SIAM J. ALG. DISC. METH.
Vol. 8, No. 1, January 1987

EMBEDDING GRAPHS IN BOOKS: A LAYOUT PROBLEM WITH APPLICATIONS TO VLSI DESIGN*

FAN R. K. CHUNG†, FRANK THOMSON LEIGHTON‡ AND ARNOLD L. ROSENBERG§

Abstract. We study the graph-theoretic problem of embedding a graph in a book with its vertices in a line along the spine of the book and its edges on the pages in such a way that edges residing on the same page do not cross. This problem abstracts layout problems arising in the routing of multilayer printed circuit boards and in the design of fault-tolerant processor arrays. In devising an embedding, one strives to minimize both the number of pages used and the "cutwidth" of the edges on each page. Our main results (1) present optimal embeddings of a variety of families of graphs; (2) exhibit situations where one can achieve small pagenumber only at the expense of large cutwidth; and (3) establish bounds on the minimum pagenumber of a graph based on various structural properties of the graph. Notable in the last category are proofs that (a) every n-vertex d-valent graph can be embedded using $O(dn^{1/2})$ pages, and (b) for every $d>2$ and all large n, there are n-vertex d-valent graphs whose pagenumber is at least

$$\Omega\left(\frac{n^{1/2-1/d}}{\log^2 n}\right).$$

Key words. book embeddings, arrays of processors, fault-tolerant computing

AMS(MOS) subject classifications. 05C99, 94C15, 68C25

1. Introduction.

1.1. The problem. We study here a graph embedding problem that can be viewed in a variety of ways. We start with an undirected graph G.

Formulation 1. To embed G in a book, with its vertices on the spine of the book and its edges on the pages, in such a way that edges residing on the same page do not cross.

We seek embeddings of graphs in books that use pages that are few in number and small in width. (The *width of a page* is the maximum number of edges that cross any line perpendicular to the spine of the book. The *width of a book embedding* is the maximum width of any page of the book. The *cumulative pagewidth of a book embedding* is the sum of the widths of all the pages.) The results we present are of four types:

(1) We characterize graphs that can be embedded in books having one or two pages. For instance, the one-page graphs are precisely the outerplanar graphs. (A graph is *outerplanar* if its vertices can be placed on a circle in such a way that its edges are noncrossing chords of the circle.)

(2) We find upper bounds on the number of pages required by graphs of valence (i.e., vertex-degree) at most d, and we show that these bounds are often approached

* Received by the editors December 17, 1984, and in revised form February 20, 1986. A portion of this paper was presented at the Thirteenth International IEEE Symposium on Fault-Tolerant Computing and a portion at the Fifth International Symposium on Theory and Applications of Graphs.

† Combinatorial Optimization Department, Bell Communications Research, Morristown, New Jersey 07960.

‡ Department of Mathematics and Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139. A portion of this research was done while this author was visiting Bell Laboratories, Murray Hill, New Jersey. The research of this author was supported in part by a National Science Foundation Presidential Young Investigator Award with matching funds from Xerox, by Defence Advanced Research Projects Agency contract N00014-80-C-0622, and by Air Force contract OSR-82-0326.

§ Department of Computer Science, Duke University, Durham, North Carolina 27706. A portion of this research was done while this author was visiting Bell Laboratories, Murray Hill, New Jersey. The research of this author was supported in part by National Science Foundation grants MCS-81-16522, MCS-83-01213 and DMC-85-04308.

by specific d-valent graphs. For example, every n-vertex $(d>2)$-valent graph can be embedded in a book with $\min(n/2, O(dn^{1/2}))$ pages (graphs of valence $d \leqq 2$ require only one page); and there exist such graphs that cannot be embedded in fewer than $\Omega(n^{1/2-1/d}/\log^2 n)$ pages. (All logarithms are to the base 2.)

(3) We find optimal or near-optimal embeddings of a variety of families of graphs, including trees, grids, X-trees, cyclic shifters, permutation networks, and complete graphs. For example, every n-vertex d-ary tree can be embedded in a book having one page, of width $\lceil d/2 \rceil \cdot \log n$.

(4) We exhibit two instances of a tradeoff between the number of pages and the widths of the pages. For example, every one-page embedding of the depth-n "ladder" graph requires width $n/2$, but there are width-2 two-page embeddings for this graph.

1.2. The origins of the problem. The problem has several origins.

Sorting with parallel stacks. Even and Itai [10] and Tarjan [24] study the problem of how to realize fixed permutations of $\{1, \cdots, n\}$ with noncommunicating stacks. Initially each number is PUSHed, in the order 1 to n, onto any one of the stacks. After all the numbers are on stacks, the stacks are POPped to form the permutation. One can view this problem graph-theoretically as follows. Say we are studying permutations of $\{1, \cdots, n\}$. Then consider the bipartite graph G_n with vertices $\{a_1, \cdots, a_n, b_1, \cdots, b_n\}$ and edges connecting each a_i to b_i. The problem of realizing the permutation π on $\{1, \cdots, n\}$ with k parallel stacks is equivalent to embedding G_n in a k-page book, with its vertices embedded in the order $a_1, \cdots, a_n, b_{\pi(1)}, \cdots, b_{\pi(n)}$.

Single-row routing. In an attempt to simplify the problem of routing multilayer printed circuit boards (PCBs), So [22] decomposed the problem in the following way. In his variant, one arranges the circuit elements in a regular grid, with wiring channels separating rows and columns of elements. One then decomposes the circuit's net lists (possibly by adding new dummy elements) so that every net connects elements in a single row or in a single column. The PCB can now be routed by routing each of its rows and each of its columns independently. The variant of this scenario that does not allow a net to run from the top of a row around to its bottom nor to change layers en route [20] corresponds directly to our embedding problem applied to small-valence graphs.

Fault-tolerant processor arrays. The DIOGENES approach to the design of fault-tolerant arrays of identical processing elements (PEs, for short) [7], [21] uses "stacks of wires" to configure around faulty PEs. In broad terms, the approach works as follows. The PEs are laid out in a (logical, if not physical) line, with some number of "bundles" of wires running above the line of PEs. One then scans along the line of PEs to determine which are faulty and which are fault-free. As each good PE is encountered, it is hooked into the bundles of wires through a network of switches, thereby connecting that PE to the fault-free PEs that have already been found and preparing it for eventual connection to those that will be found. To simplify the configuration process, each bundle is made to behave like a stack, as illustrated by the following embedding of a complete depth-d binary tree (see Fig. 1). One uses a single bundle whose wires are numbered $1, \cdots, d$. After determining which of the PEs

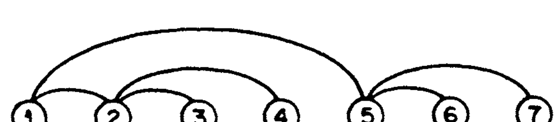


FIG. 1. *The preorder 1-page layout of the depth-3 complete binary tree.*

are good and which are faulty, one proceeds down the line of PEs from right to left. As a good PE that is to be a leaf of the tree is encountered, it is connected to line 1 in the bundle, simultaneously having lines 1 through $d-1$ "shift up," to "become" lines 2 through d; switches disconnect the left parts of the lines from the right parts so that vertex-to-vertex connectivity remains correct. The bundle has thus behaved like a stack being PUSHed. As a good PE that is to be a nonleaf of the tree is encountered, it is connected to the stack/bundle in two stages. First it is connected to lines 1 and 2 of the bundle, simultaneously having lines 3 through d "shift down" to "become" lines 1 through $d-2$; again switches ensure that proper vertex-to-vertex connectivity is maintained. The bundle here behaves like a stack being twice POPped. Second, the PE PUSHes a connection onto the stack. In this scenario, POPs amount to having a PE adopt two children that lie to its right in the line, while PUSHes amount to having the PE request to be adopted by some higher level vertex that lies to its left. The process just described lays the tree out in *preorder* and, hence, uses at most d lines.

Although not directly related to the research in this paper, the following relationship to Turing-machine graphs is also of interest.

Turing-machine graphs. One can construct a T-vertex graph that "models" a given T-step Turing machine computation, as follows. Each vertex of the graph corresponds to a step of the computation; vertices t_1 and t_2 are adjacent in the graph just if one of the machine's tape heads visits the same tape square at times t_1 and t_2, but at no intervening time. One can easily show that every k-tape Turing-machine graph is embeddable in a $2k$-page book. Hence, a characterization of graphs that are embeddable in books with a given number of pages might have applications to complexity theory. For example, a proof that such graphs have small bisection width would lead to several interesting complexity-theoretic results.

1.3. Additional formulations. Our perusal of the origins of the problem affords us additional formulations with which to hone our intuition.

Formulation 2. To place the vertices of G in a line and to assign its edges to stacks in such a way that the stacks can be used to lay out the edges.

Formulation 3. To embed the graph G so that its vertices lie on a circle and its edges are chords of the circle; to assign the chords to layers so that edges/chords on the same layer do not cross.

Formulation 3 combines the insights of [10] and [22], and yields a simple characterization of the 1-page embeddable graphs.

THEOREM 1.1 [3]. *A graph can be embedded in a one-page book if, and only if, it is outerplanar.*

Proof sketch. A graph G is outerplanar just when its vertices can be placed on a circle so that its edges become noncrossing chords of the circle.

If G is outerplanar and is laid out on a circle as above, then cutting the circle between any two vertices and opening it out to form a line yields a one-page embedding of G.

