A Polynomial Excluded-Minor Approximation of Treedepth

Ken-ichi Kawarabayashi*
National Institute of Informatics
k_keniti@nii.ac.jp

Benjamin Rossman†
University of Toronto
ben.rossman@utoronto.ca

February 18, 2017

Abstract

Treedepth (also known as vertex ranking number [28], ordered chromatic number [16], and minimum elimination tree height [21]) is a well-studied graph parameter in the family of "width measures" that includes treewidth and pathwidth. For each $k \in \mathbb{N}$, the class of graphs with treedepth $\leq k$ is closed under minors and is therefore characterized by a finite set of minimal obstructions (a.k.a. excluded minors) by the Graph Minor Theorem [23]. Unfortunately, the excluded-minor characterization of treedepth has little practical use, as the number of minimal obstructions grows enormously fast as a function of k (at least doubly exponentially [13]) and a complete classification is unknown even for small values of k (see [4, 5, 15]).

In contrast to this state of affairs, we show that just *three* non-minimal obstructions suffice for a polynomial approximation of treedepth. Our main theorem states that there is an absolute constant c such that every graph with treedepth $\geq k^c$ has one or more of the following minors:

- the $k \times k$ grid,
- the complete binary tree of height k,
- the path of order 2^k.

Since each of these graphs has treedepth $\geq k$, this result gives a polynomial approximation of treedepth. This is analogous to the recent Polynomial Grid-Minor Theorem of Chekuri and Chuzhoy [11], which shows that treewidth is polynomially approximated by the largest $k \times k$ grid minor in a graph.

Our main theorem was motivated by a specific application in complexity theory and logic. By combining our polynomial excluded-minor approximation of treedepth with lower bounds on the AC^0 formula size of detecting grids [17], paths [25] and trees [27], we obtain an $n^{\mathsf{poly}(\mathrm{treedepth}(G))}$ lower bound on the AC^0 formula size of the colored G-subgraph isomorphism problem for all graphs G. This result, in turn, has a surprising corollary in finite model theory (described in [26]): a polynomial-rank homomorphism preservation theorem on finite structures.

1 Introduction

The *treedepth* of a graph G, denoted $\mathbf{td}(G)$, is the minimum height of a rooted forest F, on the same set of vertices, such that for every edge $\{u, v\}$ in G, vertices u and v have an ancestor-descendant relationship in F (i.e. lie on a common branch). This graph parameter is also known in the

*Supported by JST ERATO Kawarabayashi Large Graph Project.
†Supported by NSERC. This work was partially carried out at the National Institute of Informatics.

literature as vertex ranking number [28], ordered chromatic number [16], and minimum elimination tree height [21]. It was systematically investigated under the name treedepth by Ossona de Mendes and Nešetřil [19] and has also been widely studied in parameterized complexity theory (see recent papers [12, 14, 22]).

Treedepth is considered to be a "width measure", a informal designation for a family of decomposition-based graph parameters that includes *treewidth* (**tw**), *pathwidth* (**pw**), and a host of others. Roughly speaking, treedepth is a measure of how "star-like" a graph is, in a similar sense that treewidth and pathwidth are measures of how "tree-like" and "path-like". These three parameters are related by inequalities

$$\mathbf{tw}(G) + 1 \leq \mathbf{pw}(G) + 1 \leq \mathbf{td}(G) \leq (\mathbf{tw}(G) + 1) \cdot \log |V(G)|. \tag{1}$$

Treedepth is also related to the order of the longest path in G, denoted $\mathbf{lp}(G)$, by

$$\log(\mathbf{lp}(G) + 1) \leq \mathbf{td}(G) \leq \mathbf{lp}(G). \tag{2}$$

(Throughout this paper $\log(\cdot)$ denotes the base-2 logarithm. See Ch. 6 of [20] for proofs of (1) and (2).)

Graph parameters **td**, **tw**, **pw** and **lp** all share the property of being monotone under the graph-minor relation (a.k.a. minor-monotone). Recall that a graph H is a *minor* of G, denoted $H \preceq G$, if H can be obtained from G by a sequence of vertex/edge deletions and edge contractions. A graph parameter $f : \{\text{graphs}\} \to \mathbb{N}$ is *minor-monotone* if $f(H) \leq f(G)$ for all $H \preceq G$. This is equivalent to the class $\{G : f(G) \leq k\}$ being minor-closed for every $k \in \mathbb{N}$, where a class $\mathcal{C}$ is *minor-closed* if $G \in \mathcal{C} \Rightarrow H \in \mathcal{C}$ for all $H \preceq G$. By the Robertson-Seymour Graph Minor Theorem [23], every minor-closed class $\mathcal{C}$ is characterized by a finite set $\mathcal{F}$ of *obstructions* (a.k.a. *excluded minors*) with the property that

$$G \in \mathcal{C} \iff (\forall F \in \mathcal{F})(F \not\preceq G)$$

for all graphs G; moreover, $\mathcal{F}$ is unique (up to isomorphism of its elements) subject to *minimality* (i.e. $F \not\preceq F'$ for all distinct $F, F' \in \mathcal{F}$). It follows that every minor-monotone graph parameter f is characterized by the sequence $(\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_k, \dots)$ of finite minimal obstruction sets $\mathcal{F}_k$ for the class $\{G : f(G) \leq k\}$.

Understanding the exact minimal obstruction sets $\mathcal{F}_k$ (computing, classifying, counting, etc.) for specific minor-monotone graph parameters is an active topic of research in graph theory (see [1, 10]). When it comes to treedepth, minimal obstructions have been studied by two sets of authors [4, 5, 13, 15]. However, a complete classification of minimal obstructions for treedepth $\leq k$ remains elusive even for small values of k (less than 10). Moreover, Dvořák et al [13] showed that the number of minimal obstructions grows enormously fast (at least doubly exponentially) as a function of k [13]. The situation is similar for other "width measures" like treewidth. This severely limits the usefulness of minimal obstructions in applications, such as parameterized algorithms on bounded tree-depth graphs.

On the other hand, there are applications where having a reasonable *approximation* of a parameter like treedepth or treewidth serves a good enough purpose. (We describe one such application in Section 6, which was the original motivation for the results of this paper.) The question arises whether one (or a bounded number of) "uniform" families of non-minimal obstructions suffice for a polynomial approximation of a given minor-monotone graph parameter. A recent breakthrough of Chekuri and Chuzhoy [11] gave precisely such a result for treewidth (resolving a longstanding conjecture in graph minor theory).

Theorem 1 (Polynomial Grid-Minor Theorem for Treewidth [11])**.** *There is an absolute constant c such that every graph with treewidth $\geq k^c$ has a $k \times k$ grid minor.*

Since the $k \times k$ grid has treewidth k, Theorem 1 establishes the treewidth of a graph is polynomially related to the size of its largest grid minor. (Prior to Theorem 1, treewidth only known to be exponential in the size of the largest grid minor.) In this paper, we establish an analogous "polynomial excluded-minor approximation" of treedepth in terms of *three* uniform families of obstructions: grids, complete binary trees, and paths.

Theorem 2 (Polynomial Grid/Tree/Path-Minor Theorem for Treedepth)**.** *There is an absolute constant c such that every graph with treedepth $\geq k^c$ has one or more of the following minors:*

- *the $k \times k$ grid,*
- *the complete binary tree of height k,*
- *the path of order 2^k.*

Since each of the above graphs has treedepth $\geq k$, this result gives the claimed polynomial approximation of $\mathbf{td}(G)$. (The only previous result along similar lines is the observation that the longest path in a graph G gives an exponential approximation of $\mathbf{td}(G)$ via inequality (2).) Theorem 2 is actually obtained as a corollary of the following:

Theorem 3 (Main Theorem)**.** *There is an absolute constant C such that every graph G with treedepth $\geq Ck^5 \log^2 k$ satisfies one or more of the following conditions:*

- *G has treewidth $\geq k$,*
- *G has the complete binary tree of height k as a minor,*
- *G contains a path of order 2^k.*

Notice that Theorem 2 follows immediately from Theorems 1 and 3. Our proof of Theorem 3 is entirely contained in this paper (i.e. it does not rely on outside lemmas or the heavy machinery involved in Theorem 1).

The rest of this paper is organized as follows. In Section 2 we state some basic definitions. In Section 3 we prove some lemmas on tree decompositions. In Section 4 we prove some additional lemmas on rooted trees (essentially proving Theorem 3 in the case where G is a tree). In Section 5 we present the proof of Theorem 3. In Section 6 we describe a surprising application of Theorem 3 in circuit complexity and logic, which was the motivation for this paper. Finally, we conclude with some observations and open problems in Section 7.

2 Preliminaries

$\mathbb{N} = \{0, 1, 2, \dots\}$. For $n \in \mathbb{N}$, $[n] = \{1, \dots, n\}$. $\log(\cdot)$ is the base-2 logarithm.

All graphs in this paper are finite simple graphs. Formally, a *graph* is a pair $G = (V(G), E(G))$ where $E(G) \subseteq \binom{V(G)}{2}$. A *tree* is a connected acyclic graph. A tree is *subcubic* if it has maximum degree at most 3. Examples of subcubic trees include paths and binary trees.

Definition 4 (Tree Decompositions, Treewidth, Pathwidth)**.**

- A *tree decomposition* of a graph G is a pair $(T, \mathcal{W})$ where T is a tree and $\mathcal{W} = \{W_t\}_{t \in V(T)}$ is a family of sets $W_t \subseteq V(G)$ such that
  - $\bigcup_{t \in V(T)} W_t = V(G)$, and every edge of G has both ends in some W_t,
  - if $t, t', t'' \in V(T)$ and t' lies on the path in T between t and t'', then $W_t \cap W_{t''} \subseteq W_{t'}$.
- The *width* of a tree decomposition $(T, \mathcal{W})$ is defined as $\max_{t \in V(T)} |W_t| - 1$.
- The *treewidth* of G, denoted $\mathbf{tw}(G)$, is the minimum width of a tree decomposition for G.
- The *pathwidth* of G, denoted $\mathbf{pw}(G)$, is the minimum width of a tree decomposition $(T, \mathcal{W})$ for G such that T is a path.

