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

November 1, 2017

Abstract

Treedepth is a well-studied graph invariant in the family of "width measures" that includes treewidth and pathwidth. Understanding these invariants in terms of excluded minors has been an active area of research. The recent Grid Minor Theorem of Chekuri and Chuzhoy [12] establishes that treewidth is polynomially approximated by the largest $k \times k$ grid minor. In this paper, we give a similar polynomial excluded-minor approximation for treedepth in terms of three basic obstructions: grids, tree, and paths. Specifically, we show that there is a constant c such that every graph of treedepth $\geq k^c$ contains one of the following minors (each of treedepth $\geq k$):

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

Let us point out that we cannot drop any of the above graphs for our purpose. Moreover, given a graph G we can, in randomized polynomial time, find either an embedding of one of these minors or conclude that treedepth of G is at most k^c.

This result has potential applications in a variety of settings where bounded treedepth plays a role. In addition to some graph structural applications, we describe a surprising application in circuit complexity and finite model theory from recent work of the second author [28].

1 Introduction

Treedepth is a well-studied graph invariant with several equivalent definitions. It appears in the literature under various names including vertex ranking number [30], ordered chromatic number [17], and minimum elimination tree height [23], before being systematically studied under the name treedepth by Ossona de Mendes and Nešetřil [20]. Bounded treedepth graphs play an important role in areas such as the theory of sparse graph classes [21, 22], parameterized complexity theory [13, 15, 24], and model theory [28, 29].

Formally, the *treedepth* of an undirected graph 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). The forest F may be regarded as

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

a decomposition of G into subgraphs of cliques that lie along the branches of F. This type of decomposition is related to the more familiar tree-decompositions in the definition of treewidth (see Section 3). Indeed, treedepth is a close relative of "width measures" like treewidth and pathwidth. Intuitively, treedepth measures how "star-like" a graph is (note that graphs of treedepth 1 are disjoint unions of stars), whereas treewidth and pathwidth measure the extent to which a graph is "tree-like" and "path-like". These three invariants, denoted $\mathbf{td}(G)$, $\mathbf{tw}(G)$ and $\mathbf{pw}(G)$, 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 [21] for proofs of (1) and (2).)

These four graph invariants — **td**, **tw**, **pw** and **lp** — 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 deletions, edge deletions and edge contractions. A graph invariant $f : \{\text{graphs}\} \to \mathbb{N}$ 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 [25], 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 invariant 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 invariants is an active topic of research in graph theory (see [1, 11]). When it comes to treedepth, minimal obstructions have been studied by two sets of authors [4, 5, 14, 16]. 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 [14] showed that the number of minimal obstructions grows enormously fast (at least doubly exponentially) as a function of k [14]. 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 7, 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 invariant. A recent breakthrough of Chekuri and Chuzhoy [12] gave precisely such a result for treewidth (resolving a longstanding conjecture in graph minor theory).

Theorem 1.1 (Polynomial Grid-Minor Theorem for Treewidth [12])**.** *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 1.1 establishes the treewidth of a graph is polynomially related to the size of its largest grid minor. (Prior to Theorem 1.1, 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 basic obstructions: grids, complete binary trees, and paths.

Theorem 1.2 (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 $\geq k$, the largest such obstruction gives a polynomial approximation of $\mathbf{td}(G)$. Moreover, all three obstructions in Theorem 1.2 are necessary for a polynomial approximation. (In light of (2), the length of the longest path in G gives a weaker exponential approximation of $\mathbf{td}(G)$.)

Theorem 1.2 is obtained by a combination of Theorem 1.1 and the following result, which is the technical main theorem of this paper.

Theorem 1.3 (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.*

Our proof of Theorem 1.3 is entirely self-contained (in particular, we do not rely on Theorem 1.1). Due to the constructive nature of the proofs, we get an algorithmic version of Theorem 1.3. Combined with the algorithmic version of Theorem 1.1 from [12], we get a randomized polynomial-time algorithm which, given a graph G, either determines that G is of small treedepth or outputs a certificate of one of the three cases in Theorem 1.2 (i.e., a minor-embedding of a grid, tree, or path. For details see Section 6).