Conversely, given a one-page embedding of G, passing a line through the vertices of G in their order in the embedding and joining the ends of the line together to form a circle demonstrates G's outerplanarity. □

This characterization suggests yet another formulation.

Formulation 4. To decompose G into outerplanar graphs all of whose outerplanarity is witnessed by the same embedding of G's vertices.

1.4. Reflections from the facets. The many formulations of our problem suggest at least two variants: the first assumes that the layout of the vertices is fixed (as in sorting with parallel stacks and single-row routing); the second leaves the arrangement of the vertices as part of the problem (as in the construction of fault-tolerant processor arrays). We focus in this paper on the harder version of the problem, in which the placement of the vertices is not given.

The many facets of our problem further allow us to draw on results obtained in a variety of contexts.

The first result follows from Tarjan's analysis of the number of stacks that are required to compute a given permutation of $\{1, \cdots, n\}$. We translate the result to our graph-theoretic setting.

THEOREM 1.2 [24]. *Let the graph G have vertices* $\{a_1, \cdots, a_n, b_1, \cdots, b_n\}$ *and edges connecting each* a_i *to* b_i. *Let* π_1 *and* π_2 *be permutations of* $\{1, \cdots, n\}$. *Let the vertices of G be placed in a line in the order* $a_{\pi_1(1)}, \cdots, a_{\pi_1(n)}, b_{\pi_2(n)}, \cdots, b_{\pi_2(1)}$. *The number of pages needed to embed G given this placement of its vertices is precisely the length of the longest sequence of b-vertices whose indices are similarly ordered with their a-mates.*

The next result is immediate from the following important observation by Even and Itai [10]: The problem

> To minimize the number of pages required to embed a graph G in a book, when the ordering of G's vertices along the spine of the book is prespecified

is equivalent to the problem

> To find a minimum vertex-coloring for a *circle graph* (which is the intersection-graph for chords of a circle).

The correspondence between the two problems is best seen from Formulation 3 of the book-embedding problem. Garey et al. [13] show that the coloring problem for circle graphs is NP-complete.

THEOREM 1.3 [10], [13]. *The following problem is* NP-*complete*: *Given a graph G, an ordering of the vertices of G, and an integer k, decide whether or not G can be embedded in a k-page book when its vertices are placed along the spine of the book in the specified order.*

See [1] for a related result.

2. Sample embeddings and helpful principles. The problems of embedding small-valence graphs and of analyzing given embeddings are harder than they seem at first. In order to help the reader develop intuition for the remaining sections, we now present helpful strategies for obtaining bounds, and we illustrate them with sample embeddings and their analyses.

2.1. An embedding strategy. Formulation 3 of our problem suggests a strategy for embedding graphs in books, that is valuable both in finding and describing embeddings. In order to embed the graph G in a book, the strategy advocates:

1. embedding the vertices of G in a circle by finding a hamiltonian cycle in G or in some *edge-augmentation* of G (that is, a graph obtained from G by adding zero or more new edges);

2. assigning the edges of G (which are easily transformed into chords of the circle) to pages in some noncrossing manner, perhaps by coloring the vertices of the associated circle graph.

Reinforcing the intuition behind this heuristic is the fact that hamiltonian cycles add virtually no cost to an embedding: a cycle adds only 1 to the cutwidth of a layout (since one snips it), and it does not interfere with any other edges, so it does not increase the pagenumber of the embedding.

2.2. Two strategies for lower bounds. The first strategy for bounding pagenumber from below resides in the following result, which follows from Theorem 4.1 (q.v.).

THEOREM 2.1. *If the graph G is not planar, then it cannot be embedded in fewer than three pages.*

The second bounding strategy revolves around the properties of *matching graphs.* For our purposes a matching graph is a regular univalent graph (hence has an even number of vertices). If we view a matching graph as being bipartite, we can naturally associate with it a permutation π: the graph's "input" vertices are labelled $1, \cdots, n$ and are connected, respectively, to "output" vertices $\pi(1), \cdots, \pi(n)$. We shall encounter situations when analyzing a specific layout or a class of layouts of a graph G wherein we can assert that G must contain as a subgraph a matching graph G^* such that

1. the input vertices of G^* all lie to one side of its output vertices;
2. the input and output vertices of G^* are *similarly ordered,* in the sense that, if the inputs are laid out in the order $v_1, v_2, \cdots, v_n$, then the outputs appear in the order $\pi(v_1), \pi(v_2), \cdots, \pi(v_n)$.

When the existence of such a G^* can be established, we can infer that this (class of) embedding(s) of G requires n pages. The reasoning leading to this conclusion bears a strong kinship with the reasoning that Tarjan [24] and Even and Itai [10] used when studying sequences of integers that can be sorted using n stacks.

The lower bounds we obtain via matching subgraphs are among the best we derive in the paper.

2.3. Sample embeddings.

2.3.1. The pinwheel graph. The embeddings we shall be presenting in the course of our study will bear out the value of the hamiltonian-cycle embedding strategy. The following example illustrates how careful one must be to search for a *good* hamiltonian cycle.

The *depth-n pinwheel graph* $P(n)$ has $2n$ vertices

$$\{a_1, a_2, \cdots, a_n\}$$

and

$$\{b_1, b_2, \cdots, b_n\}$$

and edges connecting each pair of vertices of the form

$$\begin{aligned} &a_i - b_i, && 1 \leqq i \leqq n,\\ &a_i - b_{n-i+1}, && 1 \leqq i \leqq n,\\ &a_i - a_{i+1}, && 1 \leqq i < n,\\ &b_i - b_{i+1}, && 1 \leqq i < n. \end{aligned}$$

See Fig. 2.

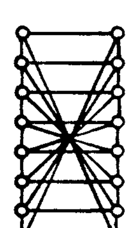


FIG. 2. *The depth-8 pinwheel graph $P(8)$.*

When $n>2$, the graph $P(n)$ is not planar: $P(3)=K_{3,3}$. We shall see in § 4 (Theorem 4.1) that this nonplanarity precludes $P(n)$'s being embedded in fewer than three pages. Can one do this well? The obvious hamiltonian cycle—that goes "down the a's and up the b's"—leads to an embedding using roughly n pages, one of width proportional to n. However, if one studies the structure of pinwheels more carefully, then one discovers a hamiltonian cycle that leads to a 3-page embedding for $P(n)$, independent of n, in which the three pages have widths 2, 4, and 4, respectively.

PROPOSITION 2.2. *The graph $P(n)$ is 3-page embeddable in such a way that one page has width 2 and the other two have width 4 each.*[1]

Proof. The embedding. One obtains the desired cycle by rearranging the "butterflies" that comprise $P(n)$, as follows. We use asterisks to divide the cycle into segments that facilitate the analysis of the induced embedding. Assume for simplicity that n is even.

$$a_1-b_1-a_n-b_n-b_{n-1}-a_{n-1}-b_2-a_2-*-a_3-b_3-a_{n-2}-b_{n-2}-b_{n-3}-a_{n-3}-b_4-a_4-*-$$

$$\cdots-*-a_{n/2-1}-b_{n/2-1}-a_{n/2+2}-b_{n/2+2}-b_{n/2+1}-a_{n/2+1}-b_{n/2}-a_{n/2}.$$

Each segment of the cycle comprises two adjacent butterflies, the second recorded in reverse order of the first. Let us linearize the vertices of $P(n)$ by snipping the cycle between a_1 and $a_{n/2}$, as suggested by the way we have written the cycle.

The analysis. For each segment, we need one width-2 page to hold the butterfly edges. A second, width-4, page suffices to hold the edges that connect any single pair of adjacent butterflies. But, if this page is used for the edges that connect the ith and $(i+1)$th butterflies, it cannot also hold the edges between the $(i+1)$th and $(i+2)$th butterflies; for this next pair we need yet a third width-4 page. We need no additional pages, since the latter two can alternate joining up adjacent butterflies. Thus the cycle we have presented leads to a layout with the claimed efficiency. □

2.3.2. The sum of triangles graph. The next graph we look at is interesting because of the techniques that are needed to analyze and bound the efficiency of its embeddings. In particular, it will afford our first use of matching subgraphs to obtain a lower bound on pagenumber.

The *depth-n sum of triangles graph* $T(n)$ has vertices

$$\{a_i,\, b_i,\, c_i\colon 1\leqq i\leqq n\}$$

and edges connecting each triple a_i, b_i, c_i into a triangle.

THEOREM 2.3. *The graph $T(n)$ is 1-page embeddable, with width 2. However, if one insists that $T(n)$ be laid out "by columns", so that the vertices $\{a_i\}$ are all contiguous, and so are the vertices $\{b_i\}$ and the vertices $\{c_i\}$, then $T(n)$ is $3n^{1/3}$-page embeddable, and this is optimal, within a factor of* 3.

Proof. The unrestricted layout of $T(n)$ being obvious (triangle by triangle), we restrict attention to layouts of $T(n)$ that keep all the a-vertices, all the b-vertices, and all the c-vertices contiguous, so we can refer with no ambiguity to the *a-block* of vertices, the *b-block*, and the *c-block*. We shall henceforth assume such a layout without further explicit mention. We shall also assume, for simplicity, that n is a perfect cube.

[1] One of the referees has found a 3-page embedding of $P(n)$ with pagewidths 4, 3, and 1, respectively.

