On the pagenumber of k-trees

Jennifer Vandenbussche*, Douglas West†, Gexin Yu‡

February 14, 2009

Abstract

A *p-page embedding* of a graph G is a vertex-ordering π of $V(G)$ (along the "spine" of a book) and an assignment of edges to p half-planes (called "pages") such that no page contains crossing edges (alternating endpoints) relative to π. The *pagenumber* of G is the least p such that G has a p-page embedding. We disprove a conjecture of Ganley and Heath by showing that when $k \geq 3$, there are k-trees that do not embed in k pages. We also present an algorithm that produces k-page embeddings for k-trees in a special class.

1 Introduction

Embeddings of graphs in "books" were introduced by Bernhart and Kainen [1]. A *p-page embedding* of a graph G is a vertex ordering π of $V(G)$ (along the "spine" of a book) and an embedding of the edges of G into p half-planes sharing the spine (called "pages") such that no page contains crossing edges. Since the ends of the spine may be considered to be joined by a curve enclosing the edges embedded on a page, we may equivalently view π as a cyclic ordering of the vertices, and then each page is an outerplanar embedding of a subgraph of G with the vertices ordered by π on the outer face. These subgraphs decompose G. The pagenumber of G, denoted $\mathrm{bt}(G)$, is the least p such that G has a p-page embedding. We say that G "embeds in p pages" when $\mathrm{bt}(G) \leq p$.

Note that $\mathrm{bt}(G) = 1$ if and only if G is outerplanar. Bernhart and Kainen [1] showed that $\mathrm{bt}(G) \leq 2$ if and only if G is a subgraph of a Hamiltonian planar graph. Pagenumber has been studied on several classes of graphs, including planar graphs [10], graphs with genus g [5, 6], and complete bipartite graphs [3, 7].

*Mathematics Department, Southern Polytechnic State University, Marietta GA, jvandenb@spsu.edu. This work supported in part by the 2004 REGS program (Research Experiences for Graduate Students) of the Mathematics Department at the University of Illinois

†Mathematics Department, University of Illinois, west@math.uiuc.edu. Supported in part by NSA grant H98230-06-1-0065.

‡Department of Mathematics, College of William and Mary gyu@wm.edu. Supported in part by NSF grant DMS-0852452.

In this paper, we study the pagenumber of k-trees. The inductive definition of k-trees is the most convenient for our arguments. A *k-tree* is either the complete graph K_k or a graph obtained from a k-tree G by adding one vertex whose neighborhood is a k-clique in G, where a *k-clique* is a set of k pairwise adjacent vertices. The 1-trees are simply the trees, which are outerplanar, and hence they have pagenumber 1. Chung, Leighton, and Rosenberg [2] showed that the pagenumber of every 2-tree is at most 2. Ganley and Heath [4] exhibited k-trees that require k pages and proved that if G is a k-tree, then $\mathrm{bt}(G) \le k+1$. They conjectured that every k-tree embeds in k pages; we disprove this conjecture.

Theorem 1.1. *For $k \ge 3$, there is a k-tree that does not embed in k pages.*

After proving this main theorem, we present an algorithm that produces k-page embeddings for many k-trees. Let $G[X]$ denote the subgraph of a graph G induced by a set X of vertices. A *tree-decomposition* of G consists of a *host tree* T and a family $\{X_i : i \in V(T)\}$ of subsets of $V(G)$ such that (1) $G = \bigcup_{i \in V(T)} G[X_i]$ and (2) for each $v \in V(G)$, the set $\{i : v \in X_i\}$ induces a subtree of T. To avoid confusing vertices of T with those of G, we use *node* for a vertex of the host tree, and we call the set of vertices in G corresponding to a node of T a *bag* (this term may be due to Bruce Reed). We use $(T, \mathbf{X})$ to denote a tree-decomposition in which T is the host tree and $\mathbf{X}$ is the set of bags (with a given correspondence of bags to nodes).

The *width* of a tree-decomposition $(T, \mathbf{X})$ is $\max_{i \in V(T)}\{|X_i| - 1\}$. The *treewidth* of G is the minimum width among all tree-decompositions of G. (Since every graph has a tree-decomposition with all vertices in one bag, treewidth is well-defined.)

Further motivation for studying k-trees is that the subgraphs of k-trees are precisely the graphs with treewidth at most k; this equivalence was noted at least as early as [8]. Since deleting edges does not increase pagenumber, upper bounds on the pagenumber of k-trees are also upper bounds on the pagenumber of graphs with treewidth at most k, and the k-trees are the hardest graphs with treewidth k to embed.

A tree-decomposition of width k is *smooth* if every bag has size exactly $k+1$ and the bags for any two adjacent nodes have exactly k common elements. By the inductive definition, a k-tree has a smooth tree-decomposition such that every bag is a $(k+1)$-clique.

Togasaki and Yamazaki [9] showed that if G is a k-tree and has a smooth tree-decomposition of width k whose host tree is a path, then $\mathrm{bt}(G) \le k$. We enlarge the family of k-trees where the conclusion holds.

