Trade-Off Between Spread and Width for Tree Decompositions

Hans L. Bodlaender
Department of Information and Computing Sciences, Utrecht University, The Netherlands

Carla Groenland
Faculty of Electrical Engineering, Mathematics and Computer Science, Technical University Delft, The Netherlands

Abstract

We study the trade-off between (average) spread and width in tree decompositions, answering several questions from Wood [arXiv:2509.01140]. The spread of a vertex v in a tree decomposition is the number of bags that contain v. Wood asked for which $c > 0$, there exists c' such that each graph G has a tree decomposition of width $c\,\mathrm{tw}(G)$ in which each vertex v has spread at most $c'(d(v)+1)$. We show that $c \geq 2$ is necessary and that $c > 3$ is sufficient. Moreover, we answer a second question fully by showing that near-optimal average spread can be achieved simultaneously with width $O(\mathrm{tw}(G))$.

2012 ACM Subject Classification Mathematics of computing → Trees; Theory of computation → Fixed parameter tractability

Keywords and phrases Tree decomposition, spread, domino treewidth

Digital Object Identifier 10.4230/LIPIcs.WG.2026.9

Related Version *Preprint*: `https://arxiv.org/abs/2601.04040` [8]

Funding *Carla Groenland*: Research supported by the Dutch Research Council (NWO, VI.Veni. 232.073).

1 Introduction

Tree decompositions have been widely studied in both structural and algorithmic contexts, for example due to their importance to graph minor theory [27] and the fact that many problems can be solved in linear time on graphs of bounded treewidth [2, 11].

The main criterion of "optimality" for a tree decomposition is usually its width, or, equivalently, the size of the largest bag in the tree decomposition. Recently, Wood [30] proved several results concerning tree decompositions which have additional nice properties, while not sacrificing too much of the width. Our work continues on this, answering several of the problems posed by Wood.

Besides the width, our first parameter of interest is the spread. The *spread* of a vertex v in a tree-decomposition $(T, (B_x)_{x \in V(T)})$ is the number of bags v is in: $|\{x \in V(T) : v \in B_x\}|$. Since all neighbours of a vertex v must appear together with v in some bag, when the width is held constant, the spread of v must necessarily grow with its degree.

Wood [30] posed the following problem.

▶ **Problem 1.** *What is the infimum of c such that for some c', every graph G with treewidth k has a tree-decomposition with width at most $(c + o(1))k$, in which each vertex $v \in V(G)$ has spread at most $c'(d(v) + 1)$?*

Wood [30] already proved that the answer is at most 14 (with $c' = 1$).

▶ **Theorem 2** (Theorem 2 in [30]). *Every graph G with treewidth k has a tree-decomposition with width at most $14k + 13$, such that each vertex $v \in V(G)$ has spread at most $d(v) + 1$.*

52nd International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2026).
Editors: Jan Goedgebeur and Paweł Rzążewski; Article No. 9; pp. 9:1–9:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

We improve the lower and upper bound for Problem 1 with the following two results, which show that the value of c of Problem 1 is in the interval $[2, 3]$.

▶ **Theorem 3.** *For each $c > 3$, there is a constant $c' > 0$ such that every graph G with treewidth k has a tree-decomposition with width at most ck, in which each vertex $v \in V(G)$ has spread at most $c'(d(v) + 1)$*

▶ **Theorem 4.** *For each $c < 2$ and for each $c' > 0$, there is a graph G of treewidth k such that in every tree-decomposition with width at most ck, some vertex $v \in V(G)$ has spread strictly more than $c'(d(v) + 1)$.*

Wood also proposed a lower bound example for Problem 1, conjecturing there are no constants $\varepsilon, c > 0$ such that the $(n \times n)$-grid has a tree-decomposition of width at most $(2 - \varepsilon)n$ and spread at most c. This conjecture is false and in fact for any $m \geq n \geq 1$ and $\varepsilon > 0$, the $(n \times m)$-grid admits a tree-decomposition of width at most $(1 + \varepsilon)n$ in which each vertex has spread at most $O(1/\varepsilon)$ (see Lemma 10).

When storing a tree-decomposition $(T, (B_x)_{x \in V(T)})$ of a graph G, the memory required depends on $\sum_{x \in V(T)} |B_x|$. Our second parameter of interest is the average spread $\frac{1}{|V(G)|} \sum_{x \in V(T)} |B_x|$. Regarding this, Wood [30] raised the following problem.

▶ **Problem 5.** *What is the infimum of the $c' \in \mathbb{R}$ such that for some c every graph G has a tree-decomposition of width at most $c\,\mathrm{tw}(G)$ and average spread at most c'?*

Wood already proved the upper bound 3. We show that the answer to this problem is 1.

▶ **Theorem 6.** *Let $c' > 1$. Then there is a constant $c > 0$ such that every graph G has a tree-decomposition of width at most $c\,\mathrm{tw}(G)$ and average spread at most c'.*

Related work: domino treewidth, tree-partition-width and persistence

Bienstock [1] asked whether it is possible for each graph G to find a tree decomposition $(T, (B_t)_{t \in V(T)})$ in which T is a binary tree and the width as well as the maximum diameter of T_v (the subtree of T induced on the bags containing v, for $v \in V(G)$), are bounded by a constant depending only on $\mathrm{tw}(G)$ and $\Delta(G)$. Ding and Oporowski [12] answered this question positively in a stronger fashion, providing the first bounds for spread that we are aware of.

▶ **Theorem 7** (Theorem 1.3 in [12]). *Every graph G has a tree decomposition of width at most $2^{\mathrm{tw}(G)+1}(\mathrm{tw}(G) + 1) - 1$ such that each vertex $v \in V(G)$ has spread at most $1 + 2d(v)3^{2^{\mathrm{tw}(G)}}$*

Ding and Oporowski [12] also studied a special case of tree decompositions with bounded spread, namely those where each vertex has spread at most 2. They showed that each graph admits such a tree decomposition of width at most $O(\mathrm{tw}(G)^2\Delta(G)^3)$, where $\mathrm{tw}(G)$ and $\Delta(G)$ denote the treewidth and maximum degree of the graph G respectively. Bodlaender and Engelfriet [6] first used the term *domino treewidth* for the optimal width of a tree decomposition in which each vertex has spread 2. We write $\mathrm{dtw}(G)$ for the domino treewidth of G. Bodlaender [4] improved the upper bound on domino treewidth to $\mathrm{dtw}(G) = O(\mathrm{tw}(G)\Delta(G)^2)$ and gave examples of graphs G with $\mathrm{dtw}(G) = \Omega(\mathrm{tw}(G)\Delta(G))$.

A *tree-partition* of a graph G is a partition of $V(G)$ into (disjoint) bags $(B_i)_{i \in V(T)}$, where T is a tree, such that $uv \in V(G)$ implies that the bags of u and v are the same or adjacent in T. The width is the size of the largest bag, and the *tree-partition-width* $\mathrm{tpw}(G)$ of G is found by taking the minimum width over all tree-partitions of G. The notion was first introduced by Seese [28] under the name *strong treewidth*.

