Tree-Partitions of k-Trees with Applications in Graph Layout *

Vida Dujmović
School of Computer Science
McGill University
Montréal, Canada
vida@cs.mcgill.ca

David R. Wood
School of Computer Science
Carleton University
Ottawa, Canada
davidw@scs.carleton.ca

September 23, 2002

Abstract

A *tree-partition* of a graph is a partition of its vertices into 'bags' such that contracting each bag into a single vertex gives a forest (after deleting loops and replacing parallel edges by a single edge). It is proved that every k-tree has a tree-partition such that each bag induces a $(k-1)$-tree, amongst other properties. Applications of this result to two well-studied models of graph layout are presented. First it is proved that graphs of bounded tree-width have bounded *queue-number*, thus resolving an open problem due to Ganley and Heath [15] and disproving a conjecture of Pemmaraju [25]. This result provides renewed hope for the positive resolution of a number of open problems in the theory of queue layouts. In a related result, it is proved that graphs of bounded tree-width have *three-dimensional straight-line grid drawings* with linear volume, which represents the largest known class of graphs with such drawings.

keywords: graph, tree-partition, k-tree, tree-width, queue layout, queue-number, three-dimensional graph drawing, ordered layering.

1 Introduction

Tree-decompositions have become an indispensable tool in algorithmic graph theory. A related structure is that of a tree-partition (defined formally in Section 1.2). A *tree-partition* of a graph is a partition of its vertices into 'bags' such that contracting each bag to a single vertex gives a forest (after deleting

*Technical Report TR-02-03, School of Computer Science, Carleton University, Ottawa, Canada, 2002. Research supported by NSERC and FCAR.

loops and replacing parallel edges by a single edge). We prove that every k-tree has a tree-partition such that each bag induces a connected $(k-1)$-tree, amongst other properties.

Two applications of this theorem in graph layout are then presented. The first relates to queue layouts of graphs (defined in Section 1.3). Queue layouts, which are closely related to stack layouts (also called book embeddings), have been extensively studied [19, 23, 25, 28, 34, 35] and have applications in fault-tolerant processing, matrix computations, sorting networks, and graph drawing. We prove that graphs of bounded tree-width have bounded queue-number, thus solving an open problem due to Ganley and Heath [15]. This result, which also disproves a conjecture of Pemmaraju [25], has significant implications for other open problems in the field.

In a related result we prove that graphs of bounded tree-width have three-dimensional straight-line grid drawings with linear volume, which is the largest known class of graphs admitting such drawings. Motivated by applications in information visualisation, VLSI layout, and software engineering (see [12]), there is a growing body of research in three-dimensional straight-line graph drawing [4, 6, 8, 12, 13, 24, 26, 35].

The remainder of the paper is organised as follows. Section 1.1 recalls a number of definitions and well-known results. In Sections 1.2, 1.3 and 1.4 we survey and state our results for tree-partitions, queue layouts and three-dimensional graph drawings, respectively. In Section 2 we prove the above-mentioned theorem concerning tree-partitions of k-trees. In Section 3 we establish a number of results concerning 'ordered layerings', which will lead to proofs of our theorems for queue layouts and three-dimensional drawings. In Section 4 we refine our bound on the queue-number of a k-tree, to match the best known existing result in the case of $k = 2$. We conclude in Section 5.

1.1 Preliminaries

Throughout this paper all graphs G are undirected, simple and finite with vertex set $V(G)$ and edge set $E(G)$. The number of vertices and the maximum degree of G are respectively denoted by $n = |V(G)|$ and $\Delta(G)$.

A linear order $\sigma = (v_1, v_2, \dots, v_n)$ of $V(G)$ is called a *vertex-ordering* of G. Suppose G is connected. The *depth* of a vertex v_i in σ is the distance between v_1 and v_i in G. We say σ is a *breadth-first* vertex-ordering if for all vertices v_i and v_j with $i < j$, the depth of v_i in σ is no more than the depth of v_j in σ. Vertex-orderings, and in particular, vertex-orderings of trees will be used extensively in this paper. Consider a breadth-first vertex-ordering σ of a tree T such that vertices at depth $d \geq 1$ are ordered with respect to the ordering of vertices at depth $d-1$. In particular, if v and x are vertices at depth d with respective parents w and y at depth $d-1$ with $w <_\sigma y$ then $v <_\sigma x$. Such a vertex-ordering is called a *lexicographical* breadth-first vertex-ordering of T.

A graph H is a *minor* of a graph G if H is isomorphic to a graph obtained from G by contracting edges, deleting edges, and deleting isolated vertices. A family of graphs closed under taking minors is *proper* if it is not the class of all graphs.

Let G be a graph and let T be a tree. An element of $V(T)$ is called a *node*. Let $\{T_x \subseteq V(G) : x \in V(T)\}$ be a set of subsets of $V(G)$ indexed by the nodes of T. Each T_x is called a *bag*. The pair $(T, \{T_x : x \in V(T)\})$ is a *tree-decomposition* of G if:

- $\displaystyle\bigcup_{x \in V(T)} T_x = V(G)$ (that is, every vertex of G is an at least one bag),
- for every edge $vw \in E(G)$, there is a node x of T such that $v \in T_x$ and $w \in T_x$, and
- for all nodes $x, y, z \in V(T)$, if y is on the path from x to z in T, then $T_x \cap T_z \subseteq T_y$.

The *width* of a tree-decomposition is one less than the maximum cardinality of a bag. A *path-decomposition* is a tree-decomposition where the tree T is a path. The *path-width* (respectively, *tree-width*) of a graph G, denoted by $\mathsf{pw}(G)$ ($\mathsf{tw}(G)$), is the minimum width of a path- (tree-) decomposition of G. Path-width and tree-width were first introduced by Robertson and Seymour [29]. A graph G is said to have *bounded path-width* (*tree-width*) if $\mathsf{pw}(G) \leq k$ ($\mathsf{tw}(G) \leq k$) for some constant k.

A *k-tree* for some $k \in \mathbb{N}$ is defined recursively as follows. The empty graph is a k-tree, and the graph obtained from a k-tree by adding a new vertex v adjacent to each vertex of a clique C with at most k vertices is also a k-tree. This definition of a k-tree is by Rautenbach and Reed [27]. The following more restrictive definition of a k-tree, which we call 'strict', was introduced by Arnborg and Proskurowski [1] and is more often used in the literature. A k-clique is a *strict k-tree*, and the graph obtained from a strict k-tree by adding a new vertex v adjacent to each vertex of a k-clique is also a strict k-tree. Obviously the strict k-trees are a proper sub-class of the k-trees. A subgraph of a k-tree is called a *partial k-tree*, and a subgraph of a strict k-tree is called a *partial strict k-tree*. The following result is well known (see for example [2, 27]).

Lemma 1. *Let G be a graph. The following are equivalent:*

(1) *G has tree-width $\mathsf{tw}(G) \leq k$,*

(2) *G is a partial k-tree,*

(3) *G is a partial strict k-tree,*

(4) *G is a subgraph of a chordal graph with no clique on $k+2$ vertices.*

Proof. Scheffler [31] proved that (1) and (3) are equivalent. That (1) and (4) are equivalent is due to Robertson and Seymour [29]. That (2) and (4) are equivalent is the characterisation of chordal graphs in terms of 'perfect elimination' vertex-orderings due to Fulkerson and Gross [14]. ☐

1.2 Tree-Partitions

As in the definition of a tree-decomposition, let G be graph and let $\{T_x \subseteq V(G) : x \in V(T)\}$ be a set of subsets of $V(G)$ (called *bags*) indexed by the nodes of a tree T. The pair $(T, \{T_x : x \in V(T)\})$ is a *tree-partition* of G if

- for all distinct nodes x and y of T, $T_x \cap T_y = \emptyset$, and
- for every edge vw of G, either
  - there is a node x of T such that both $v \in T_x$ and $w \in T_x$ (vw is called an *intra-bag* edge), or
  - there is an edge xy of T such that $v \in T_x$ and $w \in T_y$ (vw is called an *inter-bag* edge).