Theorem 1.2. *If a k-tree G has a smooth tree-decomposition of width k such that the host tree has maximum degree at most 3, then $bt(G) \le k$.*

The k-tree we construct for Theorem 1.1 has a smooth tree-decomposition of width k whose host tree has maximum degree $k+2$. This leaves open the question of determining the maximum l such that every k-tree having a smooth tree-decomposition of width k whose host tree has maximum degree at most l has a book embedding in k pages. Letting $m(k)$ be the maximum such l, our results yield $3 \le m(k) \le k+1$ (see Section 4).

2 A k-Tree With No k-Page Embedding

In this section, we use the phrasing of pagenumber based on a cyclic arrangement of the vertices. We construct a k-tree G that does not embed in k pages. Given any cyclic ordering of $V(G)$, we use pigeonholing arguments to produce an induced subgraph of G that cannot be embedded in k pages under that ordering. This suffices, since a k-page embedding of G contains a k-page embedding of every induced subgraph.

Construction 2.1. *Construction of* G. Our graph G has a central k-clique X with vertices $x_1, \ldots, x_k$. Next we add vertices $y_1, \ldots, y_{kn}$, where $n = k^2 + k + 5$, each adjacent to all of X. Finally, we add vertices called *children*. Each child is adjacent to $k-1$ vertices in X and one y_i. A child has *type* (i, j) if it is adjacent to y_i and nonadjacent to x_j. There are $k^2 n$ different types of children. We create M children of each type, where $M = 3k(nk + k + n)$, so G altogether has $k^2 nM$ children. We refer to all children adjacent to vertex x_r (or y_i) as the *children of* x_r (or y_i). Each vertex of X has $(k-1)knM$ children, each y_i has kM children, and altogether G has $k^2 nM + kn + k$ vertices. □

Fix a cyclic ordering π of $V(G)$. If G has a k-page embedding under π, then any induced subgraph of G has a k-page embedding under the corresponding subordering of π. Our plan is to discard vertices of G in stages until we extract an appropriate induced subgraph that will not embed in k pages under the ordering inherited from π. This contradiction will imply that G has no k-page embedding under π.

Lemma 2.2. *The subscripts on vertices of* G *can be permuted within* X *and within* $y_1, \ldots, y_{kn}$ *so that* $x_1, \ldots, x_k, y_1, \ldots, y_n$ *appear in that order in* π.

Proof. The vertices of X cut the cyclic arrangement π into k segments. By the Pigeonhole Principle, some segment contains at least n vertices of $\{y_1, \ldots, y_{kn}\}$. □

Having permuted subscripts as described in Lemma 2.2, delete from G the vertices $y_{k+1}, \ldots, y_{kn}$ and all their children to obtain an induced subgraph G_1 with vertex ordering π_1 inherited from π. Let $Y = \{y_1, \ldots, y_n\}$, and call the vertices of $X \cup Y$ the *parents*.

Lemma 2.3. *Within* π_1*, there is a subordering consisting of* $X \cup Y$ *and* $3k$ *children of each type in* G_1*, such that for each* (i, j) *the children of type* (i, j) *appear consecutively.*

Proof. We select vertices by an algorithm. In each step either we select $3k$ children of a type not yet selected, or we select a parent vertex. We start just after x_1, moving toward x_2 in π_1. Each new step starts with the first vertex after the previous step ends. A step ends when a parent vertex is reached or when $3k$ children have been seen with the same unselected type. In the latter case, these $3k$ children are selected for that type, and all other vertices seen in the step are discarded. In the former case, only that parent vertex is selected.

The selected vertices form a subordering with the claimed property if all types are selected. If type (i, j) is never selected, then each step sees at most $3k-1$ vertices of type (i, j). Since $k + n$ steps end at parents, the number of steps is $r + k + n$, where r is the number of types selected. If type (i, j) is not selected, then $r < kn$, and at most $(3k-1)(nk-1+k+n)$ children of type (i, j) are seen. Since there are $3k(nk + k + n)$ children of each type, type (i, j) must be selected at some step. □

Let G_2 be the subgraph of G_1 induced by the parents and the children picked in Lemma 2.3. Since the $3k$ children of the same type in G_2 appear consecutively in the inherited ordering π_2, we can speak of the *central* k children among these $3k$ of the same type in π_2. We will discard the non-central children, but first we show that the central children of a given type behave the same way in any k-page embedding of G_2 with ordering π_2.

Vertices $a_1, \ldots, a_m$ and $b_1, \ldots, b_m$ form a *twist of size* m in a cyclic vertex ordering if they appear in the order $a_1, \ldots, a_m, b_1, \ldots, b_m$ and all edges $a_i b_i$ for $1 \leq i \leq m$ exist. If a vertex ordering has a twist of size m, then every book embedding using that ordering requires at least m pages, since the m edges $a_1b_1, \ldots, a_mb_m$ pairwise cross and require distinct pages. For clarity, we specify such a twist by separating the two parts and saying that $a_1, \ldots, a_m$ *form a twist with* $b_1, \ldots, b_m$.