▶ **Observation 8.** *For any graph G,* $\operatorname{tpw}(G) - 1 \leq \operatorname{dtw}(G) \leq (\Delta + 1)\operatorname{tpw}(G)$.

The upper bound follows because after rooting the tree T of a tree-partition of G, we may add a vertex v to its parent bag B_p (where p is the parent of c and $v \in B_c$) whenever v has a neighbour in B_p. This gives a domino tree decomposition of at most $\Delta + 1$ times the width. For the lower bound, note that after rooting a domino tree decomposition, we can remove each vertex from a bag whenever it is not the bag closest to the root containing it. This yields a tree-partition, since for any edge uv, at least one of the two is still present in the highest bag they originally shared. (The construction was given by Bodlaender and Engelfriet [6].)

It was first shown that $\operatorname{tpw}(G) \leq 24(\operatorname{tw}(G) + 1)\Delta(G)$ by an anonymous referee of [12] and the constant was later improved (to approximately 17.5) by Wood [29]. The lower bound $2\operatorname{tpw}(G) \geq \operatorname{tw}(G) + 1$ was shown by Seese [28]. Tree-partitions for which the underlying tree has bounded maximum degree have been studied as well by Distel and Wood [14].

Bodlaender, Groenland and Jacob [9] showed that Domino Treewidth with the domino treewidth as parameter is complete for the parameterized complexity class XALP: deciding if a given graph has a tree decomposition of width at most k and spread at most 2 is XALP-complete. They proved a similar result for Tree-Partition-Width and provided an FPT-approximation algorithm for this as well.

The parameter *treespan* can be defined in terms of the occupation time in a graph searching game [17] and is polynomially tied to domino treewidth (see [21, Theorem 54]).

Downey and McCartin [15, 16] also studied the notion of spread for path decompositions under the name *persistence*. They motivated this notion as being a more natural notion for a “path-like” graph, particularly for online computational problems. They showed amongst others that deciding if a given graph has a path decomposition of given width and persistence is W[t]-hard for all t via a reduction from Bandwidth, with the sum of the width and persistence as parameter. This reduction can be easily turned into an XNLP-hardness proof, using the recent result by Bodlaender, Groenland, Nederlof and Swennenhuis [10] that Bandwidth is XNLP-complete. An XNLP-membership proof can be done in the same fashion as the XNLP-membership of Bandwidth [10]. Thus Bounded Persistence Pathwidth with sum of width and persistence as parameter is XNLP-complete.

2 Preliminaries

We use the short-cut notation $[n] = \{1, \ldots, n\}$. A *tree decomposition* of a graph G is a pair $(T, (B_t)_{t \in V(T)})$ with T a tree and *bags* $B_t \subseteq V(G)$ for each $t \in V(T)$ such that

- $\bigcup_{t \in V(T)} B_t = V(G)$,
- for all edges $vw \in E(G)$, there is a t with $v, w \in B_t$, and
- for all v, the nodes $\{t \in V(T) \mid v \in B_t\}$ form a subtree (denoted T_v) of T.

The *width* of the decomposition is $\max_{t \in V(T)} |B_t| - 1$, and the *treewidth* of a graph G is the minimum width over all tree decompositions of G. A *path decomposition* is a tree decomposition $(T, (B_t)_{t \in V(T)})$ for which T is a path and the *pathwidth* is the minimum width over all path decompositions of G. The *spread* of a vertex v in a tree-decomposition $(T, (B_t)_{t \in V(T)})$ is the number of bags v is in: $|\{t \in V(T) : v \in B_t\}|$.

The following lemma is standard (see e.g. [30, Corollary 9]).

▶ **Lemma 9** (Folklore). *Let $(T, (B_t)_{t \in V(T)})$ be a tree decomposition of G. Let $w : V(G) \to \mathbb{R}_{\geq 0}$ be a weight function. Then there is $t \in V(T)$ such that for each component C of $G - B_t$, we find that $w(C) \leq w(V(G))/2$.*

3 Grid graphs and extensions

The $(n \times m)$-grid is the graph on vertex set $[n] \times [m]$ and edge set

$$\{(i,j)(i+1,j) : i \in [n-1]\} \cup \{(i,j)(i,j+1) : j \in [m-1]\}.$$

We first provide a near-optimal width tree decomposition for grids with constant spread.

▶ **Lemma 10.** *Let $c > 1$ and let $n, m \in \mathbb{N}$. Then the $(n \times m)$-grid has a path decomposition of width cn and spread at most $\lceil 1/(c-1) \rceil + 1$.*

We provide a "proof by picture" in Figure 1 and leave a formal proof until Section 6.

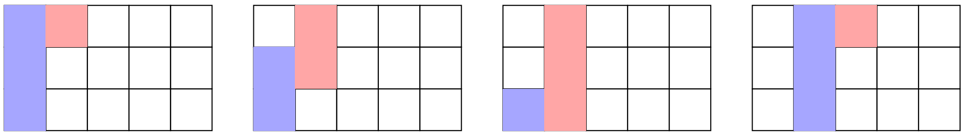

Figure 1 Illustration of the construction in the proof of Theorem 10.

Figure 1 above shows four consecutive bags for $c = 1 + 1/3$ via the coloured regions. Each column denotes a single column of the $(n \times m)$-grid, whereas any row in the picture corresponds to the union of at most $\lceil n/3 \rceil$ consecutive rows of the grid.

Let $D_{n,m}$ denote the graph obtained from the $(n \times m)$-grid by adding the same diagonal edge in each square, that is, we add the edges

$$\{(i,j)(i+1,j+1) : i \in [n-1], j \in [m-1]\}.$$

When $n = m$, this can also be seen as the dual graph of the hexagonal grid.

As depicted in Figure 2, we also define the *east*, *west*, *north* and *south* sets as

$$E = \{(i,m) : i \in [n]\},$$
$$W = \{(i,1) : i \in [n]\},$$
$$N = \{(1,j) : j \in [m]\},$$
$$S = \{(n,j) : j \in [m]\}.$$

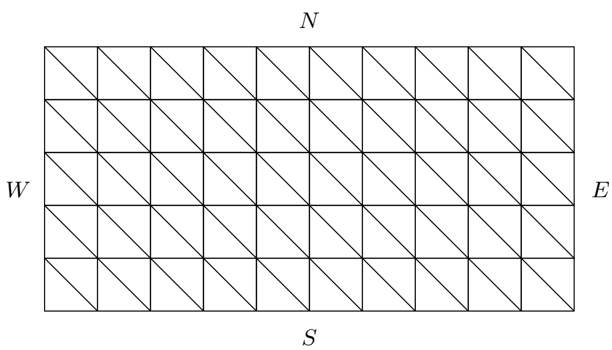

Figure 2 $D_{6,11}$ with west, north, east and south sets.

The following result is well-known (see e.g. [18] for the case $n = m$ and the introduction of [20] for a sketch of the result below).