The main property of tree-partitions which has been studied in the literature is the maximum size of a bag, called the *width* of the tree-partition [3, 10, 11, 18, 33]. The minimum width over all tree-partitions of a graph G is the *tree-partition-width*[1] of G, denoted by $\mathsf{tpw}(G)$. A graph with bounded degree has bounded tree-partition-width if and only if it has bounded tree-width [11]. In particular, for every graph G, Ding and Oporowski [10] proved that $\mathsf{tpw}(G) \leq 24\,\mathsf{tw}(G)\Delta(G)$, and Seese [33] proved that $\mathsf{tw}(G) \leq 2\,\mathsf{tpw}(G) - 1$.

Theorem 4 in this paper (see Section 2) provides a tree-partition of a k-tree G with additional features besides small width. First, the subgraph induced by each bag is a connected $(k-1)$-tree. This allows us to perform induction on k. Second, in each non-root bag T_x the set of vertices in the parent bag of x with a neighbour in T_x form a clique. This feature will be crucial in the intended applications. Finally the tree-partition has width at most $\max\{1, k(\Delta(G)-1)\}$, which represents a constant-factor improvement over the above result by Ding and Oporowski [10] in the case of k-trees.

1.3 Queue Layouts

A *queue layout* of a graph G consists of a vertex-ordering σ of G, and a partition of $E(G)$ into *queues*, such that no two edges in the same queue are *nested* with respect to σ. That is, there are no edges vw and xy in a single queue with $v <_\sigma x <_\sigma y <_\sigma w$. The minimum number of queues in a queue layout of G is called the *queue-number* of G, and is denoted by

[1] Tree-partition-width has also been called *strong tree-width* [3, 33].

$\mathsf{qn}(G)$. A similar concept is that of a *stack layout* (or *book embedding*), which consists of a vertex-ordering σ of G, and a partition of $E(G)$ into *stacks* (or *pages*) such that there are no edges vw and xy in a single stack with $v <_\sigma x <_\sigma w <_\sigma y$. The minimum number of stacks in a stack layout of G is called the *stack-number* (or *page-number*) of G, and is denoted by $\mathsf{sn}(G)$. A queue (respectively, stack) layout with k queues (stacks) is called a k*-queue* (k*-stack*) *layout*. Note that queue layouts of directed graphs [21, 22, 25] and posets [20, 25] have also been investigated.

Heath and Rosenberg [23] characterised graphs admitting 1-queue layouts as the 'arched leveled planar' graphs, and proved that it is NP-complete to recognise such graphs. This result is in contrast to the situation for stack layouts — the graphs admitting 1-stack layouts are precisely the outerplanar graphs, which can be recognised in polynomial time. On the other hand, it is NP-hard to minimise the number of stacks in a stack layout which respects a given vertex-ordering [16]. However the analogous problem for queue layouts can be solved as follows. As illustrated in Figure 1, a k*-rainbow* in a vertex-ordering σ consists of a matching $\{v_iw_i : 1 \leq i \leq k\}$ such that $v_1 <_\sigma v_2 <_\sigma \cdots <_\sigma v_k <_\sigma w_k <_\sigma w_{k-1} <_\sigma \cdots <_\sigma w_1$.

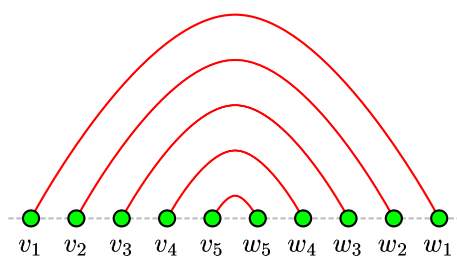

Figure 1: A rainbow of size 5 in a vertex-ordering.

A vertex-ordering containing a k-rainbow needs at least k queues. A straightforward application of Dilworth's Theorem [9] proves the converse. That is, a fixed vertex-ordering admits a k-queue layout where k is the size of the largest rainbow. (Heath and Rosenberg [23] describe an $O(m \log\log n)$ time algorithm to compute the queue assignment.) Thus determining $\mathsf{qn}(G)$ can be viewed as the following vertex layout problem.

Lemma 2. [23] *The queue-number* $\mathsf{qn}(G)$ *of a graph* G *is the minimum, taken over all vertex-orderings* σ *of* G*, of the maximum size of a rainbow in* σ*.*

Stack and/or queue layouts of k-trees have previously been investigated in [5, 15, 28, 35]. A 1-tree has queue-number at most one, since in a lexicographical breadth-first vertex-ordering of a tree no two edges are nested. Chung *et al.* [5] proved that in a depth-first vertex-ordering of a tree no two edges cross. Thus 1-trees have stack-number at most one. Rengarajan and Veni Madhavan [28] proved that a graph with tree-width at most two, also called a *series-parallel* graph[2], has a 2-stack layout and a 3-queue layout. More generally, Ganley and Heath [15] proved that the stack-number $\mathsf{sn}(G) \leq \mathsf{tw}(G)+1$ (using a depth-first traversal of a tree-decomposition), and asked whether the queue-number of a graph is bounded by its tree-width? Partial answers to this question were obtained by Wood [35]. It was shown that the queue-number of a graph is at most its path-width, and is bounded by its tree-partition-width. In particular, $\mathsf{qn}(G) \leq \frac{3}{2}\mathsf{tpw}(G)$ for every graph G, and hence $\mathsf{qn}(G) \leq 36\,\mathsf{tw}(G)\Delta(G)$ by the result of Ding and Oporowski [10] discussed in Section 1.2. In this paper we answer the question of Ganley and Heath [15] in the affirmative.

Theorem 1. *Every graph G with bounded tree-width has $O(1)$ queue-number. In particular,*

$$\mathsf{qn}(G) \;\leq\; 3^{\mathsf{tw}(G)} \cdot 6^{(4^{\mathsf{tw}(G)}-3\mathsf{tw}(G)-1)/9} - 1 \;\;.$$

Further consequences of this result are discussed in Section 5.

1.4 Three-Dimensional Graph Drawings

A *three-dimensional straight-line grid drawing* of a graph, henceforth called a *three-dimensional drawing*, represents the vertices by distinct points in $\mathbb{Z}^3$, and represents each edge as a line-segment between its end-vertices, such that edges only intersect at common end-vertices. In contrast to the case in the plane, it is well known that every graph has a three-dimensional drawing. We therefore are interested in optimising certain measures of the aesthetic quality of a drawing. If a three-dimensional drawing is contained in an axis-aligned box with side lengths $X-1$, $Y-1$ and $Z-1$, then we speak of a $X \times Y \times Z$ drawing with *volume* $X \cdot Y \cdot Z$. We are interested in three-dimensional drawings with small volume.

The volume of three-dimensional drawings has been extensively studied [4, 6, 8, 12, 13, 24, 26, 35]. Cohen, Eades, Lin, and Ruskey [6] proved that every graph has a three-dimensional drawing with $O(n^3)$ volume, and this bound is asymptotically tight for the complete graph K_n. It is therefore of interest to identify fixed graph parameters which allow for three-dimensional drawings with $O(n^2)$ or $O(n)$ volume.

The first such parameter to be studied was the chromatic number [4, 24]. Pach, Thiele, and Tóth [24] proved that graphs of bounded chromatic number have three-dimensional drawings with $O(n^2)$ volume, and that this bound is asymptotically optimal for the complete bipartite graph with equal sized bipartitions. In particular, the volume bound is $O(k^2n^2)$ for k-colourable graphs.

[2]'Series-parallel digraphs' are often defined in terms of certain 'series' and 'parallel' composition operations. Bodlaender [2] proved that the underlying undirected graph of such a 'series-parallel digraph' has tree-width at most two.

The first non-trivial $O(n)$ volume bound was established by Felsner, Wismath, and Liotta [13] in the case of outerplanar graphs. Dujmović, Morin, and Wood [12] proved that graphs of bounded path-width also have three-dimensional drawings with $O(n)$ volume. In particular, if $\mathsf{pw}(G)$ denotes the path-width of a graph G, the volume bound is $O(\mathsf{pw}(G)^2 \cdot n)$. This implies $O(n \log^2 n)$ volume drawings for graphs of bounded tree-width.

Wood [35] proved the following fundamental relationship between queue layouts and three-dimensional drawings.

