h_{\mathcal{F}}$ be an integer-valued function, such that the following holds. For any integer k, and any graph G with $\mathrm{tw}(G) \geq h_{\mathcal{F}}(k)$, G contains k disjoint subgraphs $G_1, \ldots, G_k$, each of which is isomorphic to a member of $\mathcal{F}$. Then $\mathcal{F}$ has the Erdos-Pósa property with $f_{\mathcal{F}}(k) \leq k \cdot h_{\mathcal{F}}(k)$.*

Lemma 5.4 *Let $\mathcal{F}$ be any family of connected graphs, and let $h_{\mathcal{F}}$ be an integer-valued function, such that the following holds. For any integer k, and any graph G with $\mathrm{tw}(G) \geq h_{\mathcal{F}}(k)$, G contains k disjoint subgraphs $G_1, \ldots, G_k$, each of which is isomorphic to a member of $\mathcal{F}$. Moreover, suppose that $h_{\mathcal{F}}(\cdot)$ is superadditive[7] and satisfies the property that $h_{\mathcal{F}}(k+1) \leq \alpha h_{\mathcal{F}}(k)$ for all $k \geq 1$ where α is some universal constant. Then $\mathcal{F}$ has the Erdos-Pósa property with $f_{\mathcal{F}}(k) \leq \beta h_{\mathcal{F}}(k) \log(k+1)$ where β is a universal constant.*

One way to prove that $p_{\mathcal{F}}(G) \geq k$ whenever $\mathrm{tw}(G) \geq h_{\mathcal{F}}(k)$ is via the following proposition, that is based on the Grid-Minor Theorem. It is often implicitly used; see [Ree97].

Proposition 5.1 *Let $\mathcal{F}$ be any family of connected graphs, and assume that there is an integer g, such that any graph containing a $g \times g$ grid as a minor is guaranteed to contain a sub-graph isomorphic to a member of $\mathcal{F}$. Let $h(g')$ be the treewidth that guarantees the existence of a $g' \times g'$ grid minor in any graph. Then $f_{\mathcal{F}}(k) \leq O(k \cdot h(g\sqrt{k}))$. In particular $f_{\mathcal{F}}(k) \leq 2^{O(g^5 k^{2.5})}$.*

We improve the exponential dependence on k in the preceding proposition to near-linear. We state a more general theorem and then derive the improvement as a corollary.

Theorem 5.4 *Let $\mathcal{F}$ be any family of connected graphs, and assume that there is an integer r, such that any graph of treewidth at least r is guaranteed to contain a sub-graph isomorphic to a member of $\mathcal{F}$. Then $f_{\mathcal{F}}(k) \leq \tilde{O}(kr^2)$.*

[7] We say that an integer-valued function h is superadditive if for all $x, y \in \mathbb{Z}^+$, $h(x) + h(y) \leq h(x+y)$.

Proof: Let G be any graph with $\text{tw}(G) \geq kr^2 \text{poly}\log(kr)$. Theorem 1.1 guarantees that G can be partitioned into k node-disjoint subgraphs $G_1, \ldots, G_k$, such that for each i, $\text{tw}(G_i) \geq r$. From the assumption in the theorem statement, each G_i has a subgraph isomorphic to a member of $\mathcal{F}$. Therefore G contains k such subgraphs. We have thus established that if $\text{tw}(G) \geq \tilde{O}(kr^2)$, then $p_{\mathcal{F}}(G) \geq k$. We apply Lemma 5.4 to conclude that $f_{\mathcal{F}}(G) \leq \tilde{O}(kr^2)$. □

Combining the preceding theorem with the Grid-Minor Theorem gives the following easy corollary.

Corollary 5.2 *Let $\mathcal{F}$ be any family of connected graphs, such that for some integer g, any graph containing a $g \times g$ grid as a minor is guaranteed to contain a sub-graph isomorphic to a member of $\mathcal{F}$. Then $f_{\mathcal{F}}(k) \leq 2^{O(g^5)}\tilde{O}(k)$.*

Some concrete results: For a fixed graph H, let $\mathcal{F}(H)$ be the family of all graphs that contain H as a minor. Robertson and Seymour [RS86], as one of the applications of their Grid-Minor Theorem, showed that $\mathcal{F}(H)$ has the Erdos-Pósa property iff H is planar. The if direction can be deduced as follows. Every planar graph H is a minor of a $g \times g$ grid, where $g = O(|V(H)|^2)$. We can then use Proposition 5.1 to obtain a bound on $f_{\mathcal{F}(H)}$, which is super-exponential in k. However, by directly applying Corollary 5.2, we get the following improved near-linear dependence on k.

Theorem 5.5 *For any fixed planar graph H, the family $\mathcal{F}(H)$ of graphs has the Erdos-Pósa property with $f_{\mathcal{F}(H)}(k) = O(k \cdot \text{poly}\log(k))$.*

For any integer $m > 0$, let $\mathcal{F}_m$ be the family of all cycles whose length is 0 modulo m. Thomassen [Tho88] showed that $\mathcal{F}_m$ has the Erdos-Pósa property, with $f_{\mathcal{F}_m} = 2^{m^{O(k)}}$. We can use Corollary 5.2 to obtain a bound of $f_{\mathcal{F}_m} = \tilde{O}(k) \cdot 2^{\text{poly}(m)}$, using the fact that a graph containing a grid minor of size $2^{\text{poly}(m)}$ must contain a cycle of length 0 modulo m.

Acknowledgements: CC thanks Tasos Sidiropoulos for discussions on grid minors and bidemsionality, and Alina Ene for discussions on the decomposition result in [Chu12] and for pointing out [FST11]. This paper is partly inspired by the lovely article of Bruce Reed on treewdith and their applications [Ree97]. We thank Paul Wollan for pointers to recent work on the Grid-Minor theorem and Erdos-Pósa type theorems.

References

[Ami10] Eyal Amir. Approximation algorithms for treewidth. *Algorithmica*, 56(4):448–479, 2010.

[And10] Matthew Andrews. Approximation algorithms for the edge-disjoint paths problem via Raecke decompositions. In *Proceedings of IEEE FOCS*, pages 277–286, 2010.

[ARV09] Sanjeev Arora, Satish Rao, and Umesh V. Vazirani. Expander flows, geometric embeddings and graph partitioning. *J. ACM*, 56(2), 2009.

[BDFM12] Hans L Bodlaender, Rod Downey, Fedor V Fomin, and Dániel Marx, editors. *The Multivariate Algorithmic Revolution and Beyond.* Essays Dedicated to Michael R. Fellows on the Occasion of His 60th Birthday. Springer-Verlag New York Incorporated, May 2012.