Definition 5 (Rooted Trees)**.** A *rooted tree* is a tree T with a designated root vertex. The *height* of T is the maximum number of vertices on a root-to-leaf path. We use the following notation:

- $\vec{E}(T)$ is the set of ordered pairs xy such that x is a child of y in T. (We write xy instead of (x, y) and think of this pair as a directed edge.)
- $<_T$ is the partial order on $V(T)$ defined by $x <_T y$ iff x is a proper descendent of y; we write $x \leq_T y$ iff $x <_T y$ or $x = y$; for $W \subseteq V(T)$, we write $W \leq_T x$ iff $w \leq_T x$ for all $w \in W$.
- The *closure* of T, denoted $\mathrm{Clos}(T)$, is the graph with vertex set $V(T)$ and edge set $\{\{x, y\} : x <_T y \text{ or } y <_T x\}$. (In other words, two vertices are joined by an edge in $\mathrm{Clos}(T)$ iff they lie on a common branch in T.)

The following lemma characterizes partial orders of the form $<_T$ (proof is left as an exercise).

Lemma 6. *For any finite partially ordered set* $(X, <)$*, statements* (i) *and* (ii) *are equivalent:*

(i) *There exists a rooted tree T with $V(T) = X$ and $< \,=\, <_T$ (i.e. $x < y \Leftrightarrow x <_T y$ for all $x, y \in X$).*

(ii) $(X, <)$ *has a unique maximal element and, for each $x \in X$, the set $\{y \in X : x < y\}$ is totally ordered by $<$.*

Definition 7 (Treedepth)**.** The *treedepth* of a connected graph G, denoted $\mathbf{td}(G)$, is the minimum height of a rooted tree T such that $G \subseteq \mathrm{Clos}(T)$. The *treedepth* of a disconnected graph is the maximum treedepth of its connected components.[1]

Definition 8 (Graph Minors and Minor-Monotonicity)**.**

- A graph F is a *minor* of G, denoted $F \preceq G$, if F is isomorphic to a graph that can be obtained from G by a sequence of edge deletions and edge contractions.
- A graph parameter $f : \{\text{graphs}\} \to \mathbb{N}$ is *minor-monotone* if $f(F) \leq f(G)$ for all graph $F \preceq G$.

Width measures $\mathbf{tw}(\cdot)$, $\mathbf{pw}(\cdot)$ and $\mathbf{td}(\cdot)$ are easily seen to be minor-monotone. The parameter $\mathbf{lp}(\cdot)$, the order of the longest path, is minor-monotone as well.

[1] Treedepth of general graphs G can be defined as the minimum height of a rooted forest F such that $G \subseteq \mathrm{Clos}(F)$ (where $\mathrm{Clos}(F)$ is defined similarly as $\mathrm{Clos}(T)$). Elsewhere in the literature, rooted forests F satisfying $G \subseteq \mathrm{Clos}(F)$ are called *treedepth decompositions* of G. We avoid this terminology in this paper, to avoid confusion with the more common notion of tree decompositions.

3 Lemmas on Tree Decompositions

Our first lemma bounds the treedepth of a graph G in terms of the width of one of its tree decomposition $(T, \mathcal{W})$ and the treedepth of T. Although the proof is fairly simple, we are not aware if this observation has appeared before in the literature.

Lemma 9. *If $(T, \mathcal{W})$ is a width-w tree decomposition of a graph G, then*

$$\mathbf{td}(G) \leq (w+1) \cdot \mathbf{td}(T).$$

Proof. Suppose $(T, \mathcal{W})$ be a width-w tree decomposition of a graph G. We will construct a rooted R of height at most $(w+1) \cdot \mathbf{td}(T)$ such that $G \subseteq \mathrm{Clos}(R)$. (The construction is illustrated in Figure 1. The tree decomposition $(T, \mathcal{W})$ in that example happens to be a path.)

By definition of treedepth, there exists a rooted tree S such that $T \subseteq \mathrm{Clos}(S)$ and $\mathbf{td}(T) = \mathrm{height}(S)$. Without loss of generality, we may assume that $V(S) = V(T)$ (by contracting any vertices of $V(S) \setminus V(T)$).

Recall that $\mathcal{W}$ is a family $\{W_t\}_{t \in V(T)}$ where $W_t \subseteq V(G)$. For each $t \in V(T)$, define the set $U_t \subseteq W_t$ by $U_t := W_t \setminus \bigcup_{u\,:\,t <_S u} W_u$. Let $\mathcal{U} := \{U_t\}_{t \in V(T)}$ and note that $\mathcal{U}$ forms a partition of $V(G)$ (where some of sets U_t may be empty).

For each $t \in V(T)$, fix an arbitrary linear order $<_t$ on U_t. Define partial order $<^\star$ on $V(G)$ by

$$x <^\star y \overset{\text{def}}{\iff} \Big(\bigwedge_{t \in V(T)} x, y \in U_t \text{ and } x <_t y \Big) \vee \Big(\bigwedge_{t,u \in V(T)\,:\,t <_S u} x \in U_t \text{ and } y \in U_u \Big).$$

That is, we have $x <^\star y$ iff either x, y belong to the same set U_t and $x <_t y$, or x, y belong to distinct U_t, U_u respectively where $t <_S u$.

It is easy to see that $<^\star$ is equivalent to $<_R$ for a unique rooted R with $V(R) = V(G)$. (This follows from the observation that $<^\star$ is a partial order on $V(G)$; it has a unique maximal element (namely, the $<_t$-maximal element of U_t $(= W_t)$ where $t = \mathrm{root}(S)$); and for every $x \in V(G)$, the set $\{y : x <^* y\}$ is totally ordered by $<^*$.) Note that

$$\mathbf{td}(G) \leq \mathrm{height}(R) \leq \max_{t \in V(T)} |W_t| \cdot \mathrm{height}(S) = (w+1) \cdot \mathbf{td}(T).$$

To complete the proof, it remains to establish that $G \subseteq \mathrm{Clos}(R)$. Consider an edge $\{x, y\} \in E(G)$. By definition of $(T, \mathcal{W})$ being a tree decomposition of G, the set $\{t \in V(T) : \{x, y\} \subseteq W_t\}$ is non-empty; let p be any $<_S$-maximal element in this set. Consider the set $\{u \in V(T) : p \leq_S u \text{ and } \{x, y\} \cap W_t \neq \emptyset\}$; let q be the unique $<_S$-maximal element in this set. There are now two cases to consider:

- Assume $p = q$. Then $x, y \in U_p$. W.l.o.g., $x <_p y$. Then we have $x <_R y$ and hence $\{x, y\} \in E(\mathrm{Clos}(R))$.
- Assume $p \neq q$. Then $|\{x, y\} \cap W_q| = 1$. W.l.o.g., $\{x, y\} \cap W_q = \{y\}$. Then we have $x \in U_p$ and $y \in U_q$ and $p <_S q$. It follows that $x <_R y$ and hence $\{x, y\} \in E(\mathrm{Clos}(R))$.

Since $\{x, y\} \in E(\mathrm{Clos}(R))$ in both cases, we conclude that $G \subseteq \mathrm{Clos}(R)$. $\square$

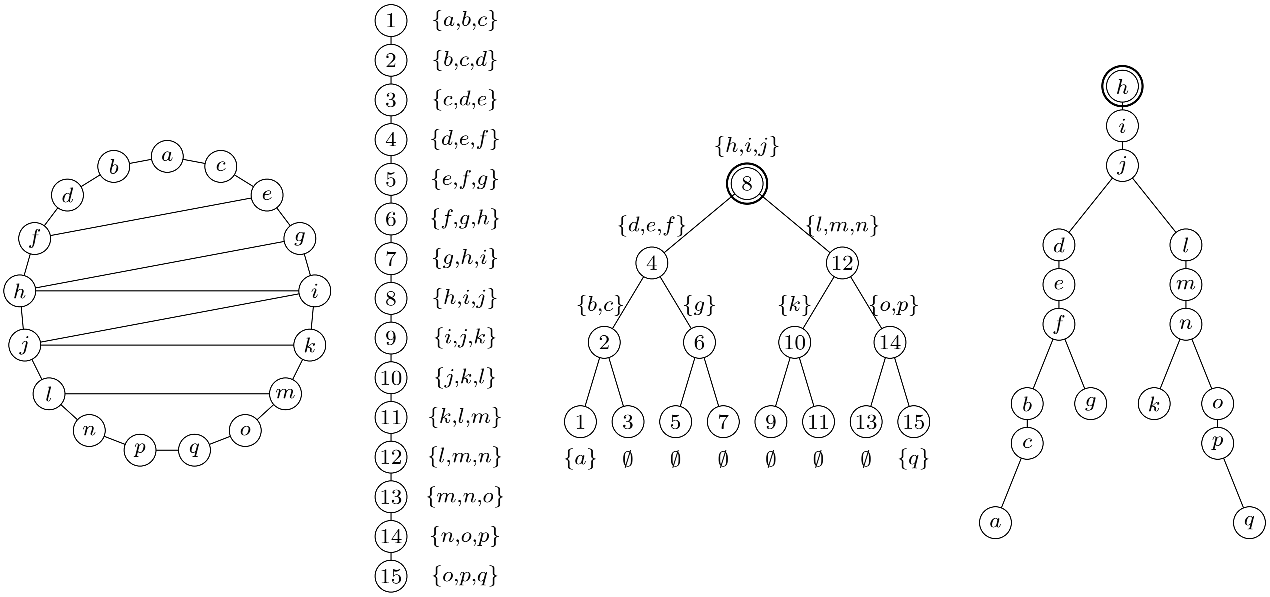

Figure 1: From left to right: G, $(T,\mathcal{W})$, $(S,\mathcal{U})$, R

We next introduce a normal form for tree decompositions of connected graphs, which witnesses tight upper bounds for both treewidth and treedepth (as shown in Lemmas 13 and 14).