The upper bound. Begin with each of the blocks of vertices in order: the a-block lies in the order

$$a_1, a_2, \cdots, a_n,$$

the b-block lies in the order

$$b_1, b_2, \cdots, b_n,$$

and the c-block lies in the order

$$c_1, c_2, \cdots, c_n.$$

Partition each of these blocks into $n^{1/3}$ *segments*, each segment being further subdivided into $n^{1/3}$ *runs* of $n^{1/3}$ vertices each. Each block thus has the form:

$$(R_1 \cdots R_d)(R_{d+1} \cdots R_e) \cdots (R_{f+1} \cdots R_g),$$

where runs are grouped by parentheses into segments. To this point, corresponding runs in corresponding segments are similarly ordered in each block.

Now begin rearranging the vertices within blocks as follows. Assume without loss of generality that the a-block lies to the left of the b-block, which lies to the left of the c-block.

(a) Leave the a-block as is.

(b) Rearrange the b-block by reversing the order of its segments, and reversing the order of the runs within each segment (but keeping vertices within runs in order, as before). The block will now look like:

$$(R_g \cdots R_{f+1}) \cdots (R_e \cdots R_{d+1})(R_d \cdots R_1).$$

(c) Rearrange the c-block by reversing the order of the runs in each segment and reversing the order of vertices within each run (but keeping the original order of the segments). If we let $\bar{R}$ denote the run obtained by reversing the vertices of the run R, then the block will now look like:

$$(\bar{R}_d \cdots \bar{R}_1)(\bar{R}_e \cdots \bar{R}_{d+1}) \cdots (\bar{R}_g \cdots \bar{R}_{f+1}).$$

Now let us add in the edges of $T(n)$ and keep track of how many pages we can get by with. When we add the edges that connect the a-block to the b-block, we note that a single page will accommodate one edge from each a-run to its corresponding b-run; since each a-run emits $n^{1/3}$ edges to the b-block, we need only this many pages to realize the a-to-b edges. When we add the edges that connect the b-block to the c-block, we note that a single page will accommodate the edges from one b-run per segment to its corresponding c-run; since there are $n^{1/3}$ runs per segment, we need only this many pages to realize the b-to-c edges. When we add the edges that connect the a-block to the c-block, we note that a single page will accommodate all the edges from one a-segment to its corresponding c-segment. Since there are only $n^{1/3}$ segments per block, we need only this many pages to implement the a-to-c edges. We have thus used $3n^{1/3}$ pages to implement all of $T(n)$'s edges.

The lower bound. Without loss of generality, say that we have $T(n)$ laid out in an a-block, a b-block, and a c-block, in that order. If we concentrate on any pair of blocks, we have a subgraph of $T(n)$ that is a matching graph whose "inputs" and "outputs" are laid out disjointly. Using the obvious correspondence between similarly (resp., oppositely) ordered inputs and outputs on the one hand, and increasing (resp., decreasing) subsequences of an integer sequence on the other hand, we note the following variant of a well-known result of Erdos and Szekeres [9].

LEMMA 2.4 [9]. *Let A and B be orderings of the integers* $\{1, 2, \cdots, n\}$. *If sequences A and B share no similarly ordered subsequence of length greater than k, then they share an oppositely ordered subsequence of length at least* n/k.

Now assume for contradiction that our layout of $T(n)$ requires fewer than $n^{1/3}$ pages. As we have noted in § 2.2, this implies that the a-block and the b-block share no similarly ordered subsequence of vertices of length as great as $n^{1/3}$. By Lemma 2.4, therefore, these blocks must share an oppositely ordered subsequence of length greater than $n^{2/3}$. Look now at the length-$n^{2/3}$ subsequence of the c-block that corresponds to the oppositely ordered subsequence of the a-block and the b-block. By Lemma 2.4, this subsequence of the c-block must share with the corresponding subsequence of the a-block either a similarly ordered subsequence of length

$$(n^{2/3})^{1/2} = n^{1/3}$$

or an oppositely ordered subsequence of the same length. In the former case, the edges between the a-block and the c-block cannot be realized with fewer than $n^{1/3}$ pages; in the latter case, the edges between the b-block and the c-block require this many pages. This contradicts our assumption that fewer than $n^{1/3}$ pages suffices to realize the layout of $T(n)$. □

3. Specific efficient layouts. Our attention to this point has been on establishing general analysis techniques and bounds. We now turn to the task of finding efficient layouts of a number of familiar graph families. We shall find in § 4 that these families have much more modest pagenumber demands than random graphs.

3.1. Trees. In § 1.2 we presented an embedding of the complete binary tree that turns out to be optimal in both pagenumber (one) and pagewidth ($\log n$). (Optimality of width follows from [5].) It is not hard to show that all trees enjoy embeddings that are approximately as efficient as those of complete trees.

PROPOSITION 3.1. *Every n-vertex d-ary tree can be embedded in one page of width at most*

$$\min\left(n-1, \left\lceil \frac{d}{2} \right\rceil \cdot \left\lceil \frac{\log n}{\log 3/2} \right\rceil\right).$$

Proof sketch. Let G be a graph. One *adds a fringe* to a vertex v of G by appending to v a line of (possibly 0) vertices:

$$v - v_1 - v_2 - \cdots - v_r, \qquad r \geqq 0.$$

A *fringing of G* is a graph obtained by adding a fringe to each vertex of G.

Concentrate on a single vertex v of G. Say that when G is laid out, v is flanked by vertices u and w. Let v have two fringes, $v_1, \cdots, v_r$ and $v_1', \cdots, v_s'$ (one or both of which may be empty). Lay the fringes out either in the indicated order between v and w or in reverse order between u and v. To choose the side of v: place the first fringe on that side of v where the fewest edges of G cross or meet v (as in the conventional definition of cutwidth); place the second fringe using the same criterion in the now-augmented embedding. This strategy increases the cumulative width of the embedding by at most 1, while leaving the number of pages (one) unchanged.

An easy induction verifies that any d-ary tree T can be "built" by levels, by starting with a single vertex and "double"-fringing the graph at most

$$\left\lceil \frac{d}{2} \right\rceil \cdot \left\lceil \frac{\log |T|}{\log 3/2} \right\rceil$$

times. Our bounds on pagewidth follow from this method of constructing the tree and from the fact that the tree has at most $n-1$ edges. Details are left to the reader. □

Proposition 3.1 seeks to optimize the worst-case tree embedding. Dolev and Trickey [8] present an algorithm for finding a width-optimal one-page embedding for an individual tree.

3.2. Square grids. Square grids are planar and subhamiltonian, hence 2-page embeddable. (We verify this claim in Theorem 4.1.) The augmented hamiltonian cycle formed by row-by-row alternated east-to-west and west-to-east sweeps, as indicated in Fig. 3(a), leads to the 2-page embedding shown in Fig. 3(b). This embedding is optimal both in number of pages—the grid is not outerplanar—and in the cumulative width of the pages—the $n \times n$ grid has minimum bisection width n.

PROPOSITION 3.2. *The $n \times n$ square grid admits a 2-page embedding, each page of width n. This embedding is optimal in pagenumber and is within a factor of 2 of optimal in pagewidth.*

3.3. X-Trees. The *depth-d X-tree* $X(d)$ is the edge augmentation of the depth-d complete binary tree that adds edges going across each level of the tree in left-to-right order (see Fig. 4(a)).

X-trees are planar and subhamiltonian, hence admit 2-page embeddings. While it is easy to find a 2-page embedding for $X(d)$—the cycle that runs across levels in alternating orders yields one such—it is difficult to find one that has width $o(n)$ (where $n=2^d-1$ is the number of vertices in $X(d)$), despite the fact that $X(d)$ has a bisector of size d. However, the edge-augmentation of the X-tree depicted in Fig. 4(a), with

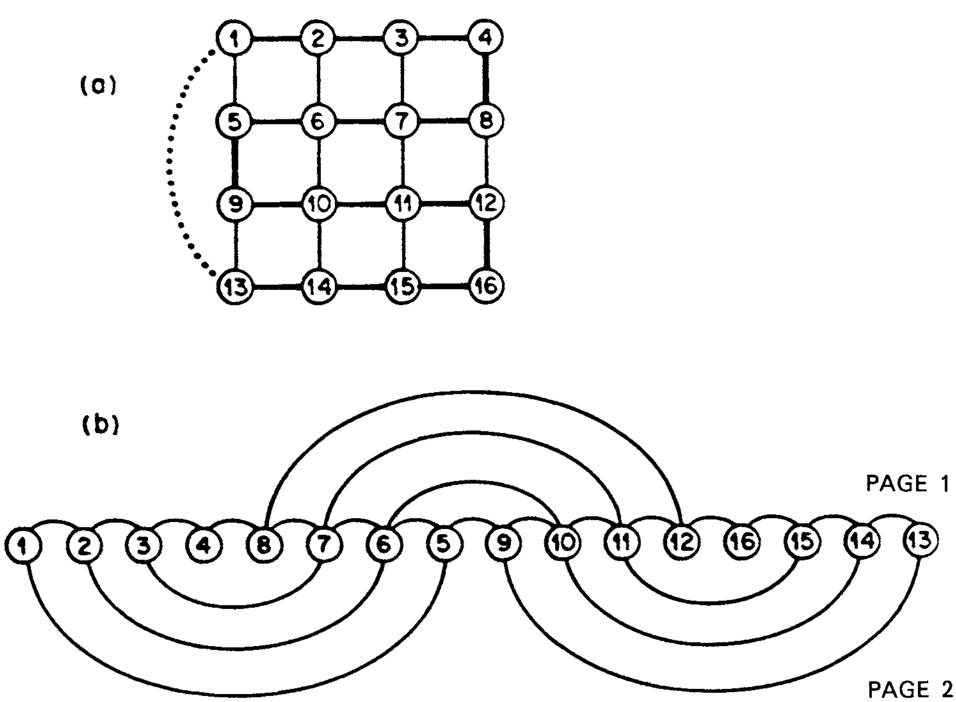


FIG. 3. (a) *The* 4×4 *grid and its efficient hamiltonian cycle.* (*In all the figures, the edges added to create an efficient cycle are shown as dotted lines; the graph edges comprising the cycles are thickened.*) (b) *The* 2-*page layout of the grid induced by the cycle.*