By construction, two children z and z' of the same type in G_2 have the same neighborhood. In a k-page embedding of G_2, we say that z and z' *have the same edge assignment* if for every vertex v in their neighborhood, the edges from v to z and z' lie on the same page.

Lemma 2.4. *In a k-page embedding of G_2 under π_2, the central k children of any one type all have the same edge assignment.*

Proof. Let $v_1, \ldots, v_k$ be the neighbors of the children of type (i, j), in order of their appearance in π (note that $\{v_1, \ldots, v_k\} = (X - \{x_j\}) \cup \{y_i\})$. Group the $3k$ consecutive children of type (i, j) into three groups A, B, C of consecutive vertices in π_2, each of size k. Thus the central k children of type (i, j) are the vertices of B. For $1 \leq r \leq k$, we show that all edges from v_r to B lie on the same page in the given embedding.

Fix r. Choose $a_1, \ldots, a_{r-1}$ in A and $c_{r+1}, \ldots, c_k$ in C, in order under π_2. Given $z \in B$, note that $a_1, \ldots, a_{r-1}, z, c_{r+1}, \ldots, c_k$ form a twist of size k with $v_1, \ldots, v_k$. The edges involving v_r in the twists for various choices of z are all the edges from v_r to B. Each such edge must avoid the $k - 1$ pages of the other edges in the twist, and those edges are the same for $z, z' \in B$. Hence all edges from v_r to B lie on the same page.

Since this holds for all r, the vertices of B have the same edge assignment. □

Let G_3 be the subgraph of G_2 induced by the parents and the k central children of each type, with the inherited ordering π_3. Using the next observation, we will further restrict the vertex set to only five vertices of Y, the children of three of them, and X.

Lemma 2.5. *Let $x_0 = y_n$ and $x_{k+1} = y_1$, and consider a k-page embedding of G_3 under π_3. For $0 \leq j \leq k$, at most k vertices of Y have children between x_j and x_{j+1}, where the region between x_j and x_{j+1} is the portion of π_3 bounded by x_j and x_{j+1} that contains no parents.*

Proof. Suppose that $\{y_{i_1}, \ldots, y_{i_{k+1}}\}$ have children between x_j and x_{j+1}, with $i_1 < \cdots < i_{k+1}$, and let z be a child of $y_{i_{j+1}}$ between x_j and x_{j+1}. Now $y_{i_1}, \ldots, y_{i_{k+1}}$ form a twist of size $k+1$ with $x_1, x_2, \ldots, x_j, z, x_{j+1}, \ldots, x_k$, preventing G_3 from embedding in k pages. □

In Lemma 2.4, we proved that in a k-page embedding of G_3 under π, the children of any one type have the same edge assignment (and appear consecutively). By Lemma 2.5, at most $k(k+1)$ vertices of Y have children (in G_3) along the part of the circle from y_n to y_1 that contains X. Since $n = k^2 + k + 5 = k(k+1) + 5$, at least five vertices of Y have all their (k types of) children along the part of the circle from y_1 to y_n.

In particular, there are at least three such vertices of Y other than y_1 and y_n. Let y_a, y_b, y_c be three such vertices, with $a < b < c$. Let $Z_{i,j}$ denote the set of k children of type (i,j) in G_3 (adjacent to y_i and not to x_j), and let $Z = \bigcup_{(i,j)\in\{a,b,c\}\times[k]} Z_{i,j}$. Let G_4 be the subgraph of G_3 induced by $X \cup \{y_1, y_a, y_b, y_c, y_n\} \cup Z$.

It suffices to show that G_4 does not embed in k pages under the inherited ordering π_4. Assume henceforth that such an embedding exists.

The sets $Z_{i,j}$ for $j \in [k]$ and $i \in \{a,b,c\}$ are located along the part of the circle from y_1 to y_n that avoids X. We say that $Z_{i,r}$ is *before* $Z_{i,s}$ if it is encountered first when following this part of the circle from y_1 to y_n (similarly define *after*). We use $N(v)$ for the set of neighbors of a vertex v.

Lemma 2.6. *For $1 \le r < s \le k$ and $i \in \{a,b,c\}$, if $Z_{i,r}$ and $Z_{i,s}$ are on the same side of y_i (both before y_i or both after y_i), then $Z_{i,r}$ is before $Z_{i,s}$.*

Proof. We state the proof for when $Z_{i,r}$ and $Z_{i,s}$ are both before y_i; the other argument is symmetric. Suppose that $Z_{i,s}$ is before $Z_{i,r}$. Choose $S \subseteq Z_{i,s}$ and $R \subseteq Z_{i,r}$ with $|S| = s$ and $|R| = k+1-s$. Since all of $Z_{i,j}$ is adjacent to all of $X - \{x_j\}$, we have $S \subseteq N(x_r)$ and $R \subseteq N(x_s)$. We conclude that $y_i, x_1, \ldots, x_k$ form a twist of size $k+1$ with $S \cup R$, contradicting the k-page embedding. □

The *earlier* children of y_i are those before y_i; the others are its *later* children.