[CE13] Chandra Chekuri and Alina Ene. Poly-logarithmic approximation for maximum node disjoint paths with constant congestion. In *Proc. of ACM-SIAM SODA*, 2013.

[Chu12] Julia Chuzhoy. Routing in undirected graphs with constant congestion. In *Proc. of ACM STOC*, pages 855–874, 2012.

[CKS04] Chandra Chekuri, Sanjeev Khanna, and F. Bruce Shepherd. The all-or-nothing multicommodity flow problem. In *Proc. of ACM STOC*, pages 156–165, 2004.

[CKS05] Chandra Chekuri, Sanjeev Khanna, and F. Bruce Shepherd. Multicommodity flow, well-linked terminals, and routing problems. In *Proc. of ACM STOC*, pages 183–192, 2005.

[CL12] Julia Chuzhoy and Shi Li. A polylogarithimic approximation algorithm for edge-disjoint paths with congestion 2. In *Proc. of IEEE FOCS*, 2012.

[DF07] R. Downey and M.R. Fellows. *Fixed-parameter Tractability.* Theory of computing systems. Springer, 2007.

[DFHT05] Erik D Demaine, Fedor V Fomin, MohammadTaghi Hajiaghayi, and Dimitrios M Thilikos. Subexponential parameterized algorithms on bounded-genus graphs and H-minor-free graphs. *J. ACM*, 52(6):866–893, 2005.

[DH07a] E.D. Demaine and M Hajiaghayi. The Bidimensionality Theory and Its Algorithmic Applications. *The Computer Journal*, 51(3):292–302, November 2007.

[DH07b] Erik D Demaine and MohammadTaghi Hajiaghayi. Quickly deciding minor-closed parameters in general graphs. *European Journal of Combinatorics*, 28(1):311–314, January 2007.

[DH08] Erik Demaine and Mohammadtaghi Hajiaghayi. Linearity of grid minors in treewidth with applications through bidimensionality. *Combinatorica*, 28:19–36, 2008.

[DHK09] Erik Demaine, MohammadTaghi Hajiaghayi, and Ken-ichi Kawarabayashi. Algorithmic graph minor theory: Improved grid minor bounds and wagners contraction. *Algorithmica*, 54:142–180, 2009.

[EP65] P. Erdos and L. Pósa. On independent circuits contained in a graph. *Canadian Journal of Mathematics. Journal Canadien de Mathématiques*, 17:347–352, 1965.

[FG10] J. Flum and M. Grohe. *Parameterized Complexity Theory.* Texts in Theoretical Computer Science. an Eatcs Series. Springer, 2010.

[FHL08] U. Feige, M.T. Hajiaghayi, and J.R. Lee. Improved approximation algorithms for minimum weight vertex separators. *SIAM Journal on Computing*, 38:629–657, 2008.

[FST11] Fedor V. Fomin, Saket Saurabh, and Dimitrios M. Thilikos. Strengthening Erdos-Pósa property for minor-closed graph classes. *Journal of Graph Theory*, 66(3):235–240, 2011.

[GM09] Martin Grohe and Dániel Marx. On tree width, bramble size, and expansion. *J. Comb. Theory, Ser. B*, 99(1):218–228, 2009.

[HS70] A. Hajnal and E. Szemerédi. Proof of a conjecture of P. Erdos. *Combinatorial Theory and its Applications*, pages 601–623, 1970.

[KK12] K. Kawarabayashi and Y. Kobayashi. Linear min-max relation between the treewidth of H-minor-free graphs and its largest grid minor. In *Proc. of STACS*, 2012.

[KRV09] Rohit Khandekar, Satish Rao, and Umesh Vazirani. Graph partitioning using single commodity flows. *J. ACM*, 56(4):19:1–19:15, July 2009.

[KS06] Petr Kolman and Christian Scheideler. Improved bounds for the unsplittable flow problem. *J. Algorithms*, 61(1):20–44, 2006.

[KT10] Stephan Kreutzer and Siamak Tazari. On brambles, grid-like minors, and parameterized intractability of monadic second-order logic. In *Proc. of ACM-SIAM SODA*, pages 354–364, 2010.

[LLR95] Nathan Linial, Eran London, and Yuri Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15:215–245, 1995.

[LR99] F. T. Leighton and S. Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *Journal of the ACM*, 46:787–832, 1999.

[Nie06] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms.* Oxford Lecture Series in Mathematics And Its Applications. Oxford University Press, 2006.

[OSVV08] L. Orecchia, L.J. Schulman, U.V. Vazirani, and N.K. Vishnoi. On partitioning graphs via single commodity flows. In *Proc. of ACM STOC*, pages 461–470. ACM, 2008.

[Räc02] Harald Räcke. Minimizing congestion in general networks. In *Proc. of IEEE FOCS*, pages 43–52, 2002.

[Ree97] Bruce Reed. *Surveys in Combinatorics*, chapter Treewidth and Tangles: A New Connectivity Measure and Some Applications. London Mathematical Society Lecture Note Series. Cambridge University Press, 1997.

[RS86] Neil Robertson and P D Seymour. Graph minors. V. Excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 41(1):92–114, August 1986.

[RST94] N Robertson, P Seymour, and R Thomas. Quickly Excluding a Planar Graph. *Journal of Combinatorial Theory, Series B*, 62(2):323–348, November 1994.

[RW12] Bruce A Reed and David R Wood. Polynomial treewidth forces a large grid-like-minor. *European Journal of Combinatorics*, 33(3):374–379, April 2012.

[RZ10] Satish Rao and Shuheng Zhou. Edge disjoint paths in moderately connected graphs. *SIAM J. Comput.*, 39(5):1856–1887, 2010.

[Sey12] Paul Seymour. The tree-width bound from excluding a planar minor. Third Workshop on Graphs and Matroids, August 2012.

[ST93] P. D. Seymour and Robin Thomas. Graph searching and a min-max theorem for tree-width. *J. Comb. Theory Ser. B*, 58(1):22–33, May 1993.

[Tho88] C Thomassen. On the presence of disjoint subgraphs of a specified type. *Journal of Graph Theory*, 12(1):101–111, 1988.