Definition 10 (Greedy Rooted Tree Decomposition)**.**

- A *greedy rooted tree decomposition* of a connected graph G is a rooted tree T with the following properties:

  1. $V(T) = V(G)$,
  2. $G \subseteq \mathrm{Clos}(T)$,
  3. for every child-parent pair $xy \in \vec{E}(T)$, there exists $w \leq_T x$ such that $\{w, y\} \in E(G)$.

  (Given (1) and (2), note that condition (3) is equivalent to the following: for every $x \in V(T)$, the induced subgraph of G on $\{w : w \leq_T x\}$ is connected.)

- For each $x \in V(G)$, we define the set $\mathrm{Bag}_{T,G}(x) \subseteq V(G)$ by

$$\mathrm{Bag}_{T,G}(x) := \{x\} \cup \{y : \text{there exists } w \text{ such that } w \leq_T x <_T y \text{ and } \{w, y\} \in E(G)\}.$$

- The *width* of T with respect to G is defined by $\max_{x \in V(G)} |\mathrm{Bag}_{T,G}(x)| - 1$.

Remark 11. Our notion of greedy rooted tree decompositions is defined only for connected graphs for simplicity. However, Definition 10 extends naturally to general graphs by considering rooted forests instead of rooted trees.

The same notion appears at least once before in the literature: in [12] where it is called *good treedepth decomposition*. An even "greedier" class of tree decompositions was introduced in [14] under the name *minimal rooted trees*. Every minimal rooted tree for a connected graph G (in the sense of [14]) is a greedy rooted tree decomposition of G (in our sense), but not vice-versa. For our application, it is crucial that we work with greedy rooted tree decompositions (in particular, Lemma 14 is false with respect to minimal rooted trees).

The next lemma establishes that greedy rooted tree decompositions are, in fact, tree decompositions (in the sense of Definition 4).

Lemma 12. *If T is a greedy rooted tree decomposition of a connected graph G, then T together with $\{\mathrm{Bag}_{T,G}(x)\}_{x \in V(G)}$ is a tree decomposition of G.*

Proof. Straightforward from definitions. □

The next two lemmas show that height-optimal (resp. width-optimal) greedy rooted tree decomposition witness the treedepth (resp. treewidth) of connected graphs.

Lemma 13. *Every connected graph G has a greedy rooted tree decomposition of height $\mathbf{td}(G)$.*

Proof. By definition of treedepth, there exists a rooted tree T of height $\mathbf{td}(G)$ such that $G \subseteq \mathrm{Clos}(T)$. W.l.o.g., we may assume that $V(T) = V(G)$ (by contracting any vertices in $V(T) \setminus V(G)$). Thus, T satisfies conditions (i) and (ii) of Definition 10. If T satisfies condition (iii), then we are done. So we assume that T violates condition (iii).

Consider any child-parent pair $xy \in \vec{E}(T)$ witnessing the violation of condition (iii), that is, y is the parent of x in T and there is no edge in G between y and any element of $\{w : w \leq_T x\}$. Note that y cannot be the root of T (since it would then follow from $G \subseteq \mathrm{Clos}(T)$ that G is disconnected). Let z be the parent of y in T. Let T' be the rooted tree obtained from T by removing the edge $\{x, y\}$ and adding the edge $\{x, z\}$. Note the following:

1. T' satisfies conditions (i) and (ii) (that is, $V(T') = V(G)$ and $G \subseteq \mathrm{Clos}(T')$).

2. $\mathrm{height}(T') \leq \mathrm{height}(T)$.

3. $\mathrm{width}(T', G) \leq \mathrm{width}(T, G)$.

4. We have $\phi(T') < \phi(T)$ where $\phi : \{\text{rooted trees}\} \to \mathbb{N}$ is the potential function $\phi(S) := \sum_{v \in V(S)} \mathrm{depth}_S(v)$ where $\mathrm{depth}_S(v)$ is the distance between v and the root of S. This is clear, since $V(T') = V(T)$ and

$$\mathrm{depth}_{T'}(v) = \begin{cases} \mathrm{depth}_T(v) - 1 & \text{if } v \leq_T x, \\ \mathrm{depth}_T(v) & \text{otherwise.} \end{cases}$$

It follows from observations (1)–(4) that finitely many operations $T \mapsto T'$ transform T into a greedy rooted tree decomposition of G of at most the same height and width. In particular, the height is at most $\mathbf{td}(T)$, which proves the lemma. □

Lemma 14. *Every connected graph G has a greedy rooted tree decomposition of width $\mathbf{tw}(G)$.*

Proof. By definition of treewidth, there exists a tree decomposition $(T, \mathcal{W})$ of G of width $\mathbf{tw}(G)$. W.l.o.g., we may assume that W_t is nonempty for all $t \in V(T)$. We now make T into a rooted tree by arbitrary fixing a choice of $\mathrm{root}(T) \in V(T)$. Without increasing width, we can massage[2] the tree decomposition $(T, \mathcal{W})$ in order that

[2]If $|W_{\mathrm{root}(T)}| = \{v_1, \dots, v_k\}$ where $k \geq 2$, then replace $\mathrm{root}(T)$ with a path on fresh vertices $t_1, \dots, t_k$ where $W_{t_i} = \{v_1, \dots, v_i\}$; if $|W_s \setminus W_t| = k \geq 2$ for some $st \in \vec{E}(T)$, then replace the edge $\{s, t\}$ in T by a path of length $k - 1$ with appropriate sets W_u at the newly created vertices u.

- $|W_{\mathrm{root}(T)}| = 1$,
- $|W_s \setminus W_t| = 1$ for all every child-parent pair $st \in \vec{E}(T)$.

We may now identify $V(T)$ with $V(G)$ by identifying root(T) with the unique element of $W_{\mathrm{root}(T)}$ and identifying each non-root t with the unique element of $W_t \setminus W_u$ where u is the parent of t.

Thus identified, the rooted tree T now satisfies conditions (i) and (ii), that is, $V(T) = V(G)$ and $G \subseteq \mathrm{Clos}(T)$. Moreover, we have $\mathrm{width}(T, G) \leq \mathrm{width}(T, \mathcal{W})$. Finally, we repeat the same operation $T \mapsto T'$ as in the proof of Lemma 13 until T satisfies condition (iii) with respect to G. Since this operation does not increase width, we obtain a greedy rooted tree decomposition of G of width at most $\mathbf{tw}(G)$, which proves the lemma. $\square$

4 Lemmas on Rooted Trees

In this section we some prove results about rooted trees. In particular, we prove the special case of our main theorem: every tree with treedepth k contains a path of length $2^{\Omega(\sqrt{k})}$ or a complete binary tree of height $\Omega(\sqrt{k})$ as a minor. We begin with some basic definitions.

Definition 15 (The rooted minor relation $\preceq_{\text{rooted}}$)**.** For rooted trees S and T, we say that S is a *rooted minor* of T, denoted $S \preceq_{\text{rooted}} T$, if S is isomorphic to a rooted tree obtained from T by deleting non-root leaves and contracting edges.

Definition 16 (Rooted Trees P_k and B_h)**.**

- For $k \geq 1$, let P_k denote the path of order k rooted at one of its endpoints.
- For $h \geq 1$, let B_h denote the rooted complete binary tree of height h (with $2^h - 1$ vertices).

Note that P_1 and B_1 are both the rooted tree of size 1 (i.e. an isolated root).

The next definition gives useful notation describing the structure of rooted trees.

Definition 17 (Rooted Tree-Building Operations $*$ and $\langle\rangle$)**.**

- For rooted trees S and T, let $S * T$ denote the rooted tree formed by taking the disjoint union of S and T and identifying the two roots. (For example, $P_2 * \cdots * P_2$ is a star rooted at its central vertex.) This operation is associative and commutative with identity element P_1. For a sequence of rooted trees $T_1, \ldots, T_m$ ($m \in \mathbb{N}$), we adopt the convention that $T_1 * \cdots * T_m = P_1$ if $m = 0$.
- For a rooted tree T, let $\langle T \rangle$ denote the rooted tree obtained from T by creating a new root ρ and drawing an edge between ρ and the old root of T.
- For a sequence of rooted trees $T_1, \ldots, T_m$, let

$$\langle T_1, \ldots, T_m \rangle := \langle T_1 * \langle T_2 * \ldots \langle T_{m-1} * \langle T_m \rangle \rangle \ldots \rangle \rangle.$$

  That is, $\langle T_1, \ldots, T_m \rangle$ is the rooted tree obtained by identifying the root of T_i with the ith vertex from the root on the rooted path P_{m+1}.

These operations on rooted trees are illustrated in Figure 2, below.

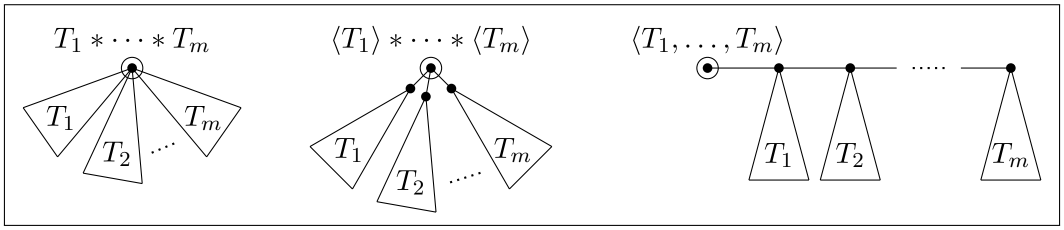

Figure 2

Note that for $k, h \geq 1$,

$$P_k = \langle \underbrace{P_1, \ldots, P_1}_{k-1 \text{ times}} \rangle \quad \text{and} \quad B_h = \langle B_{h-1} \rangle * \langle B_{h-1} \rangle$$

where as special cases we define $\langle\rangle = \langle B_0 \rangle := P_1$ (i.e. the isolated root). (We only speak of B_0 in the context of $\langle B_{h-1} \rangle$ where $h-1$, as the base case in inductions; here B_0 is treated as the empty tree with 0 vertices.)