▶ **Theorem 11** (Variant of Hex theorem). *Let $n, m \geq 1$ be integers. For any two sets H and V covering $V(D_{n,m})$, either there is a path in $D_{n,m}[H]$ from W to E or there is a path in $D_{n,m}[V]$ from N to S.*

We call such a path from N to S in the theorem statement above a *north-south-path*. From this result, we obtain the following corollary.

▶ **Corollary 12.** *If $A \subseteq V(D_{n,m})$ does not contain a north-south-path, then $D_{n,m} - A$ has a connected component of size at least $n(m - |A|)$.*

Proof. We apply Theorem 11 with $V = A$ and $H = V(D_{n,m}) \setminus A$ to find that there is a connected component C of $D_{n,m} - A$ which contains a vertex on every column of $D_{n,m}$. Since the set A contains vertices in at most $|A|$ columns, for any other column j, the vertices $\{(i,j) : i \in [n]\}$ in this column will all be in C. Hence $|C| \geq n(m - |A|)$. ◀

Here it becomes clear why it is convenient to move to unbalanced grids: when m is much larger than n, any "balanced" separator now must contain a north-south-path.

Let $m = n^8$ and split $[m]$ into $b = n^4$ consecutive intervals $J_1, \dots, J_b$ of length $m' = n^4$. For each $i \in [b]$, we choose $y_i \in J_i$. Let $D^+_{n,m}$ be the graph obtained from $D_{n,m}$ by adding a vertex v_0 with neighbourhood

$$\{(1, y_1), (1, y_{n^2+1}), \dots, (1, y_{b-n^2+1})\}.$$

In particular, $d(v_0) = b/n^2 = n^2$. We call v_0 the *special vertex*. A visualisation is given in Figure 3.

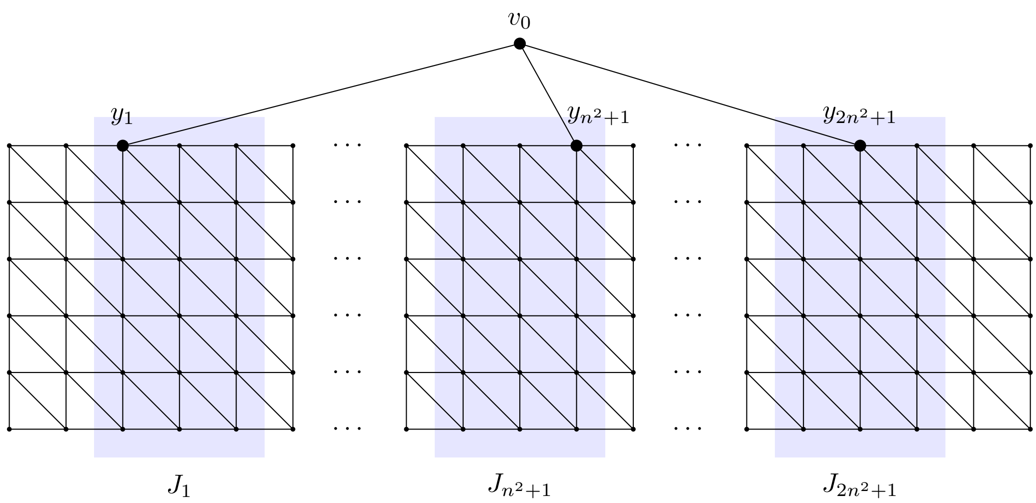

Figure 3 A visualisation of $D^+_{n,m}$.

▶ **Lemma 13.** *For n sufficiently large, with $m = n^8$, for any tree decomposition $(T, (B_x)_{x \in V(T)})$ of $D^+_{n,m}$ of width at most $2n - 2$, the special vertex v_0 has spread at least $\frac{1}{2}n^2(d(v_0) + 1)$ in the tree decomposition.*

Proof. Let $(T, (B_x)_{x \in V(T)})$ be a tree decomposition of $D^+_{n,m}$ of width at most $2n - 2$. Let v_0 denote the special vertex of $D^+_{n,m}$.

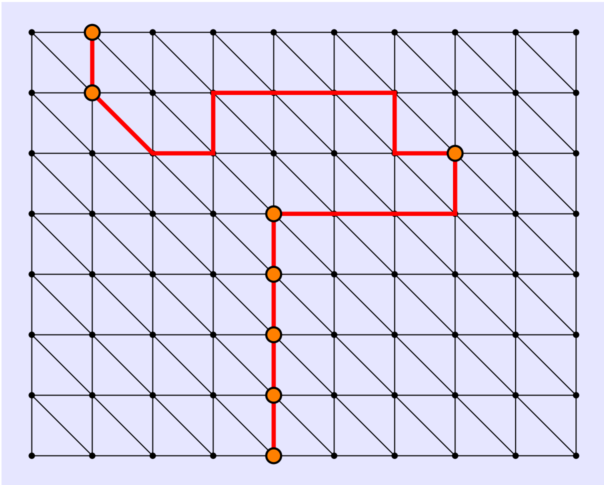

Figure 4 The figure shows a potential north-south path (in red) in J_a and highlights the vertices chosen in X_a (in orange).

For $a \in [b]$, let $w_a : V(D^+_{n,m}) \to \{0,1\}$ be defined by $w_a((i,j)) = 1$ if $j \in J_a$ and 0 otherwise. For $a \in [b]$, applying Lemma 9 with w_a as weighting, provides $x(a) \in V(T)$ such that each component C of $D^+_{n,m} - B_{x(a)}$ satisfies $w_a(C) \leq w_a(V(G))/2$. Since for n sufficiently large,

$$n(m' - |B_{x(a)}|) \geq n(n^4 - (2n-1)) > n^5/2 = nm'/2,$$

Corollary 12 applied to the copy of $D_{n,m'}$ induced on $[n] \times J_a$, shows that $B_{x(a)}$ must contain a north-south-path which stays within $[n] \times J_a$. Let $X_a \subseteq [n] \times J_a$ denote n vertices in $B_{x(a)}$ chosen so that each $i \in [n]$ occurs exactly once as first coordinate. A visualisation is given in Figure 4.

Let P denote the path in T between $x(1)$ and $x(b)$. For each $a \in [b]$, we let $y(a)$ denote the node in P closest to $x(a)$. (For example, $y(1) = x(1)$ and $y(b) = x(b)$.)

▷ Claim 14. The nodes $y(a)$ are distinct and appear in the order $y(1), \ldots, y(b)$ in P.

Proof. First, we exclude the option of the existence of $1 \leq i_1 < i_2 < i_3 \leq b$ such that the node $x(i_3)$ lies on the path between $x(i_1)$ and $x(i_2)$ in T. Within $D_{n,m}$, there are n vertex disjoint paths from X_{i_1} to X_{i_2} that do not pass through X_{i_3}. In particular, $B_{x(i_3)}$ now must contain one vertex for each such path, plus the n vertices in X_{i_3}. But then $|B_{x(i_3)}| \geq 2n$, contradicting the fact that all bags have size at most $2n-1$. A similar argument applies to exclude the possibility that $x(i_1)$ lies in between $x(i_2)$ and $x(i_3)$.