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 1.3 in the case where G is a tree). In Section 5 we present the proof of Theorem 1.3. In Section 6 we describe polynomial-time algorithms which give effective versions of our main theorems. In Section 7 we describe a surprising application of Theorem 1.3 in circuit complexity and logic, which was the motivation for this paper. Finally, we conclude with some observations and open problems in Section 8.

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 2.1 (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 2.2 (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 $x \le_T y$ iff $x <_T y$ or $x = y$; for $W \subseteq V(T)$, we write $W \le_T x$ iff $w \le_T x$ 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.)

Definition 2.3 (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 2.4 (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 invariant $f : \{\text{graphs}\} \to \mathbb{N}$ is *minor-monotone* if $f(F) \le f(G)$ 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 this lemma is essentially folklore (it is implicit in proofs of the inequality $\mathbf{td}(G) \leq (\mathbf{tw}(G)+1)\log|V(G)|$ [9, 21]), we could not find a proof in the literature, so include one for completeness.

Lemma 3.1. *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 deleting 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

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

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 $\{y : x <^\star y\}$ is totally ordered by $<^\star$.) 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 $G \subseteq \mathrm{Clos}(R)$. □

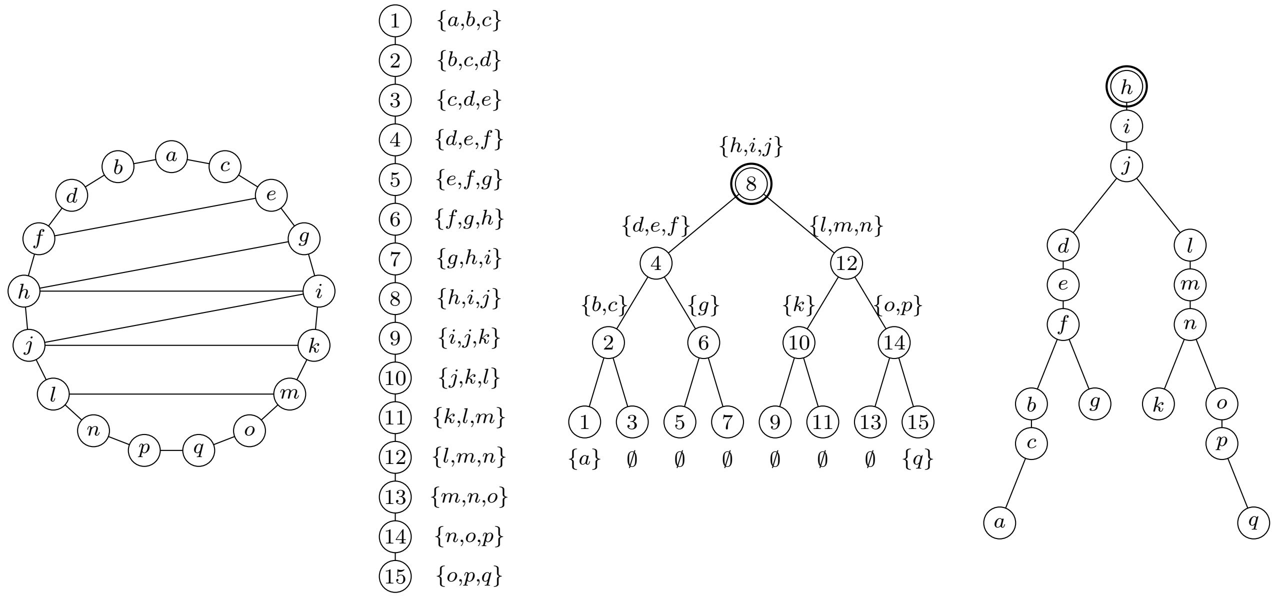

Figure 1: From left to right: G, $(T, \mathcal{W})$, $(S, \mathcal{U})$, R

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 3.5 and 3.6).

Definition 3.2 (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 3.3. Our notion of greedy rooted tree decompositions is defined only for connected graphs for simplicity. However, Definition 3.2 extends naturally to general graphs by considering rooted forests instead of rooted trees.

The same notion appears at least twice in the literature: in [13] under the name *good treedepth decomposition* and in [10] under the name *reduced separation forest*. An even "greedier" class of tree decompositions appears in [15] under the name *minimal rooted trees*. Every minimal rooted tree for a connected graph G (in the sense of [15]) is a greedy rooted tree decomposition of G (in our sense), but not conversely. (The notion of minimal rooted trees would not work for our purposes, as Lemma 3.6 is false with respect to this more restrictive class of tree decompositions.)