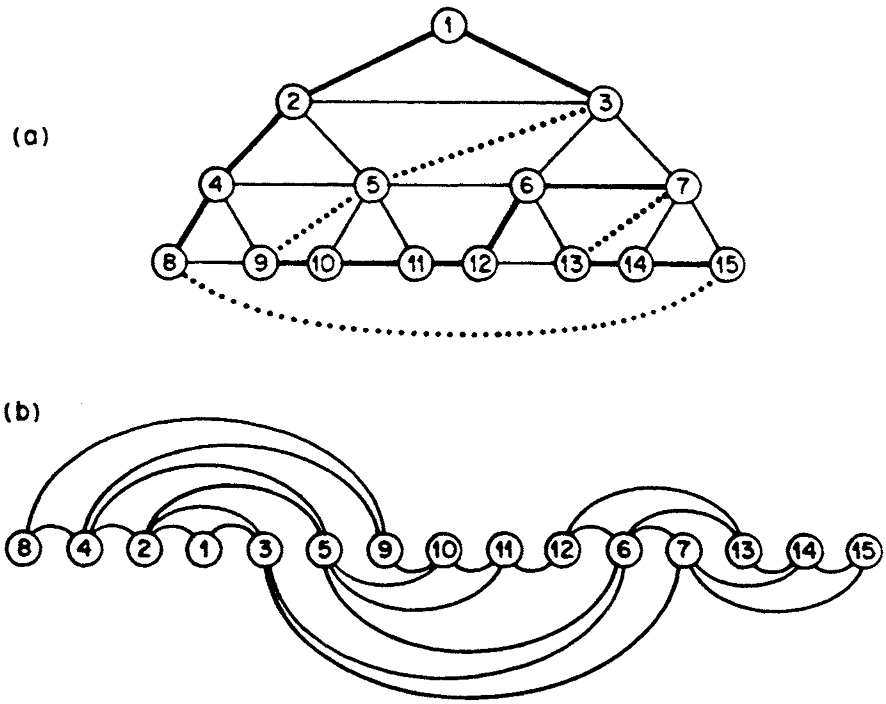


FIG. 4. (a) *An edge-augmentation of the depth-*4 *X-tree and an efficient hamiltonian cycle.* (b) *The layout of the X-tree induced by the cycle of* (a).

the indicated hamiltonian cycle, leads to the width-$O(d)$ 2-page embedding of $X(d)$ depicted in Fig. 4(b).

PROPOSITION 3.3. *The depth-d X-tree admits a 2-page embedding, with one page of width* $2d$ *and one of width* $3d$. *This embedding is optimal in pagenumber and is within a factor of* 5 *of optimal in cumulative pagewidth.*

Proof. Optimality in number of pages is immediate since $X(d)$ is not outerplanar for $d \geqq 3$. The (near-) optimality of the claimed cutwidth follows from the proof in [17] that $X(d)$ has no bisector of size less than d, coupled with the demonstration that this implies a similar bound on cutwidth.

It remains only to verify that the widths of the pages in the prescribed embedding do indeed satisfy the claimed bounds. The verification proceeds by induction, but requires some detail about the layout of $X(d)$. Say that we have a 2-page embedding of $X(d-1)$ with the claimed pagewidths and the following form. We depict the embedding schematically by its linearization of $X(d)$'s vertices, together with a few relevant edges. For simplicity we draw page 1 above the line of vertices and page 2 below the line.

$\alpha srt\beta$

LAYOUT 1

Here r, s, t are, respectively, the root of $X(d-1)$ and its left and right sons; α and β are the strings comprising the rest of $X(d-1)$'s vertices. Assume for induction that in Layout 1:

(1) the left spine vertices (which are the leftmost vertices at each level) of $X(d-1)$ appear consecutively in leaf-to-root order in α;

(2) the right spine vertices (which are the rightmost vertices at each level) appear, not necessarily consecutively, in root-to-leaf order in β;

(3) the vertices r, s, t and all of the left and right spine vertices are *exposed* on page 2, in the sense that no edge of $X(d-1)$ passes totally over them (i.e., under them in the picture);

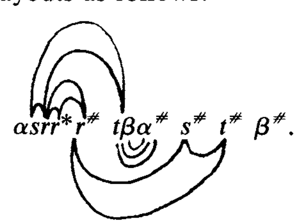
(4) the width of page 1 is at most $2d-2$;

(5) the width of page 2 is 0 below the left spine vertices, and is less than $3k-3$ to the right of the level-$(d-k-1)$ spine vertices.

Now take a second copy of Layout 1:

LAYOUT 2

The prescribed layout of $X(d)$—whose set of vertices is just the union of the sets of vertices of its two depth-$(d-1)$ sub-X-trees, in addition to r^*, its root vertex—is obtained from the indicated layouts as follows:



LAYOUT 3

A careful analysis of the composite layout extends the induction: Conditions (1), (2) are immediate since the left (resp., right) spine of $X(d)$ is contained in the string αsr (resp., the string $r^{\#}\ t^{\#}\ \beta^{\#}$), whose order is inherited from Layout 1 (resp., from Layout 2). Condition (3) is clear from the depiction of Layout 3: no edges are placed in the forbidden regions. Conditions (4), (5) are verified by simple counting.

Analysis of small X-trees establishes the base of the induction, thereby completing the proof. □

3.4. Benes permutation networks and their relatives. We now consider families of graphs whose structure is materially more complicated than the ones we have considered so far. These families are all very similar in structure and arise in a variety of contexts. They include the *FFT networks* whose structure represents the computational dependencies in the Fast Fourier Transform algorithm, *Banyan networks* whose structure approximates that of the Boolean n-cube while retaining bounded vertex-degrees, and the Benes rearrangeable permutation network [2], which is shown in Fig. 5(a). We concentrate on the Benes network, since it is a supergraph of the others, hence the hardest of the group to embed efficiently.

Let n be a power of 2. The *n-input Benes network* $B(n)$ is the graph defined inductively as follows.

1. $B(2)$ is the complete bipartite graph $K_{2,2}$ on two *input* vertices $i_{1,1}$ and $i_{1,2}$ and two *output* vertices $o_{1,1}$ and $o_{1,2}$.

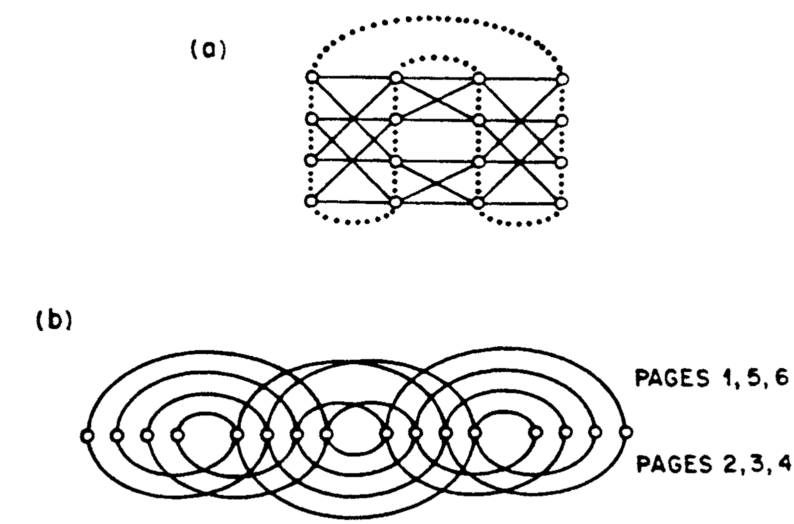


FIG. 5. (a) *The* 4*-input Benes network.* (b) *A* 6*-page layout of two levels of the network.*

2. $B(n)$ is obtained by taking two copies of $B(n/2)$ as well as n new input vertices, $i_{n,1}, i_{n,2}, \cdots, i_{n,n}$ and n new output vertices, $o_{n,1}, o_{n,2}, \cdots, o_{n,n}$. For each $1 \leqq k \leqq n$, one adds edges that create one copy of $K_{2,2}$ with "inputs" $i_{n,k}$ and $i_{n,k+n/2}$ and "outputs" $i_{n/2,k}$ and $i'_{n/2,k}$ (the primed vertices coming from the second of the two copies of $B(n/2)$) and one copy of $K_{2,2}$ with "inputs" $o_{n/2,k}$ and $o'_{n/2,k}$ and "outputs" $o_{n,k}$ and $o_{n,k+n/2}$ (again, primed vertices come from the second copy of $B(n/2)$).