Lemma 2.7. *All edges joining y_i to its earlier children lie on the same page. Symmetrically, those joining y_i to its later children lie on the same page.*

Proof. Consider the earlier children of y_i. By Lemma 2.4, the vertices of a set $Z_{i,j}$ have the same edge assignment. Hence it suffices to show that edges from y_i to two different types of earlier children are on the same page.

We may assume that $Z_{i,r}$ is before $Z_{i,s}$, which in turn is before y_i. Choose $w \in Z_{i,r}$, and let z be the first vertex of $Z_{i,s}$. We have picked z so that all edges from X to $Z_{i,s} - \{z\}$ cross $y_i z$ (and also $y_i w$). The $k-1$ vertices of $Z_{i,s} - \{z\}$ form a twist with the $k-1$ vertices of $X - \{x_s\}$. Therefore, only one page remains for $y_i z$ and $y_i w$. □

Lemma 2.8. *If $x_1, \ldots, x_k$ form twists with both $v_1, \ldots, v_k$ and $w_1, \ldots, w_k$, where $v_1, \ldots, v_k$ come before $w_1, \ldots, w_k$ except possibly for $v_k = w_1$, then for $1 \le r \le k$ the edges incident to x_r in the two twists are on the same page.*

Proof. Observe that $x_1, \ldots, x_{r-1}, x_{r+1}, \ldots, x_k$ form a twist with $v_1, \ldots, v_{r-1}, w_{r+1}, \ldots, w_k$. The edges $x_r v_r$ and $x_r w_r$ cross all $k-1$ edges formed by this twist. □

We now consider two cases, forbidding a k-page embedding in each.

Lemma 2.9. *If $Z_{i,1}$ is before $Z_{i,k}$ for some i in $\{a, b, c\}$, then G_4 does not embed in k pages under π_4.*

Proof. The vertices $x_1, \ldots, x_k$ form twists with both $\{y_1\} \cup Z_{i,1}$ and $Z_{i,k} \cup \{y_n\}$, where we drop one vertex each of $Z_{i,1}$ and $Z_{i,k}$. By Lemma 2.8, the edges incident to x_r in the two twists are on the same page, which we call page r, for $1 \le r \le k$. By Lemma 2.4, the edges from x_r to all of $Z_{i,1} \cup Z_{i,k}$ are on page r.

Since $k \ge 3$, we may choose j with $2 \le j \le k-1$. If $Z_{i,j}$ is after $Z_{i,1}$ and before $Z_{i,k}$, then any edge from x_r to $Z_{i,j}$ crosses the edges from $x_1, \ldots, x_{r-1}$ to $\{y_1\} \cup Z_{i,1}$ and from $x_{r+1}, \ldots, x_k$ to $Z_{i,k} \cup \{y_n\}$. Therefore, all edges from x_r to $Z_{i,j}$ lie on page r.

If $Z_{i,j}$ is after $Z_{i,k}$, then by Lemma 2.6 both $Z_{i,k}$ and $Z_{i,1}$ are earlier than y_i. If $Z_{i,j}$ is before $Z_{i,1}$, then we have the symmetric situation with $Z_{i,1}$ and $Z_{i,k}$ later than y_i. Therefore, by symmetry we may assume that there are two types of children before y_i such that all edges from x_r to these two types lie on page r, for $1 \le r \le k$. The two sets of children are $Z_{i,1}$ and $Z_{i,j}$, where it may be that $j = k$.

Let z be the first vertex of $Z_{i,j}$. Since $y_i z$ crosses the edges from $X - \{x_j\}$ to the last vertex of $Z_{i,j}$, edge $y_i z$ lies on page j. Let z' be the first vertex of $Z_{i,1}$. Since $y_i z'$ crosses the edges from $X - \{x_1\}$ to the last vertex of $Z_{i,1}$, edge $y_i z'$ lies on page 1. Since $j \ne 1$, this contradicts Lemma 2.7. We conclude that G_4 does not embed in k pages under π. □

Lemma 2.10. *If $Z_{i,k}$ is before $Z_{i,1}$ for all i in $\{a, b, c\}$, then G_4 does not embed in k pages under π_4.*

Proof. For each i in $\{a, b, c\}$, Lemma 2.6, implies that y_i is after $Z_{i,k}$ and before $Z_{i,1}$. Again consider j with $2 \le j \le k-1$, and recall that $a < b < c$. The set $Z_{b,j}$ occurs before or after y_b; by symmetry, we may assume that $Z_{b,j}$ is before y_b (hence also before $Z_{b,k}$, by Lemma 2.6).