The following three lemmas establish the key properties of greedy rooted tree decompositions. (These properties are also noted in [10, 13].) The first lemma establishes that greedy rooted tree decompositions are, in fact, tree decompositions in the sense of Definition 2.1.

Lemma 3.4. *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 3.5. *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 deleting any vertices in $V(T) \setminus V(G)$). Thus, T satisfies conditions (i) and (ii) of Definition 3.2. 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 3.6. *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 $\mathrm{root}(T)$ 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 3.5 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. □

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 for trees. Namely, we show that 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 a few definitions.

Definition 4.1 (The Rooted Minor Relation $\preceq_{\mathrm{rooted}}$)**.** For rooted trees S and T, we say that S is a *rooted minor* of T, denoted $S \preceq_{\mathrm{rooted}} T$, if S is isomorphic to a rooted tree obtained from T by deleting non-root leaves and contracting edges.

Definition 4.2 (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 some useful notation for describing the structure of rooted trees.

Definition 4.3 (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 $T_1, \dots, T_m$ ($m \in \mathbb{N}$), 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 $T_1, \dots, T_m$ ($m \geq 1$), let

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

  That is, $\langle T_1, \dots, T_m \rangle$ 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.

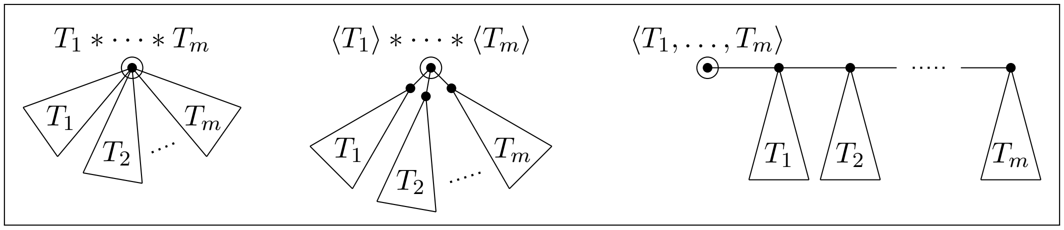

Figure 2

As a matter of notation, let $\langle\rangle$ and $\langle B_0 \rangle$ both denote the rooted tree P_1 (i.e., a single isolated root). Note that for $k, h \geq 1$,

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

(Note: The reader may regard B_0 as the empty tree with 0 vertices; this is not a rooted tree. On the other hand, $\langle B_0 \rangle$ is a rooted tree with 1 vertex.)

Lemma 4.4. *Every rooted tree T has a unique decomposition the form $\langle T_1 \rangle * \cdots * \langle T_l \rangle$ for some $l \in \mathbb{N}$ and rooted trees $T_1, \ldots, T_l$ (unique up to ordering).*

Proof. Straightforward from definitions. Here l is the degree of root(T) and $T_1, \ldots, T_l$ are the subtrees rooted at the children of root(T) (see Figure 1). (Note that $l = 0$ in this decomposition if, and only if, T is an isolated root.) □

The next two lemmas characterize the rooted minor relation in terms of the decomposition given by Lemma 4.4.

Lemma 4.5. *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 4.6. *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 $\langle B_h \rangle$ minors

The next lemmas characterize the structure of rooted trees T that omit binary trees $\langle B_h \rangle$ as rooted minors. (We use these results soon in §4.3.)

Lemma 4.7. *If T is a rooted tree such that $B_h \not\preceq_{\text{rooted}} T$ and $\langle B_h\rangle \not\preceq_{\text{rooted}} T$, then there exist $m \geq 1$ 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. Assume $B_h \not\preceq_{\text{rooted}} T$ and $\langle B_h\rangle \not\preceq_{\text{rooted}} T$ and note that this implies $h \geq 2$ (since $B_1 \preceq_{\text{rooted}} T$). We argue by induction on $|V(T)|$. In the base case $T = P_1$, we set $m := 1$ and $S_1 := T$.