Lemma 3. [35] *Let G be a graph from a proper minor-closed family. Then G has a $O(1) \times O(1) \times O(n)$ drawing if and only if G has queue-number $\mathsf{qn}(G) \in O(1)$.*

Lemma 3 somewhat hides the magnitude of the constants involved. For example, applying the result of Rengarajan and Veni Madhavan [28] discussed in Section 1.3, Wood [35] proved that every 2-tree has a three-dimensional drawing with $O(n)$ volume, but with a constant of at least 10^{16}. For particular sub-classes of 2-trees, improved constants have been obtained [8, 26]. As another example, Wood [35] proved that graphs of bounded tree-partition-width (which include those of bounded tree-width and bounded maximum degree) have three-dimensional drawings with $O(n)$ volume, although the actual volume bound is approximately $O(\mathsf{tw}(G)^4 (\mathsf{tw}(G)^2\,\mathsf{tpw}(G))^{\mathsf{tw}(G)^2} \cdot n)$.

In this paper we prove the following two bounds on the volume of three-dimensional drawings. The first represents a substantial improvement in the dependence on $\mathsf{tpw}(G)$ compared with the above-mentioned result.

Theorem 2. *Every graph G with bounded tree-partition-width, which includes graph of bounded tree-width and bounded degree, has a three-dimensional drawing with linear volume. In particular, the drawing is*

$$O(\mathsf{tpw}(G)) \times O(\mathsf{tpw}(G)) \times O(n) \ ,$$

which is

$$O(\mathsf{tw}(G)\,\Delta(G)) \times O(\mathsf{tw}(G)\,\Delta(G)) \times O(n) \ .$$

Our main result for three-dimensional graph drawing is the following.

Theorem 3. *Every graph G with bounded tree-width has a three-dimensional drawing with $O(n)$ volume. In particular, the drawing is*

$$O(3^{\mathsf{tw}(G)} \cdot 6^{(4^{\mathsf{tw}(G)} - 3\mathsf{tw}(G) - 1)/9}) \times O(3^{\mathsf{tw}(G)} \cdot 6^{(4^{\mathsf{tw}(G)} - 3\mathsf{tw}(G) - 1)/9}) \times O(n) \ .$$

The family of graphs of bounded tree-width is the largest known family to admit three-dimensional drawings with linear volume. It includes most of the graphs previously known to admit such drawings (e.g., outerplanar and series-parallel graphs), and also includes many graph families for which the previous best volume bound was $O(n \log^2 n)$, such as Halin graphs, the control-flow graphs of structured programs, k-outerplanar graphs, bandwidth-k graphs, and cutwidth-k graphs.

2 Tree-Partitions

Theorem 4. *Let G be a k-tree with maximum degree Δ. Then G has a rooted tree-partition $(T, \{T_x : x \in V(T)\})$ such that for all nodes x of T,*

- *if x is a non-root node of T and y is the parent node of x, then the set of vertices in T_y with a neighbour in T_x form a clique C_x of G, and*
- *the induced subgraph $G[T_x]$ is a connected $(k-1)$-tree.*

Furthermore the width of $(T, \{T_x : x \in V(T)\})$ is at most $\max\{1, k(\Delta-1)\}$.

Proof. We assume G is connected, since if G is not connected then a tree-partition of G which satisfies the theorem can be determined by adding a new root node with an empty bag which is adjacent to the root node of a tree-partition of each connected component of G.

It is well-known that for every vertex r of the k-tree G, there is a vertex-ordering $\sigma = (v_1, v_2, \dots, v_n)$ of G with $v_1 = r$ such that for all $i \in \{1, 2, \dots, n\}$,

(i) if G^i is the induced subgraph $G[\{v_1, v_2, \dots, v_i\}]$, then G^i is connected and the vertex-ordering of G^i induced by σ is a breadth-first vertex-ordering of G^i.

(ii) the neighbours of v_i in G^i form a clique $C_i = \{v_j : v_iv_j \in E(G), j < i\}$ with $1 \leq |C_i| \leq k$ (unless $i = 1$ in which case $C_i = \emptyset$), and

In the language of chordal graphs, σ is a (reverse) 'perfect elimination' vertex-ordering and can be determined, for example, by the Lex-BFS algorithm by Rose *et al.* [30] (also see [17]).

Let r be a vertex of minimum degree[3] in G. Then $\deg(r) \leq k$. Let $\sigma = (v_1, v_2, \dots, v_n)$ be a vertex-ordering of G with $v_1 = r$, and satisfying (i) and (ii). By (i) the depth of each vertex v_i in σ is the same as the depth of v_i in the vertex-ordering of G^j induced by σ, for all $j \geq i$. We therefore simply speak of *the* depth of v_i. Let V_d be the set of vertices of G at depth d.

Claim: For all $i \in \{1, 2, \dots, n\}$, for all $d \geq 1$, and for every connected component Z of $G^i[V_d]$, the set of vertices at depth $d-1$ with a neighbour in Z form a clique of G.

Proof. We proceed by induction on i. The result is trivially true for $i = 1$. Suppose it is true for $i - 1$.

Let d be the depth of v_i. Each vertex in C_i is at depth $d-1$ or d. Let C_i' be the set of vertices in C_i at depth d, and let C_i'' be the set of vertices in C_i at depth $d-1$. Thus C_i' and C_i'' are both cliques with $C_i = C_i' \cup C_i''$. Furthermore, if $i > 1$ then v_i must have a neighbour at depth $d-1$, and thus $C_i'' \neq \emptyset$.

[3] We choose r to have minimum degree simply to prove a slightly improved bound on the width of the tree-partition. If we choose r to be an arbitrary vertex then the width is at most $\max\{1, \Delta, k(\Delta-1)\}$, and the remainder of Theorem 4 holds.

Let X be the vertex set of the connected component of $G^i[V_d]$ such that $v_i \in X$. By induction, for all $d' \leq d$, the claim holds for all connected components of Y of $G^i[V_{d'}]$ with $Y \neq X$, since such a Y is also a connected component of $G^{i-1}[V_{d'}]$.

Case 1. $C'_i = \emptyset$: Then v_i has no neighbours in G^i at depth d; that is, $X = \{v_i\}$. Thus the set of vertices at depth $d-1$ with a neighbour in X is precisely the clique $C_i = C''_i$.

Case 2. $C'_i \neq \emptyset$: The neighbourhood of v_i in X forms a non-empty clique (namely C'_i). Thus $X \setminus v_i$ is the vertex-set of a connected component of $G^{i-1}[V_d]$. Let Y be the set of vertices at depth $d-1$ with a neighbour in $X \setminus v_i$. By induction, Y is a clique. Since $C''_i \cup C'_i$ is a clique, $C''_i \subseteq Y$. Thus the set of vertices at depth $d-1$ with a neighbour in X is the clique Y.

This completes the proof of the claim. ☐

Define a graph T and a partition $\{T_x : x \in V(T)\}$ of $V(G)$ indexed by the nodes of T as follows. There is one node x in T for every connected component of each $G[V_d]$, whose bag T_x is the vertex-set of the corresponding connected component. We say x and T_x are at *depth* d. Clearly a vertex in a depth-d bag is also at depth d. The (unique) node of T at depth zero is called the *root* node. Let two nodes x and y of T be connected by an edge if there is an edge vw of G with $v \in T_x$ and $w \in T_y$. Thus $(T, \{T_x : x \in V(T)\})$ is a 'graph-partition'.

We now prove that in fact T is a tree. First observe that T is connected since G is connected. By definition, nodes of T at the same depth d are not adjacent. Moreover nodes of T can be adjacent only if their depths differ by one. Thus T has a cycle only if there is a node x in T at some depth d, such that x has at least two distinct neighbours in T at depth $d-1$. However this is impossible since by the above claim (with $i = n$), the set of vertices at depth $d-1$ with a neighbour in T_x form a clique (which we call C_x), and are hence in a single bag at depth $d-1$. Thus T is a tree and $(T, \{T_x : x \in V(T)\})$ is a tree-partition of G (see Figure 2).

We now prove that each bag T_x induces a connected $(k-1)$-tree. This is true for the root node which only has one vertex. Suppose x is a non-root node of T at depth d. Each vertex in T_x has at least one neighbour at depth $d-1$. Thus in the vertex-ordering of T_x induced by σ, each vertex $v_i \in T_x$ has at most $k-1$ neighbours $v_j \in T_x$ with $j < i$. Thus $G[T_x]$ is a partial $(k-1)$-tree. An induced subgraph of a k-tree is itself a k-tree. Thus $G[T_x]$ is a $(k-1)$-tree. By definition each $G[T_x]$ is connected.