A Proofs Omitted from Section 2

A.1 Proof of Claim 2.1

We assume without loss of generality that $x_1 \geq x_2 \geq \cdots \geq x_n$, and process the integers in this order. When x_i is processed, we add i to A if $\sum_{j\in A} x_j \leq \sum_{j \in B} x_j$, and we add it to B otherwise. We claim that at the end of this process, $\sum_{i\in A} x_i, \sum_{i \in B} x_i \geq N/3$. Indeed, if $x_1 \geq N/3$, then 1 is added to A, and, since $x_1 \leq 2N/3$, it is easy to see that both subsets of integers sum up to at least $N/3$. Otherwise, $|\sum_{i\in A} x_i - \sum_{i\in B} x_i| \leq \max_i\{x_i\} \leq x_1 \leq N/3$.

A.2 Proof of Corollary 2.1

We build a bramble $\mathcal{B}$ of order at least $\frac{\alpha\cdot|T|}{3\Delta}$, as follows. Let X be any subset of fewer than $\frac{\alpha\cdot|T|}{3\Delta}$ vertices, and let $\mathcal{C}$ be the set of all connected components of $G \setminus X$. We claim that at least one connected component in $\mathcal{C}$ must contain more than $\frac{1}{2}|T|$ of the vertices of T.

Assume otherwise. Let $\mathcal{C} = \{C_1, \ldots, C_\ell\}$; let $C_{\ell+1} = X$, and let $\mathcal{C}' = \mathcal{C} \cup \{C_{\ell+1}\}$. Then, using Claim 2.1, we can find a partition $(\mathcal{A}, \mathcal{B})$ of the sets in $\mathcal{C}'$, with $\sum_{C\in\mathcal{A}} |C \cap T|, \sum_{C\in\mathcal{B}} |C \cap T| \geq |T|/3$. Let $A = \bigcup_{C\in\mathcal{A}} C$, $B = \bigcup_{C\in\mathcal{B}} C$. From the well-linkedness of the set T of vertices, $|E(A, B)| \geq \alpha|T|/3$ must hold. However, $E(A, B)$ only contains edges incident to the vertices of X, and their number is bounded by $|X| \cdot \Delta < \alpha|T|/3$, a contradiction.

We are now ready to define the bramble $\mathcal{B}$. For each subset X of fewer than $\frac{\alpha\cdot|T|}{3\Delta}$ vertices, we add the unique connected component C_X of $G \setminus X$, containing more than half the vertices of T to $\mathcal{B}$. It is easy to see that $\mathcal{B} = \{C_X \mid X \subseteq V(G); |X| < \frac{\alpha\cdot|T|}{3\Delta}\}$ is indeed a bramble. The order of $\mathcal{B}$ is at least $\frac{\alpha|T|}{3\Delta}$, since for each set S of fewer than $\frac{\alpha|T|}{3\Delta}$ vertices, there is a graph in $\mathcal{B}$ that does not contain any vertices of S - the graph C_S. Therefore, $\mathrm{BN}(G) \geq \frac{\alpha|T|}{3\Delta}$, and so $\mathrm{tw}(G) \geq \frac{\alpha|T|}{3\Delta}$-1.

A.3 Proof of Theorem 2.2

We use the $\alpha_{\text{ARV}}(k')$-approximation algorithm $\mathcal{A}_{\text{ARV}}$ for the sparsest cut problem; if a polynomial-time algorithm is not needed we can use an exact algorithm for the sparsest cut problem. Throughout the algorithm, we maintain a partition $\mathcal{W}$ of the input set S of vertices, where for each $R \in \mathcal{W}$, $|\mathrm{out}(R)| \leq |\mathrm{out}(S)|$. At the beginning, $\mathcal{W}$ consists of the subsets of S defined by the connected components of $G[S]$.

Let $R \in \mathcal{W}$ be any set in the current partition, and let $(G_R, \mathcal{T}'_R)$ be the instance of the sparsest cut problem corresponding to R, defined as follows. We start with graph G, and subdivide every edge $e \in \mathrm{out}_G(R)$ with a new vertex t_e, letting $\mathcal{T}'_R = \{t_e \mid e \in \mathrm{out}_G(R)\}$. Let G_R be the sub-graph of the resulting graph, induced by $R \cup \mathcal{T}'_R$. We then consider the instance of the sparsest cut on graph G_R, with the set $\mathcal{T}'_R$ of terminals. We say that a cut (A', B') in G_R is sparse, iff its sparsity is less than $\alpha \cdot \alpha_{\text{ARV}}(k')$. We apply the algorithm $\mathcal{A}_{\text{ARV}}$ to the instance $(G_R, \mathcal{T}'_R)$ of sparsest cut. If the algorithm returns a cut (A', B'), that is a sparse cut, then let $A = A' \setminus \mathcal{T}'_R$, and $B = B' \setminus \mathcal{T}'_R$. We remove R from $\mathcal{W}$, and add A and B to it instead. Let $T_A = \mathrm{out}(R) \cap \mathrm{out}(A)$, and $T_B = \mathrm{out}(R) \cap \mathrm{out}(B)$, and assume without loss of generality that $|T_A| \leq |T_B|$. Then $|E(A, B)| < \alpha \cdot \alpha_{\text{ARV}}(k')|T_A|$ must hold, and in particular, $|\mathrm{out}(A)| \leq |\mathrm{out}(B)| \leq |\mathrm{out}(R)| \leq |\mathrm{out}(S)|$. For accounting purposes, each edge in set T_A is charged $\alpha \cdot \alpha_{\text{ARV}}(k')$ for the edges in $E(A, B)$. Notice that the total charge to the edges in T_A is $\alpha \cdot \alpha_{\text{ARV}}(k')|T_A| \geq |E(A, B)|$. Notice also that since $|T_A| \leq |\mathrm{out}(R)|/2$ and $|E(A, B)| \leq \alpha \cdot \alpha_{\text{ARV}}(k')|T_A| \leq 0.1|T_A|$, we have that $|\mathrm{out}(A)| \leq 0.51|\mathrm{out}(R)|$.

The algorithm stops when for each set $R \in \mathcal{W}$, the procedure $\mathcal{A}_{\text{ARV}}$ returns a cut that is not sparse. We argue that this means that each set $R \in \mathcal{W}$ is α-good. Assume otherwise, and let $R \in \mathcal{W}$ be a set that is not α-good. Then the corresponding instance of the sparsest cut problem must have a cut of sparsity less than α. The algorithm $\mathcal{A}_{\text{ARV}}$ should then have returned a cut whose sparsity is less than $\alpha \cdot \alpha_{\text{ARV}}(k')$, that is a sparse cut.