Lemma 18. *Every rooted tree T has a unique decomposition the form $\langle T_1 \rangle * \cdots * \langle T_m \rangle$ for some $m \in \mathbb{N}$ and rooted trees $T_1, \ldots, T_m$ (unique up to ordering).*

Proof. Straightforward from definitions. Here m is the degree of root(T) and $T_1, \ldots, T_m$ are the subtrees rooted at the children of root(T) (see Figure 1). □

The next two lemmas characterize the rooted minor relation in terms of the decomposition given by Lemma 18.

Lemma 19. *For rooted trees S and T, we have $\langle S \rangle \preceq_{\text{rooted}} \langle T \rangle$ if, and only if, $S \preceq_{\text{rooted}} T$ or $\langle S \rangle \preceq_{\text{rooted}} T$.*

Proof. Assume $\langle S \rangle \preceq_{\text{rooted}} \langle T \rangle$ and consider the edge in $\langle T \rangle$ between root($\langle T \rangle$) and root(T). If this edge is not contracted in the minor isomorphic to S, then $S \preceq_{\text{rooted}} T$. If this edge is contracted, then $\langle S \rangle \preceq_{\text{rooted}} T$.

The other direction is clear. If $S \preceq_{\text{rooted}} T$, then clearly $\langle S \rangle \preceq_{\text{rooted}} \langle T \rangle$ (by the same sequence of deletions and contractions). If $\langle S \rangle \preceq_{\text{rooted}} T$, then $\langle S \rangle \preceq_{\text{rooted}} \langle T \rangle$ (by $T \preceq_{\text{rooted}} \langle T \rangle$ and transitivity of $\preceq_{\text{rooted}}$). □

Lemma 20. *Suppose $S = \langle S_1 \rangle * \cdots * \langle S_l \rangle$ and $T = \langle T_1 \rangle * \cdots * \langle T_m \rangle$. Then $S \preceq_{\text{rooted}} T$ if, and only if, there exists a one-to-one function $j : [l] \hookrightarrow [m]$ such that $\langle S_i \rangle \preceq_{\text{rooted}} \langle T_{j(i)} \rangle$ for all $i \in [l]$.*

Proof. Straightforward from definitions. □

4.1 Rooted trees that exclude B_h minors

The next lemmas characterize rooted trees T that omit complete binary trees as rooted minors.

Lemma 21. *For every rooted tree T and $h \geq 1$, the following statements are equivalent:*

(1) $\langle B_h \rangle \npreceq_{\text{rooted}} T$ *and* $B_h \npreceq_{\text{rooted}} T$,

(2) *$T = P_1$ or there exist rooted trees S and T' such that*

$$T = S * \langle T' \rangle, \quad \langle B_{h-1} \rangle \not\preceq_{\text{rooted}} S, \quad \langle B_h \rangle \not\preceq_{\text{rooted}} T', \quad B_h \not\preceq_{\text{rooted}} T',$$

(3) *there exist $m \in \mathbb{N}$ and rooted trees $S_1, \dots, S_m$ such that $T = S_1 * \langle S_2, \dots, S_m \rangle$ and $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} S_i$ for all $i \in [m]$.*

Proof. By Lemma 18, $T = \langle T_1 \rangle * \cdots * \langle T_l \rangle$ for some $l \in \mathbb{N}$ and $T_1, \dots, T_l$. We first prove the equivalence of (1) and (2).

Case 1: $l = 0$. Then $T = P_1$ (a single isolated root) and clearly both (1) and (2) hold.

Case 2: $l \geq 1$ and $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} \langle T_i \rangle$ for all $i \in [l]$. We have $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} T$ by Lemmas 19 and 20. Therefore, (1) holds (by transitivity of $\preceq_{\text{rooted}}$ since $\langle B_{h-1} \rangle \preceq_{\text{rooted}} \langle B_h \rangle$ and $\langle B_{h-1} \rangle \preceq_{\text{rooted}} B_h$). Let $T' := T_1$ and $S := \langle T_2 \rangle * \cdots * \langle T_l \rangle$. Then (2) holds with respect to this S and T'. (Note that $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} S$ by Lemma 20, since $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} \langle T_i \rangle$ for all $i \in \{2, \dots, l\}$.)

Case 3: There exist distinct $i_1, i_2 \in [l]$ such that $\langle B_{h-1} \rangle \preceq_{\text{rooted}} \langle T_{i_1} \rangle$ and $\langle B_{h-1} \rangle \preceq_{\text{rooted}} \langle T_{i_2} \rangle$. In this case, we have $B_h \preceq_{\text{rooted}} T$. Therefore, (1) does not hold. To see that (2) does not hold, consider any S and T' such that $T = S * \langle T' \rangle$. Then $T' = T_i$ for unique $i \in [l]$ (by Lemma 20). Either $i \neq i_1$ or $i \neq i_2$. Suppose w.l.o.g. that $i \neq i_1$. Then $\langle T_{i_2} \rangle \preceq_{\text{rooted}} S$ and hence $\langle B_{h-1} \rangle \preceq S$. Therefore, (2) does not hold.

Case 4: There exists $i_1 \in [l]$ such that $\langle B_{h-1} \rangle \preceq_{\text{rooted}} \langle T_{i_1} \rangle$ and $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} \langle T_i \rangle$ for all $i \in [l] \setminus \{i_1\}$. W.l.o.g. assume $i_1 = 1$. We consider two subcases.

Subcase 4a: $\langle B_h \rangle \preceq_{\text{rooted}} \langle T_1 \rangle$. In this case, (1) does not hold. To see that (2) does not hold, consider any S and T' such that $T = S * \langle T' \rangle$. If $T' \neq T_1$, then we have $\langle T_1 \rangle \preceq_{\text{rooted}} S$ and hence $\langle B_{h-1} \rangle \preceq_{\text{rooted}} \langle B_h \rangle \preceq_{\text{rooted}} S$. If $T = T_1$, then $\langle B_h \rangle \preceq_{\text{rooted}} \langle T_1 \rangle$ implies either $\langle B_h \rangle \preceq_{\text{rooted}} T_1$ or $B_h \preceq_{\text{rooted}} T_1$ (by Lemma 19). Therefore, (2) does not hold.

Subcase 4b: $\langle B_h \rangle \not\preceq_{\text{rooted}} \langle T_1 \rangle$. In this case, both (1) and (2) hold. To see that (1) holds, we cannot have $\langle B_h \rangle \preceq_{\text{rooted}} T$ as this implies that $\langle B_h \rangle \preceq_{\text{rooted}} \langle T_i \rangle$ for some $i \in [l]$: $i = 1$ is ruled out by the assumption $\langle B_h \rangle \not\preceq_{\text{rooted}} \langle T_1 \rangle$, and $i \neq 1$ is ruled out by the assumption $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} \langle T_i \rangle$ for $i \in [l] \setminus \{1\}$. We also cannot have $B_h \preceq_{\text{rooted}} T$ as this implies that $\langle B_{h-1} \rangle \preceq_{\text{rooted}} \langle T_{i_1} \rangle$ and $\langle B_{h-1} \rangle \preceq_{\text{rooted}} \langle T_{i_2} \rangle$ for distinct $i_1, i_2 \in [l]$. To see that (2) holds, let $T' := T_1$ and $S := \langle T_2 \rangle * \cdots * \langle T_l \rangle$. We have $T = S * \langle T' \rangle$ and $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} S$. Finally, the assumption $\langle B_h \rangle \not\preceq_{\text{rooted}} \langle T_1 \rangle$ implies that $\langle B_h \rangle \not\preceq_{\text{rooted}} T'$ and $B_h \not\preceq_{\text{rooted}} T'$.

As the above cases cover all possibilities, this shows that (1) $\iff$ (2). To show the equivalence with (3), consider the following procedure. Assume that (1) holds, that is, $\langle B_h \rangle \not\preceq_{\text{rooted}} T$ and $B_h \not\preceq_{\text{rooted}} T$. If $T = P_1$, then (3) is satisfied with $m = 0$ (under the convention that the $*$ of zero rooted trees is P_1). Otherwise, (2) gives a decomposition $T = S * \langle T' \rangle$. Set $S_1 := S$ and obtain $S_2, \dots, S_m$ from T'.

Finally, to show that (3) implies (2), assume that $T = S_1 * \langle S_2, \dots, S_m \rangle$ as in (3). If $m = 0$, then $T = P_1$. So assume $m \geq 1$ and let $S := S_1$ and $T' := S_2 * \langle S_3, \dots, S_m \rangle$. Note that $T = S * \langle T' \rangle$ and

$\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} S$. It is easy to see that $\langle B_h\rangle \not\preceq_{\text{rooted}} T'$ and $B_h \not\preceq_{\text{rooted}} T'$, as either $\langle B_h\rangle \preceq_{\text{rooted}} T'$ or $B_h \preceq_{\text{rooted}} T'$ imply that $\langle B_{h-1}\rangle \preceq_{\text{rooted}} S_i$ for some $i \in \{2,\dots,l\}$. □

Lemma 22. *Let T be a rooted tree and $h \geq 1$. Then $\langle B_h\rangle \not\preceq_{\text{rooted}} T$ if, and only if, there exist $m \in \mathbb{N}$ and $l_1,\dots,l_m \in \mathbb{N}$ and rooted trees $S_{i,j}$ ($i \in [m]$, $j \in [l_i]$) such that*

$$T = \langle S_{1,1},\dots,S_{1,l_1}\rangle * \cdots * \langle S_{m,1},\dots,S_{m,l_m}\rangle$$

and $\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} S_{i,j}$ for all $i \in [m]$ and $j \in [l_i]$.