Finally, consider the size of a bag in T. We claim that each bag contains at most $\max\{1, k(\Delta-1)\}$ vertices. The root bag has one vertex. Let x be a non-root node of T with parent node y. Suppose y is the root node. Then $T_y = \{r\}$, and thus $|T_x| \leq \deg(r) \leq k \leq k(\Delta-1)$ assuming $\Delta \geq 2$. If $\Delta \leq 1$ then all bags have one vertex. Now assume y is a non-root node. The set of vertices in T_y with a neighbour in T_x forms the clique C_x. Let $k' = |C_x|$.

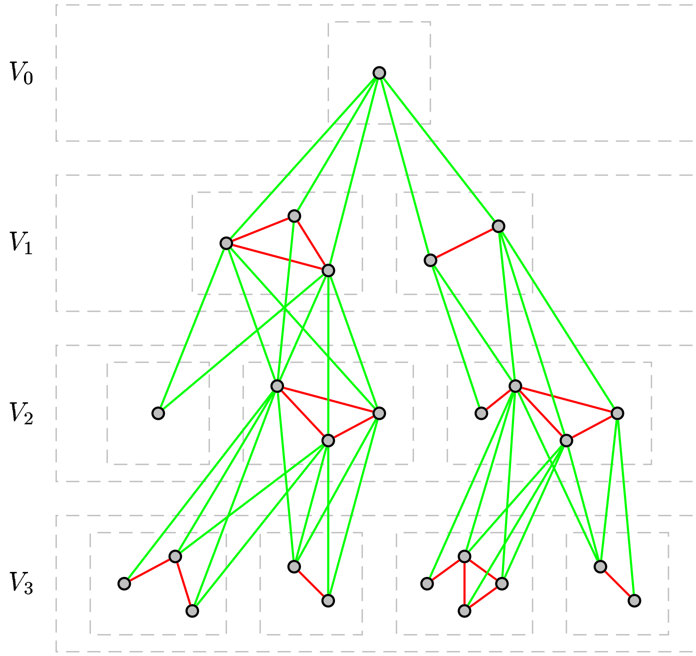

Figure 2: Illustration for Theorem 4 in the case of $k = 3$.

Thus $k' \geq 1$, and since $C_x \subseteq T_y$ and $G[T_y]$ is a $(k-1)$-tree, $k' \leq k$. A vertex $v \in C_x$ has $k' - 1$ neighbours in C_x and at least one neighbour in the parent bag of y. Thus v has at most $\Delta - k'$ neighbours in T_x. Hence the number of edges between C_x and T_x is at most $k'(\Delta - k')$. Every vertex in T_x is adjacent to a vertex in C_x. Thus $|T_x| \leq k'(\Delta - k') \leq k(\Delta - 1)$. This completes the proof. □

3 Ordered Layerings

Dujmović, Morin, and Wood [12] introduced the following structure[4]. An *ordered* $|I|$*-layering* of a graph G, said to be *indexed* by a set I, is a set $\{(V_i, <_i) : i \in I\}$ such that:

- $\bigcup_{i \in I} V_i = V(G)$,
- $\forall$ distinct $i, j \in I$, $V_i \cap V_j = \emptyset$,
- $\forall i \in I$, $<_i$ is a total order of V_i, and
- $\forall$ edges $vw \in E(G)$, if $v \in V_i$ and $w \in V_j$ then $i \neq j$.

[4]Dujmović *et al.* [12] actually consider a slightly more general structure which allows for edges between consecutive vertices in a layer.

It will be convenient to refer to both an element $i \in I$ and a set of vertices V_i as a *layer*. It will always be clear from the context whether a layer is an index i or a set of vertices V_i.

An ordered ℓ-layering of a graph G can be thought of as a proper vertex-colouring of G with ℓ colours such that each colour class is equipped with a total ordering of its vertices.

An *X-crossing* in an ordered layering consists of two edges vw and xy such that for distinct layers i and j, $v <_i x$ and $y <_j w$. Dujmović *et al.* [12] proved that ordered layerings with no X-crossings are inherently related to three-dimensional drawings with small volume.

Lemma 4. [12] *If a graph G has an ordered ℓ-layering with no X-crossing then G has a $O(\ell) \times O(\ell) \times O(n)$ three-dimensional drawing (see Figure 3). Conversely, if G has an $A \times B \times C$ drawing then G has an ordered layering with no X-crossing and $O(AB)$ layers.*

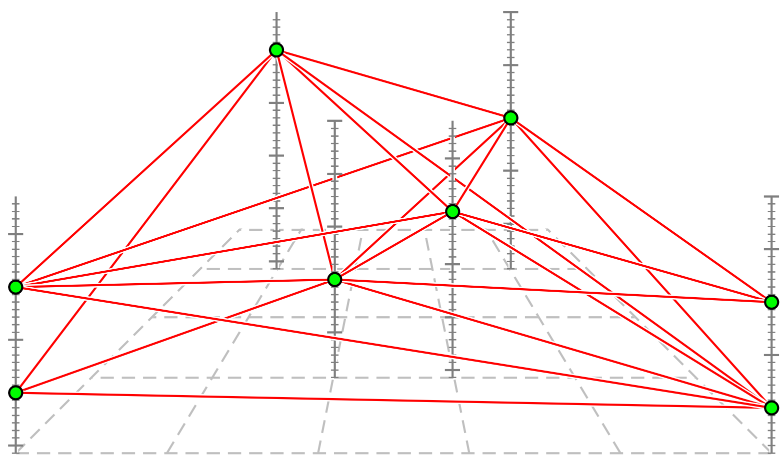

Figure 3: A three-dimensional drawing produced from an ordered layering with no X-crossing; vertices in each layer are placed on a vertical 'rod'.

Wood [35] proved that ordered layerings with no X-crossing are also inherently related to queue layouts.

Lemma 5. [35] *If a graph G has an ordered ℓ-layering with no X-crossing, then $\mathsf{qn}(G) \leq \ell - 1$. Conversely, if G is from a proper minor-closed family and $\mathsf{qn}(G) \in O(1)$, then G has an ordered layering with no X-crossing and $O(1)$ layers.*

Proof Sketch. Suppose the ordered layering is $\{(V_i, <_i) : 1 \leq i \leq \ell\}$. Let σ be the vertex-ordering $V_1, V_2, \ldots, V_\ell$, with each V_i ordered by $<_i$. Between each pair of layers there is at most one edge in a rainbow of σ, as otherwise there would be an X-crossing in the ordered layering. Thus there is at most

$\ell - 1$ edges in rainbow of σ (see Figure 4). By Lemma 2, $\mathsf{qn}(G) \leq \ell - 1$. For the proof of the converse see [35]. □

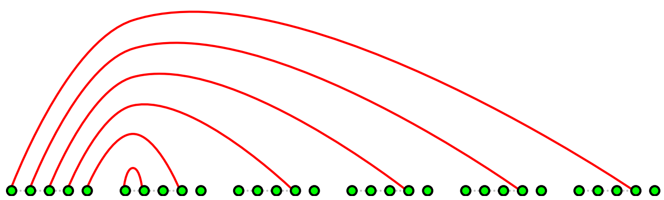

Figure 4: Maximum rainbow in a vertex-ordering from an ordered layering.

Note that Lemmata 4 and 5 together prove Lemma 3. The following result is implicit in the work of Felsner *et al.* [13].

Lemma 6. [13] *Every 1-tree T has an ordered 3-layering with no X-crossing.*

Proof. Clearly we can assume that T is connected. Root T at an arbitrary node r. Let σ be a lexicographical breadth-first vertex-ordering of T starting at r, as described in Section 1.1. For $i \in \{0, 1, 2\}$, let V_i be the set of nodes of T with depth $d \equiv i \pmod 3$ in σ. With each V_i ordered by σ, we have an ordered 3-layering of T. Clearly adjacent vertices are in distinct layers. Since no two edges are nested in σ, there is no X-crossing (see Figure 5). □

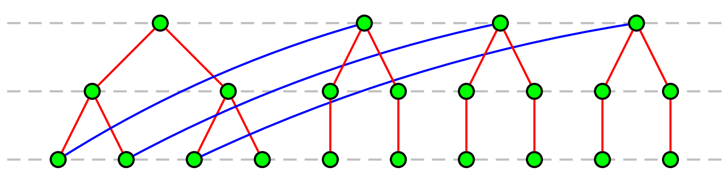

Figure 5: An ordered 3-layering of a tree with no X-crossing.