Benes networks and their relatives are nonplanar, so they require at least three pages. Games [12] has recently discovered an elegant embedding that achieves this pagenumber. In order to illustrate a strategy that is often useful for finding good book embeddings, we describe now a simple 6-page embedding, which is built upon the hamiltonian cycle that alternates running up and down the "columns" of inputs and outputs of $B(n)$; see Fig. 5(b). In this embedding, one uses three pages to realize the "butterflies" that connect each "column" of vertices to the next "column." The fact that the embedding uses only a bounded number of pages is due to its reusing pages as it proceeds down the columns of $B(n)$. This strategy of reusing independent pages is a central feature of efficient embeddings (cf. [6], [15], [29]). It is somewhat surprising that any graph capable of "computing" all permutations can be realized with any bounded number, let alone 3, of pages.

PROPOSITION 3.4 [12]. *The Benes network $B(n)$ admits a* 3*-page embedding, with each page having width n. This embedding is optimal in pagenumber and within a factor of* 3 *of optimal in pagewidth.*

3.5. The Boolean *n*-cube. Our next family of graphs also has a rich interconnection structure which follows the communication structure of a broad class of algorithms. This family has been proposed as a desirable network architecture for a highly parallel computer; indeed, many of the other networks discussed in the literature—the shuffle-exchange, the banyan, and the cube-connected-cycles, for example—arose as bounded-valence stand-ins for our next graph. The *Boolean n-cube* $C(n)$ has as vertices the set of all binary strings of length n. The edges of $C(n)$ connect string-vertices x and y just when x and y are unit Hamming distance apart, i.e., when there exist binary strings

α, β, of collective length $n-1$, such that

$$\{x, y\}=\{\alpha 0\beta, \alpha 1\beta\}.$$

Thus $C(n)$ has 2^n vertices and $n2^{n-1}$ edges. Since $C(n)$ is hard to visualize for $n>3$, its efficient embedding is more easily described inductively in string-oriented terms, rather than via a hamiltonian cycle.

PROPOSITION 3.5. *The graph $C(n)$ $(n\geqq 2)$ admits an $(n-1)$-page embedding, with one page of width 2^i for each $1\leqq i\leqq n-1$. This embedding is within a factor of 2 of optimal in both pagenumber and cumulative pagewidth.*

Proof. The lower bound on pagenumber is immediate from the facts that

(a) the pagenumber of $C(n)$ is at least as big as the minimum number of outerplanar graphs into which $C(n)$ can be decomposed (Theorem 1.1);

(b) an N-vertex outerplanar graph can have at most N "noncircle" edges [23];

(c) $C(n)$ has $n2^{n-1}=(1/2)N\cdot \log N$ edges.

The lower bound on cumulative pagewidth follows from the easily derived fact that $C(n)$ has minimum bisection width 2^{n-1}.

The upper bound is seen easily by describing inductively the linearization of the vertices of $C(n)$.

* The vertices of $C(2)$ are laid out as follows:

$$00 \quad 01 \quad 11 \quad 10$$

hence $C(2)$ is embeddable in one width-2 page.

* Assume that $C(n)$ is realized with $n-1$ pages of widths $2, 4, \cdots, 2^{n-1}$, via the linearization

$$\beta_1\beta_2\cdots\beta_N$$

where each β_i is a distinct length-n binary word. Then the following layout for $C(n+1)$:

$$0\beta_1 0\beta_2\cdots 0\beta_N 1\beta_N\cdots 1\beta_2 1\beta_1$$

is realizable with just one more page, of width N. This extends the induction and completes the proof. □

3.6. The complete graph K_n. Finally, we analyze the complete graph on n vertices, K_n, in which every pair of vertices is adjacent. To simplify our analysis, without losing any of the germane ideas, let us assume that n is even.

PROPOSITION 3.6. *The complete graph K_n is embeddable in $n/2$ pages, each of width at most n. This embedding is optimal in pagenumber and in cumulative pagewidth.*

Proof. We establish the claims in reverse order.

Optimality in cumulative pagewidth is immediate since, by symmetry, all layouts of K_n have the same cutwidth.

Optimality in number of pages is deducible from our principle about matching subgraphs. Lay the vertices of K_n out on a line; call the vertices $0, 1, \cdots, n-1$ in left-to-right order. Note that K_n contains as a subgraph the matching graph M_n whose input vertices are $0, 1, \cdots, (n/2)-1$, and whose output vertices are given by: $\pi(v)= v+n/2$ for $0\leqq v<n/2$. Since the inputs and outputs of M_n are similarly ordered in this embedding, this embedding requires $n/2$ pages. Since all embeddings of K_n are isomorphic, the bound on pagenumber follows.

To see the upper bounds, consider the following way to lay out K_n. Place the vertices $0, 1, \cdots, n-1$ evenly spaced on a circle. For each vertex v, $0\leqq v<n/2$, draw

the line-graph L_v as indicated in the following illustration, in which all arithmetic is modulo n and in which double dashes denote edges of the line:

$$v = v+1 = v-1 = v+2 = v-2 = \cdots = v+(n/2)-1 = v-(n/2)+1 = v+(n/2).$$

It is not hard to verify the following facts.

(1) Each such line is composed of noncrossing chords of the circle; hence, by Theorem 1.1, each is embeddable on a single page.

(2) Every edge of K_n appears in precisely one line: to verify this, note that each vertex w is an endpoint of (hence, has valence 1 in) precisely one line, namely, $L_{w \bmod n/2}$ and has valence 2 in all other lines, so that in all, $n-1$ edges leave w; moreover, no two lines share an edge since, in the circle picture, all the lines emanating from vertex w have different slopes.

These two facts establish that, if one snips the circle between any two vertices, thereby laying K_n out in a line, and if one colors the edges of K_n according to which line L_v they lie in, one obtains an embedding of K_n in an $n/2$-page book. By the symmetry of K_n, this embedding has optimal cumulative pagewidth. □

3.7. The mesh of cliques. The $n \times n$ *mesh of cliques* $M(n)$ is the graph whose vertex-set is $\{1, 2, \cdots, n\} \times \{1, 2, \cdots, n\}$ and whose edges connect each row $\{i\} \times \{1, 2, \cdots, n\}$ into an n-vertex clique and each column $\{1, 2, \cdots, n\} \times \{i\}$ into an n-vertex clique. While we do not know how efficiently $M(n)$ can be embedded in a book in general, we can show that any embedding that places $M(n)$'s vertices along the spine row by row must use $n^{4/3}$ pages. The proof follows the inspiration of Theorem 2.3; details are left to the reader. Any nontrivial bound (particularly a lower bound) on the pagenumber of $M(n)$ would be interesting.

As a closing note to this section, Muder [18] and West [30] have a number of nontrivial bounds on the pagenumber of complete bipartite graphs $K_{n,n}$, that improve our results in [7].

4. Graph structure and pagenumber. In this section, we look at certain structural features of a graph, that are related to the number of pages required to embed the graph in a book. We find certain unexpected effects as well as the absence of certain expected ones.

4.1. Planarity. Theorem 1.1 indicates that the outerplanarity of a graph has a material effect on its pagenumber. It is easy to show that planarity has a not-dissimilar effect, but only when it is accompanied by a second structural property.

THEOREM 4.1 [3]. *The graph G admits a 2-page embedding if, and only if, it is subhamiltonian, i.e., a subgraph of a planar hamiltonian graph.*

Proof sketch. A graph is subhamiltonian just if it is embeddable in the plane so that (1) its vertices lie on a circle; (2) each of its edges lies either totally within the circle or totally without it; and (3) no edges cross in the layout.

Given such a "circular" embedding of a subhamiltonian graph G, cutting the circle between any two of G's vertices yields a planar embedding of G in a line, with each edge lying either totally above the line (i.e., on page 1) or totally below it (i.e., on page 2).

Conversely, given a 2-page embedding of the graph G, we view this embedding as placing G in a line with each edge lying totally above the line (page 1) or totally below it (page 2), and with no edges crossing. Pasting together the ends of the line containing G's vertices yields a "circular" embedding of G that witnesses G's subhamiltonian planarity. □

In the several years since the appearance of [3], the question of how many pages an arbitrary planar graph requires has attracted considerable attention. Buss and Shor [6] were the first to demonstrate that planar graphs can be embedded in a bounded number of pages; their elegant layout technique embeds an arbitrary planar graph in 9 pages. Heath [15], [16] used a quite different technique that improves this bound to 7 pages. Yannakakis [29] has recently settled the issue by proving coincident upper and lower bounds of 4 pages.

THEOREM 4.2 [29]. *Every planar graph admits a* 4*-page embedding. Moreover, there exist planar graphs requiring* 4 *pages.*

Returning to the consequences of Theorem 4.1, we observe that every series-parallel graph is 2-page embeddable. The class of *series-parallel graphs* is defined inductively as follows.

1. The 2-vertex graph with one source vertex s adjacent to one target vertex t is a series-parallel graph.

2. If G is a series-parallel graph with source vertex s and target vertex t and if G' is a series-parallel graph with source vertex s' and target vertex t', then the graph G'' obtained by "identifying" vertices t and s' is a series-parallel graph with source vertex s and target vertex t'. (This is an example of "series composition.")

3. If $G_1, \cdots, G_n$ are series parallel graphs with source vertices $s_1, \cdots, s_n$ and target vertices $t_1, \cdots, t_n$, respectively, then the graph G^* obtained by: taking a new source vertex s and adding edges between s and each of the s_i; and taking a new target vertex t and adding edges between t and each of the t_i is a series-parallel graph with source vertex s and target vertex t. (This is an example of "parallel composition.")