Since the $x(a)$ must be distinct (by the width assumption), the claim above shows that for each $a \in \{2, \ldots, b-1\}$, the node $y(a)$ must lie strictly between $y(1)$ and $y(b)$ on P.

We next show that all $y(a)$ are distinct. Suppose that there are $1 < a < a' < b$ with $y = y(a) = y(a')$. Then B_y is a vertex separator between $2n$ vertex-disjoint paths, n from X_1 to X_a and n from $X_{a'}$ to X_b. This contradicts the width assumption since that would imply that $|B_y| \geq 2n$.

Finally, we exclude the option that for some $1 < a < a' < b$, the nodes appear in the wrong order: $y(1), y(a'), y(a), y(b)$. Again, both $y(a)$ and $y(a')$ are now a vertex separator between $2n$ vertex-disjoint paths, n from X_1 to X_a and n from $X_{a'}$ to X_b.

So the nodes $y(1), \ldots, y(b)$ are distinct and appear in this particular order. ◁

Next, we show that

$$v_0 \in B_{y(n\cdot n^2+1)} \cap B_{y((n+1)n^2+1)} \cap \cdots \cap B_{y(b-n\cdot n^2+1)}. \tag{1}$$

Let $z(1), \ldots, z(b')$ denote all nodes of P chosen in order with $z(1) = y(1)$. If the equation above fails, then there are two options. First, if $v_0 \notin B_z$ for all $z \in V(P)$, then let $z(i_0)$ denote the (unique) closest node in P to a bag containing v_0. Second, if $v_0 \in B_z$ for some $z \in V(P)$, then let $z_{\min}$ (respectively $z_{\max}$) be the $z \in [b']$ closest to $y(1)$ (respectively $y(b)$) such that $v_0 \in B_z$. If $z_{\min}$ appears before $y(n \cdot n^2 + 1)$ (or is equal to it) and $z_{\max}$ appears after $y(b - n \cdot n^2 + 1)$ (or is equal to it), then (1) holds. If $z_{\min}$ appears strictly after $y(n \cdot n^2 + 1)$ on P then we define $z(i_0) = z_{\min}$. Otherwise, $z_{\max}$ must appear strictly before $y(b - n \cdot n^2 + 1)$ and we define $z(i_0) = z_{\max}$.

Either way, $z(i_0)$ is defined such that it either appears strictly after $y(n \cdot n^2 + 1)$ on P and is the "first" node closest to a bag containing v_0, or strictly before $y(b - n \cdot n^2 + 1)$ and is the "last" such node. Our argument will be analogous in both cases and we will assume henceforth that the former holds.

Let $j \in [b]$ be the smallest such that $y(j) = z(i_0)$ or $y(j)$ appears after $z(i_0)$. By assumption, $j > n \cdot n^2 + 1$. For $\ell \in \{0, 1, \ldots, n-1\}$, we choose a vertex $x_\ell \in X_{\ell n^2+1}$ and define a path P_ℓ within $J_{\ell n^2+1} \cup \{v_0\}$ running from v_0 to x_ℓ via the neighbour $(1, y_{\ell n^2+1})$ of v_0. The paths P_ℓ intersect only in v_0 and are disjoint from $J_{j-1} \cup J_j \cup J_{j+1}$. There are n vertex-disjoint paths from X_{j-1} to X_{j+1} entirely contained in $J_{j-1} \cup J_j \cup J_{j+1}$ (that keep their coordinate fixed) and since $z(i_0)$ lies on the path between $x(j-1)$ and $x(j+1)$ (by definition of j), $B_{z(i_0)}$ contains at least one vertex from each of these n paths. Since $B_{z(i_0)} \cap B_{z(i_0-1)}$ is a vertex separator between X_ℓ and $\{v_0\}$ which does not contain v_0, for each $\ell \in \{0, 1, \ldots, n-1\}$, $B_{z(i_0-1)} \cap B_{z(i_0)}$ must contain a vertex of $P_\ell \setminus \{v_0\}$. Using the disjointness constraints, we obtain $2n$ different vertices in $B_{z(i_0)}$, contradicting the upper bound on the width.

In conclusion, we established that v_0 is present in at least $b - 2(n+1)n^2 = n^4 - 2(n+1)n^2$ bags while its degree is $b/n^2 = n^2$. For n sufficiently large, the spread of v_0 is at least $n^4 - 2(n+1)n^2 > \frac{1}{2}n^2(n^2+1) = \frac{1}{2}n^2(d(v_0)+1)$. ◀

The lemma above implies Theorem 4 for all $c < 2$ since $D^+_{n,m}$ has treewidth at most $n+2$ (since adding v_0 to all bags in a tree decomposition of $D_{n,m}$ provides a tree decomposition for $D^+_{n,m}$).

4 Well-behaved spread using thrice the width

In this section we prove Theorem 3 which is immediately implied by the result below.

▶ **Theorem 15.** *Let $b \geq 5$ be an integer. Every graph G admits a tree decomposition of width at most $(3 + 8/b)(\mathrm{tw}(G) + 1)$ for which each vertex $v \in V(G)$ is in at most $2b(d(v) + 1)$ bags.*

A key idea in the best bound of 14 from Wood [30] is the definition of a *slick* tree decomposition $(T, (B_x)_{x \in V(T)})$, in which the tree T is rooted and the following property holds: if a vertex $v \in V(G)$ is in $B_x \cap B_y$ where x is the parent of y, then v has a neighbour in $B_y \setminus B_x$. This property automatically ensures that each vertex has spread at most $d(v) + 1$.

Using separators, a rooted tree decomposition can be constructed using induction on the number of vertices in the graph. After the first separator S is obtained, a recursive call can be placed on the components of $G - S$; the aim is then to glue together the tree decompositions obtained via recursion using a bag containing S. At the next recursion depth, we are given a special set S that needs to become part of the root bag (thinking ahead of our gluing operation), and we find a "balanced" separator X for S, passing on X and parts of S to the next recursion depth as special set. Such an approximation scheme for treewidth has been used since the 1990s [7, 26], see [5] for a recent overview.

Wood [30] cleverly noticed that in this routine, before passing a special set S' to a recursive call, we can also replace it by a set S where we add some neighbours of vertices in S in order to ensure the tree decomposition will become slick. In order to keep the bag size smaller compared to their approach, we will only add neighbours for a "negligible" fraction of the vertices, while ensuring all vertices have a neighbour added once-in-a-while to upper bound the spread.

To facilitate the reasoning about the spread of the vertices, we introduce the notion of a *b-marking* for a rooted tree decomposition $(T,(B_t))$ of G. This is a partial function $m : V(G) \times V(T) \to \{1,\dots,b\}$ such that:

- $m(v,t)$ is defined if and only if $v \in B_t$;
- if v is in $B_c \cap B_p$ for c the child of p in T, then either $m(v,c) < m(v,p)$ or v has a neighbour in B_c which it does not have in B_p;
- if $v \in B_\ell$ and ℓ does not have a child c with $v \in B_c$ (and ℓ is not the root) then v has a neighbour in B_ℓ that it does not have in the parent bag of ℓ.