Let $\{(V_i, <_i) : i \in I\}$ be an ordered layering of a graph G. We say a clique C of G *spans* the set of layers $\{i \in I : C \cap V_i \neq \emptyset\}$. Let S be a set of cliques of G. Suppose there exists a total order $\preceq$ on S such that for all cliques $C_1, C_2 \in S$, if there exists a layer $i \in I$, and vertices $v \in V_i \cap C_1$ and $w \in V_i \cap C_2$ with $v <_i w$, then $C_1 \prec C_2$. In this case, we say $\preceq$ is *nice*, and S is *nicely ordered* by the ordered layering.

Lemma 7. *Let $L \subseteq I$ be a set of layers in an ordered layering $\{(V_i, <_i) : i \in I\}$ of a graph G with no X-crossing. If S is a set of cliques, each of which spans L, then S is nicely ordered by the given ordered layering.*

Proof. Define a relation $\preceq$ on S as follows. For every pair of cliques $C_1, C_2 \in S$, define $C_1 \preceq C_2$ if $C_1 = C_2$ or there exists a layer $i \in L$ and vertices $v \in C_1$ and $w \in C_2$ with $v <_i w$. Clearly all cliques in S are comparable.

Suppose that $\preceq$ is not antisymmetric; that is, there exists distinct cliques $C_1, C_2 \in S$, distinct layers $i, j \in L$, and distinct vertices $v_1, w_1 \in C_1$ and $v_2, w_2 \in C_2$, such that $v_1 <_i v_2$ and $w_2 <_j w_1$. Since C_1 and C_2 are cliques, the edges v_1w_1 and v_2w_2 form an X-crossing, which is a contradiction. Thus $\preceq$ is antisymmetric.

We claim that $\preceq$ is transitive. Suppose there exist cliques $C_1, C_2, C_3 \in S$ such that $C_1 \preceq C_2$ and $C_2 \preceq C_3$. We can assume that C_1, C_2 and C_3 are pairwise distinct. Thus there are vertices $u_1 \in C_1$, $u_2 \in C_2$, $v_2 \in C_2$ and $v_3 \in C_3$, such that $u_1 <_i u_2$ and $v_2 <_j v_3$ for some pair of (not necessarily distinct) layers $i, j \in L$. Since C_3 has a vertex in V_i and since $C_3 \not\preceq C_2$, there is a vertex $u_3 \in C_3$ with $u_2 \leq_i u_3$. Thus $u_1 <_i u_3$, which implies that $C_1 \preceq C_3$. Thus $\preceq$ is transitive.

Hence $\preceq$ is a total order on S, which by definition is nice. □

Consider the problem of partitioning the cliques of a graph into sets such that each set is nicely ordered by a given ordered layering. The following immediate corollary of Lemma 7 says that there exists such a partition where the number of sets does not depend upon the size of the graph.

Corollary 1. *Let G be a graph with maximum clique size k. Given an ordered ℓ-layering of G with no X-crossing, there is a partition of the cliques of G into $\sum_{i=1}^{k} \binom{\ell}{i}$ sets, each of which is nicely ordered by the given ordered layering.* □

We will not actually use Corollary 1 in the following result, but the idea of partitioning the cliques into nicely ordered sets is central to its proof.

Theorem 5. *For every integer $k \geq 0$, there is a constant $\ell_k = 3^k \cdot 6^{(4^k-3k-1)/9}$ such that every graph G with tree-width $\mathsf{tw}(G) \leq k$ has an ordered ℓ_k-layering with no X-crossing.*

Proof. If the input graph G is not a k-tree then add edges to G to obtain a k-tree containing G as a subgraph. It is well-known that a graph with tree-width at most k is a *spanning* subgraph of a k-tree. These extra edges can be deleted once we are done. We proceed by induction on k with the following induction hypothesis.

For all $k \in \mathbb{N}$, there exists constants t_k and ℓ_k, and sets I and S such that

1. *$|I| = \ell_k$ and $|S| = t_k$,*
2. *each element of S is a subset of I, and*
3. *every k-tree G has an ordered ℓ_k-layering indexed by I, and with no X-crossing, such that for every clique C of G, the set of layers which C spans is in S.*

Consider the base case with $k = 0$. A 0-tree G has no edges and thus has an ordered 1-layering with no X-crossing. Let $I = \{1\}$ and order $V_1 = V(G)$ arbitrarily. Thus $\ell_0 = 1$, $t_0 = 1$, and $S = \{\{1\}\}$ satisfy the hypothesis for every 0-tree. Now suppose the result holds for $k-1$, and G is a k-tree. Let $(T, \{T_x : x \in V(T)\})$ be a tree-partition of G described in Theorem 4, where T is rooted at r.

By Theorem 4 each induced subgraph $G[T_x]$ is a $(k-1)$-tree. By induction, there are sets I and S with $|I| = \ell_{k-1}$ and $|S| = t_{k-1}$, such that for every node x of T, the induced subgraph $G[T_x]$ has an ordered ℓ_{k-1}-layering indexed by I and with no X-crossing. For every clique C of $G[T_x]$, if C spans $L \subseteq I$ then $L \in S$. Assume $I = \{1, 2, \ldots, \ell_{k-1}\}$ and $S = \{S_1, S_2, \ldots, S_{t_{k-1}}\}$. By Theorem 4, for each non-root node x of T, if p is the parent node of x, then the set of vertices in T_p with a neighbour in T_x form a clique C_x. Let $\alpha(x) = i$ where C_x spans S_i. For the root node r of T, let $\alpha(r) = 1$.

Layering of T

To construct an ordered layering of G we first construct an ordered layering of the tree T indexed by the set $\{(d, i) : d \geq 0, 1 \leq i \leq t_{k-1}\}$, where the layer $L_{d,i}$ consists of nodes x of T at depth d with $\alpha(x) = i$. Here the *depth* of a node x is the distance in T from the root node r to x. We order the nodes of T within the layers by increasing depth. There is only one node at depth $d = 0$. Suppose we have determined the order of the nodes up to depth $d-1$ for some $d \geq 1$.

Let $i \in \{1, 2, \ldots, t_{k-1}\}$. The nodes in $L_{d,i}$ are ordered primarily with respect to the relative positions of their parent nodes (at depth $d-1$). More precisely, let $\rho(x)$ denote the parent node of each node $x \in L_{d,i}$. For all nodes x and y in $L_{d,i}$, if $\rho(x)$ and $\rho(y)$ are on the same layer and $\rho(x) < \rho(y)$ in that layer, then $x < y$ in $L_{d,i}$. For x and y with $\rho(x)$ and $\rho(y)$ on distinct layers, the relative order of x and y is not important. It remains to specify the order of nodes in $L_{d,i}$ with a common parent.

Suppose P is a set of nodes in $L_{d,i}$ with a common parent node p. By construction, for every node $x \in P$, the parent clique C_x spans S_i in the ordered layering of $G[T_p]$. By Lemma 7 the cliques $\{C_x : x \in P\}$ are nicely ordered by the ordered layering of $G[T_p]$. Let the order of P in the layer $L_{d,i}$ be specified by a nice ordering of $\{C_x : x \in P\}$, as illustrated in Figure 6.

This construction defines a partial order on the nodes in layer $L_{d,i}$, which can be arbitrarily extended to a total order. Hence we have an ordered layering of T. Since the nodes in each layer are ordered primarily with respect to the relative positions of their parent nodes in the previous layers, there is no X-crossing in the ordered layering of T.