Proof. For the "only if" direction, assume $\langle B_h\rangle \not\preceq_{\text{rooted}} T$. Let $T = \langle T_1\rangle * \cdots * \langle T_m\rangle$ be the decomposition of T given by Lemma 18. Then for all $i \in [m]$, we have $\langle B_h\rangle \not\preceq_{\text{rooted}} T_i$ and $B_h \not\preceq_{\text{rooted}} T_i$ by Lemmas 19 and 20. By Lemma 21, there exist $l_i \in \mathbb{N}$ and rooted trees $S_{i,1},\dots,S_{i,l_i}$ such that $T_i = S_{i,1} * \langle S_{i,2},\dots,S_{i,l_i}\rangle$ and $\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} S_{i,j}$ for all $j \in [l_i]$. We have $\langle T_i\rangle = \langle S_{i,1},\dots,S_{i,l_i}\rangle$, and hence $T = \langle S_{1,1},\dots,S_{1,l_1}\rangle * \cdots * \langle S_{m,1},\dots,S_{m,l_m}\rangle$.

For the "if" direction, we run this argument in reverse. Assume $T = \langle S_{1,1},\dots,S_{1,l_1}\rangle * \cdots * \langle S_{m,1},\dots,S_{m,l_m}\rangle$ where $\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} S_{i,j}$ for all $i \in [m]$ and $j \in [l_i]$. For $i \in [m]$, let $T_i := S_{i,1} * \langle S_{i,2},\dots,S_{i,l_i}\rangle$. By Lemma 21, we have $\langle B_h\rangle \not\preceq_{\text{rooted}} T_i$ and $B_h \not\preceq_{\text{rooted}} T_i$. Therefore, $\langle B_h\rangle \not\preceq_{\text{rooted}} \langle T_i\rangle$ by Lemma 19 and $\langle B_h\rangle \not\preceq_{\text{rooted}} \langle T_1\rangle * \cdots * \langle T_m\rangle = T$ by Lemma 20. □

4.2 Treedepth bounds

The next lemmas give bounds on the treedepth of the *underlying graph* of T (that is, ignoring the root).

Lemma 23 (See [19, 20])**.** *For all $k, h \geq 1$, we have $\mathbf{td}(P_k) = \lceil \log(k+1)\rceil$ and $\mathbf{td}(B_h) = h$.* □

Note that the embedding $P_{15} \subseteq \text{Clos}(B_4)$, which witnesses the bound $\mathbf{td}(P_{15}) \leq 4$, is depicted in Figure 1.

Lemma 24. *For all $m \in \mathbb{N}$ and rooted trees $T_1,\dots,T_m$,*

$$\mathbf{td}(T_1 * \cdots * T_m) \leq \max\{\mathbf{td}(T_1),\dots,\mathbf{td}(T_m)\} + 1.$$

Proof. Let $\mathbf{td}_{\text{rooted}}(T)$ denote the minimum height of a rooted tree T' with $\text{root}(T') = \text{root}(T)$ and $E(T) \subseteq E(\text{clos}(T'))$. It is easy to see that $\mathbf{td}(T) \leq \mathbf{td}_{\text{rooted}}(T)$ and $\mathbf{td}_{\text{rooted}}(T_1 * \cdots * T_m) = \max\{\mathbf{td}(T_1),\dots,\mathbf{td}(T_m)\} + 1$. □

Lemma 25. *For all $m \in \mathbb{N}$ and rooted trees $T_1,\dots,T_m$,*

$$\mathbf{td}(\langle T_1,\dots,T_m\rangle) \leq \lceil \log(m+2)\rceil + \max\{\mathbf{td}(T_1),\dots,\mathbf{td}(T_m)\}.$$

Proof. For each $i \in [m]$, fix a rooted tree T'_i of height $\mathbf{td}(T_i)$ such that $E(T_i) \subseteq E(\text{clos}(T'_i))$. Invoking Lemma 23, let T'_0 be a rooted tree of height $\lceil \log(m+2)\rceil$ such that $E(P_{k+1}) \subseteq E(\text{clos}(T'_0))$. Label the vertices of P_{m+1} as $v_0,\dots,v_k$ with v_0 being the root. Let T' be the rooted tree, with root v_0, obtained from the disjoint union of $T'_0,\dots,T'_k$ by identifying vertices v_i and $\text{root}(T'_i)$ for each $i \in [m]$. Note that $E(\langle T_1,\dots,T_k\rangle) \subseteq E(\text{clos}(T'))$ and

$$\text{height}(T') \leq \text{height}(T'_0) + \max_{i\in[m]} \text{height}(T'_i) = \lceil \log(m+2)\rceil + \max_{i\in[m]} \mathbf{td}(T_i).$$ □

Lemma 26. *For every rooted tree T and $h \geq 0$ and $k \geq 1$, if $\langle B_h \rangle \not\preceq_{\text{rooted}} T$ and $P_k \not\preceq_{\text{rooted}} T$, then*

$$\mathbf{td}(T) \leq h \cdot (\lceil \log(k+1) \rceil + 1).$$

Proof. By induction on h. The base case $h = 0$ is vacuous, since $\langle B_0 \rangle = P_1$ is a rooted minor of every rooted tree. For the induction step, let $h \geq 1$ and assume $\langle B_h \rangle \not\preceq_{\text{rooted}} T$ and $P_k \not\preceq_{\text{rooted}} T$. By Lemma 22, there exist $m \in \mathbb{N}$ and $l_1, \dots, l_m \in \mathbb{N}$ and rooted trees $S_{i,j}$ ($i \in [m]$, $j \in [l_i]$) such that

$$T = \langle S_{1,1}, \dots, S_{1,l_1} \rangle * \cdots * \langle S_{m,1}, \dots, S_{m,l_m} \rangle$$

and $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} S_{i,j}$ for all $i \in [m]$ and $j \in [l_i]$. We also clearly have $l_i < k$ and $P_k \not\preceq_{\text{rooted}} S_{i,j}$ for all $i \in [m]$ and $[l_i]$. By the induction hypothesis, $\mathbf{td}(S_{i,j}) \leq (h-1) \cdot \lceil \log(k+1) \rceil$. By Lemma 25, we have

$$\begin{aligned}
\mathbf{td}(\langle S_{i,1}, \dots, S_{i,l_i} \rangle) &\leq \lceil \log(l_i + 2) \rceil + \max\{\mathbf{td}(S_{i,1}), \dots, \mathbf{td}(S_{i,l_i})\} \\
&\leq \lceil \log(k+1) \rceil + (h-1) \cdot (\lceil \log(k+1) \rceil + 1) \\
&= h \cdot (\lceil \log(k+1) \rceil + 1) - 1.
\end{aligned}$$

Finally, by Lemma 24, we have

$$\begin{aligned}
\mathbf{td}(T) &\leq \max\{\mathbf{td}(\langle S_{1,1}, \dots, S_{1,l_1} \rangle), \dots, \mathbf{td}(\langle S_{m,1}, \dots, S_{m,l_m} \rangle)\} + 1 \\
&\leq h \cdot (\lceil \log(k+1) \rceil + 1).
\end{aligned}$$ □

Lemma 27. *Every rooted tree with treedepth $\geq d$ contains a subcubic rooted subtree of order $\geq 2^{\sqrt{d}-2}$.*

Proof. We prove the contrapositive. Suppose T is a rooted tree that does not contain a subcubic rooted subtree of order $\geq 2^{\sqrt{d}-2}$. In particular, T does not have $\langle B_h \rangle$ or P_k as a rooted minor where $h = \lceil \sqrt{d} - 2 \rceil$ and $k = 2^h$. By Lemma 26, it follows that

$$\mathbf{td}(T) \leq h \cdot (\lceil \log(k+1) \rceil + 1) \leq (\sqrt{d} - 1)(\lceil \log(2^{\sqrt{d}-1} + 1) \rceil + 1) < d.$$ □

4.3 Large connected graphs with small degree have a large P_k or B_h minor

Lemma 28. *Let $h, k, c \geq 1$ and suppose T is a rooted tree such that $\langle B_h \rangle \not\preceq_{\text{rooted}} T$ and $P_k \not\preceq_{\text{rooted}} T$ and every vertex of T has at most c children. Then $|V(T)| \leq (ck)^h$.*

Proof. By induction on h. In the base case $h = 1$, we have T is a subdivided star (since $P_2 * P_2 = \langle B_1 \rangle \not\preceq_{\text{rooted}} T$). Moreover, we have $T = \langle P_{k_1} \rangle * \cdots * \langle P_{k_m} \rangle$ where $0 \leq m \leq c$ and $1 \leq k_i \leq k-1$ for all $i \in [m]$. Therefore,

$$|V(T)| = 1 + \sum_{i=1}^{m} k_i \leq 1 + c(k-1) \leq ck.$$

For the induction step, suppose $h \geq 2$. By Lemma 22, there exist $m \in \mathbb{N}$ and $l_1, \dots, l_m \in \mathbb{N}$ and $S_{i,j}$ ($i \in [m]$, $j \in [l_m]$) such that $T = \langle S_{1,1}, \dots, S_{1,l_1} \rangle * \cdots * \langle S_{m,1}, \dots, S_{m,l_m} \rangle$ and $\langle B_{h-1} \rangle \not\preceq_{\text{rooted}} S_{i,j}$ for all $i \in [m]$ and $j \in [l_m]$. Note that $m \leq c$ and $l_i \leq k-1$ and $P_{k-1} \not\preceq_{\text{rooted}} S_{i,j}$ for all $i \in [m]$ and $j \in [l_i]$. By the induction hypothesis, we have $|V(S_{i,j})| \leq (c(k-1))^{h-1}$ for all i and j. Therefore,

$$|V(T)| = 1 + \sum_{i=1}^{m} \sum_{j=1}^{l_i} |V(S_{i,j})| \leq 1 + c(k-1)(c(k-1))^{h-1} = 1 + (c(k-1))^h \leq (ck)^h.$$ □

Lemma 29. *Every connected graph G with maximum degree $\leq c+1$ and order $\geq c^k$ has a path of order $c^{\lceil\sqrt{k}\rceil-2}$ or $B_{\lceil\sqrt{k}\rceil-2}$ as a minor.*

Proof. Assume that G contains no path of length $c^{\lceil\sqrt{k}\rceil-2}$ and no $B_{\lceil\sqrt{k}\rceil-2}$-minor. Let T be any spanning tree of G rooted at any of its leaves. Since G has maximum degree $c+1$, it follows that every node has at most c children in T. Moreover, T does not have $P_{c^{\lceil\sqrt{k}\rceil-2}}$ or $B_{\lceil\sqrt{k}\rceil-2}$ as a rooted minor. By Lemma 28, T has order at most $(c^{\lceil\sqrt{k}\rceil-1})^{\lceil\sqrt{k}\rceil-2} < c^k$. It follows that G has order $< c^k$ since $V(G) = V(T)$. □

5 Proof of Theorem 3

We prove Theorem 3 by establishing the following, stronger theorem.