Note that this last property can always be ensured, since if T_v denotes the subtree of T consisting of t with $v \in B_t$, and ℓ is a leaf of T_v, then we may simply remove v from B_ℓ if v has no neighbours in B_ℓ.

The use of this notion is via the following observation.

▶ **Lemma 16.** *If a rooted tree decomposition admits a b-marking, then each vertex v is in at most $b(d(v)+1)$ bags.*

Proof. Let $(T,(B_t)_{t\in V(T)})$ be a tree decomposition of G with b-marking m. Let $v \in V(G)$. Let T_v be the subtree of T corresponding to the bags containing v. We call a node t of T_v *happy* if there is a neighbour of v in B_t which was not in B_p for p the parent of t. (The root is also happy.) Note that:

- each leaf node of T_v is happy (by the third marking property);
- any rooted path with b vertices has at least one happy vertex.

The second property holds since such a path $t_1,\dots,t_b$ must pass from parent to child in each step and therefore the marking goes down by 1, or a happy node occurs, by the second marking property. We use this to show that the fraction of nodes in $V(T_v)$ that is happy is at least $1/b$. Since the number of non-root happy nodes is at most the number of neighbours of v (as u is the reason a $t \in V(T_v)$ is happy only if t is the closest vertex to the root with $u \in B_t$), this shows that $|V(T_v)| \le b(d(v)+1)$.

The remainder of the proof is straightforward. We initialise $T' = T_v$ and iteratively shrink it as follows. Starting from a leaf v_1 (which is happy by definition), we take a longest rooted path $v_k, v_{k-1},\dots,v_2,v_1$ such that $v_2,\dots,v_k$ are unhappy (allowing for $v_k = v_1$). In particular, the parent v_{k+1} of v_k (which could be the root) is happy and $v_k,\dots,v_2$ forms a rooted path of unhappy vertices, and so $k \le b$. We change T' by removing $v_1,\dots,v_k$ and attaching any neighbours of these vertices to v_{k+1}. Since v_{k+1} is happy, this cannot create rooted paths on at least b vertices without a happy vertex. Repeating this, we eventually end up with only the root, and reversing the process shows that the fraction of happy vertices is at least $1/b$. ◀

To prove Theorem 15, we prove a slightly stronger claim by induction on n below.

▶ **Lemma 17.** *Let $b \geq 5$ and $k \geq 2$ be integers. Let G be a graph with treewidth at most $k-1$. Let $S \subseteq V(G)$ be given with*

$$2k + 1 \leq |S| \leq (2 + 8/b)k + 1$$

together with $m' : S \to \{1, \ldots, b+1\}$ such that at most $\lceil (1/b)(2 + 8/b)k \rceil$ vertices in S are marked i by m' for all $i \in [b]$. Then G admits a tree decomposition $(T, (B_x)_{x \in V(T)})$ such that

- *$|B_x| \leq (3 + 8/b)k + 1$ for all $x \in V(T)$,*
- *T is rooted in r and $S \subseteq B_r$,*
- *there is a $(b+1)$-marking $m : V(G) \times V(T) \to [b+1]$ for which $m(s, r) = m'(s)$ for all $s \in S$.*

Proof. If $|V(G)| \leq (3 + 8/b)k + 1$ then the lemma holds since we may put all vertices in the same bag.

Suppose that $|V(G)| > (3 + 8/b)k + 1$ and that the lemma has been shown already for graphs on fewer vertices. Let S, m' be given as in the lemma statement. We define a weight function w on G by giving all vertices in S weight 1 and the remaining vertices weight 0. We apply Lemma 9 on (G, w) using any tree decomposition for G of optimal width, in order to find a vertex set X with $|X| \leq \text{tw}(G) + 1 \leq k$ such that each component C of $G - X$ satisfies $|C \cap S| \leq |S|/2$.

For each component C of $G - X$, we add vertices to a set S_C and define a marking $m'_C : S_C \to [b+1]$ as follows:

- For all vertices v in $S \cap C$ or $X \cap S$ marked 1 by m' that have a neighbour in $C \setminus S$, we add v and a fixed neighbour u of v in $C \setminus S$ to S_C and set $m'_C(v) = m'_C(u) = b + 1$.
- All other vertices $s \in S \cap C$ or $X \cap S$ with a neighbour in $C \setminus S$ are added to S_C with mark $m'_C(s) = m'(s) - 1$.
- All vertices $x \in X \setminus S$ with a neighbour in $C \setminus S$ are added to S_C add with mark $m_C(x) = b$.

In particular, vertices in X or S without neighbours in $C \setminus S$ are not added to S_C. Note that

$$\begin{aligned} |S_C| &\leq |S \cap C| + |X| + |\{v \in S : m'(v) = 1\}| \\ &\leq \lfloor |S|/2 \rfloor + k + \lceil (1/b)(2 + 8/b)k \rceil \\ &\leq 2k + \lfloor 1/2 + 4k/b \rfloor + \lceil 4k/b \rceil \\ &\leq (2 + 8/b)k + 1. \end{aligned}$$

where we used that $8/b \leq 2$ for $4 \leq b$ and that $\lfloor 1/2 + x \rfloor + \lceil x \rceil \leq 2x + 1$ for all real $x \geq 0$.

We wish to apply the induction hypothesis on $G_C = G[(C \setminus S) \cup S_C]$ with special set S_C and the marking m'_C constructed above. If $|V(G_C)| \leq 3k$, then this is not needed and we may simply define $R_C = S_C$. Otherwise, we can ensure that $|S_C| \geq 2k + 1$; if needed, we can add vertices from C to S_C and mark these $b + 1$ by m'_C. Note that $|V(G_C)| < |V(G)|$ since at most $|S \cap C| + |X \cap S| \leq |S|/2 + k < |S|$ vertices of S remain present in G_C.

In order to ensure the marking m'_C for S_C satisfies the assumptions in the lemma statement, we can decrease some of the marks if needed. Any vertex in S_C marked i by m'_C for some $i \in [b-1]$, is marked $i + 1$ in S by m', so there are not too many of those. Moreover, at most

$$|S \cap C| + |X| \leq \lfloor |S|/2 \rfloor + k \leq 2k + \lfloor 1/2 + 4k/b \rfloor \leq 2k + 8k/b$$

vertices can ever receive a mark $\leq b$. This means that if too many vertices received mark b at this point, we can safely decrease some of their marks ("increasing their priority").

We are now ready to apply the induction hypothesis on $G_C = G[(C \setminus S) \cup S_C]$ with special set S_C and the marking m'_C adjusted as above (if needed). This yields a tree decomposition T_C with root bag R_C containing S_C for each component C together with a $(b+1)$-marking m_C.

We make one additional bag $B_r = S \cup X$ and make it adjacent to the various R_C bags. Note that $|B_r| \leq |S| + k \leq (3 + 8/b)k + 1$. This creates a tree decomposition which we root in B_r. We let the marking m be defined as by

- $m(s, r) = m'(s)$ for all $s \in S$;
- $m(x, r) = b + 1$ for all $x \in X \setminus S$;
- $m(v, t) = m_C(v, t)$ for $t \in V(T_C)$ and $v \in B_t$.