Layering of G

To construct an ordered layering of G from the ordered layering of T, replace each layer $L_{d,i}$ by ℓ_{k-1} 'sub-layers', and for each node x of T, insert the ordered layering of $G[T_x]$ in place of x on the sub-layers corresponding to

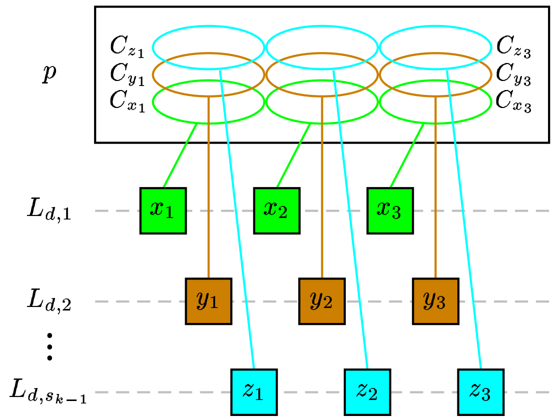

Figure 6: Ordered layering of nodes with a common parent p.

the layer containing x in the ordered layering of T. More formally, the ordered layering of G is indexed by the set

$$\{(d,i,j) : d \geq 0, 1 \leq i \leq t_{k-1}, 1 \leq j \leq \ell_{k-1}\} \ .$$

Each layer $V_{d,i,j}$ consists of those vertices v of G such that, if T_x is the bag containing v, then x is at depth d in T, $\alpha(x) = i$, and v is on layer j in the ordered layering of $G[T_x]$. If x and y are distinct nodes of T with $x < y$ in $L_{d,i}$, then $v < w$ in $V_{d,i,j}$, for all vertices $v \in T_x$ and $w \in T_y$ on layer j. If v and w are vertices of G on layer j in bag T_x at depth d, then the relative order of v and w in $V_{d,\alpha(x),j}$ is the same as in the ordered layering of $G[T_x]$.

Clearly adjacent vertices of G are in distinct layers. Thus we have defined an ordered layering of G. We claim there is no X-crossing. Clearly an intra-bag edge of G is not in an X-crossing with an edge not in the same bag. By induction, there is no X-crossing between intra-bag edges in a common bag. Since there is no X-crossing in the ordered layering of T, inter-bag edges of G which are mapped to edges of T without a common parent node, are not involved in an X-crossing.

Consider a parent node p in T. For each child node x of p, the set of vertices in T_p adjacent to a vertex in T_x forms the clique C_x. Thus there is no X-crossing between a pair of edges both from C_x to T_x, since the vertices of C_x are on distinct layers. Consider two child nodes x and y of p. For there to be an X-crossing between an edge from T_p to T_x and and an edge from T_p to T_y, the nodes x and y must be on the same layer in the ordered layering of T. Suppose $x < y$ in this layer. By construction, C_x and C_y span the same set of layers, and $C_x \preceq C_y$ in the corresponding nice ordering. Thus for any layer containing vertices $v \in C_x$ and $w \in C_y$, $v \leq w$ in that layer. Since all the vertices in T_x are to the left of the vertices in T_y (on a common layer), there is no X-crossing between an edge from T_p to T_x and and an edge from T_p to T_y. Therefore there is no X-crossing in the ordered layering of G.

Wrapped layering of G

Now 'wrap' the layering of G in the spirit of Lemma 6. In particular, define an ordered layering of G indexed by

$$\big\{(d',i,j) : d' \in \{0,1,2\}, 1 \leq i \leq t_{k-1}, 1 \leq j \leq \ell_{k-1}\big\} \ ,$$

where each layer

$$W_{d',i,j} \ = \ \bigcup \{V_{d,i,j} : d \equiv d' \pmod 3\} \ .$$

If $v \in V_{d,i,j}$ and $w \in V_{d+3,i,j}$ then $v < w$ in the order of $W_{d',i,j}$ (where $d' = d \bmod 3$). The order of each $V_{d,i,j}$ is preserved in $W_{d',i,j}$. The set of layers $\{W_{d',i,j} : d' \in \{0,1,2\}, 1 \leq i \leq t_{k-1}, 1 \leq j \leq \ell_{k-1}\}$ forms an ordered layering of G.

For every edge vw of G, the depths of the bags in T containing v and w differ by at most one. Thus in the wrapped layering of G, adjacent vertices remain on distinct layers, and there is no X-crossing. The number of layers is $3 \cdot t_{k-1} \cdot \ell_{k-1}$.

Every clique C of G is either contained in a single bag of the tree-partition or is contained in two adjacent bags. Let

$$S' = \big\{\{(d',i,h) : h \in S_j\} : d' \in \{0,1,2\}, 1 \leq i,j \leq t_{k-1}\big\} \ .$$

For every clique C of G contained in a single bag, the set of layers containing C is in S'. Let

$$S'' = \big\{\{(d',i,h) : h \in S_j\} \cup \{((d'+1) \bmod 3, p, h) : h \in S_q\} : \\ d' \in \{0,1,2\}, 1 \leq i,j,p,q \leq t_{k-1}\big\} \ .$$

For every clique C of G contained in two bags, the set of layers containing C is in S'. Observe that $S' \cup S''$ is independent of G. Hence $S' \cup S''$ satisfies the hypothesis for k.

Now $|S'| = 3t_{k-1}^2$ and $|S''| = 3t_{k-1}^4$, and thus $|S' \cup S''| = 3t_{k-1}^2(t_{k-1}^2 + 1)$. Therefore any solution to the following set of recurrences satisfies the theorem:

$$\begin{aligned} t_0 &\geq 1 \\ \ell_0 &\geq 1 \\ t_k &\geq 3t_{k-1}^2(t_{k-1}^2+1) \\ \ell_k &\geq 3t_{k-1} \cdot \ell_{k-1} \ . \end{aligned} \tag{1}$$

We claim that

$$t_k = 6^{(4^k-1)/3} \text{ and } \ell_k = 3^k \cdot 6^{(4^k-3k-1)/9}$$

is a solution to (1). Observe that $t_0 = 1$ and $\ell_0 = 1$. Now

$$3t_{k-1}^2(t_{k-1}^2+1) \ \leq \ 6t_{k-1}^4 \ , \tag{2}$$

and

$$6(6^{(4^{k-1}-1)/3})^4 \;=\; 6^{1+4(4^{k-1}-1)/3} \;=\; 6^{(4^k-1)/3} \;=\; t_k \;\;.$$

Thus the recurrence for t_k is satisfied. Now

$$\begin{aligned} 3 \cdot t_{k-1} \cdot \ell_{k-1} &= 3 \cdot 6^{(4^{k-1}-1)/3} \cdot 3^{k-1} \cdot 6^{(4^{k-1}-3(k-1)-1)/9} \\ &= 3^k \cdot 6^{(3 \cdot 4^{k-1}-3+4^{k-1}-3k+3-1)/9} \\ &= 3^k \cdot 6^{(4^k-3k-1)/9} \\ &= \ell_k \;\;. \end{aligned}$$

Thus the recurrence for ℓ_k is satisfied. This completes the proof. □

Observe that Theorem 5 together with Lemmata 4 and 5 proves Theorems 3 and 1, respectively.

There are two places in the proof of Theorem 5 where the analysis is weak. In Eqn. (2) we bound $3t_{k-1}^2(t_{k-1}^2+1)$ by $6t_{k-1}^4$, and in S we include sets of layers with more than $k+1$ layers, even though a k-tree has no clique on more than $k+1$ vertices. Thus the following set of recurrences provides a refined solution to our problem, where $t_{k,i}$ denotes the number of sets of layers in S with exactly i layers, $1 \le i \le k+1$. We denote by $t'_{k,i}$ (respectively, $t''_{k,i}$) the number of sets of layers in S with exactly i layers which contain cliques from one (two) bags.

$$\begin{aligned} \ell_0 &= 1 \\ t_{0,1} &= 1 \\ t'_{k,k+1} &= 0 \\ t'_{k,i} &= 3 \cdot t_{k-1} \cdot t_{k-1,i} && (1 \le i \le k) \\ t''_{k,i} &= 3(t_{k-1})^2 \sum_{j\,=\,1}^{i-1} t_{k-1,j} \cdot t_{k-1,i-j} && (1 \le i \le k+1) \\ t_{k,i} &= t'_{k,i} + t''_{k,i} && (1 \le i \le k+1) \\ t_k &= \sum_{i=1}^{k+1} t_{k,i} \\ \ell_k &= 3 \cdot t_{k-1} \cdot \ell_{k-1} && (3) \end{aligned}$$