Finally, we need to bound $\sum_{R\in\mathcal{W}} |\mathrm{out}(R)|$. We use the charging scheme defined above. Consider some iteration where we partition the set R into two subsets A and B, with $|T_A| \leq |T_B|$. Recall that each edge in T_A is charged $\alpha \cdot \alpha_{\text{ARV}}(k')$ in this iteration, while $|\mathrm{out}(A)| \leq 0.51|\mathrm{out}(R)|$ holds. Consider some edge $e = (u, v)$. Whenever e is charged via the vertex u, the size of the set $\mathrm{out}(R)$, where $u \in R \in \mathcal{W}$ goes down by the factor of at least 0.51. Therefore, e can be charged at most

$2 \log k'$ times via each of its endpoints. The total charge to e is then at most $4\alpha \cdot \alpha_{\text{ARV}}(k') \log k' < \frac{1}{2}$ (since $\alpha < \frac{1}{8\alpha_{\text{ARV}}(k') \cdot \log k'}$). This however only accounts for the *direct* charge. For example, some edge $e' \notin \text{out}(S)$, that was first charged to the edges in $\text{out}(S)$, can in turn be charged for some other edges. We call such charging *indirect*. If we sum up the indirect charge for every edge $e \in \text{out}(S)$, we obtain a geometric series, and so the total direct and indirect amount charged to every edge $e \in \text{out}(S)$ is at most $8\alpha \cdot \alpha_{\text{ARV}}(k') \log k'$. Therefore, $\sum_{R \in \mathcal{W}} |\text{out}(R)| \le k'(1 + 16\alpha \cdot \alpha_{\text{ARV}}(k') \log k')$ (we need to count each edge $e \in \left(\bigcup_{R \in \mathcal{W}} \text{out}(R)\right) \setminus \text{out}(S)$ twice: once for each its endpoint).

A.4 Proof of Theorem 2.3

Since G has treewidth k, we can efficiently find a set X of $\Omega(k)$ vertices of G with properties guaranteed by Lemma 2.2. Assume for simplicity that $|X|$ is even. We use the cut-matching game of Khandekar, Rao and Vazirani [KRV09], defined as follows.

We are given a set V of nodes, where $|V|$ is even, and two players, the cut player and the matching player. The goal of the cut player is to construct an edge-expander in as few iterations as possible, whereas the goal of the matching player is to prevent the construction of the edge-expander for as long as possible. The two players start with a graph $\mathcal{X}$ with node set V and an empty edge set. The game then proceeds in iterations, each of which adds a set of edges to $\mathcal{X}$. In iteration j, the cut player chooses a partition (Y_j, Z_j) of V such that $|Y_j| = |Z_j|$ and the matching player chooses a perfect matching M_j that matches the nodes of Y_j to the nodes of Z_j. The edges of M_j are then added to $\mathcal{X}$. Khandekar, Rao, and Vazirani [KRV09] showed that there is a strategy for the cut player that guarantees that after $O(\log^2 |V|)$ iterations the graph $\mathcal{X}$ is a $1/2$-edge-expander. Orecchia et al. [OSVV08] strengthened this result by showing that after $O(\log^2 |V|)$ iterations the graph $\mathcal{X}$ is a $\Omega(\log |V|)$-edge-expander. We use $\gamma_{\text{CMG}}(n)$ to denote the number of iterations of the cut-matching game required in the proof of the preceding theorem for $|V| = n$. Note that the resulting expander is regular with vertex degrees equal to $\gamma_{\text{CMG}}(n)$.

Using the cut-matching game we can embed an expander $H = (X, F)$ into G as follows. Each iteration j of the cut-matching game requires the matching player to find a matching M_j between a given partition of X into two equal-sized sets Y_j, Z_j. From Lemma 2.2, there exist a collection P_j of paths from Y_j to Z_j, that cause congestion at most $1/\alpha^*$ on the vertices of G; these paths naturally define the required matching M_j. The game terminates in $\gamma_{\text{CMG}}(|X|)$ steps. Consider the collection of paths $\mathcal{P} = \cup_j P_j$ and let G' be the subgraph of G induced by the union of the edges in these paths and let $H = (X, F)$ be the expander on X created by the union of the edges in $\cup_j M_j$. By the construction, for each j, any node v of G appears in at most $1/\alpha^*$ paths in P_j. Therefore, the maximum degree in G' is at most $2\gamma_{\text{CMG}}(|X|)/\alpha^* = O(\log^3 k)$ and moreover the node (and hence also edge) congestion caused by the edges of H in G is also upper bounded by the same quantity. We claim that $\text{tw}(G') = \Omega(k/\log^6 k)$. Since $H = (X, F)$ is an edge-expander, X is α-edge-well-linked in H for a fixed constant α. Since H is embedded in G' with congestion at most $2\gamma_{\text{CMG}}(|X|)/\alpha^*$, X is $\frac{\alpha \cdot \alpha^*}{2\gamma_{\text{CMG}}(|X|)}$-edge-well-linked in G'. Since the maximum degree in G' is at most $2\gamma_{\text{CMG}}(|X|)/\alpha^*$, we can apply Corollary 2.1 to see that $\text{tw}(G') = \Omega\left(\frac{|X|(\alpha^*)^2}{(\gamma_{\text{CMG}}(|X|))^2}\right) = \Omega(k/\log^6 k)$.

B Proofs Omitted from Section 3

B.1 Proof of Theorem 3.2

Given the graph H, we build a new graph H', as follows: Subdivide every edge $e \in E(H)$ with a new vertex v_e; add a new vertex t_e and connect it to v_e with an edge. The set of vertices of this new graph H' can be partitioned into three subsets: $V_1 = V(H)$; $V_2 = \{v_e \mid e \in E(H)\}$ and $\mathcal{T} = \{t_e \mid e \in E(H)\}$. Let $S = V_1 \cup V_2$. We perform a well-linked decomposition of S in graph H', by applying Theorem 2.2 to it, with parameter $\alpha = \frac{1}{160 \log m \cdot \alpha_{\text{ARV}}(m)}$. Let $\mathcal{W}$ be the resulting well-linked decomposition of S. We define a partition $\mathcal{W}'$ of the vertices of H as follows: for each $W \in \mathcal{W}$, we add $W' = W \cap V_1$ to $\mathcal{W}'$. Our final partition of $V(H)$ is $\mathcal{W}'$.