In order to show this is a $(b + 1)$-marking, we only need to consider what happens to $v \in B_p \cap B_c$ where $B_p = B_r = S \cup X$ is the root bag and $B_c = R_C$ for some component C. If $v \in X \setminus S$, then

$$m(v, c) = m_C(v, c) = m'_C(v) \leq b = (b + 1) - 1 = m(v, p) - 1$$

as desired. If $v \in S$ then

$$m(v, c) = m_C(v, c) = m'_C(v) = m'(v) - 1 = m(v, p) - 1,$$

unless $m'(v) = 1$ in which case a neighbour of v in C which was not in B_r will have been added. This shows that m is a $(b + 1)$-marking, as desired. ◀

Proof of Theorem 15. Let $b \geq 5$ be an integer and let G be a graph and let $k = \text{tw}(G)+1 \geq 2$. The statement is immediately true when $|V(G)| \leq 2k+1$. Otherwise, we let S contain $2k+1$ vertices of G and set $m'(s) = b+1$ for all $s \in S$. Lemma 17 now provides a tree decomposition of width at most $(3 + 8/b)k = (3 + 8/b)(\text{tw}(G) + 1)$ that admits a $(b + 1)$-marking. By Lemma 16 each vertex v has spread at most $(b + 1)(d(v) + 1) \leq 2b(d(v) + 1)$ in this. ◀

5 Proof of Theorem 6

We will use the following auxiliary result.

▶ **Lemma 18** (Lemma 22 in [30]). *For any integer $k \geq 2$, every rooted tree T with $|V(T)| \geq k$ with root r has a sequence $(T_1, \dots, T_m)$ of pairwise edge-disjoint rooted subtrees of T such that:*

- $T = T_1 \cup \dots \cup T_m$;
- $r \in V(T_1)$ *and for* $i \in \{2, \dots, m\}$, *if* r_i *is the root of* T_i *then* $V(T_i) \cap V(T_1 \cup \dots \cup T_{i-1}) = \{r_i\}$;
- $|V(T_i)| \in \{k, \dots, 2k - 2\}$ *for each* $i \in \{1, \dots, m\}$.

In fact, our proof of Theorem 6 will follow an approach from Wood [30], for which we apply the lemma with a different parameter. Only the analysis of the spread in this construction is new.

Proof of Theorem 6. Let $c' > 1$. Choose an integer $t \geq 2$ such that

$$t/(t - 1) \leq c'.$$

Let $c = 3 + 4t$. We will prove that every graph G has a tree-decomposition $(T, (B_t)_{t \in V(T)})$ of width at most $c\,\text{tw}(G)$ and average spread

$$\frac{1}{|V(G)|} \sum_{x \in V(T)} |B_x| \leq c'.$$

Let G be a graph of treewidth $w \geq 1$. Then there is a tree decomposition $(T, (B_x)_{x \in V(T)})$ of G with width w, which we can root such that if x is the parent of y in T, then $|B_y \setminus B_x| = 1$, that is, each bag contributes exactly one new vertex. We apply Lemma 18 to T with $k = t \cdot (w+1) + 1$ to obtain $(T_1, \ldots, T_m)$. (If the lemma does not apply, we are already done by putting all vertices in a single bag.) Let

$$B_1' = \cup_{s \in V(T_1)} B_s$$

and for $i \in \{2, \ldots, m\}$, let

$$B_i' = \cup_{s \in V(T_i) \setminus \{r_i\}} B_s.$$

In [30, Lemma 23] it is shown that such a *quotient* yields a tree decomposition $(F, (B_x')_{x \in V(F)})$ of G, for any tree F with vertex-set $\{1, \ldots, m\}$, rooted at vertex 1, where for $i \in \{2, \ldots, m\}$, the parent of i is any number $j \in \{1, \ldots, i-1\}$ such that $r_i \in V(T_j)$.

Note that for any $i \in [m]$,

$$|B_i'| \leq (w+1) + (|V(T_i)| - 1) \leq w + 2k = (1 + 4t + 2)w = c \cdot \mathrm{tw}(G),$$

since moving from a parent to a child bag introduces *at most* one new vertex. Moreover,

$$|B_i'| \geq |V(T_i)| - 1 \geq k - 1 = t \cdot (w+1).$$

since each bag introduces *at least* one new vertex.

We say that vertex v is *new* in bag B_i' if i is the smallest $j \in [m]$ for which $v \in B_j'$.

▷ Claim 19. For all $i \in [m]$, B_i' has at least $(1 - 1/t)|B_i'|$ new vertices.

Proof. The new vertices in B_i' are exactly those in $B_i' \setminus B_{r_i}$. Indeed, T_i only intersects $T_1, \ldots, T_{i-1}$ in r_i and since $(T, (B_s)_{s \in V(T)})$ is a tree decomposition,

$$\left(\cup_{s \in V(T_i) \setminus \{r_i\}} B_s\right) \cap \left(\cup_{s \in V(T_1) \cup \cdots \cup V(T_{i-1})} B_s\right) \subseteq B_{r_i}.$$

Using that for each $s \in V(T_i) \setminus \{r_i\}$, there is at least one vertex in B_s that was not in the bag of its parent, we find that

$$|B_{r_i} \cap B_i'|/|B_i'| \leq (w+1)/(|V(T_i)| - 1) \leq (w+1)/((w+1)t) = 1/t,$$

and so there are at least $(1 - 1/t)|B_i'|$ new vertices in B_i'. ◁

We now show how the claim above implies that the average spread $\sum_{x \in V(F)} |B_x'|/|V(G)|$ of the tree decomposition is at most

$$t/(t-1) = 1 + 1/(t-1)$$

For $v \in V(G)$, let i_v denote the (unique) index i such that $v \in B_i'$ is new. Then

$$|V(G)| = |\{(v, B_{i_v}') : v \in V(G)\}| = \sum_{i=1}^{m} |\{(v, B_i') : i = i_v, v \in V(G)\}|.$$

By the claim, for each i, since the summand is exactly the number of new vertices,

$$|\{(v, B_i') : i = i_v, v \in V(G)\}| \geq (1 - 1/t)|B_i'|.$$

This proves

$$|V(G)| \geq \sum_{i=1}^{m} |B'_i|(1 - 1/t).$$

Rearranging shows that the average spread

$$\sum_{i=1}^{m} |B'_i|/|V(G)| \leq 1/(1 - 1/t) = 1/((t-1)/t) = t/(t-1).$$

Hence we provided the desired tree decomposition. ◀

6 Near-optimal spread in the grid

Recall that Lemma 10 states that for any $c > 1$, any $(n \times m)$-grid has a path decomposition of width cn and spread at most $\lceil 1/(c-1) \rceil + 2$.