A further refinement is possible. If we replace "k-tree G" in the inductive hypothesis in the proof of Theorem 5 by "partial strict k-tree G", then by adding edges, G can be assumed to be a strict k-tree, and thus each parent clique C_x is a k-clique. The layers for G are $\{V_{d',i,j} : d' \in \{0,1,2\}, 1 \le i \le t_{k-1,k}, 1 \le j \le \ell_{k-1}\}$, and we obtain the following set of recurrences.

$$\begin{aligned}
\ell_0 &= 1 \\
t_{0,1} &= 1 \\
t'_{k,k+1} &= 0 \\
t'_{k,i} &= 3 \cdot t_{k-1,k} \cdot t_{k-1,i} && (1 \leq i \leq k) \\
t''_{k,i} &= 3(t_{k-1,k})^2 \sum_{j=1}^{i-1} t_{k-1,j} \cdot t_{k-1,i-j} && (1 \leq i \leq k+1) \\
t_{k,i} &= t'_{k,i} + t''_{k,i} && (1 \leq i \leq k+1) \\
\ell_k &= 3 \cdot t_{k-1,k} \cdot \ell_{k-1}
\end{aligned} \tag{4}$$

From an algorithmic point of view, the disadvantage of using strict k-trees is that at each recursive step, extra edges must be added to enlarge the graph from a partial strict k-tree into a strict k-tree, whereas when using (non-strict) k-trees, extra edges need only be added at the beginning of the algorithm.

We have solved (3) and (4) for small values of k. Table 1 compares these solutions to that presented in Theorem 5.

Table 1: Values of ℓ_k (rounded up for $k \geq 3$)

k	Theorem 5	Eqns. (3)	Eqns. (4)
0	1	1	1
1	3	3	3
2	54	54	27
3	2^{21}	2^{19}	2^{16}
4	2^{77}	2^{66}	2^{53}
5	2^{298}	2^{238}	2^{192}
6	$2^{1,181}$	2^{879}	2^{707}
7	$2^{4,711}$	$2^{3,301}$	$2^{2,655}$
8	$2^{18,829}$	$2^{12,601}$	$2^{10,134}$
9	$2^{75,299}$	$2^{48,145}$	$2^{38,721}$
10	$2^{301,177}$	$2^{184,588}$	$2^{148,454}$

We now show that if the degree as well as the tree-width of the graph is bounded then the dependence on the tree-width in the number of layers of an ordered layering is substantially reduced.

Lemma 8. *Every graph G has an ordered layering with no X-crossing and at most $3\,\mathsf{tpw}(G) \leq 72\,\Delta(G)\mathsf{tw}(G)$ layers.*

Proof. Let $(T, \{T_x : x \in V(T)\})$ be a tree-partition of G with width $\mathsf{tpw}(G)$. By Lemma 6, T has an ordered 3-layering with no X-crossing. Replace each layer by $\mathsf{tpw}(G)$ layers, and for each node x in T, place the vertices in bag T_x on the layers replacing the layer containing x, with at most one vertex in T_x on a single layer. The ordering within each layer preserves the ordering of the original layering of T. We obtain an ordered layering of G. In the layering of T, adjacent nodes were on distinct layers, and there is no X-crossing. Thus there is no X-crossing in the layering of G. The number of layers is $3\,\mathsf{tpw}(G)$, which is at most $72\,\Delta(G)\mathsf{tw}(G)$ by the theorem of Ding and Oporowski [10] discussed in Section 1.2. $\square$

Theorem 2 immediately follows from Lemmata 4 and 8.

4 Series-Parallel Graphs

Graphs with tree-width at most two are called *series-parallel*. Three-dimensional drawings of series-parallel graphs are of particular interest [8, 26]. By Eqns. (4), such a graph has an ordered 27-layering with no X-crossing. This bound can be improved as follows.

Lemma 9. *Every series-parallel graph has an ordered* 18*-layering with no X-crossing.*

Proof. Consider an ordered 3-layering of a 1-tree T, indexed by $\{1, 2, 3\}$ and produced by Lemma 6. Let $S_1 = \{\{1, 2\}, \{2, 3\}, \{2\}\}$ and $S_2 = \{\{1, 3\}, \{1\}, \{3\}\}$. For each $i \in \{1, 2\}$, let C_i be the set of cliques of T which span a set of layers in S_i. Clearly every clique of T is in $C_1 \cup C_2$. It is easily checked that each C_i is nicely ordered by the ordered 3-layering of T. Thus in Theorem 5 we can use $t_1 = 2$. Since $\ell_3 = 3 \cdot t_1 \cdot \ell_1$ and $\ell_1 = 3$, we have $\ell_2 = 18$. Thus a series-parallel graph has an ordered 18-layering. $\square$

Applying Lemma 4, we find that every series-parallel graph has a $36 \times 37 \times 37\lceil \frac{n}{36} \rceil$ drawing (see [12] for how to determine the exact constants). This represents a considerable improvement in the constant of the $O(n)$ volume bound, compared to the result of Wood [35] discussed in Section 1.

Theorem 1 gives a bound on the queue-number of $\mathsf{qn}(G) \leq 53$ in the case of a series-parallel graph G. By Lemmata 5 and 9 we have that $\mathsf{qn}(G) \leq 17$. Rengarajan and Veni Madhavan [28] proved that $\mathsf{qn}(G) \leq 3$. We now provide a simple proof of the same bound.

Lemma 10. *Every series-parallel graph G has queue-number* $\mathsf{qn}(G) \leq 3$.

Proof. Assume G is a 2-tree. Otherwise add edges to G so that it becomes a 2-tree. Deleting these extra edges does not increase the queue-number. Let $(T, \{T_x : x \in V(T)\}$ be a rooted tree-partition of G from Theorem 4. For all nodes x of T, the induced subgraph $G[T_x]$ is a tree. Compute a 1-queue layout of each $G[T_x]$ using a lexicographical breadth-first vertex-ordering, as described in Section 1.1. For each node x at depth d the set of vertices at depth $d-1$ with a neighbour in T_x form the clique C_x, which is simply a vertex or an edge. Compute a breadth-first vertex-ordering of T starting at r, and replace each node x by the vertex-ordering in the queue layout of $G[T_x]$. Nodes x at depth $d \geq 1$ are ordered with respect to the rightmost vertex in C_x in the existing ordering of vertices at depth $d-1$.

Clearly, the intra-bag edges can share the 'first' queue. Let vw be an inter-bag edge with $v \in C_x$ and $w \in T_x$. If v is the rightmost vertex in C_x then assign vw to the 'second' queue. Otherwise assign vw to the 'third' queue. By the ordering chosen for the nodes at depth d, no two edges in the 'second' queue are nested. Suppose edges vw and pq in the third 'queue' are nested with $v < p < q < w$ in the vertex-ordering. Then v is the leftmost vertex of an intra-bag edge vu, and p is the leftmost vertex of an intra-bag edge pr. Note that $r \neq u$ since each vertex has at most one intra-bag edge whose other endpoint is to the left in the ordering (since each $G[T_x]$ is a tree). Since q was chosen to be to the left of w, r is to the left of u. Thus the intra-bag edges vu and pr are nested, which is a contradiction. Thus no two edges in the 'third' queue are nested. We obtain a 3-queue layout of G. □

5 Conclusion and Open Problems

One reason why tree-decompositions are important in algorithmic graph theory is that, by using dynamic programming on a tree-decomposition, many NP-hard problems can be solved efficiently for graphs of bounded tree-width. It would be of interest to see if similar results can be obtained by recursive dynamic programming on the tree-partition of Theorem 4.

The following are some open problems regarding queue layouts:

1. Does every planar graph have bounded queue-number?
2. Does every graph with bounded stack-number have bounded queue-number?

Note that since planar graphs have stack-number at most four [36], the second question is more general than the first. Heath *et al.* [19, 23] conjectured that both of these questions have an affirmative answer. More recently however, Pemmaraju [25] conjectured that the 'stellated K_3', a planar 3-tree, has $\Theta(\log n)$ queue-number, and provided evidence to support this conjecture (also see [15]). This suggested that the answers to the above questions were both negative. In particular, Pemmaraju [25] and Heath [private communication, 2002] conjectured that planar graphs have $O(\log n)$ queue-number. However, Theorem 1 provides a queue-layout of *any* 3-tree, and thus the stellated K_3, with $O(1)$ queues. Hence our result disproves the first conjecture of Pemmaraju [25] mentioned above, and renews hope in an affirmative answer to the above open problems.