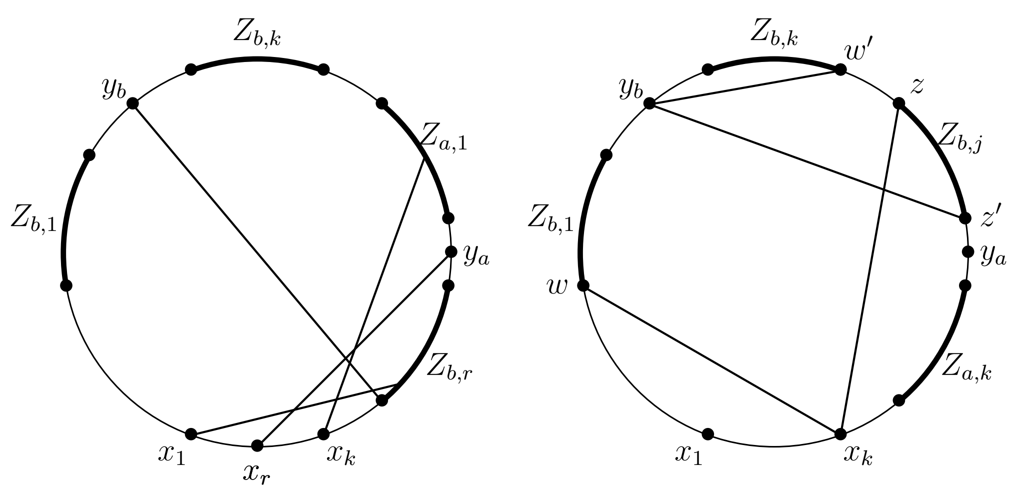

Figure 1: The cases of Lemma 2.10 (twist of size $k + 1$, crossing on a page).

Case 1: y_a is after some child of y_b (on the left in Fig. 1). Let $Z_{b,r}$ be the last set of k children of y_b before y_a. Note that $r > 1$. Now $y_b, x_1, \ldots, x_k$ form a twist of size $k + 1$ with

r vertices of $Z_{b,r}$, y_a, and $k-r$ vertices of $Z_{a,1}$ (we have observed that $Z_{a,1}$ is after y_a; this contribution is empty if $r=k$). Hence in this case G_4 does not embed in k pages under π_4.

Case 2: y_a is before all children of y_b (on the right in Fig. 1). Thus y_a is before $Z_{b,j}$, and $Z_{a,k}$ is before y_a. Since $j<k$, vertices $x_1,\ldots,x_k$ form a twist with $k-1$ vertices of $Z_{a,k}$ and the last vertex of $Z_{b,j}$ (call it z). Also $x_1,\ldots,x_k$ form a twist with $\{y_b\}\cup Z_{b,1}$. By Lemma 2.8, x_kz and x_kw lie on the same page, where w is the last vertex of $Z_{b,1}$.

Let w' be the first vertex of $Z_{b,k}$. Note that $x_1,\ldots,x_k$ form a twist with $(Z_{b,k}-\{w'\})\cup\{w\}$. Since y_bw' crosses the $k-1$ edges of this twist other than x_kw, edges y_bw' and x_kw lie on the same page.

Finally, by Lemma 2.7, y_bw' lies on the same page with y_bz', where z' is the first vertex of $Z_{b,j}$. Now y_bz' and x_kz lie on the same page, but they cross. Hence in this case also G_4 does not embed in k pages under π_4. □

Theorem 2.11. *The k-tree G does not embed in k pages.*

Proof. As argued earlier, it suffices to show that G_4 does not embed in k pages under the vertex ordering π_4. Lemmas 2.9 and 2.10 eliminate all possibilities for k-page embeddings and complete the proof. □

3 Construction of k-Page Embeddings

In this section, we present an algorithm that produces a k-page embedding of a k-tree G from a smooth tree-decomposition of width k in which the host tree has maximum degree at most 3. In the previous section it was important to view the vertex ordering for a book embedding as cyclic; here it is equally important to take the linear interpretation. We produce both a linear vertex ordering and an embedding of the k-tree in k pages under that ordering.

Since the bags in a tree-decomposition correspond bijectively to the nodes of the host tree, we may view the host tree as a tree in which the bags *are* the nodes. In the notation $(T,\mathbf{X})$, now T records the adjacency relation on the family of bags. This allows us to speak of "neighboring bags", and we can more clearly distinguish between vertices of G and nodes of T.

The inductive definition of a k-tree G with more than k vertices yields natural smooth tree-decompositions with width k. Adding one vertex to the trivial k-tree K_k creates a $(k+1)$-clique, which becomes a single bag for a host tree with one node. Each vertex x added to G then forms a new $(k+1)$-clique X, and X shares k vertices with a previous $(k+1)$-clique X' (there may be many choices for X'). Adding a new bag X adjacent to X' completes a smooth tree-decomposition of width k for the enlarged graph. It is easy to see inductively that every smooth tree-decomposition of a k-tree arises in this way.

A tree-decomposition constructed in this way has $n-k$ bags, where $n=|V(G)|$. To simplify the presentation of our algorithm and proof, we seek a tree-decomposition where the nodes correspond bijectively to the vertices of G. To achieve this, we modify the construction above by first growing the initial k-clique one vertex at a time.