For the induction step, assume $|V(T)| \geq 1$ and let $T = \langle T_1\rangle * \cdots * \langle T_l\rangle$ be the decomposition given by Lemma 4.4. Observe that there exists at most one $i \in [l]$ such that $\langle B_{h-1}\rangle \preceq_{\text{rooted}} \langle T_i\rangle$ (since otherwise we would have $B_h = \langle B_{h-1}\rangle * \langle B_{h-1}\rangle \preceq_{\text{rooted}} T$).

Consider the case that $\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} \langle T_i\rangle$ for all $i \in [l]$. In this case, we have $\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} T$ by Lemma 4.6. Therefore, the condition in the lemma is satisfied with $m := 1$ and $S_1 := T$.

Finally, consider the case that there exists a unique $i \in [l]$ such that $\langle B_{h-1}\rangle \preceq_{\text{rooted}} \langle T_i\rangle$. Without loss of generality, assume $i = l$. Let $S_1 := \langle T_1\rangle * \cdots * \langle T_{l-1}\rangle$ and $T' := T_l$. Observe that $\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} S_1$ and $|V(T)| \geq 1+|V(T')|$ (since $\langle T'\rangle$ is a subtree of T). By the induction hypothesis applied to T', there exists $m \geq 2$ and rooted trees $S_2,\dots,S_m$ such that $T' = S_2 * \langle S_3,\dots,S_m\rangle$ and $\langle B_{h-1}\rangle \not\preceq_{\text{rooted}} S_i$ for all $i \in [m]$. We are now done, as $T = S_1 * \langle T'\rangle = S_1 * \langle S_2,\dots,S_m\rangle$. □

Lemma 4.8. *If T is a rooted tree such that $\langle B_h\rangle \not\preceq_{\text{rooted}} T$, then there exist $m \geq 0$ and $l_1,\dots,l_m \geq 1$ 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. Assume $\langle B_h\rangle \not\preceq_{\text{rooted}} T$. Let $T = \langle T_1\rangle * \cdots * \langle T_m\rangle$ be the decomposition given by Lemma 4.4. 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 4.5 and 4.6. By Lemma 4.7, 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 $T = \langle S_{1,1},\dots,S_{1,l_1}\rangle * \cdots * \langle S_{m,1},\dots,S_{m,l_m}\rangle$. □

4.2 Treedepth bounds

The next lemmas give bounds on the treedepth of the *underlying graph* of a tree T (that is, ignoring the root). These lemmas play a role in the proof of Theorem 1.3 in §5.

Lemma 4.9 ([20, 21])**.** *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 4.10. *For all $m \geq 0$ 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 4.11. *For all $m \geq 0$ 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 $E(T_i) \subseteq E(\mathrm{clos}(T_i'))$. Invoking Lemma 4.9, let T_0' be a rooted tree of height $\lceil \log(m+2) \rceil$ such that $E(P_{k+1}) \subseteq E(\mathrm{clos}(T_0'))$. Label the vertices of P_{m+1} as $v_0, \ldots, v_k$ with v_0 being the root. Let T' be the rooted tree, with root v_0, obtained from the disjoint union of $T_0', \ldots, T_k'$ by identifying vertices v_i and $\mathrm{root}(T_i')$ for each $i \in [m]$. Note that $E(\langle T_1, \ldots, T_k \rangle) \subseteq E(\mathrm{clos}(T'))$ and

$$\mathrm{height}(T') \leq \mathrm{height}(T_0') + \max_{i \in [m]} \mathrm{height}(T_i') = \lceil \log(m+2) \rceil + \max_{i \in [m]} \mathbf{td}(T_i). \qquad \square$$

Lemma 4.12. *For every rooted tree T and $h \geq 0$ and $k \geq 1$, if $\langle B_h \rangle \not\preceq_{\mathrm{rooted}} T$ and $P_k \not\preceq_{\mathrm{rooted}} T$, then*

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

Proof. The lemma is proved 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 $\langle B_h \rangle \not\preceq_{\mathrm{rooted}} T$ and $P_k \not\preceq_{\mathrm{rooted}} T$. By Lemma 4.8, there exist $m \in \mathbb{N}$ and $l_1, \ldots, l_m \in \mathbb{N}$ and rooted trees $S_{i,j}$ ($i \in [m]$, $j \in [l_i]$) such that

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

and $\langle B_{h-1} \rangle \not\preceq_{\mathrm{rooted}} S_{i,j}$ for all $i \in [m]$ and $j \in [l_i]$. We also clearly have $l_i < k$ and $P_k \not\preceq_{\mathrm{rooted}} S_{i,j}$ 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 4.11, we have

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

Finally, by Lemma 4.10, we have

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

Lemma 4.13. *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 4.12, it follows that

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

4.3 Bounded-degree graphs that omit P_k and B_h minors

The final two lemmas of this section bound the size of bounded-degree graphs that omit P_k and B_h minors.