A graph is series-parallel just when its being so follows from provisos 1–3.

PROPOSITION 4.3. *Every series-parallel graph is* 2*-page embeddable.*

Proof. It is clear that every series-parallel graph is planar. By Theorem 4.1, then, we need only show that each such graph is subhamiltonian. This is easily proved by induction on the number of vertices in the graph, using the following inductive hypothesis.

> Given a series-parallel graph G with source vertex s and target vertex t, there is a planar edge-augmentation of G that has a hamiltonian path starting at s and ending at t.

The indicated path can then be completed to a cycle by an edge from t to s, without endangering planarity, thus establishing that the graph is subhamiltonian.

We sketch the easy induction. (1) Trivially, the unique 2-vertex series-parallel graph satisfies the claim. (2) If the graphs G and G' with source vertices s and s' and target vertices t and t' each satisfies the claim, then so also does their series composition: the desired hamiltonian path goes from s through G to t, which is identified with s', and thence through G' to t'. (3) If the graphs $G_1, \cdots, G_n$ are series-parallel, with source vertices $s_1, \cdots, s_n$ and target vertices $t_1, \cdots, t_n$, then the parallel composition of the graphs satisfies the claim: the desired hamiltonian path goes from s to s_1, thence through G_1 to t_1, to s_2, thence through G_2 to $t_2, \cdots$, from t_{n-1} to s_n, thence through G_n to t_n, and finally to t. Details are left to the reader. □

The final corollary of Theorem 4.1 is a direct consequence of Wigderson's result that the problem of deciding whether or not a maximal planar graph is hamiltonian is NP-complete [28].

COROLLARY 4.4. *The problem of deciding* 2*-page embeddability is* NP*-complete.*

4.2. Bisection width. The next structural property we consider measures the ease of recursively cutting a graph into two equal size subgraphs. We find that this measure yields a nontrivial upper bound on pagenumber but does not provide any nontrivial lower bound.

For our purposes, the simplest measure of the ease of bisecting a graph resides in the Bhatt–Leighton [4] notion of bifurcator: The graph G has a *ρ-bifurcator of size* B (B an integer and $\rho > 1$) either if G has fewer than B edges or if G admits a *decomposition tree* with the following property. The root of the tree (which is the sole vertex at level 0 of the tree) is the graph G. Each graph H at level $k \geqq 0$ of the tree that has more than one vertex gives rise to two disjoint graphs at level $k+1$, having the following properties: (a) each graph contains at least one vertex; (b) their union is H; and (c) each is connected to the other by no more than $B\rho^{-k}$ edges.

THEOREM 4.5. *If the graph G has a ρ-bifurcator of size B, then it is embeddable in $(\rho/(\rho-1))B$ pages.*

Proof. Let G have a ρ-bifurcator of size B. One begins the process of embedding G in a book by forming G's decomposition tree. One now lays G's vertices in a line (which will be the spine of the book) in the same order in which they appear as leaves of the decomposition tree. One assigns edges to pages as follows. At each level k of the tree, one creates $B\rho^{-k}$ new pages. One proceeds through all of the subgraphs of G that are split at that level, and one assigns one "cut" edge from each such subgraph to each of the new pages. No crossings can be introduced by such an assignment strategy since (a) edges that belong to the same level-k subgraph are assigned to different pages, and (b) edges that are assigned to the same page belong to disjoint intervals of vertices (because of the way vertices were laid out in the spine). It remains only to count the number of pages used in the embedding. This number is clearly bounded above by

$$\sum_{k \geqq 0} B\rho^{-k} = \left(\frac{\rho}{\rho-1}\right)B. \qquad \square$$

An immediate corollary of this result is that every small-degree n-vertex planar graph is embeddable in $O(n^{1/2})$ pages. This was the best upper bound known before the work of Buss and Shor [6], Heath [15], [16], and Yannakakis [29].

Theorem 4.5 indicates that the size of a graph's bifurcator places a nontrivial upper bound on the number of pages it requires. For the most part, this does not work in the other direction. By Theorem 4.1, every n-vertex 2-page embeddable graph has a $2^{1/2}$-bifurcator of size $O(n^{1/2})$, but once we get to 3-page embeddable graphs, knowledge of a graph's pagenumber no longer yields a nontrivial bound on the size of its bifurcators.

PROPOSITION 4.6. *There exist n-vertex 3-page embeddable graphs whose smallest ρ-bifurcators have size $\Omega(n/\log n)$ for all $\rho > 1$.*

Proof. Games [12] has shown that the n-input Benes network can be embedded in a 3-page book. A straightforward application of Thompson's lower bound proof technique [25] shows that every ρ-bifurcator of the $O(n \cdot \log n)$-vertex 3-page embeddable graph $B(n)$ has size $\Omega(n)$. $\square$

The bound in Proposition 4.6 has recently been strengthened by Galil, Kannan and Szemeredi [11], but it is still not known whether or not there exist n-vertex 3-page embeddable graphs whose smallest ρ-bifurcators have size $\Theta(n)$. As we mentioned in § 1.2, showing the existence of such graphs could have interesting consequences in classical complexity theory.

4.3. Valence. The final structural property we study is the valence of a graph. We find that this property affords us nontrivial upper and lower bounds on pagenumber.

These bounds are not very close for small or large valences, but they are close for moderate-valence graphs.

The graph G has valence d if no vertex of G has degree exceeding d. G is *regular* if all its vertices have the same degree.

4.3.1. An upper bound for d-valent graphs.

THEOREM 4.7. *Let d be any positive integer, and let ε be any positive constant. Say that G is a d-valent graph with n vertices, where*

$$n > \left(\frac{\ln((d+1)n^{1/2})}{\varepsilon}\right)^4.$$

If $d \leqq 2$, then G is 1-page embeddable. For any values of d and ε, G is $F(\varepsilon, d, n)$-page embeddable, where

$$F(\varepsilon, d, n) = \min\left[\frac{n}{2}, (1+\varepsilon)(2+2^{1/2})(d+1)n^{1/2}\right].$$

Proof. The cases $d \leqq 2$ are simple, for if $d = 1$, G is a matching graph, and if $d = 2$, G consists of disjoint paths and cycles.

We turn now to the case of arbitrary valence d. Say that we are given an n-vertex graph G of valence d. We note first that G is embeddable in $n/2$ pages, since K_n is (Proposition 3.6); hence we need look only at the second term in the expression for $F(\varepsilon, d, n)$. We shall justify this term (nonconstructively) by showing that not all embeddings of G in books can be "bad," in the sense of using too many pages.

We begin by decomposing G into at most $d+1$ matching graphs, $G_0, \cdots, G_d$, each having at most n vertices, by means of an edge-coloring algorithm (this is always possible by Vizing's Theorem [26]). Now consider all possible permutations of G's vertices (or, equivalently, all possible layouts of the vertices in the spine of a book).

Focus on an arbitrary permutation π and on its "behavior" on one of G's constituent matching graphs G_i. Consider those edges of G_i that connect a vertex in the left half of the layout with a vertex in the right half; say there are k such edges. These edges can be viewed (as we have noted earlier) as specifying a permutation on k integers. Since we have assumed nothing about the layout nor the edges, this permutation can be viewed as a random permutation on k integers. By a fundamental result of Hammersley [14, Thm. 6], the fraction of such permutations that have an increasing sequence of length exceeding $k^{1/2} + \varepsilon(n/2)^{1/2}$ is strictly less than

$$\exp\left(-2\varepsilon\left(\frac{n}{2}\right)^{1/2}\right).$$

This means (as we have noted before, by analogy with work of Tarjan [24]) that at most this small fraction of the layouts will require as many as $(1+\varepsilon)(n/2)^{1/2}$ pages to realize the edges of G_i that connect a vertex in the left half of the layout to a vertex in the right half (since $k \leqq n/2$).

Recall that increasing (resp., decreasing) sequences in a permutation correspond to similarly ordered (resp., oppositely ordered) sequences of inputs and outputs of our matching graph. Moreover, one can show via a strengthened analogue of Lemma 2.4 that the existence of a length-p increasing sequence in a permutation implies that the permutation can be partitioned into p decreasing sequences. The residents of each of the pages in the layout are the edges corresponding to one of these decreasing sequences.