Construction 3.1. *Modified tree-decomposition of a k-tree; construction and notation.*

Given an inductively constructed k-tree G, let the initial k-clique be $\{a_1, \dots, a_k\}$, and let $A_i = \{a_1, \dots, a_i\}$ for $1 \le i \le k$. Begin the tree-decomposition with a k-vertex path, letting the ith bag on the path be A_i. Treat A_1 as the root of the tree. Subsequently, add a bag for each new vertex of G as described above. When adding a vertex to G with a lower-case name, such as x, we use the corresponding upper-case designation (X) for the new bag. The vertex x in G is the one vertex of X that does not belong to the neighbor of X on the path to the root. We refer to that vertex as the *distinguished vertex* for the bag. This terminology is consistent also for the initial path, with a_i being the distinguished vertex of A_i.

Let $\mathbf{A} = A_1, \dots, A_k$. Deleting $\mathbf{A}$ yields a smooth tree-decomposition $(T', \mathbf{X}')$ of width k that corresponds to the inductive construction of G, where the initial k-clique is A_k and our modification makes A_k adjacent to the first node in T'. Call this node A_{k+1}; deleting one element of A_{k+1} yields A_k. By choosing A_{k+1} to be a leaf of T', we generate a modified tree-decomposition with the host tree T having the same maximum degree as T'.

In a rooted tree, the *parent* of a node other than the root is the next node on its path toward the root. Henceforth, we use $(T, \mathbf{X})$ to denote the modified tree-decomposition of G (obtained from the smooth decomposition $(T', \mathbf{X}')$) in which the host tree has root A_1 and the distinguished vertex of every bag other than A_1 is the unique vertex not in its parent (and the distinguished vertex of A_1 is a_1). Given also that T has maximum degree at most 3, we will produce a k-page embedding.

At times, we obtain a vertex of G as a difference of sets, without having a lower-case name for it. In this situation, to name the bag for which it is the distinguished vertex, we use an overbar operator. For example, when X is the parent of Y in T and $X \notin \mathbf{A}$, we have $|X - Y| = 1$, since $|X \cap Y| = k$, but the unique vertex v of $X - Y$ often is not the distinguished vertex of X. We use $\overline{X - Y}$ (or $\overline{v}$) to refer to the bag for which it is the distinguished vertex. If x *is* the distinguished vertex of X, then simply $\overline{x} = X$. □

The algorithm uses the modified tree-decomposition $(T, \mathbf{X})$ and its bijection from the bags in $\mathbf{X}$ to the vertices of G to produce a vertex ordering and a k-edge-coloring of G so that the endpoints of two edges with the same color do not occur alternately in the vertex ordering. Viewing the edge-coloring as an assignment of edges to k pages, this condition is precisely the condition for avoiding crossings on the pages. The idea is to use the correspondence between vertices and bags to color the edges of T using $k + 1$ colors, and then use the $(k + 1)$-edge-coloring of T to produce the k-edge-coloring of G.

In a graph, a *u, v-path* is a path with endpoints u and v. We say that X is an *ancestor* of Y and Y is a *descendant* of X if X lies on the A_1, Y-path in T. In a rooted tree containing a node X, the *subtree rooted at X* is the induced subtree with root X containing all nodes whose path to the root contains X.

We will use the next lemma to define the edge-coloring of G. Recall that x and y refer to the distinguished vertices of X and Y, respectively, and that the specification of distinguished vertices is a bijection from $\mathbf{X}$ to $V(G)$.

Lemma 3.2. *If $xy \in E(G)$, then one of $\{X, Y\}$ is an ancestor of the other in T.*

Proof. If $xy \in E(G)$, then x and y appear in a common bag. Since the bags containing a vertex of G induce a subtree of T, every bag in the X,Y-path in T contains x or y. The distinguished vertex of a bag Z cannot appear in any bag that is an ancestor of Z in T; it appears only in bags of the subtree rooted at Z. Therefore, we conclude that one of $\{X, Y\}$ is an ancestor of the other, and the distinguished vertex of the ancestor appears in all bags on the path down to the descendant. □

In a rooted tree, the parent of a node is its neighbor on the path to the root; its *children* are its other neighbors. When T has maximum degree at most 3 (and the root has degree 1), each node has at most two children. Designate the children of a bag in T as its left and right children, arbitarily (either may be missing). We refer to the subtrees of T rooted at the left and right children of a node X as the left and right *subtrees of* X. A *breadth-first search* of a rooted tree explores its nodes in order of distance from the root.

Algorithm 3.3. *k-page embedding of k-trees having a modified tree-decomposition* $(T, \mathbf{X})$ *obtained from smooth* $(T', \mathbf{X}')$ *of width k and maximum degree at most* 3.

First we produce a linear vertex ordering π from T. Initialize π to (a_1). Begin a breadth-first search of T from bag A_1. When exploring bag X, having already assigned its distinguished vertex x a position in π, place the vertex corresponding to its left child (if it has one) immediately before x in π and the vertex corresponding to its right child (if it has one) immediately after x in π.