Lemma 4.14. *Let $h, k, c \geq 1$ and suppose T is a rooted tree such that $\langle B_h \rangle \not\preceq_{\mathrm{rooted}} T$ and $P_k \not\preceq_{\mathrm{rooted}} T$ and every vertex of T has at most c children. Then $|V(T)| \leq (ck)^{h-1}$.*

Proof. The lemma is proved by induction on h. In the base case $h = 1$, the condition $\langle B_1 \rangle \not\preceq_{\text{rooted}} T$ implies that T is an isolated root (since $\langle B_1 \rangle = P_2$). Therefore $|V(T)| = 1 = (ck)^{h-1}$.

For the induction step, suppose $h \geq 2$. By Lemma 4.8, there exist $m \in \mathbb{N}$ and $l_1, \ldots, l_m \in \mathbb{N}$ and rooted trees $S_{i,j}$ such that $T = \langle S_{1,1}, \ldots, S_{1,l_1} \rangle * \cdots * \langle S_{m,1}, \ldots, S_{m,l_m} \rangle$ 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 $|V(S_{i,j})| \leq (c(k-1))^{h-2}$ for all i and j. Therefore,

$$\begin{aligned} |V(T)| &= 1 + \sum_{i=1}^{m} \sum_{j=1}^{l_i} |V(S_{i,j})| \\ &\leq 1 + c(k-1)(c(k-1))^{h-2} = 1 + (c(k-1))^{h-1} \leq (ck)^{h-1}. \end{aligned}$$

□

Lemma 4.15. *Let $h, c \geq 1$ and suppose G is a connected graph with maximum degree $\leq c+1$ such that $B_h \not\preceq G$ and $P_{c^h} \not\preceq G$. Then $|V(G)| \leq c^{h^2}$.*

Proof. Let T be any spanning tree of G rooted at any of its leaves. Since G has maximum degree $c+1$, every vertex has at most c children in T. The assumption that $P_{c^h} \not\preceq G$ and $B_h \not\preceq G$ implies that $P_{c^h} \not\preceq_{\text{rooted}} T$ and $\langle B_h \rangle \not\preceq_{\text{rooted}} T$. Therefore, by Lemma 4.14,

$$|V(G)| = |V(T)| \leq (c^{h+1})^{h-1} \leq c^{h^2}.$$

□

5 Proof of Theorem 1.3

In this section, we prove Theorem 1.3 by showing the following stronger result.

Theorem 5.1. *Every graph G contains a path of order 2^h or has a B_h-minor where*

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

(Obs: Note that the ratio r is at most 1 by inequality (1).)

Proof of Theorem 1.3 assuming Theorem 5.1. Let $k \geq 1$ and suppose G is a graph of treewidth $< k$ which does not contains a path of order 2^k nor a B_k-minor. Theorem 5.1 implies

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

It follows that $\mathbf{td}(G) \leq O(k^5 \log^2 k)$. □

The rest of this section is devoted the proof of Theorem 5.1.

Proof of Theorem 5.1. It clearly suffices to prove the theorem for connected graphs G. Let G be any connected graph and let $r = \mathbf{td}(G)/(\mathbf{tw}(G)+1)$. We must show that G contains a path of length 2^h or a B_h-minor where $h = \Omega(r^{1/4}/\log^{1/2}(\mathbf{tw}(G)+1))$.