In order to bound $\sum_{W' \in \mathcal{W}'} |\text{out}(W')|$, observe that each edge $e \in \text{out}(W')$ contributes at least 1 to $\text{out}(W)$. In addition, $\text{out}(W)$ contains edges connecting some vertices in V_2 to the vertices in $\mathcal{T}$. Such edges do not belong to H and do not contribute to $|\text{out}(W')|$. The total number of such edges in graph H' is m. Therefore, $\sum_{W' \in \mathcal{W}'} |\text{out}(W')| \leq \sum_{W \in \mathcal{W}} |\text{out}(W)| - m \leq m(1 + 16\alpha\alpha_{\text{ARV}}(m)\log m) - m = m/10$.

Finally, we claim that for each $W' \in \mathcal{W}'$, $\Psi(H[W']) \geq \alpha$. Consider any partition (A, B) of W', and assume without loss of generality that $|E_H(A)| \leq |E_H(B)|$. It is enough to prove that $|E_H(A, B)| \geq \alpha |E_H(A)|$. We build a partition (A', B') of W, as follows. Set A' contains all vertices of $V_1 \cap A$, and all vertices v_e where both endpoints of e belong to A. (We assume that if both endpoints of an edge e belong to W, then so does the vertex v_e: otherwise, vertex v_e must belong to a separate cluster $W_e = \{v_e\}$ in $\mathcal{W}$, and by merging W and W_e we obtain a valid partition.) All other vertices of W belong to B. Clearly, $|E_H(A, B)| = |E_{H'}(A', B')|$. Moreover, for every edge $e \in E_H(A)$, we have $v_e \in A'$ and $t_e \in \text{out}_{H'}(A') \cap \text{out}_{H'}(W)$, and for every edge $e \in E_H(B)$, we have $v_e \in B'$ and $t_e \in \text{out}_{H'}(B') \cap \text{out}_{H'}(W)$. In particular, $|\text{out}_{H'}(A') \cap \text{out}_{H'}(W)|, |\text{out}_{H'}(B') \cap \text{out}_{H'}(W)| \geq |E_H(A)|$. Therefore, from the α-well-linkedness of W', $|E_H(A, B)| = |E_{H'}(A', B')| \geq \alpha \min\{|\text{out}_{H'}(A') \cap \text{out}_{H'}(W)|, |\text{out}_{H'}(A') \cap \text{out}_{H'}(W)|\} \geq \alpha |E_H(A)|$.

B.2 Proof of Theorem 3.4

Definition B.1 *We say that a graph $G = (V, E)$ is an α-expander, iff* $\min_{\substack{X \subseteq V: \\ |X| \leq |V|/2}} \{\frac{|E(X, \overline{X})|}{|X|}\} \geq \alpha$.

We will use the result of Leighton and Rao [LR99], who show that any multicommoditly flow instance in an expander graph that can be routed with no congestion, can also be routed on relatively short paths with a small edge-congestion. In order to use their result, we need to turn H' into a constant-degree expander[8]. We do so as follows.

We process the vertices of H' one-by-one. Let v be any such vertex, let d be its degree, and let $e_1, \ldots, e_d$ be the edges adjacent to v. We replace v with a degree-3 expander X_v on d vertices, whose expansion parameter is some constant $\alpha' < 1$. Each edge $e_1, \ldots, e_d$ now connects to a distinct vertex of X_v. Let H'' denote the graph obtained after each super-node of H' has been processed. Notice that the maximum vertex degree in H'' is bounded by 4. We next show that graph H'' is an α_0-expander, for $\alpha_0 = \alpha' \cdot \Psi(H')/12$.

Claim B.1 *Graph H'' is an α_0-expander, for $\alpha_0 = \alpha' \cdot \Psi(H')/12$.*

[8]An alternative to using constant degree expanders is to argue about short paths by appealing to large conductance and product multicommodity flows.

Proof: Assume otherwise, and let (A,B) be a violating cut, that is, $|E_{H''}(A,B)| < \alpha_0 \cdot \min\{|A|,|B|\}$. We use the cut (A,B) to define a partition (A',B'), of $V(H')$, and show that $|E_{H'}(A',B')| < \Psi(H') \cdot \min\{|E_{H'}(A')|,|E_{H'}(B')|\}$, contradicting the definition of conductance.

Partition (A',B') is defined as follows. For each vertex $v \in V(H')$, if at least half the vertices of X_v belong to A, then we add v to A'; otherwise we add v to B'.

We claim that $|E_{H'}(A',B')| \le |E_{H''}(A,B)|/\alpha'$. Indeed, consider any vertex $v \in V(H')$, and consider the partition (A_v,B_v) of the vertices of X_v defined by the partition (A,B): that is, $A_v = A \cap V(X_v)$, $B_v = B \cap V(X_v)$. Assume without loss of generality that $|A_v| \le |B_v|$. Then the contribution of the edges of X_v to $E_{H''}(A,B)$ is at least $\alpha' \cdot |A_v|$. After vertex v is processed, we add at most $|A_v|$ edges to the cut. Therefore,

$$|E_{H'}(A',B')| \le \frac{|E_{H''}(A,B)|}{\alpha'} \le \frac{\alpha_0}{\alpha'} \cdot \min\{|A|,|B|\} = \frac{\Psi(H')}{12} \min\{|A|,|B|\}$$

Assume without loss of generality that $\sum_{v \in A'} d_{H'}(v) \le \sum_{v \in B'} d_{H'}(v)$, so $|E_{H'}(A')| \le |E_{H'}(B')|$. Consider the set A of vertices of H'', and let $A_1 \subseteq A$ be the subset of vertices, that belong to expanders X_v, where $|V(X_v) \cap A| \le |V(X_v) \cap B|$. Notice that from the expansion properties of graphs X_v, $|E_{H''}(A,B)| \ge \alpha'|A_1|$, and so $|A_1| \le \frac{|E_{H''}(A,B)|}{\alpha'} \le \frac{\alpha_0}{\alpha'}|A| \le \frac{|A|}{8}$. As every vertex in $A \setminus A_1$ contributes at least 1 to the final summation $\sum_{v \in A'} d_{H'}(v)$, we get that $\sum_{v \in A'} d_{H'}(v) \ge \frac{7}{8}|A|$, while, as observed above, $|E_{H'}(A',B')| \le \frac{\Psi(H')}{12}|A| \le 0.1|A|$. Therefore, $|E_{H'}(A')| = \left(\sum_{v \in A'} d_{H'}(v) - |E_{H'}(A',B')|\right)/2 \ge 0.2|A|$. We conclude that

$$|E_{H'}(A',B')| \le \frac{\Psi(H')}{12}|A| < 0.2\Psi(H')|A| \le \Psi(H')|E_{H'}(A')|,$$

contradicting the definition of conductance. □

The following theorem easily follows from the results of Leighton and Rao [LR99], and its proof can be found in [Chu12] (see also [KS06]).