When Z is an ancestor of Y, let $Z\!:\!Y$ denote the edge incident to Z on the Z,Y-path in T. Define a $(k+1)$-coloring f of $E(T)$ as follows. For an edge XY in T with X being the parent of Y (and x being the distinguished vertex of X), let

$$f(XY) = \begin{cases} j, & \text{if } XY = A_jA_{j+1} \text{ for } 1 \le j \le k \\ k+1, & \text{if } X \notin \mathbf{A} \text{ and } x \notin Y; \\ f(\overline{X-Y}\!:\!Y), & \text{if } X \notin \mathbf{A} \text{ and } x \in Y. \end{cases}$$

(Here $\overline{X-Y}$ is the bag whose distinguished vertex is the one vertex of $X - Y$.)

If $xy \in E(G)$, then by Lemma 3.2, we may assume by symmetry that X is an ancestor of Y. Define the coloring g on $E(G)$ by $g(xy) = f(X\!:\!Y)$. □

A fundamental property of f will make it easy to show that g does not use color $k+1$.

Lemma 3.4. *If X is the parent of Y in T, then $f(XY) = k+1$ if and only if $x \notin Y$, where x is the distinguished vertex of X.*

Proof. Consider a bag X closest to A_1 at which the claim fails. If $X \in \mathbf{A}$, then $x \in Y$ and $f(XY) \le k$, so we may assume that $X \notin \mathbf{A}$. Now $|X - Y| = 1$. If the one vertex of $X - Y$ is x, then by definition $f(XY) = k + 1$. Otherwise, $X - Y = \{z\} \ne \{x\}$, where z is the distinguished vertex for Z. Here Z is an ancestor of X, and $f(XY) = f(Z\!:\!Y)$. Let W be the child of Z on the Z,Y-path in T, so $Z\!:\!Y = ZW$. The choice of X yields $f(ZW) = k+1$ if and only if $z \notin W$. However, $z \in Z \cap X$, so $z \in W$, and $f(XY) = f(ZW) \ne k + 1$. □

Lemma 3.5. *No edge in G has color $k+1$ under g.*

Proof. As noted previously, when $xy \in E(G)$ we may assume by symmetry that X is an ancestor of Y, and then $g(xy) = f(X\!:\!Y)$. By the definition of f, we have $g(xy) = f(XZ)$, where Z is the child of X on the X,Y-path in T. By Lemma 3.4, we have $g(xy) = k+1$ if and only if $x \notin Z$. If $x \notin Z$, then x appears in no bag in the subtree of X rooted at Z; in this case x and y could not appear in a bag together and could not form an edge. □

Lemma 3.6. *If X is an ancestor of Y such that $x \in Y$, then $f(X\!:\!Y)$ appears on no edge of the X,Y-path in T except the initial edge $X\!:\!Y$.*

Proof. Consider a bag X closest to A_1 at which the claim fails. Let $j = f(X\!:\!Y)$. Since $x \in Y$, Lemma 3.4 implies that $j \le k$. If color j appears again on the X,Y-path, then let ZZ' be the edge on which it first reappears, with Z the parent of Z'. Since this is a reappearance of j, the bag Z cannot be A_j.

Since $j \le k$, the definition of f yields $j = f(W\!:\!Z')$, where $\{w\} = Z - Z'$. Since $w \notin Z'$, also $w \notin Y$. Since $x \in Y$, we have $w \ne x$, and therefore $W \ne X$. We conclude that W is an ancestor of X, since ZZ' was the first reappearance of j on the X,Y-path. Since $j = f(W\!:\!Z')$ and $X\!:\!Y$ is an edge of the W,Z'-path in T, we have contradicted the choice of X as the failure closest to A_1. □

Theorem 3.7. *Algorithm 3.3 produces a k-page embedding.*

Proof. By Lemma 3.5, g is a k-edge-coloring of G. It remains to show that g does not give the same color to edges whose endpoints alternate in π. Let xy and uv be edges whose endpoints alternate in π. It suffices to show that $g(xy) \ne g(uv)$.

By Lemma 3.2 and symmetry, we may assume that X is an ancestor of Y and U is an ancestor of V. Now $g(xy) = f(X\!:\!Y)$ and $g(uv) = f(U\!:\!V)$. Furthermore, since x and y must appear in a bag together, and y appears only in bags in the subtree rooted at Y, we have $x \in Y$. Similarly, $u \in V$.

By the construction of π in Algorithm 3.3, the vertices for bags in the left subtree of a given bag Z comprise a consecutive segment in π immediately before its distinguished vertex z, and those for bags in the right subtree of Z comprise a consecutive segment immediately after z in π. Since the algorithm is symmetric with respect to left and right, we may assume that Y is in the right subtree of X, and therefore $\pi(x) < \pi(y)$. Since the vertices for bags in the right subtree of X immediately follow x in π and uv crosses xy, the right subtree of X must contain U or V.

Case 1: U is in the right subtree of X. Here V is also in the right subtree of X, since U is an ancestor of V. If V is in the left subtree of U, then $\pi(x) < \pi(v) < \pi(y) < \pi(u)$. Since the vertices for bags in the left subtree of U (including v) appear just before u in π, also Y is in the left subtree of U. Now U lies on the X,Y-path in T. By Lemma 3.6, $f(X\!:\!Y)$ cannot reappear on the X,Y-path in T. Therefore, $g(uv) \ne g(xy)$.