Proof of Lemma 10. We divide $[n]$ into at most $b = \lceil 1/(c-1) \rceil$ consecutive intervals of size at most $\lceil (c-1)n \rceil$, say $[n] = A_1 \sqcup A_2 \sqcup \cdots \sqcup A_b$. Let $A_{[i,j]} = A_i \cup A_{i+1} \cup \cdots \cup A_j$. We claim the following sequence of bags forms the desired path decomposition. The sequence starts with

$$\begin{gathered}
(A_{[1,b]} \times \{1\}) \cup (A_1 \times \{2\}) \\
(A_{[2,b]} \times \{1\}) \cup (A_{[1,2]} \times \{2\}) \\
(A_{[3,b]} \times \{1\}) \cup (A_{[1,3]} \times \{2\}) \\
\vdots \\
(A_b \times \{1\}) \cup (A_{[1,b]} \times \{2\}) \\
(A_{[1,b]} \times \{2\}) \cup (A_1 \times \{3\})
\end{gathered}$$

and continues in this fashion. To be precise, for $i \in [b]$ and $j \in [m-1]$, let

$$B_{i,j} = (A_{[i,b]} \times \{j\}) \cup (A_{[1,i]} \times \{j+1\})$$

where $[1,1] = \{1\}$ and $[b,b] = \{b\}$. The sequence is

$$B_{1,1}, B_{2,1}, \ldots, B_{b,1}, B_{1,2}, \ldots, B_{b,2}, \ldots, B_{1,m-1}, \ldots, B_{b,m-1}.$$

Note that each bag $B_{i,j}$ has size at most $n + \lceil (c-1)n \rceil = \lceil cn \rceil$, so the width is at most cn as claimed. Each vertex of the $n \times m$ grid is in $A_i \times \{j\}$ for exactly one value of (i,j). Each $A_i \times \{j\}$ appears in the sequence above at most $b+1$ times. So the spread of each vertex is at most $b + 1 = \lceil 1/(c-1) \rceil + 1$.

Finally, we verify that the sequence of bags forms a path decomposition directly from the definition: the bags containing a vertex form a subpath and every vertex is covered. All edges are of one of the following forms:

$$\{(x,j),(x+1,j)\},\ \{(x,j),(x,j+1)\}.$$

Let i be given such that $x \in A_i$. Then $(x,j),(x+1,j),(x,j+1) \in B_{i,j}$, so there is an edge containing both types of edges above. ◀

7 Conclusion

In this paper we increased our understanding of the trade-off between width and spread for tree decompositions. We discuss the algorithmic aspects and two conjectures below.

Algorithmic aspects

The proof of Theorem 15 can easily be made algorithmic. Given a c-approximation algorithm for finding balanced separators, for any integer $b \geq 1$ a tree decomposition can be computed for any graph G of width at most $c(3 + 8/b)(\mathrm{tw}(G) + 1)$ for which each vertex $v \in V(G)$ is in at most $2b(d(v) + 1)$ bags, at only polynomial overhead cost. Similarly, the proof of Theorem 6 can also be made algorithmic, showing that for any $c' > 1$, there is a constant $c > 0$ and a polynomial time algorithm which computes for any graph G a tree-decomposition of width at most $c\,\mathrm{tw}(G)$ and average spread at most c'.

Although notions related to spread have been motivated several times for potential algorithmic use, we are not currently aware of any algorithms that explicitly use tree decompositions with small spread and width. A recent application of a result on tree decompositions with additional properties, namely when the tree of the decomposition is relatively small, is in the construction of universal graphs [23].

Conjecture for Problem 1

In terms of the solution to Problem 1, we expect that the answer is 3. The following example could perhaps be used to provide a lower bound which matches our upper bound from Theorem 3. In Section 3 we defined $D^+_{n,m}$ for $m = n^8$, which is obtained from the $(n \times m)$-grid by consistently adding one diagonal per square and adding a special vertex with neighbours spread among the first row. Let H be the graph obtained from n^2 copies of $D^+_{n,m}$, which are glued in a sequence by gluing the "west boundary" of the ith copy on the "east boundary" of the $(i+1)$th, with a vertex v_0 added which is adjacent to $n^2 - 1$ vertices that lie on the first row of these "glued boundaries". Let $v_0^{(i)}$ denote the special vertex of the ith copy of $D^+_{n,m}$. Suppose that a tree decomposition of H has width $< 3n$. We expect that if any bag contains two north-south paths that have many neighbours of $v_0^{(i)}$ in between them, then the spread of $v_0^{(i)}$ is forced to be large. However, when not allowing such a bag, it is difficult to ensure that the bags containing the neighbours of v_0 are all "close to each other" and thereby the spread of v_0 becomes large. We did not see a nice way of formalising this intuition and so we leave open what the exact answer is.

Spread versus width in the grid

Wood [30] also conjectures that every tree-decomposition of the $(n \times n)$-grid with width n, has some vertex with spread $\Omega(n)$. We propose the following stronger conjecture, which would be best possible up to the multiplicative constant.

▶ **Conjecture 20.** *There is a constant $\delta > 0$ such that for all integers $m \geq n \geq 1$, every tree-decomposition of the $(n \times m)$-grid with width $n + a$, has a vertex with spread at least $\delta(n/a)$.*

We sketch why this conjecture holds in the case that m is much larger than n. We may assume that $a \leq \varepsilon n$ (since we may choose δ and each vertex has spread at least 1). Using Lemma 9, we can find a bag B_t that is a balanced separator. Using a similar argument to the proof of Corollary 12, we find that $\{(i, x_i) : i \in [n]\} \subseteq B_t$ for some choice of $x_1, \dots, x_n \in [m/3, 2m/3]$.

Let $I \subseteq [n]$ be the set of $i \in [n]$ for which (i, x_i) is the unique vertex of the form (i, j) in the bag B_t. Then $|I| \geq n - a$. For m much larger than n, all vertices in $\{(i, x_i + 1) : i \in I\}$ then must also be in the same connected component, and so there is a neighbour t_2 of t with $\{(i, x_i) : i \in I\} \subseteq B_{t_2}$. We can then find a neighbour t_3 of t_2 that loses at most a more vertices from the set I, and repeat this n/a before I is empty. This shows that some vertex of the form (i, x_i) is in at least n/a bags.

For the $(n \times n)$-grid, we find it likely that a similar situation occurs, but in this case the separator structure becomes more complicated. Although it is no longer the case that each balanced separator contains a north-south path, we remark that a weaker analogue of this still holds.

▶ **Lemma 21** (Bodlaender [3, Claim 88a]). *Let $(T, (B_x)_{x \in V(T)})$ be a tree decomposition of an $(n \times n)$-grid of treewidth n. Then there is a $t \in V(T)$, such that at least one of the following two holds:*

1. *B_t contains a vertex from each row of the grid.*
2. *B_t contains a vertex from each column of the grid.*

We remark that for the $(n \times n)$-grid, it is already highly non-trivial to prove the rank-width is $n - 1$ (as shown by Jelínek [22]), that the cliquewidth is $n + 1$ (as shown by Golumbic and Rotics [19]) or that its edge-bandwidth is $2n - 1$ (as shown by Pikhurko and Wojciechowski [25]).