By Lemma 3.6, we may fix a greedy rooted tree decomposition T of width $\mathbf{tw}(G)$ for G. By Lemma 3.1, 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 $V(S) \subseteq V(Q)$. By Lemma 4.15, we 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 3.2, there exists a function $x \mapsto \breve{x} : V' \to V'$ such that $\breve{x} \le_T x$ 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 $\breve{x} \le_T x$ 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 $|V(G)| \ge 2$ so that $x \in V'$) we have $\breve{x} = x$ (this is forced by the requirement $\breve{x} \le_T x$) and therefore $\{\breve{x}, \widehat{x}\} \in E(F)$ and hence $\breve{x} \in V(F)$. ◄Claim 1

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 $u \le_T x <_T v$), or $u = \widehat{x}$ and $v \in \breve{x}$ (in which case $v \le_T x <_T u$). ◄Claim 2

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

► 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 p_i be the unique $<_T$-maximum element in $\{p_0, \dots, p_t\}$. Toward a contradiction assume $i \neq 0$ (that is, $p_i \neq \widehat{x}$). Then $i \in \{1, \dots, t-1\}$ 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 $p_0, \dots, p_{i-1} \le_T u$ and $p_{i+1}, \dots, p_t \le_T w$.[3] Since $\widehat{x} = p_0$ and $x = p_t$, this means that $\widehat{x} \le_T u$ and $x \le_T w$. 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. ◄Claim 3

The rooted subtree $S \subseteq T$:

- By Lemma 4.13, 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 $\le \mathbf{tw}(G) + 2$.

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

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 $\{P_x \subseteq F\}_{x\in W'}$ 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., $\{w_1, \dots, w_k\} = \{w \in W : wx \in \vec{E}(S)\}$ where x may have additional children in T) such that $Q_{w_1}, \dots, Q_{w_k}$ and $P_{w_1}, \dots, P_{w_k}$ are already defined. (Obs: $k \leq 2$ since S is subcubic.) 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 $\{x^\star \in V(P_x) \cap V(Q_x)\}_{x\in W'}$:

- Finally, let $Q = Q_{\mathrm{root}(S)}$. Note that $Q = \bigcup_{x\in W'} P_x$ by the above definition.
- For each $x \in W'$, let $x^\star$ be the unique element in $V(P_x) \cap V(Q_x)$. (That is, $x^\star$ 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 $x^\star$ (the first vertex in $V(Q_x)$ encountered on the unique path in F from $\widehat{x}$ to x).

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

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

$$\begin{aligned}\deg_Q(q) &= \sum_{x\in W'} \deg_{P_x}(q) \\ &= 2\cdot|\{x \in W' : \deg_{P_x}(q) = 2\}| + |\{x \in W' : q = \widehat{x}\}| + |\{x \in W' : q = x^\star\}|.\end{aligned}$$

We now note:

- $|\{x \in W' : q = \widehat{x}\}| \leq 2$ (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 from these observations. ◀Claim 4

Claim 5. *For all $q \in V(Q)$, we have $|\{x \in W' : q = x^\star\}| \le |\mathrm{Bag}_{T,G}(q)| - 1 \le \mathbf{tw}(G)$.*

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

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

For each $x \in W'$ such that $q = x^\star$, define the set

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

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 $p_i \le_T q$. Then $p_i = \breve{p}_{i-1}$ and $p_i \le_T q <_T p_{i-1}$. Therefore, U_x is nonempty.
- We have $V(P_x) \cap V(P_y) = \{q\}$ for all distinct $x, y \in W'$ such that $q = x^\star = y^\star$. Therefore, U_x and U_y are disjoint.

It follows from these three observations that $|\{x \in W' : q = x^\star\}| \le |\mathrm{Bag}_{T,G}(q)| - 1$. Finally, recall that T was chosen such that $\mathrm{width}(T, G) = \max_{x \in V(G)} |\mathrm{Bag}_{T,G}(x)| - 1 = \mathbf{tw}(G)$. ◀Claim 5

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

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

It now follows from Lemma 4.15 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 5.1. □

6 Algorithmic Results

In this section, we describe the algorithmic versions of our main results. From the constructive nature of the proof of Theorem 5.1, we have the following

Corollary 6.1 (Algorithmic Version of Theorem 5.1)**.** *There is a polynomial-time algorithm which, given a graph G and a width-w tree decomposition of G, outputs a minor embedding of either P_{2^h} or B_h where*

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

Results of Bodlaender et al [9, 8] give a polynomial-time algorithm which, given a graph G, outputs a tree decomposition of G of width $O(\mathbf{tw}(G)^2)$. (This is actually a combination of two polynomial-time approximation algorithms for treewidth: an $O(\log n)$-approximation for arbitrary n-vertex graphs G [9] and a 5-approximation algorithm in the case where $\mathbf{tw}(G) \le \log n$ [9].) Combining this algorithm with Corollary 6.2, we get

Corollary 6.2. *There is a polynomial-time algorithm which, given a graph G, outputs a minor embedding of P_{2^h} or B_h where*

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

To obtain the algorithmic version of Theorem 1.2, we combine Corollary 6.2 with the randomized polynomial-time algorithm of Chekuri and Chuzhoy [12] which, given a graph G, a minor embedding of the $k \times k$ grid where $k = \mathbf{tw}(G)^{\Omega(1)}$.