By Lemma 3, the first question above is almost equivalent to an open problem due to Felsner *et al.* [13], who ask whether every planar graph has a three-dimensional drawing with $O(n)$ volume? A celebrated result independently due to de Fraysseix, Pach, and Pollack [7] and Schnyder [32] states that every planar graph has a two-dimensional straight-line grid drawing with $O(n^2)$ area, and that $\Omega(n^2)$ area is necessary for certain planar graphs. Even whether every planar graph has a three-dimensional drawing with $o(n^2)$ volume is an open problem.

References

[1] S. ARNBORG AND A. PROSKUROWSKI, Linear time algorithms for NP-hard problems restricted to partial k-trees. *Discrete Appl. Math.*, **23(1)**:11–24, 1989.

[2] H. L. BODLAENDER, A partial k-arboretum of graphs with bounded treewidth. *Theoret. Comput. Sci.*, **209(1-2)**:1–45, 1998.

[3] H. L. BODLAENDER AND J. ENGELFRIET, Domino treewidth. *J. Algorithms*, **24(1)**:94–123, 1997.

[4] T. CALAMONERI AND A. STERBINI, 3D straight-line grid drawing of 4-colorable graphs. *Inform. Process. Lett.*, **63(2)**:97–102, 1997.

[5] F. R. K. CHUNG, F. T. LEIGHTON, AND A. L. ROSENBERG, Embedding graphs in books: a layout problem with applications to VLSI design. *SIAM J. Algebraic Discrete Methods*, **8(1)**:33–58, 1987.

[6] R. F. COHEN, P. EADES, T. LIN, AND F. RUSKEY, Three-dimensional graph drawing. *Algorithmica*, **17(2)**:199–208, 1996.

[7] H. DE FRAYSSEIX, J. PACH, AND R. POLLACK, How to draw a planar graph on a grid. *Combinatorica*, **10(1)**:41–51, 1990.

[8] E. DI GIACOMO, G. LIOTTA, AND S. WISMATH, Drawing series-parallel graphs on a box. In S. WISMATH, ed., *Proc. 14th Canadian Conf. on Computational Geometry (CCCG '02)*, The University of Lethbridge, Canada, 2002.

[9] R. P. DILWORTH, A decomposition theorem for partially ordered sets. *Ann. of Math. (2)*, **51**:161–166, 1950.

[10] G. DING AND B. OPOROWSKI, Some results on tree decomposition of graphs. *J. Graph Theory*, **20(4)**:481–499, 1995.

[11] G. DING AND B. OPOROWSKI, On tree-partitions of graphs. *Discrete Math.*, **149(1-3)**:45–58, 1996.

[12] V. DUJMOVIĆ, P. MORIN, AND D. R. WOOD, Pathwidth and three-dimensional straight-line grid drawings of graphs. In M. GOODRICH, ed., *Proc. 10th International Symp. on Graph Drawing (GD '02)*, Lecture Notes in Comput. Sci., Springer, to appear.

[13] S. FELSNER, S. WISMATH, AND G. LIOTTA, Straight-line drawings on restricted integer grids in two and three dimensions. In P. MUTZEL, M. JÜNGER, AND S. LEIPERT, eds., *Proc. 9th International Symp. on Graph Drawing (GD '01)*, vol. 2265 of *Lecture Notes in Comput. Sci.*, pp. 328–342, Springer, 2002.

[14] D. R. FULKERSON AND O. A. GROSS, Incidence matrices and interval graphs. *Pacific J. Math.*, **15**:835–855, 1965.

[15] J. L. GANLEY AND L. S. HEATH, The pagenumber of k-trees is $O(k)$. *Discrete Appl. Math.*, **109(3)**:215–221, 2001.

[16] M. R. GAREY, D. S. JOHNSON, G. L. MILLER, AND C. H. PAPADIMITRIOU, The complexity of coloring circular arcs and chords. *SIAM J. Algebraic Discrete Methods*, **1(2)**:216–227, 1980.

[17] M. C. GOLUMBIC, *Algorithmic graph theory and perfect graphs*. Academic Press, 1980.

[18] R. HALIN, Tree-partitions of infinite graphs. *Discrete Math.*, **97**:203–217, 1991.

[19] L. S. HEATH, F. T. LEIGHTON, AND A. L. ROSENBERG, Comparing queues and stacks as mechanisms for laying out graphs. *SIAM J. Discrete Math.*, **5(3)**:398–412, 1992.

[20] L. S. HEATH AND S. V. PEMMARAJU, Stack and queue layouts of posets. *SIAM J. Discrete Math.*, **10(4)**:599–625, 1997.

[21] L. S. HEATH AND S. V. PEMMARAJU, Stack and queue layouts of directed acyclic graphs. II. *SIAM J. Comput.*, **28(5)**:1588–1626, 1999.

[22] L. S. HEATH, S. V. PEMMARAJU, AND A. N. TRENK, Stack and queue layouts of directed acyclic graphs. I. *SIAM J. Comput.*, **28(4)**:1510–1539, 1999.

[23] L. S. HEATH AND A. L. ROSENBERG, Laying out graphs using queues. *SIAM J. Comput.*, **21(5)**:927–958, 1992.

[24] J. PACH, T. THIELE, AND G. TÓTH, Three-dimensional grid drawings of graphs. In G. DI BATTISTA, ed., *Proc. 5th International Symp. on Graph Drawing (GD '97)*, vol. 1353 of *Lecture Notes in Comput. Sci.*, pp. 47–51, Springer, 1998.

[25] S. V. PEMMARAJU, *Exploring the Powers of Stacks and Queues via Graph Layouts*. Ph.D. thesis, Virginia Polytechnic Institute and State University, Virginia, U.S.A., 1992.

[26] T. PORANEN, A new algorithm for drawing series-parallel digraphs in 3D. Tech. Rep. A-2000-16, Dept. of Computer and Information Sciences, University of Tampere, Finland, 2000.

[27] D. RAUTENBACH AND B. REED, Tree width and graph minors. In *Topics in Graph Theory*, Cambridge University Press.

[28] S. RENGARAJAN AND C. E. VENI MADHAVAN, Stack and queue number of 2-trees. In D. DING-ZHU AND L. MING, eds., *Proc. 1st Annual International Conf. on Computing and Combinatorics (COCOON '95)*, vol. 959 of *Lecture Notes in Comput. Sci.*, pp. 203–212, Springer, 1995.

[29] N. ROBERTSON AND P. D. SEYMOUR, Graph minors. II. Algorithmic aspects of tree-width. *J. Algorithms*, **7(3)**:309–322, 1986.

[30] D. J. ROSE, R. E. TARJAN, AND G. S. LEUKER, Algorithmic aspects of vertex elimination on graphs. *SIAM J. Comput.*, **5**:266–283, 1976.

[31] P. SCHEFFLER, *Die Baumweite von Graphen als ein Maß für die Kompliziertheit algorithmischer Probleme*. Ph.D. thesis, Akademie der Wissenschaften der DDR, Berlin, 1989.

[32] W. SCHNYDER, Planar graphs and poset dimension. *Order*, **5(4)**:323–343, 1989.

[33] D. SEESE, Tree-partite graphs and the complexity of algorithms. In L. BUDACH, ed., *Proc. International Conf. on Fundamentals of Computation Theory*, vol. 199 of *Lecture Notes in Comput. Sci.*, pp. 412–421, Springer, 1985.

[34] F. SHAHROKHI AND W. SHI, On crossing sets, disjoint sets, and pagenumber. *J. Algorithms*, **34(1)**:40–53, 2000.

[35] D. R. WOOD, Queue layouts, tree-width, and three-dimensional graph drawing. In *Proc. of 22nd Foundations of Software Technology and Theoretical Computer Science (FSTTCS '02)*, Lecture Notes in Comput. Sci., Springer, to appear.

[36] M. YANNAKAKIS, Embedding planar graphs in four pages. *J. Comput. System Sci.*, **38**:36–67, 1986.