Update (April 2026)

Sergey Norin [24] proved Conjecture 20 and together with Neil Rahman showed the answer to Problem 1 is indeed 3. Moreover, Marc Distel, Neel Kaul, Raj Kaul and David Wood [13] gave examples of graphs with treewidth k, maximum degree Δ and domino treewidth $\Omega(k\Delta^2)$ for $k \geq 2$.

References

1. Dan Bienstock. On embedding graphs in trees. *Journal of Combinatorial Theory, Series B*, 49(1):103–136, 1990. `doi:10.1016/0095-8956(90)90066-9`.
2. Hans L. Bodlaender. A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM Journal on Computing*, 25(6):1305–1317, 1996. `doi:10.1137/S0097539793251219`.
3. Hans L. Bodlaender. A partial k-arboretum of graphs with bounded treewidth. *Theoretical Computer Science*, 209(1-2):1–45, 1998. `doi:10.1016/S0304-3975(97)00228-4`.
4. Hans L. Bodlaender. A note on domino treewidth. *Discrete Mathemathics and Theoretical Computer Science*, 3(4):141–150, 1999. `doi:10.46298/DMTCS.256`.
5. Hans L. Bodlaender. Approximation algorithms for treewidth, pathwidth, and treedepth — A short survey. In Daniel Král and Martin Milanic, editors, *50th International Workshop on Graph-Theoretic Concepts in Computer Science, WG 2024*, volume 14760 of *Lecture Notes in Computer Science*, pages 3–18. Springer, 2024. `doi:10.1007/978-3-031-75409-8_1`.
6. Hans L. Bodlaender and Joost Engelfriet. Domino treewidth. *Journal of Algorithms*, 24:94–123, 1997. `doi:10.1006/jagm.1996.0854`.
7. Hans L. Bodlaender, John R. Gilbert, H. Hafsteinsson, and Ton Kloks. Approximating treewidth, pathwidth, frontsize, and minimum elimination tree height. *Journal of Algorithms*, 18:238–255, 1995. `doi:10.1006/JAGM.1995.1009`.
8. Hans L. Bodlaender and Carla Groenland. Trade-off between spread and width for tree decompositions. *arXiv:2601.04040*, 2026. `doi:10.48550/arXiv.2601.04040`.

9 Hans L. Bodlaender, Carla Groenland, and Hugo Jacob. On the parameterized complexity of computing tree-partitions. *Discrete Mathematics & Theoretical Computer Science*, 26:3, 2025. doi:10.46298/dmtcs.12540.

10 Hans L. Bodlaender, Carla Groenland, Jesper Nederlof, and Céline M. F. Swennenhuis. Parameterized problems complete for nondeterministic FPT time and logarithmic space. *Information and Computation*, 300:105195, 2024. doi:10.1016/J.IC.2024.105195.

11 Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Information and Computation*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.

12 Guoli Ding and Bogdan Oporowski. Some results on tree decomposition of graphs. *Journal of Graph Theory*, 20(4):481–499, 1995. doi:10.1002/JGT.3190200412.

13 Marc Distel, Neel Kaul, Raj Kaul, and David R. Wood. Tree-partitions and small-spread tree-decompositions. *arxiv:2604.05690*, 2026. doi:10.48550/arXiv.2604.05690.

14 Marc Distel and David R. Wood. Tree-partitions with bounded degree trees. In David R. Wood, Jan de Gier, and Cheryl E. Praeger, editors, *2021–2022 MATRIX Annals*, volume 5 of *MATRIX Book Series*, pages 203–212. Springer, 2024. doi:10.1007/978-3-031-47417-0_11.

15 Rodney G. Downey and Catherine McCartin. Online problems, pathwidth, and persistence. In Rodney G. Downey, Michael R. Fellows, and Frank K. H. A. Dehne, editors, *1st International Workshop on Parameterized and Exact Computation, IWPEC 2004*, volume 3162 of *Lecture Notes in Computer Science*, pages 13–24. Springer, 2004. doi:10.1007/978-3-540-28639-4_2.

16 Rodney G. Downey and Catherine McCartin. Bounded persistence pathwidth. In Mike D. Atkinson and Frank K. H. A. Dehne, editors, *11th Symposium on Computing: the Australasian Theory Symposium, CATS 2005*, volume 41 of *CRPIT*, pages 51–56. Australian Computer Society, 2005. URL: http://crpit.scem.westernsydney.edu.au/abstracts/CRPITV41Downey.html.

17 Fedor V. Fomin, Pinar Heggernes, and Jan Arne Telle. Graph searching, elimination trees, and a generalization of bandwidth. *Algorithmica*, 41(2):73–87, 2005. doi:10.1007/S00453-004-1117-Y.

18 David Gale. The game of Hex and the Brouwer fixed-point theorem. *The American Mathematical Monthly*, 86(10):818–827, 1979. doi:10.2307/2320146.

19 Martin Charles Golumbic and Udi Rotics. On the clique-width of some perfect graph classes. *International Journal of Foundations of Computer Science*, 11(3):423–443, 2000. doi:10.1142/S0129054100000260.

20 Joel David Hamkins and Davide Leonessi. Infinite Hex is a draw. *Integers: Electronic Journal of Combinatorial Number Theory*, 23(G3), 2023. doi:10.5281/zenodo.10075843.

21 Hugo Jacob, William Lochet, and Christophe Paul. On a tree-based variant of bandwidth and forbidding simple topological minors. *arXiv:2502.11674*, 2025. doi:10.48550/arXiv.2502.11674.

22 Vít Jelínek. The rank-width of the square grid. *Discrete Applied Mathematics*, 158(7):841–850, 2010. doi:10.1016/j.dam.2009.02.007.

23 Neel Kaul, Jaehoon Kim, Minseo Kim, and David Wood. On universal graphs for trees and tree-like graphs. *arXiv:2511.22358*, 2025. doi:10.48550/arXiv.2511.22358.

24 Sergey Norin. Personal communication.

25 Oleg Pikhurko and Jerzy Wojciechowski. Edge-bandwidth of grids and tori. *Theoretical Computer Science*, 369(1):35–43, 2006. doi:10.1016/j.tcs.2006.06.018.

26 Neil Robertson and Paul D. Seymour. Graph minors. XIII. The disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63:65–110, 1995. doi:10.1006/JCTB.1995.1006.

27 Neil Robertson and P.D Seymour. Graph minors. III. Planar tree-width. *Journal of Combinatorial Theory, Series B*, 36(1):49–64, 1984. doi:10.1016/0095-8956(84)90013-3.

28 Detlef Seese. Tree-partite graphs and the complexity of algorithms. In Lothar Budach, editor, *5th International Conference on Fundamentals of Computation Theory, FCT 1985*, volume 199 of *Lecture Notes in Computer Science*, pages 412–421. Springer, 1985. doi:10.1007/BFB0028825.

29 David R. Wood. On tree-partition-width. *European Journal of Combinatorics*, 30(5):1245–1253, 2009. doi:10.1016/J.EJC.2008.11.010.

30 David R. Wood. Tree decompositions with small width, spread, order and degree. *arXiv:2509.01140*, 2025. doi:10.48550/arXiv.2509.01140.