Corollary 6.3 (Algorithmic Version of Theorem 1.2)**.** *There is a randomized polynomial-time algorithm which, given a graph G, outputs a minor embedding of one of the following graphs where $k = \mathbf{td}(G)^{\Omega(1)}$:*

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

The algorithm of Corollary 6.3 finds a $k \times k$ grid minor via the Chekuri-Chuzhoy algorithm and a P_{2^h} or B_h minor via Corollary 6.2. It outputs the grid minor if $k > h$ and the P_{2^h} or B_h minor otherwise.

7 Applications in Complexity and Logic

The main result of this paper, Theorem 1.3, 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 [18], paths [27] and trees [29], we obtain an $n^{\mathrm{poly}(\mathbf{td}(G))}$ 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 [28] for details.

7.1 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 7.1. 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

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

Definition 7.2. 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 [18] shows that the complexity of $\mathrm{SUB}(G)$ is a minor-monotone graph invariant.

Lemma 7.3. *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 7.3, the function $G \mapsto \chi(\mathrm{SUB}(G,n))$ is a minor-monotone graph invariant 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 7.4 (AC^0 lower bounds)**.**

1. $\mathrm{SUB}(G)$ *has* AC^0 *circuit size* $n^{\Omega(\mathbf{tw}(G)/\log \mathbf{tw}(G))}$ *for all graphs G* [18].
2. $\mathrm{SUB}(P_k)$ *has* AC^0 *formula size* $n^{\Omega(\log k)}$ *for all k* [27].
3. $\mathrm{SUB}(B_k)$ *has* AC^0 *formula size* $n^{k^{\Omega(1)}}$ *for all k* [29].

Combining Theorem 1.3 with the three lower bounds in Theorem 7.4, and using the fact that the AC^0 formula size of $\mathrm{SUB}(G)$ is minor-monotone by Lemma 7.3, we get the following:

Theorem 7.5. *There is an absolute constant $\varepsilon > 0$ such that* $\mathrm{SUB}(G)$ *has* AC^0 *formula size* $n^{\Omega(\mathbf{td}(G)^{\varepsilon})}$ *for all graphs G.*

Theorem 1.3(1) and Theorem 7.5 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.

7.2 An Improved Homomorphism Preservation Theorem on Finite Structures

Theorem 7.5 turns out to have a surprising corollary in finite model theory. The following result was proved in [28].

Theorem 7.6. *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 7.6 is based on a reduction to the AC^0 formula of $\mathrm{SUB}(G)$ and relies on Theorem 7.5 (and hence on Theorem 1.3) for the polynomial bound on the quantifier-rank of ψ. Theorem 7.6 dramatically improves an earlier result in [26], 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).

8 Open Questions

In light of Theorems 1.1 and 1.2, we conjecture the following "Polynomial Grid/Tree-Minor Theorem for Pathwidth":

Conjecture 8.1. *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 $O(k^5 \log^2 k)$ bound in Theorem 1.3. The optimal bound is likely smaller; however, examples show one cannot do better than $O(k^2)$.

Acknowledgements. We are grateful to the anonymous referees for their close reading of this paper and many helpful comments and corrections.

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. An $O(c^k n)$ 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] Mikołaj Bojańczyk and Michał Pilipczuk. Optimizing tree decompositions in MSO. In *34th Symposium on Theoretical Aspects of Computer Science*, 2017.

[11] 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.

[12] 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.

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

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

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

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

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

[18] 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.

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

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

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

[22] Jaroslav Nešetřil and Patrice Ossona de Mendez. On low tree-depth decompositions. *Graphs and combinatorics*, 31(6):1941–1963, 2015.

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

[24] 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.

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

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

[27] 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.

[28] 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.

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

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

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