If V is in the right subtree of U, then $\pi(x) < \pi(u) < \pi(y) < \pi(v)$, and Y is in the right subtree of U. Again, U lies along the X,Y-path in T, and Lemma 3.6 yields $g(uv) \ne g(xy)$.

Case 2: U is not in the right subtree of X. Since one of $\{u, v\}$ is between x and y in π, the bag V is in the right subtree of X. Since U is an ancestor of V but is not in the right subtree of X, the bag U is an ancestor of X. Now X lies along the U, V-path in T. By Lemma 3.6, again we conclude that $g(uv) \neq g(xy)$.

We have shown that g and π yield a book embedding of G in k pages. □

Given the smooth tree-decomposition used by the algorithm, the computations by which the algorithm produces the k-page embedding can easily be implemented to run in constant time per edge. Since k is fixed, this is linear in the number of vertices. A smooth tree-decomposition is generated from a simplicial elimination ordering, which for chordal graphs with bounded clique number can be found in linear time.

4 An Open Problem

In Section 3, we constructed a k-page embedding for any k-tree having a smooth tree-decomposition of width k in a host tree with maximum degree at most 3. In [9], the same conclusion was obtained when the host tree has maximum degree 2. As mentioned in the introduction, this suggests the question of determining $m(k)$, the maximum l such that every k-tree having a smooth tree-decomposition of width k in a host tree with maximum degree at most l embeds in k pages.

In Section 2, we constructed a k-tree G having no k-page embedding. It is easy to find a smooth tree-decomposition of width k for G with a host tree of maximum degree $k + 2$. For $1 \leq i \leq kn$, let $X_i = X \cup \{y_i\}$. Form a path in T with nodes $X_1, \ldots, X_{kn}$. For each X_i and x_j, grow from node X_i a path whose nodes are formed by adding to $X_i - \{x_j\}$ one child of type (i, j). This completes a smooth tree-decomposition of width k for G, and the nodes $x_2, \ldots, x_{kn-1}$ have degree $k + 2$ in the host tree.

To conclude that $3 \leq m(k) \leq k + 1$, it suffices to show that G has no smooth tree-decomposition of width k in a host tree with maximum degree less than $k + 2$. As we have remarked, every bag in a smooth tree-decomposition of width k for a k-tree is a $(k + 1)$-clique. All $(k + 1)$-cliques in G have the form $X \cup \{y_i\}$ (call these X-bags) or the form $(X - \{x_j\}) \cup \{y_i, z\}$, where z is a child of type (i, j) (call these Z-bags).

Adjacent bags in the host tree share exactly k vertices. Thus an X-bag and a Z-bag can be adjacent only if they have the same value of i. Since each child is in only one $(k+1)$-clique, Z-bags containing children of different types are not adjacent.

Since the host tree is connected, we conclude from these observations that each X-bag is adjacent to some Z-bag for each j with $1 \leq j \leq k$, giving it at least k neighbors. Also, the subgraph induced by the X-bags must be connected. Since $k \geq 3$, some X-bag has two additional neighbors among the X-bags.

Acknowledgment

We thank the referees for suggestions that led to an improved presentation of the results.

References

[1] F. Bernhart and P.C. Kainen, The book thickness of a graph. *J. Combin. Theory Ser. B* 27 (1979), 320–331.

[2] F.R.K. Chung, F.T. Leighton, A.L. Rosenberg, Embedding graphs in books: a layout problem with applications to VLSI design. *SIAM J. Algebr. Discr. Meth.* 8 (1987), 33–58.

[3] H. Enomoto, T. Nakamigawa, K. Ota, On the pagenumber of complete bipartite graphs. *J. Combin. Theory Ser. B* 71 (1997), 111–120.

[4] J. Ganley and L. Heath, The pagenumber of k-tree is $O(k)$. *Discr. Appl. Math.* 109 (2001), 215–221.

[5] L.S. Heath, S. Istrail, The pagenumber of genus g graphs is $O(g)$. *J. Assoc. Comput. Mach.* 39 (1992), 479–501.

[6] S.M. Malitz, Genus g graphs have pagenumber $O(\sqrt{g})$. *J. Algorithms* 17 (1994), 85–109.

[7] D.J. Muder, M.L. Weaver, D.B. West, Pagenumber of complete bipartite graphs. *J. Graph Theory* 12 (1988), 469–489.

[8] P. Scheffler, The graphs of tree-width k are exactly the partial k-trees, manuscript 1986; see also, Linear-time algorithms for NP-complete problems restricted to partial k-trees, Report MATH, 87-03. Akademie der Wissenschaften der DDR, Karl-Weierstrass-Institut für Mathematik (Berlin, 1987), 48 pp.

[9] M. Togasaki and K. Yamazaki, Pagenumber of pathwidth-k graphs and strong pathwidth-k graphs. *Discr. Math.* 259 (2002), 361–368.

[10] M. Yannakakis, Embedding planar graphs in four pages. 18th Annual ACM Symposium on Theory of Computing (Berkeley, CA, 1986) *J. Comput. System Sci.* 38 (1989), 36–67.