Theorem B.1 *Let G be any n-vertex α-expander with maximum vertex degree $d_{\max}$, and let M be any partial matching over the vertices of G. Then there is an efficient randomized algorithm that finds, for every pair $(u,v) \in M$, a path $P_{u,v}$ of length $O(d_{\max} \log n/\alpha)$ connecting u to v, such that the set $\mathcal{P} = \{P_{u,v} \mid (u,v) \in M\}$ of paths causes edge-congestion $O(\log^3 n/\alpha)$ in G. The algorithm succeeds with high probability.*

Let X' be any degree-3 α'-expander over $r'' = \Theta(r\Delta^2 \log^8 k)$ vertices, where α' is some constant. Our next step is to embed X' into H'', by using short paths. Specifically, we select any collection $\Gamma' = \{v_1,\ldots,v_{r''}\}$ of vertices in H'', and define an arbitrary $1:1$ matching between the vertices of X' and the vertices of Γ' (we identify these vertices and refer to vertices of X' as Γ' from now on). Notice that $r'' < r' \le |E(H')|$, so there are at least r'' vertices in H''.

Next, for every edge $e = (u,v) \in E(X')$, we find a path P_e connecting u to v in H''. This path will serve as the embedding of the edge e. In order to find the embeddings of the edges of X', we partition $E(X')$ into 5 disjoint matchings $M_1,\ldots,M_5$, using the fact that the maximum vertex degree in X' is bounded by 3. We then use Theorem B.1 to route the matchings $M_1,\ldots,M_5$ in graph H''. Let $\mathcal{P} = \{P_e \mid e \in E(X')\}$ be the resulting set of paths. Then, from Theorem B.1, the length of every path in $\mathcal{P}$ is bounded by $\ell = O(\log^3 n) = O(\log^3 k)$, and these paths cause congestion at most $\eta = O(\log^3 n)/\Psi(H') = O(\log^5 n) = O(\log^5 k)$ in H''.

We are now ready to define the set S of vertices in graph H'. We add to S every vertex v, such that at least one vertex of X_v participates in the paths in $\mathcal{P}$. Since $|\mathcal{P}| = r''$, and every path in $\mathcal{P}$ contains at most ℓ vertices, $|S| \leq r'' \cdot \ell \leq O(r\Delta^2 \log^{11} k) \leq r'$, as required. Finally, consider the graph G', obtained from $H'[S]$, by un-contracting the super-nodes in S. It now only remains to prove that $\mathrm{tw}(G') \geq r$. In order to do so, we define a subset Γ of at least r''/Δ vertices of G', and prove that these vertices are α^{**}-well-linked in G', for a suitably large α^{**}.

Consider the sub-graph H^* of H'', induced by the set S' of vertices, participating in the paths $\mathcal{P}$, and the graph G'. For every vertex $v_C \in S$, graph G' contains the sub-graph $G'[C]$, and graph H^* contains the expander X_{v_C}. So we can obtain H^* from G', by replacing every cluster $G'[C]$ with the expander X_{v_C}. Let E_0 be the set of edges in H^*, connecting vertices (x, y) that belong to distinct expanders $X_{v_C}, X_{v_{C'}}$. Then for each edge $e \in E_0$, there is a corresponding edge $e' \in E(G')$, connecting some vertex $x' \in C$ to some vertex $y \in C'$. We do not distinguish between the edges e, e', and will think about them as the same edge.

We now define a subset Γ of vertices of G', by mapping every vertex $x \in \Gamma'$ to its corresponding vertex in G'. The mapping is defined as follows. Consider some vertex $x \in \Gamma'$, and assume that $x \in X_{v_C}$. Let e be the unique edge of E_0 incident on x, and consider the same edge e in graph G'. Let x' be the endpoint of e that belongs to the cluster C. We then define $f(x) = x'$. Let $\Gamma = \{f(x) \mid x \in \Gamma'\}$. Since the degree of every vertex in G' is at most Δ, $|\Gamma| \geq r''/\Delta$. From Corollary 2.1, in order to prove that $\mathrm{tw}(G') \geq r$, it is enough to show that the set Γ of vertices is α^{**}-well-linked, for $\alpha^{**} \geq \frac{6\Delta^2 r}{r''}$.

Consider any partition (A, B) of $V(G')$, and denote $\Gamma_A = \Gamma \cap A, \Gamma_B = \Gamma \cap B$. Assume without loss of generality that $|\Gamma_A| \leq |\Gamma_B|$. We need to prove that $|E(A, B)| \geq \alpha^{**} \cdot |\Gamma_A|$.

Let $\Gamma'_A, \Gamma'_B \subseteq \Gamma'$ be subsets of vertices of Γ', corresponding to the partition (Γ_A, Γ_B) of Γ (recall that for each vertex $x' \in \Gamma'$, there can be up to Δ vertices in Γ mapped to x'. In this case we only add one of them to Γ_A or Γ_B). Since graph X' is an α'-expander, there are $|\Gamma_A|$ paths connecting the vertices in Γ'_A to the vertices of Γ'_B in X', and they cause a congestion of $1/\alpha' = O(1)$ in X'. Let $\mathcal{P}_1$ denote this set of paths. Using the embedding of X' into H'', we can build a collection $\mathcal{P}_2$ of $|\Gamma_A|$ paths in graph H'', where every path connects a distinct vertex in Γ'_A to a distinct vertex in Γ'_B, and the total congestion due to these paths is $O(\eta)$. Moreover, from the definition of H^*, all paths in $\mathcal{P}_2$ are contained in H^*. We now use the paths in $\mathcal{P}_2$ to define a flow F connecting the vertices of Γ'_A to the vertices of Γ'_B in G'. The flow F follows the paths in $\mathcal{P}_2$ on the edges that belong to set E_0. In order to complete the description of this flow, we need to show how to route it inside the clusters C for $v_C \in S$. For each such cluster C, the set $\mathcal{P}_2$ of paths defines a set D_C of $O(\eta)$-restricted demands over the edges of $\mathrm{out}(C)$. Since the cluster C is α_G-good, we can route these demands inside C with congestion at most $O(\eta \log k/\alpha_G)$. Overall, we obtain a flow F of value $|\Gamma_A|$, connecting the vertices in Γ_A to the vertices of Γ_B, with congestion $O(\eta \log k/\alpha_G)$. It follows that $|E(A, B)| \geq \frac{|\Gamma_A|\alpha_G}{\eta \log k} = \Omega\left(\frac{|\Gamma_A|}{\log^{7.5} k}\right)$. We conclude that set Γ is $\Omega(1/\log^{7.5} k)$-well-linked in G'. From Corollary 2.1, it follows that $\mathrm{tw}(G') \geq \Omega\left(\frac{|\Gamma|}{\Delta \log^{7.5} k}\right) = \Omega\left(\frac{r''}{\Delta^2 \log^{7.5} k}\right) \geq r$.