Now let us remove these edges that connect the two halves of the layout and their incident vertices. We are left with two (roughly) half-size copies of the same problem. Moreover, since we have been discussing a matching graph, the relative layout of the remaining vertices is completely independent of the layout of the vertices that were removed, so that once again, the permutations induced by the edges can be viewed as random ones, hence within the purview of Hammersley's theorem. This means that when we analyze each of the permutations specified by the edges that connect the left halves of each of the subgraphs with the right halves, we find that at most the fraction

$$\exp\left(-2\varepsilon\left(\frac{n}{4}\right)^{1/2}\right)$$

require as many as $(1+\varepsilon)(n/4)^{1/2}$ pages for their realization. We can now continue in this fashion to remove edges that have been considered, thereby reducing our concern to 2^i subproblems of size roughly $n/2^i$ each, each of which encounters "bad" layouts with probability less than

$$\exp\left(-2\varepsilon\left(\frac{n}{2^i}\right)^{1/2}\right).$$

We continue generating half-size subproblems until $n/2^i \leqq n^{1/2}$, for by that time, Proposition 3.6 assures us that every layout can be realized within $n^{1/2}$ pages (i.e., that the probability of a layout's being "bad" is 0). It is clear from the foregoing reasoning that the probability that a random layout requires more than

$$\sum_{i=1}^{(1/2)\log n} (1+\varepsilon)(n/2^i)^{1/2} \leqq (1+\varepsilon)(1+2^{1/2})n^{1/2}+n^{1/2}$$
$$< (1+\varepsilon)(2+2^{1/2})n^{1/2}$$

pages to realize one of G's component matching graphs is less than

$$\sum_{i=1}^{(1/2)\log n} 2^{i-1}\exp\left(-2\varepsilon(n/2^i)^{1/2}\right) \leqq n^{1/2}\exp\left(-\varepsilon n^{1/4}\right).$$

Since G is just the disjoint union of its component matching graphs, it follows that the probability that a random layout of G's vertices requires more than

$$(1+\varepsilon)(2+2^{1/2})(d+1)n^{1/2}$$

pages to realize all of G's component matching graphs, hence G itself, is no greater than

$$(d+1)n^{1/2}\exp\left(-\varepsilon n^{1/4}\right),$$

which is less than unity, by the assumed relationship among n, d, and ε.

We have thus shown that almost all orderings of G's vertices result in layouts using no more than $F(\varepsilon, d, n)$ pages. □

Remark. The result of Hammersley that is at the center of the preceding proof deals with the lengths of *monotonic* subsequences of permutations. We needed the result instantiated for *increasing* subsequences, for this yielded the sought bound on *pagenumber*. However, the result can also be instantiated for *decreasing* sequences, thereby giving an $O(n^{1/2})$ upper bound on *pagewidth* also. Details are left to the reader.

4.3.2. A construction for trivalent graphs. The (nonconstructive) upper bound of Theorem 4.7 holds for almost all orderings of the vertices of arbitrary d-valent graphs, but we do not have a general construction that yields a good ordering. If we restrict attention to trivalent graphs, then we do have such an explicit construction. We begin with a special case.

Let G be a trivalent graph, and let S be the set of its degree-3 vertices. We say that G is *trimmable* if G admits a matching whose removal leaves G with at most one degree-3 vertex.

LEMMA 4.8. *Every n-vertex trimmable trivalent graph can be embedded in a* $(\frac{3}{2}n^{1/2}+5)$*-page book, each page having width at most* $n^{1/2}$.

Proof. Let G be an arbitrary n-vertex trimmable trivalent graph. We shall embed G in a book via the following series of steps.

1. We remove a matching from G, plus at most one additional edge, in such a way as to be left with a bivalent subgraph of G: in fact, a set of vertex-disjoint cycles and paths that include all of G's vertices. This is possible since G is trimmable. Let us refer to the removed matching edges as *matched* edges.

2. We (tentatively) lay G out in a line, cycle/path by cycle/path. Then we reinsert the removed edges.

3. We partition the linearized version of G into $n^{1/2}$ contiguous blocks of $n^{1/2}$ vertices each, from left to right. (Assume for simplicity that n is a perfect square.)

4. Our next task is to rearrange our tentative layout so as to achieve the claimed pagenumber. Note that every block (save possibly one) has at most $n^{1/2}+4$ edges leaving it to any other block: at most $n^{1/2}$ matching edges and at most 4 *emerging* edges that go from the cycles/paths of this block to neighboring blocks. The one possible exceptional block is the one that had one additional edge removed with the matching; it could have that additional edge leaving it, too.

We rearrange the vertices in each block, from left to right, in the following way. For the first block, we sort the vertices in decreasing order of the block numbers to which their matched edges go. For each subsequent block: (a) we place those vertices whose matched edges go to leftward blocks to the left of those vertices whose matched edges go to rightward blocks; (b) we sort the leftgoing vertices in decreasing order of the block numbers to which their emerging edges go; (c) we sort the rightgoing vertices analogously; (d) within each group of leftgoing vertices that are going to the same block, we arrange the vertices in increasing order of the distance from the present block of their target vertex.

Analysis. The effect of the rearrangements in 4(a)–(d) is that now each of the $n^{1/2}$ blocks needs just one page to realize all of its rightgoing matched edges; each of these pages has width at most $n^{1/2}$. The edges that we have scrambled within each block lie totally within blocks of size $n^{1/2}$ each; hence, we need at most half this many additional pages to realize them: By Proposition 3.6, $m/2$ pages, each of width m, can realize the edges interconnecting any group of m vertices; moreover, since the blocks are mutually disjoint, we can use the same $\frac{1}{2}n^{1/2}$ pages to realize all of them. The (at most) $4n^{1/2}$ emerging edges can be realized using at most 4 new pages: Since we never move blocks, at most two of these edges connect a block to its right neighbor, and at most two connect the block to its left neighbor; hence, the only conflicts occur within a block, and 4 new pages can resolve these conflicts. (Two of the pages used with one block can be reused in its neighbor block.) Finally, at most one additional page is necessary, to realize the one non-matched edge of G that we may have had to remove at the beginning of the embedding. The result follows. □

With the help of a crucial observation by Lenny Heath [31], we can extend Lemma 4.8 into a $(\frac{3}{2}n^{1/2}+6)$-page embedding of arbitrary trivalent graphs.

LEMMA 4.9 [31]. *Every trivalent graph without cut-edges (i.e., edges whose removal disconnects the graph) is trimmable.*

Proof. If the trivalent graph G has no cut-edges, then every vertex of G has degree 2 or 3. Let us pair up the degree-2 vertices of G and add an edge between each pair.

This will augment G to a regular trivalent graph, unless G started with an odd number of bivalent vertices, in which case our pairing leaves us with one unmated degree-2 vertex, call it v. We handle this last contingency as follows. Let u–v–w be a chain in the augmented G. (If G had fewer than three vertices, it would be univalent.) Replace v and the edges (u, v) and (v, w) by the single edge (u, w). At this point, in either of the contingencies, we have augmented G to a regular trivalent graph, possibly having multiple edges, but definitely having no cut-edges (since G had none). By a well-known result of Petersen [19], the augmented G has a perfect matching, i.e., a matching whose removal renders the graph regular bivalent. If we now restore G to its original state and consider the implications of Petersen's perfect matching, we verify easily that G is trimmable. □

THEOREM 4.10. *Every n-vertex trivalent graph can be embedded in a book with* $(\frac{3}{2}n^{1/2}+6)$ *pages. Each page, save possibly one, will have width at most* $2n^{1/2}$. *The cumulative pagewidth of the embedding will at worst be proportional to n, which cannot be improved in general.*

Proof. Let us be given an arbitrary n-vertex trivalent graph G. By removing all of G's cut-edges, we decompose G into subgraphs $G_1, G_2, \cdots, G_m$, each having no cut-edges. By Lemma 4.9, each G_i is trimmable; hence, by Lemma 4.8, each G_i can be embedded in a $(\frac{3}{2}n^{1/2}+5)$-page book, each page having width at most $n^{1/2}$. Thus, any embedding of G that lays the G_i out disjointly along the line has the claimed efficiency. To prove the theorem, then, we need only show how to deal with the removed cut-edges.

We begin with two easily verified but crucial observations for which we are grateful to Lenny Heath. First, we note that if we take our layout of one of the G_i and shift the vertices cyclically, we do not change the pagenumber of the layout, and we at most double its pagewidth (since our layouts really are in circles, not lines; cf. Theorem 1.1). Second, we note that if we contract each subgraph G_i to a point, leaving only the cut-edge interconnections, then the resulting contraction of G is a tree.

Our strategy is to lay G out as a tree of subgraphs, with each subgraph laid out as in Lemma 4.8, but possibly cyclically shifted.

We begin by arbitrarily picking G_1 as the first subgraph to process. We lay G_1 out as in Lemma 4.8. Say that in the layout, the vertices

$$v_{11}, v_{12}, \cdots, v_{1k_1},$$

appearing in that order, are connected to other subgraphs by cut-edges. We place those k_1 subgraphs along the line in the reverse order of the v_{1j}. When we place each subgraph, we use the layout prescribed by Lemma 4.8; but we cyclically shift the vertices in this layout so that the leftmost cut-edge-bearing vertex is the one connected to G_1. The subgraphs just placed will remain in this order, and their layouts will stay fixed, but other subgraphs may be placed between them.

Next, we process the just-placed subgraphs recursively, from left to right. (By "recursively" here we mean the following. If we have subgraphs A and B remaining to be processed, in that order, and if in the course of processing A we place a new subgraph C between A and B, then C gets processed before B.) We process subgraph G_i, $i>1$, as follows. Say that in the layout of G_i the vertices

$$v_{i1}, v_{i2}, \cdots, v_{ik_i},$$

appearing in that order, are connected to other subgraphs by cut-edges. We place those k_i subgraphs along the line in the reverse order of the v_{ij}, *immediately to the right of* G_i (hence, to the left of all other subgraphs that have previously been placed to the right of G_i). As before, when we place each subgraph, we use the layout prescribed by Lemma 4.8; but we cyclically shift the vertices in this layout so that the leftmost cut-edge-bearing vertex is the one connected to G_i. Again, the subgraphs just placed will remain in this order, and their layouts will stay fixed, but other subgraphs may be placed between them.

The reader will recognize that we have essentially laid the contracted tree version of G out in preorder. By Proposition 3.1, then, we need only one extra page to accommodate the cut-edges. Since the contracted tree has at most n edges, the extra page has cutwidth at most n.