Theorem 30. *Every graph G contains a path of order 2^h or has a B_h-minor where*

$$h = \Omega\left(\frac{\mathbf{td}(G)^{1/4}}{w^{1/4}\log^{1/2} w}\right), \quad w = \mathbf{tw}(G) + 1.$$

Ignoring the $\log^{1/2}(\mathbf{tw}(G)+1)$ factor, observe that h is essentially $r^{1/4}$ where $r = \mathbf{td}(G)/(\mathbf{tw}(G)+1)$. (Recall that $\mathbf{tw}(G) + 1 \leq \mathbf{td}(G)$ by inequality (1), so the ratio r is always at least 1.)

Proof of Theorem 3 assuming Theorem 30. Consider any $k \geq 1$ and suppose G is a graph such that $\mathbf{tw}(G) < k$ and G contains no path of order 2^k and G contains no B_k-minor. To prove Theorem 3, we must show that $\mathbf{td}(G) = O(k^5 \log^2 k)$. If we assume Theorem 3, then we see that

$$k \geq \Omega\left(\frac{\mathbf{td}(G)^{1/4}}{w^{1/4}\log^{1/2} w}\right), \quad w = \mathbf{tw}(G) + 1 \leq k.$$

Therefore, $\mathbf{td}(G) \leq O(k^4 w \log^2 w) \leq O(k^5 \log^2 k)$. □

The rest of this section is devoted the proof of Theorem 30.

Proof of Theorem 30. It clearly suffices to prove the theorem for connected graphs G. Let G be any connected graph, and let $r = \mathbf{td}(G)/(\mathbf{tw}(G) + 1)$. Our goal is to show that G contains a path of length 2^h or a B_h-minor where $h = \Omega(r^{1/4}/\log^{1/2}(\mathbf{tw}(G)+1))$. (Here we write $\mathbf{tw}(G)+1$ instead of w, since we will use w below to stand for a vertex of G.)

By Lemma 14, G has a greedy rooted tree decomposition T of width $\mathbf{tw}(G)$. By Lemma 9, we have $\mathbf{td}(T) \geq r$.

In the rest of the proof, we will construct a sequence of three trees: first a spanning tree $F \subseteq G$, then a subcubic rooted subtree $S \subseteq T$ of order $|V(S)| = 2^{\Omega(\sqrt{r})}$, and finally a subtree $Q \subseteq F$ with maximum degree $\leq \mathbf{tw}(G) + 2$ and $V(S) \subseteq V(Q)$. Using Lemma 29, we then conclude that Q contains a path of length 2^h or a B_h-minor where $h = \Omega(r^{1/4}/\log^{1/2}(\mathbf{tw}(G) + 1))$. Since $Q \subseteq G$, this completes the proof.

The spanning tree $F \subseteq G$:

- Let $V' = V(G) \setminus \{\mathrm{root}(T)\}$. For $x \in V'$, let $\widehat{x}$ be the parent of x in T (i.e. the unique vertex in $V(G)$ $(= V(T))$ such that $x\widehat{x} \in \vec{E}(T)$).
- By condition (iii) of Definition 10, there exists a function $x \mapsto \breve{x} : V' \to V'$ such that $\breve{x} \leq_T x$ and $\{\breve{x}, \widehat{x}\} \in E(G)$ for all $x \in V'$. Fix any choice of such a function $x \mapsto \breve{x}$.
- Let $F \subseteq G$ be the subgraph of G defined by $V(F) = V(G)$ and $E(F) = \{\{\breve{x}, \widehat{x}\} : x \in V'\}$.

Claim 1. *F is a spanning tree for G (that is, F is a tree and $V(F) = V(G)$).*

▶ The fact that F is a tree follows from the observation that $\breve{x} <_T \widehat{x}$ for all $x \in V'$ (since $\breve{x} \leq_T x$ and $x <_T \widehat{x}$). To see that $V(F) = V(G)$, note that $V(G) = V(T)$ and consider any vertex $x \in V(G)$. If x is a non-leaf in T, then let w be any of its children (i.e. $x = \widehat{w}$); we have $\{\breve{w}, x\} \in E(F)$ and hence $x \in V(F)$. If x is a leaf in T, then (assuming w.l.o.g. that $|V(G)| \geq 2$ so that $x \in V'$) we have $\breve{x} = x$ (this is forced by the requirement $\breve{x} \leq_T x$) and therefore $\{\breve{x}, \widehat{x}\} \in E(F)$ and hence $\breve{x} \in V(F)$. $\square_{\text{claim}}$

Claim 2. *Each edge $\{u, v\} \in E(F)$ satisfies $u <_T v$ or $v <_T u$.*

▶ Let $\{u, v\} \in E(F)$. There exists $x \in V'$ such that $\{u, v\} = \{\breve{x}, \widehat{x}\}$. Either $u = \breve{x}$ and $v \in \widehat{x}$ (in which case $u \leq_T x <_T v$), or $u = \widehat{x}$ and $v \in \breve{x}$ (in which case $v \leq_T x <_T u$). $\square_{\text{claim}}$

Claim 3. *If $x \in V'$ and P is the unique path in F between $\widehat{x}$ and x, then $V(P) \setminus \{\widehat{x}\} \leq_T x$.*

▶ Let $(p_0, \ldots, p_t)$ be the sequence of vertices on the unique path from $\widehat{x}$ to x in F (with $p_0 = \widehat{x}$ and $p_t = x$). Let p_i be the unique $<_T$-maximum element in $\{p_0, \ldots, p_t\}$. Toward a contradiction assume $i \neq 0$ (that is, $p_i \neq \widehat{x}$). Then $i \in \{1, \ldots, t-1\}$ and, moreover, $p_{i-1} <_T p_i$ and $p_{i+1} <_T p_i$ (by Claim 2 since $\{p_{i-1}, p_i\} \in E(F)$ and $\{p_i, p_{i+1}\} \in E(G)$). Since $p_{i-1} \neq p_{i+1}$, it must be the case that $p_{i-1} = \breve{u}$ and $p_{i+1} = \breve{w}$ for distinct $u, w \in V'$ such that $\widehat{u} = \widehat{w} = p_i$. It may now be seen that $p_0, \ldots, p_{i-1} \leq_T u$ and $p_{i+1}, \ldots, p_t \leq_T w$.[3] Since $\widehat{x} = p_0$ and $x = p_t$, this means that $\widehat{x} \leq_T u$ and $x \leq_T w$. But then $\widehat{x}$ and x would be incomparable under $<_T$ (since u and w are siblings in T). This yields the desired contradiction, since $\widehat{x}$ is the parent of x in T. $\square_{\text{claim}}$

The rooted subtree $S \subseteq T$:

- By Lemma 27, T has a subcubic rooted subtree S of order $2^{\Omega(\sqrt{r})}$ (with $\mathrm{root}(S) = \mathrm{root}(T)$ and $\vec{E}(S) \subseteq \vec{E}(T)$). Fix any choice of S.
- Let $W = V(S)$ and $W' = V(S) \setminus \{\mathrm{root}(S)\}$.

The reason we need this subcubic tree S will become clear later on (in Claim 4). In short, the fact that S has degree ≤ 3 guarantees that the tree Q (which we are about to construct) will have maximum degree $\leq \mathbf{tw}(G) + 2$.

[3] To see why, if $p_j \leq_T w$ and $p_{j+1} \not\leq_T w$ for some $i+1 \leq j \leq t-1$, then it must be the case that $p_j <_T p_{j+1}$; hence p_{j+1} and p_i are comparable (since $p_j \leq_T w <_T p_i$ and $<_T$ linearly orders $\{y : p_j \leq y\}$); hence $p_i <_T p_{j+1}$, but this contradicts the maximality of p_i. A similar contradiction arises if we assume that $p_j \not\leq_T u$ for some $0 \leq j \leq i-1$.

Trees $\{Q_x \subseteq F\}_{x \in W}$ and paths $\{P_x \subseteq F\}_{x \in W'}$: By simultaneous induction (upward from the leaves of S), we define families of subgraphs $\{Q_x \subseteq F\}_{x \in W}$ and $\{P_x \subseteq P\}_{x \in W'}$ where each Q_x is a tree satisfying $x \in V(Q_x)$ and $V(Q_x) \leq_T x$ and each P_x is a path satisfying $\widehat{x} \in V(P_x)$ and $V(P_x) \setminus \{\widehat{x}\} \leq_T x$ and $|V(P_x) \cap V(Q_x)| = 1$.

- Suppose $x \in W$ is a leaf in S. Then Q_x is the single-vertex graph with $V(Q_x) = \{x\}$.