C Proof of Claim 4.1

Fix some $1 \leq j \leq h+1$. Let $\mathcal{E}_1(j)$ be the bad event that $\sum_{v \in X_j} d_H(v) \geq \frac{16m}{h}$. In order to bound the probability of $\mathcal{E}_1(j)$, we define, for each vertex $v \in V(H)$, a random variable x_v, whose value is $\frac{d_H(v)}{r'}$ if $v \in X_j$ and 0 otherwise. Notice that $x_v \in [0, 1]$, and the random variables $\{x_v\}_{v \in V(H)}$ are pairwise independent. Let $B = \sum_{v \in V(H)} x_v$. Then the expectation of B, $\mu_1 = \sum_{v \in V(H)} \frac{d_H(v)}{(h+1)r'} = \frac{2m}{(h+1)r'}$. Using the standard Chernoff bound,

$$\mathbf{Pr}\left[\mathcal{E}_1(j)\right] < \mathbf{Pr}\left[B > 8\mu_1\right] \le 2^{-8\mu_1} = 2^{-\frac{16m}{(h+1)r'}} < \frac{1}{6h}$$

since $m \ge \alpha^* k/3$ and $k > hr' \log h/\alpha^*$.

Let $\mathcal{E}_2(j)$ be the bad event that $|E_H(X_j)| < \frac{m}{8h^2}$. We next prove that $\mathbf{Pr}\left[\mathcal{E}_2(j)\right] \le \frac{1}{k}$. We say that two edges $e, e' \in E(H)$ are *independent* iff they do not share any endpoints. Our first step is to compute a partition $U_1, \ldots, U_z$ of the set $E(H)$ of edges, where $z \le 2r'$, such that for each $1 \le i \le z$, $|U_i| \ge \frac{m}{4r'}$, and all edges in set U_i are mutually independent. In order to compute such a partition, we construct an auxiliary graph Z, whose vertex set is $\{v_e \mid e \in E(H)\}$, and there is an edge $(v_e, v_{e'})$ iff e and e' are not independent. Since the maximum vertex degree in G' is at most r', the maximum vertex degree in Z is bounded by $2r' - 2$. Using the Hajnal-Szemerédi Theorem [HS70], we can find a partition $V_1, \ldots, V_z$ of the vertices of Z into $z \le 2r'$ subsets, where each subset V_i is an independent set, and $|V_i| \ge \frac{|V(Z)|}{z} - 1 \ge \frac{m}{4r'}$. The partition $V_1, \ldots, V_z$ of the vertices of Z gives the desired partition $U_1, \ldots, U_z$ of the edges of H. For each $1 \le i \le z$, we say that the bad event $\mathcal{E}_2^i(j)$ happens iff $|U_i \cap E(X_j)| < \frac{|U_i|}{2(h+1)^2}$. Notice that if $\mathcal{E}_2(j)$ happens, then event $\mathcal{E}_2^i(j)$ must happen for some $1 \le i \le z$. Fix some $1 \le i \le z$. The expectation of $|U_i \cap E(X_j)|$ is $\mu_2 = \frac{|U_i|}{(h+1)^2}$. Since all edges in U_i are independent, we can use a standard Chernoff bound to bound the probability of $\mathcal{E}_2^i(j)$, as follows:

$$\mathbf{Pr}\left[\mathcal{E}_2^i(j)\right] = \mathbf{Pr}\left[|U_i \cap E(X_j)| < \mu_2/2\right] \le e^{-\mu_2/8} = e^{-\frac{|U_i|}{8(h+1)^2}}$$

Since $|U_i| \ge \frac{m}{4r'}$, $m \ge k\alpha^*/3$, $k \ge 2^{10} h^2 r' \log k/\alpha^*$, this is bounded by $\frac{1}{k^2}$. We conclude that $\mathbf{Pr}\left[\mathcal{E}_2^i(j)\right] \le \frac{1}{k^2}$, and by using the union bound over all $1 \le i \le z$, $\mathbf{Pr}\left[\mathcal{E}_2(j)\right] \le \frac{1}{k}$.

Using the union bound over all $1 \le j \le h+1$, with probability at least $\frac{1}{2}$, none of the events $\mathcal{E}_1(j), \mathcal{E}_2(j)$ for $1 \le j \le h+1$ happen, and so for each $1 \le j \le h+1$, $|\mathrm{out}_H(X_j)| \le \sum_{v \in X_j} d_H(v) < \frac{16m}{h}$, and $|E_{G'}(X_j)| \ge \frac{m}{8h^2}$ must hold.

D Proof of Lemma 5.3

The proof closely follows the proof of Proposition 2.1 in [Tho88]. Let G be any graph, and k any integer. If $\mathrm{tw}(G) \ge h_{\mathcal{F}}(k)$, then from our assumption, $p_{\mathcal{F}}(G) \ge k$, and there is nothing to prove. So from now on, it is enough to prove the following. If G is any graph with $\mathrm{tw}(G) = w < h_{\mathcal{F}}(k)$, then either $p_{\mathcal{F}}(G) \ge k$, or $c_{\mathcal{F}}(G) \le k(w+1)$. We prove this statement by induction on k. The statement is clearly true for $k = 0$. Consider now some general value of k.

Let T be the tree-decomposition of width w of G. For each vertex $v \in V(T)$, we denote by X_v the corresponding subset of vertices of G, and recall that $|X_v| \le w+1$. For each sub-tree $T' \subseteq T$, we denote by $S_{T'} = \bigcup_{v \in V(T')} X_v$, and by $G_{T'}$ the sub-graph of G induced by $S_{T'}$.

For every vertex $v \in V(T)$, we consider all pairs (T_1, T_2) of sub-trees of T, where $T_1 \cup T_2 = T$, and $T_1 \cap T_2 = \{v\}$. Among all such triples (v, T_1, T_2), we are interested only in those where G_{T_1} contains a sub-graph isomorphic to a graph in $\mathcal{F}$, and among all triples satisfying this condition, we select the one minimizing $|V(T_1)|$. Let H be any sub-graph of G_{T_1} isomorphic to a member of $\mathcal{F}$. Then $V(H) \cap X_v \neq \emptyset$, since otherwise we can obtain a new triple (v', T_1', T_2') satisfying all the above properties, with $T_1' \subsetneq T_1$, contradicting the minimality of T_1.