  (Note that Q_x satisfies $x \in V(Q_x)$ and $V(Q_x) \leq_T x$.)

- Suppose $x \in W'$ where Q_x is already defined. Let $(p_0, \dots, p_t)$ be the sequence of vertices on the unique path from $\widehat{x}$ to x in F (with $p_0 = \widehat{x}$ and $p_t = x$). Let $s \in \{1, \dots, t\}$ be the minimum index satisfying $p_s \in V(Q_x)$. (This is well-defined since $p_t = x \in V(Q_x)$.) Then P_x is the subpath of F with $V(P_x) = \{p_0, \dots, p_s\}$ and $E(P_x) = \{\{p_0, p_1\}, \dots, \{p_{s-1}, p_s\}\}$.

  (Note that P_x satisfies $\widehat{x} = p_0 \in V(P_x)$ and $V(P_x) \setminus \{\widehat{x}\} = \{p_1, \dots, p_s\} \leq_T x$ (by Claim 3) and $V(P_x) \cap V(Q_x) = \{p_s\}$.)

- Suppose $x \in W$ is a non-leaf with children $w_1, \dots, w_k$ in S (i.e. $\{w_1, \dots, w_k\} = \{w \in W : wx \in \vec{E}(S)\}$, noting that x may have additional children in T) where $Q_{w_1}, \dots, Q_{w_k}$ and $P_{w_1}, \dots, P_{w_k}$ are already defined. We define $Q_x = (Q_{w_1} \cup \dots \cup Q_{w_k}) \cup (P_{w_1} \cup \dots \cup P_{w_k})$.

  (Note that Q_x satisfies $x \in V(Q_x)$ and $V(Q_x) \leq_T x$.)

The tree $Q \subseteq F$ and vertices $\{\widetilde{x} \in V(P_x) \cap V(Q_x)\}_{x \in W'}$:

- Finally, let $Q = Q_{\mathrm{root}(S)}$. Note that $Q = \bigcup_{x \in W'} P_x$.

- For each $x \in W'$, let $\widetilde{x}$ be the unique element in $V(P_x) \cap V(Q_x)$. (That is, $\widetilde{x}$ is the vertex p_s in the above definition of P_x.) Thus, P_x is the unique path in F between $\widehat{x}$ (the parent of x in S) and $\widetilde{x}$ (the first vertex in $V(Q_x)$ encountered on the unique path in F from $\widehat{x}$ to x).

<u>Claim</u> 4. *For all $q \in V(Q)$, we have* $\deg_Q(q) \leq |\{x \in W' : q = \widetilde{x}\}| + 2$.

▶ Consider any $q \in V(Q)$. Since $Q = \bigcup_{x \in W'} P_x$, we have

$$\deg_Q(q) = \sum_{x \in W'} \deg_{P_x}(q) = 2 \cdot |\{x \in W' : \deg_{P_x}(q) = 2\}| + |\{x \in W' : q = \widehat{x}\}| + |\{x \in W' : q = \widetilde{x}\}|.$$

Now observe the following:

- $|\{x \in W' : q = \widehat{x}\}| \leq 2$ (since S is subcubic).

- $|\{x \in W' : \deg_{P_x}(q) = 2\}| \leq 1$ (this follows from the definition of P_x).

- If $q \in W$, then $\{x \in W' : \deg_{P_x}(q) = 2\} = \emptyset$.

- If $q \notin W$, then $\{x \in W' : q = \widehat{x}\} = \emptyset$.

The claim follows. $\square_{\text{claim}}$

<u>Claim</u> 5. *For all $q \in V(Q)$, we have* $|\{x \in W' : q = \widetilde{x}\}| \leq |\mathrm{Bag}_{T,G}(q)| - 1 \leq \mathbf{tw}(G)$.

▶ Consider any $q \in V(Q)$. Recall that

$$\mathrm{Bag}_{T,G}(q) = \{q\} \cup \{y : \text{there exists } w \text{ such that } w \leq_T q <_T y \text{ and } \{w,y\} \in E(G)\}.$$

For each $x \in W'$ such that $q = \widetilde{x}$, define the set

$$U_x = \{u \in V(P_x) : \breve{u} \leq_T q <_T u\}.$$

Three simple observations:

- We have $U_x \subseteq \mathrm{Bag}_{T,G}(q) \setminus \{x\}$ (since $\{\breve{u}, u\} \in E(F) \subseteq E(G)$ for all $u \in V'$).
- Let $(p_0, \dots, p_s)$ be the unique path in F from $\widehat{x}$ to q (with $p_0 = \widehat{x}$ and $p_s = q$). Let $i \in \{1, \dots, s\}$ be the minimum index such that $p_i \leq_T q$. Then $p_i = \breve{p}_{i-1}$ and $p_i \leq_T q <_T p_{i-1}$. Therefore, U_x is nonempty.
- We have $V(P_x) \cap V(P_y) = \{q\}$ for all distinct $x, y \in W'$ such that $q = \widetilde{x} = \widetilde{y}$. Therefore, U_x and U_y are disjoint.

It follows from these three observations that $|\{x \in W' : q = \widetilde{x}\}| \leq |\mathrm{Bag}_{T,G}(q)| - 1$. Finally, recall that T was chosen such that $\mathrm{width}(T,G) = \max_{x \in V(G)} |\mathrm{Bag}_{T,G}(x)| - 1 = \mathbf{tw}(G)$. $\square_{\text{claim}}$

Claims 4 and 5 imply that Q has maximum degree $\leq \mathbf{tw}(G) + 2$. Since $V(S) \subseteq V(Q)$, we have

$$|V(Q)| \geq |V(S)| \geq 2^{\Omega(\sqrt{r})} = (\mathbf{tw}(G) + 1)^{\Omega(\sqrt{r}/\log(\mathbf{tw}(G)+1))}.$$

It now follows from Lemma 29 that Q (and hence also G since $Q \subseteq G$) contains either a path of order 2^h or a B_h-minor where $h = \Omega(r^{1/4}/\log^{1/2}(\mathbf{tw}(G)+1))$. This completes the proof of Theorem 30. $\square$

6 Application in Complexity and Logic

The main result of this paper, Theorem 3, was motivated by a specific application in circuit complexity and logic. By combining our polynomial excluded-minor approximation of treedepth with lower bounds on the AC^0 formula size of detecting grids [17], paths [25] and trees [27], we obtain an $n^{\mathsf{poly}(\mathbf{td}(G))}$ lower bound on the AC^0 formula size of the colored G-subgraph isomorphism problem for all graphs G. This result, in turn, has a surprising corollary in finite model theory: a polynomial-rank homomorphism preservation theorem on finite structures. In this section, we give a brief overview of these results (see the paper [26] for details).

6.1 The AC^0-Formula Size of Subgraph Isomorphism

In order to define the colored G-subgraph isomorphism problem, we first introduce the blow-up $G^{\uparrow n}$.

Definition 31. For a graph G and $n \in \mathbb{N}$, the *n-fold blow-up* of G is the graph $G^{\uparrow n}$ defined by

$$V(G^{\uparrow n}) = V(G) \times [n],$$
$$E(G^{\uparrow n}) = \big\{\{(v,a),(w,b)\} : \{v,w\} \in E(G),\, a,b \in [n]\big\}.$$

For $\alpha \in [n]^{V(G)}$, the subgraph $G^{(\alpha)} \subseteq G^{\uparrow n}$ (an isomorphic copy of G) is defined by

$$V(G^{(\alpha)}) = \{(v, \alpha_v) : v \in V(G)\},$$
$$E(G^{(\alpha)}) = \{\{(v, \alpha_v), (w, \alpha_w)\} : \{v, w\} \in E(G)\}.$$

Definition 32. The *colored G-subgraph isomorphism problem* is the following problem:

Given a graph $X \subseteq G^{\uparrow n}$, does there exist $\alpha \in [n]^{V(G)}$ such that $G^{(\alpha)} \subseteq X$?

To study the complexity of this problem, we view it as a sequence $\mathrm{SUB}(G) = \{\mathrm{SUB}(G,n)\}_{n\in\mathbb{N}}$ of Boolean functions $\mathrm{SUB}(G,n) : \{0,1\}^{|E(G)|\cdot n^2} \to \{0,1\}$.

The following lemma from Li et al [17] shows that the complexity of $\mathrm{SUB}(G)$ is a minor-monotone graph parameter.

Lemma 33. *If H is a minor of G, then there is a monotone-projection reduction from* $\mathrm{SUB}(H,n)$ *to* $\mathrm{SUB}(G,n)$ *for every $n \in \mathbb{N}$.*[4]

As a consequence of Lemma 33, the function $G \mapsto \chi(\mathrm{SUB}(G,n))$ is a minor-monotone graph parameter for all standard complexity measures $\chi(\cdot)$, including AC^0 circuit/formula size.[5]

It is known that $\mathrm{SUB}(G)$ is computable by AC^0 circuits of size $O(n^{\mathbf{tw}(G)+1})$, as well as by AC^0 formulas of size $O(n^{\mathbf{td}(G)})$ (moreover, depth $|V(G)|$ is sufficient in both cases).[6] The next theorem summarizes known lower bounds on the AC^0 complexity of $\mathrm{SUB}(G)$.

Theorem 34.

1. *The* AC^0 *circuit size of* $\mathrm{SUB}(G)$ *is at least* $n^{\Omega(\mathbf{tw}(G)/\log \mathbf{tw}(G))}$ *for all graphs G* [17].
2. *The* AC^0 *formula size of* $\mathrm{SUB}(P_k)$ *is at least* $n^{\Omega(\log k)}$ *for all k* [25].
3. *The* AC^0 *formula size of* $\mathrm{SUB}(B_k)$ *is at least* $n^{\Omega(\sqrt{k})}$ *for all k* [27].