Assume now that $c_{\mathcal{F}}(G) > k(w+1)$. In other words, for any subset A of $k(w+1)$ vertices in graph G, $G \setminus A$ contains a sub-graph isomorphic to a graph in $\mathcal{F}$. In particular, if we let $G' = G \setminus X_v$, then for any subset A of $(k-1)(w+1)$ vertices in this graph, $G' \setminus A$ contains a sub-graph isomorphic to a graph in $\mathcal{F}$. By the induction hypothesis, this means that G' contains $(k-1)$ disjoint sub-graphs $G_1, \ldots, G_{k-1}$, each of which is isomorphic to a graph in $\mathcal{F}$. Moreover, each such graph G_i must be disjoint from G_{T_1} (since, as observed above, any copy of a graph in $\mathcal{F}$, which is contained in G_{T_1}, must intersect X_v). Let H be any copy of a graph in $\mathcal{F}$ that is contained in G_{T_1}. Then $G_1, \ldots, G_{k-1}, H$ are k disjoint subgraphs of G, each of which is isomorphic to a graph in $\mathcal{F}$, as required.

E Proof of Lemma 5.4

The proof is inspired by the argument in [FST11].

We prove that for each $k \geq 1$, and for each graph G, if $p_{\mathcal{F}}(G) \leq k$, then $c_{\mathcal{F}}(G) \leq \beta h_{\mathcal{F}}(k) \log(k+1)$. The proof is by induction on k. The claim is trivially true for $k = 0$. We prove the statement for $k \geq 1$ assuming that it holds for all values up to $k-1$.

Let G be such that $p_{\mathcal{F}}(G) = k$ and let $T = (V_T, E_T)$ be a tree decomposition of smallest width for G. We observe that the width of T is strictly less than $h_{\mathcal{F}}(k+1)$ for otherwise $p_{\mathcal{F}}(G) > k$, contradicting our assumption. For $t \in V_T$ let $X_t \subseteq V$ be the bag of vertices at t. We root T at any vertex and use the following notation. For $t \in V_T$, T_t is the subtree of T rooted at t. $G_t = G[S_t]$ where $S_t = \cup_{t' \in T_t} X_{t'}$. $G_t^- = G_t \setminus X_t$ is the graph obtained by removing the nodes in X_t from G_t.

The induction step is based on the following claim.

Claim E.1 *There exists a separator $S \subseteq V$ such that $|S| \leq 2h_{\mathcal{F}}(k+1)$ and for each connected subgraph G' in $G \setminus S$, $p_{\mathcal{F}}(G') \leq \lfloor 2k/3 \rfloor$.*

Proof: Call a node $t \in V_T$ large if G_t^- contains a connected subgraph G' such that $p_{\mathcal{F}}(G') > \lfloor 2k/3 \rfloor$. Otherwise t is small. If the root r is small then X_r is the desired separator and we are done. Otherwise, let t be the deepest large node in T. There is a single connected component G' in $G \setminus X_t$ such that $p_{\mathcal{F}}(G') > \lfloor 2k/3 \rfloor$, otherwise it would imply that $p_{\mathcal{F}}(G) > k$. Moreover G' is contained in $G_{t'}$ for some child t' of t. We claim that $S = X_t \cup X_{t'}$ is the desired separator. If $G \setminus S$ still contains a connected component G' such that $p_{\mathcal{F}}(G') > \lfloor 2k/3 \rfloor$ then it is contained in $G_{t'}^-$ contradicting the choice that t is the deepest large node in T. □

Let S be the separator from the claim. We have $|S| \leq 2h_{\mathcal{F}}(k+1)$ which by the assumption on the function $h(\cdot)$ is at most $2\alpha h_{\mathcal{F}}(k)$. Let $G_1, G_2, \ldots, G_\ell$ be the connected components of $G \setminus S$ and let $k_i = p_{\mathcal{F}}(G_i)$. For $1 \leq i \leq \ell$, $k_i \leq \lfloor 2k/3 \rfloor < k$, and moreover $\sum_{i=1}^{\ell} k_i \leq k$. Let S_i be a minimum cardinality $\mathcal{F}$-cover for G_i. From the induction hypothesis $|S_i| \leq \beta h_{\mathcal{F}}(k_i) \log(k_i+1)$. Since $\mathcal{F}$ is a family of connected graphs, we note that $S' = S \cup (\bigcup_i S_i)$ is a $\mathcal{F}$-cover in G whose cardinality can be bounded as $2\alpha h_{\mathcal{F}}(k) + \sum_i \beta h_{\mathcal{F}}(k_i) \log(k_i+1)$. If $k = 1$ then $k_i = 0$ for all i and therefore $|S'| \leq 2\alpha h_{\mathcal{F}}(k)$ which proves the induction hypothesis for $k = 1$ if $\beta \geq 2\alpha$. We will now assume $k \geq 2$ in which case for each i, $k_i + 1 \leq \lfloor 2k/3 \rfloor + 1 \leq 3(k+1)/4$. The cardinality of S' is upper bounded as:

$$\begin{aligned}
2\alpha h_{\mathcal{F}}(k) + \sum_i \beta h_{\mathcal{F}}(k_i) \log(k_i+1) &\leq 2\alpha h_{\mathcal{F}}(k) + \sum_i \beta h_{\mathcal{F}}(k_i) \log(\frac{3}{4}(k+1)) \\
&\leq 2\alpha h_{\mathcal{F}}(k) + \beta \log(\frac{3}{4}(k+1)) \sum_i h_{\mathcal{F}}(k_i) \\
&\leq 2\alpha h_{\mathcal{F}}(k) + \beta \log(\frac{3}{4}(k+1)) \cdot h_{\mathcal{F}}(k) \qquad \text{(since } h_{\mathcal{F}}(\cdot) \text{ is superadditive)} \\
&\leq 2\alpha h_{\mathcal{F}}(k) - \beta h_{\mathcal{F}}(k) \log \frac{4}{3} + \beta h_{\mathcal{F}}(k) \log(k+1) \\
&\leq \beta h_{\mathcal{F}}(k) \log(k+1),
\end{aligned}$$

where the last inequality follows by choosing β sufficiently large compared to α. This establishes the induction step for k and finishes the proof.