Combining Theorem 3 with the three lower bounds in Theorem 34, and using the fact that the AC^0 formula size of $\mathrm{SUB}(G)$ is minor-monotone by Lemma 33, we get the following:

Theorem 35. *The* AC^0 *formula size of* $\mathrm{SUB}(G)$ *is at least* $n^{\Omega(\mathbf{td}(G)^{\varepsilon})}$ *for all graphs G where $\varepsilon > 0$ is an absolute constant.*

Theorem 3(1) and Theorem 35 lend support to the conjectures that the *unbounded-depth* circuit (resp. formula) size of $\mathrm{SUB}(G)$ is $n^{\Omega(\mathbf{tw}(G))}$ (resp. $n^{\Omega(\mathbf{td}(G))}$). Since these conjectures imply $\mathrm{P} \neq \mathrm{NP}$ and $\mathrm{NC}^1 \neq \mathrm{NL}$, it is an interesting and worthwhile first step to prove these lower bounds in the restricted bounded-depth setting.

[4] This means that $\mathrm{SUB}(H,n)$ reduces to $\mathrm{SUB}(G,n)$ via a function that maps each edge-indicator variable $Y_{e'}$ ($e' \in E(H^{\uparrow n})$) to either a constant (0 or 1) or an edge-indicator variable X_e ($e \in E(G^{\uparrow n})$).

[5] Recall that AC^0 is the class of constant-depth polynomial-size circuits in the basis $\{\mathrm{AND}_\infty, \mathrm{OR}_\infty, \mathrm{NOT}\}$. Formulas are circuits with fan-out 1. For a sequence of Boolean function $f = (f_n)$ and $d \geq 2$, the *depth-d* AC^0 *circuit/formula size of f* is the minimum number of gates in a depth-d AC^0 circuit/formula that computes f_n, as a function of n. We say that the AC^0 *circuit/formula size* of f is $O(n^c)$ (resp. $\Omega(n^c)$) if the depth-d AC^0 circuit/formula size of f is $O_d(n^c)$ (resp. $\Omega_d(n^c)$).

[6] With respect to the *uncolored* G-subgraph isomorphism problem, one obtains the essentially same upper bounds via the "color-coding" technique of Alon, Yuster and Zwick [2], which Amano [3] observed can be implemented in AC^0.

6.2 An Improved Homomorphism Preservation Theorem on Finite Structures

Theorem 35 turns out to have a surprising corollary in finite model theory. The following result was proved in [26].

Theorem 36. *Let φ be a first-order sentence of quantifier-rank r. If φ is preserved under homomorphisms on finite structures, then there is an existential-positive sentence ψ of quantifier-rank $r^{O(1)}$ such that φ and ψ are logically equivalent on finite structures.*

The proof of Theorem 36 is based on a reduction to the AC^0 formula of $\mathrm{SUB}(G)$ and relies on Theorem 35 (and hence on Theorem 3) for the polynomial bound on the quantifier-rank of ψ. Theorem 36 dramatically improves an earlier result in [24], in which the bound on quantifier-rank of ψ is a non-elementary function of r (i.e. growing faster than any constant-height tower of exponentials).

7 Conclusion

7.1 Algorithmic Version of Theorems and 2 and 3

The Polynomial Grid-Minor Theorem of Chekuri and Chuzhoy [11] is more than the existential statement given as Theorem 1: the results of [11] include a polynomial-time algorithm which, given a graph of treewidth $\geq k^c$, outputs a $k \times k$ minor. We remark that Theorems 2 and 3 have corresponding polynomial-time algorithms as well. Suppose we are given a graph G with treedepth $\geq k^c$ (for a suitable constant c). As a first step, in polynomial-time we construct a tree decomposition $(T, \mathcal{W})$ for G of width $O(\mathbf{tw}(G)^2)$ (using the 5-approximation algorithm of Bodlaender et al [8] for the case $\mathbf{tw}(G) \leq \log n$ and the $O(\log n)$-approximation algorithm of Bodlaender et al [9] for the case $\mathbf{tw}(G) > \log n$). Given this tree decomposition, we observe that all proofs in Sections 4 and 5 are constructive. Thus, we get a polynomial-time algorithm which either: certifies that G has tree-width $\geq k$, or finds a B_k-minor in G, or finds a path of length 2^k in G.

7.2 Open Questions

In light of Theorems 1 and 2, we conjecture the following "Polynomial Grid/Tree-Minor Theorem for Pathwidth":

Conjecture 6. *There is an absolute constant c such that every graph with pathwidth $\geq k^c$ has one of the following minors:*

- *the $k \times k$ grid,*
- *the complete binary tree of height k.*

The techniques introduced in this paper might be helpful in proving this conjecture. Another open problem is to improve the $Ck^5 \log^2 k$ bound in Theorem 3. The optimal bound is likely smaller (however, examples show one cannot do better than $O(k^2)$).

References

[1] Isolde Adler, Martin Grohe, and Stephan Kreutzer. Computing excluded minors. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 641–650. Society for Industrial and Applied Mathematics, 2008.

[2] Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *Journal of the ACM*, 42(4):844–856, 1995.

[3] Kazuyuki Amano. k-Subgraph isomorphism on AC^0 circuits. *Computational Complexity*, 19(2):183–210, 2010.

[4] Michael D Barrus and John Sinkovic. Minimal obstructions for tree-depth: A non-1-unique example. *arXiv preprint arXiv:1604.00550*, 2016.

[5] Michael D Barrus and John Sinkovic. Uniqueness and minimal obstructions for tree-depth. *Discrete Mathematics*, 339(2):606–613, 2016.

[6] Andreas Björklund, Thore Husfeldt, and Sanjeev Khanna. Approximating longest directed paths and cycles. In *International Colloquium on Automata, Languages, and Programming*, pages 222–233. Springer, 2004.

[7] Hans L Bodlaender, Jitender S Deogun, Klaus Jansen, Ton Kloks, Dieter Kratsch, Haiko Müller, and Zsolt Tuza. Rankings of graphs. *SIAM Journal on Discrete Mathematics*, 11(1):168–181, 1998.

[8] Hans L Bodlaender, Pal Gronas Drange, Markus S Dregi, Fedor V Fomin, Daniel Lokshtanov, and Michał Pilipczuk. A $c^k n$ 5-approximation algorithm for treewidth. *SIAM Journal on Computing*, 45(2):317–378, 2016.

[9] Hans L Bodlaender, John R Gilbert, Hjálmtyr Hafsteinsson, and Ton Kloks. Approximating treewidth, pathwidth, frontsize, and shortest elimination tree. *Journal of Algorithms*, 18(2):238–255, 1995.

[10] Kevin Cattell, Michael J Dinneen, Rodney G Downey, Michael R Fellows, and Michael A Langston. On computing graph minor obstruction sets. *Theoretical Computer Science*, 233(1):107–127, 2000.

[11] Chandra Chekuri and Julia Chuzhoy. Polynomial bounds for the grid-minor theorem. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 60–69. ACM, 2014.

[12] Li-Hsuan Chen, Felix Reidl, Peter Rossmanith, and Fernando Sánchez Villaamil. Width, depth and space. http://arxiv.org/abs/1607.00945, 2016.

[13] Zdeněk Dvořák, Archontia C Giannopoulou, and Dimitrios M Thilikos. Forbidden graphs for tree-depth. *European Journal of Combinatorics*, 33(5):969–979, 2012.

[14] Fedor V Fomin, Archontia C Giannopoulou, and Michał Pilipczuk. Computing tree-depth faster than 2^n. *Algorithmica*, 73(1):202–216, 2015.

[15] Archontia C Giannopoulou. *Tree-depth of Graphs: Characterisations and Obstructions.* PhD thesis, National and Kapodistrian University of Athens, 2009.

[16] Meir Katchalski, William McCuaig, and Suzanne Seager. Ordered colourings. *Discrete Mathematics*, 142(1):141–154, 1995.

[17] Yuan Li, Alexander Razborov, and Benjamin Rossman. On the AC^0 complexity of subgraph isomorphism. In *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*, pages 344–353. IEEE, 2014.

[18] Dániel Marx. Can you beat treewidth? In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 169–179. IEEE, 2007.

[19] Jaroslav Nešetřil and Patrice Ossona De Mendez. Tree-depth, subgraph coloring and homomorphism bounds. *European Journal of Combinatorics*, 27(6):1022–1041, 2006.

[20] Jaroslav Nešetřil and Patrice Ossona De Mendez. Sparsity (graphs, structures, and algorithms), algorithms and combinatorics, vol. 28, 2012.

[21] Alex Pothen. *The complexity of optimal elimination trees.* PhD thesis, Pennsylvania State University, Department of Computer Science, 1988.

[22] Felix Reidl, Peter Rossmanith, Fernando Sánchez Villaamil, and Somnath Sikdar. A faster parameterized algorithm for treedepth. In *International Colloquium on Automata, Languages, and Programming*, pages 931–942. Springer, 2014.

[23] Neil Robertson and Paul D Seymour. Graph minors. XX. Wagner's conjecture. *Journal of Combinatorial Theory, Series B*, 92(2):325–357, 2004.

[24] Benjamin Rossman. Homomorphism preservation theorems. *Journal of the ACM (JACM)*, 55(3):15, 2008.

[25] Benjamin Rossman. Formulas vs. circuits for small distance connectivity. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 203–212. ACM, 2014.

[26] Benjamin Rossman. An improved homomorphism preservation theorem from lower bounds in circuit complexity. In *Innovations in Theoretical Computer Science (ITCS)*, volume 67 of *LIPIcs*, 2017.

[27] Benjamin Rossman. Lower bounds for subgraph isomorphism. manuscript, 2017.

[28] Alejandro A Schäffer. Optimal node ranking of trees in linear time. *Information Processing Letters*, 33(2):91–96, 1989.

[29] Yu Wu, Per Austrin, Toniann Pitassi, and David Liu. Inapproximability of treewidth and related problems. *Journal of Artificial Intelligence Research*, 49:569–600, 2014.