We thus have an embedding of G with the parameters advertised in the statement of the theorem. The cumulative pagewidth of the embedding (which is at worst proportional to n) cannot be improved in general, as one can verify by observing that the cutwidth of a trivalent n-superconcentrator must be proportional to n. □

4.3.3. A lower bound for d-valent graphs. We have been unable to find lower bounds on the worst-case pagenumber of d-valent graphs that match the upper bounds of Theorem 4.7 and Theorem 4.10. We have, however, found nontrivial lower bounds, that we present now.

THEOREM 4.11. *For all valences $d>2$, for all sufficiently large n, there are n-vertex graphs of valence d whose pagenumber is no less than*

$$(\text{const})\,\frac{n^{(1/2)-(1/d)}}{\log^2 n}.$$

Proof. Let the valence $d>2$ of the graphs of interest be fixed. Imagine that we have a table each of whose rows is labeled with one of the $n!$ permutations of n items (= layouts of n vertices), and each of whose columns is labeled with one of the n-vertex matching graphs: the table entry corresponding to row i and column j is "FEW" if layout i uses no more than p pages on matching graph j, and is "MANY" if the layout uses more than p pages. The general strategy of our proof is to demonstrate that if p is no larger than indicated in the statement of the theorem, then some d-tuple of columns encounters at least one "MANY" in every row.

In order to get the argument going, we need to know roughly how many rows/permutations/layouts contain a "FEW" for a given column. This information is derivable from the following lemmas.

LEMMA 4.12. *At most p^{2r} permutations of r integers have no increasing sequence of length $p+1$.*

Proof. We noted in Lemma 2.4 that any permutation of $\{1, 2, \cdots, r\}$ whose longest increasing subsequence is of length p can be partitioned into p decreasing subsequences. This decomposition can be used to specify the permutation uniquely via two length-r strings over the alphabet $\{1, 2, \cdots, p\}$. The first string specifies, for each position i, which decreasing sequence occupies that position. The second string assigns the integers $\{1, 2, \cdots, r\}$ to subsequences. Since there are p^{2r} pairs of length-r strings over $\{1, 2, \cdots, p\}$, the lemma follows. □

LEMMA 4.13. *Let G be an n-vertex matching graph. The number of layouts of G that use at most p pages does not exceed*

$$P(n, p) = 2^{E(n,p)}$$

where

$$E(n, p) \leqq \frac{n}{2}\log n + n\cdot\log p + 2n\cdot\log\log n.$$

Proof. Let us count the number of layouts of G that require at most p pages. We employ the correspondence we have established between matching graphs and permutations (§ 2.2). Consider an arbitrary layout of G that has r edges passing between the leftmost $n/2$ vertices of G and the rightmost $n/2$ vertices; there are obviously no more than $n/2$ such edges. Let $\binom{x}{y}$ denote the binomial coefficient

$$\binom{x}{y}=\frac{x!}{y!(x-y)!}.$$

1. There are at most $\binom{n/2}{r}$ ways to choose the r edges that cross the center of the layout.

2. Each association (= edge) between element i and element j in a permutation can arise because $\pi(i)=j$ or because $\pi(j)=i$; hence there are 2^r ways of assigning left and right halves to each of the r edges.

3. There are at most $\binom{n/2-r}{(n/2-r)/2}$ ways to assign edges that do not cross the center to either the right or the left half of the layout.

4. Since the edges that cross the center can appear in any order, there are $r!$ ways of ordering the left endpoints of these edges.

5. By Lemma 4.12, no more than p^{2r} of the permutations specified by the r edges can be realized with only p pages, so there are at most p^{2r} ways of ordering the right endpoints of the edges that cross the center.

6. There are $\binom{n/2}{r}$ ways to place the (now ordered) endpoints of the r crossing edges on each side of the layout.

Aggregating all of these possibilities, recursing down to handle the two induced subgraphs of G to the left and to the right of the center of the layout, and allowing r to range over its possible values, we end up with the recurrence

$$P(n,p)\leqq\sum_{r\leqq n/2}\binom{n/2}{r}\cdot 2^r\cdot\binom{(n/2)-r}{[(n/2)-r]/2}\cdot r!\cdot p^{2r}\cdot\binom{n/2}{r}^2\cdot\left[P\left(\frac{n}{2}-r,p\right)\right]^2.$$

Our strategy will be to take the largest term T (say that it is the rth term) from this sum and show that nT, which certainly is no less than $P(n, p)$, is no greater than the claimed bound. We begin by representing r as

$$r=b\frac{n}{2},\qquad 0<b\leqq 1,$$

and by applying to T standard estimates for the binomial coefficients. We find that

$$\begin{aligned}P(n,p)&\leqq nT\\&\leqq \exp 2\left[\log n+\tfrac{3}{2}H(b)n+\frac{n}{2}+b\frac{n}{2}\log\left(b\frac{n}{2}\right)-b\frac{n}{2}\log e+bn\log p\right.\\&\qquad\left.+2E\left((1-b)\frac{n}{2},p\right)\right]\end{aligned}$$

where $\exp 2(x)=_{\mathrm{def}}2^x$, and where $H(b)$ is the base-2 entropy function

$$H(b)=-[b\log b+(1-b)\log(1-b)].$$

Let us now assume for induction that our claimed bound

$$E(m,p)\leqq\frac{m}{2}\log m+m\log p+2m\log\log m$$

on E (hence on P) holds for all $m<n$. It then follows from the preceding inequalities, after simplification, that

$$P(n,p)\leqq \exp 2\left[\log n+H(b)n+\frac{n}{2}\log n-b\frac{n}{2}\log e+n\log p\right.$$
$$\left.+2(1-b)n\log\log\left((1-b)\frac{n}{2}\right)\right].$$

Note that the right-hand expression can be shown to be less than

$$\exp 2\left[\frac{n}{2}\log n+n\log p+2n\log\log n\right]$$

provided only that for all $0<b\leqq 1$,

$$H(b)+\frac{\log n}{n}\leqq 2\log\log n-2(1-b)\log\log\left((1-b)\frac{n}{2}\right)+\frac{b}{2}\log e.$$

We establish this last inequality by verifying that, in fact,

$$\text{(1)}\qquad H(b)+\frac{\log n}{n}\leqq\frac{2}{\log n}+2b\log\log\left(\frac{n}{2}\right)+\frac{b}{2}\log e.$$

This will suffice since

$$\begin{aligned}2\log\log n-2(1-b)\log\log\left((1-b)\frac{n}{2}\right)&>2\log\log n-2(1-b)\log\log\left(\frac{n}{2}\right)\\&=2\log\log n-2\log\log\left(\frac{n}{2}\right)+2b\log\log\left(\frac{n}{2}\right)\\&=2\log\log n-2\log(\log n-1)+2b\log\log\left(\frac{n}{2}\right)\\&>\frac{2}{\log n}+2b\log\log\left(\frac{n}{2}\right).\end{aligned}$$

Now we must verify the final inequality (1) involving $H(b)$: Using the Taylor's series expansion for $\log(1-b)$, one can show that

$$H(b)\leqq b\log\frac{1}{b}+b\log e$$

for all $b\leqq 1$. Hence it suffices to verify that

$$b\log\frac{1}{b}+\frac{b}{2}\log e+\frac{\log n}{n}\leqq\frac{2}{\log n}+2b\log\log\left(\frac{n}{2}\right).$$

This is easily accomplished by analyzing the two cases

$$b\leqq(\log n)^{-3/2}\quad\text{and}\quad b>(\log n)^{-3/2}.$$

Thus we establish the desired inequality (1) on $H(b)$ and, through it, the desired inequality on $P(n,p)$. □

Return to proof of Theorem 4.11. Consider again our large table with entries "FEW" and "MANY". The number of "FEW" entries in each (n!-item) column of the table is at most $P(n,p)$, where p is the number of pages we are prepared to use to lay out our n-vertex d-valent graphs. Clearly, we cannot lay out all such graphs unless every d-tuple of table columns contains only "FEW" entries in at least one row of the table. (The d-tuples of this last assertion arise from the fact that every union of d matching graphs forms a d-valent graph.) These "FEW" entries have a chance of existing only if

$$c^d \leqq n!\left(\frac{P(n,p)}{n!}c\right)^d,$$

where c denotes the number of n-vertex matching graphs. The left-hand quantity is the number of d-tuples of matching graphs, while the right-hand quantity is the product of the number of rows and the number of d-tuples of "FEW" entries in each row. (The latter fact follows from the observation that, by symmetry, every row has the same number of "FEW" entries.) Simplifying, then, we can accommodate all d-valent graphs in p pages only if

$$P(n,p)^d \geqq (n!)^{d-1}.$$

By Lemma 4.13, this inequality implies (after taking logarithms)

$$dn\cdot[\tfrac{1}{2}\log n+\log p+2\log\log n]\geqq(d-1)n\log n+\Theta(n).$$

The validity of this inequality finally implies the claimed lower bound on p, namely,

$$p\geqq(\text{const})\frac{n^{1/2-1/d}}{\log^2 n}. \qquad \square$$

Our upper and lower bounds are within a few logarithmic factors apart when the valence d is logarithmic in n; they are rather far apart when d is either very big or very small. We conjecture that one of the factors of $\log n$ can be removed in the lower bound, but the tighter analysis needed is likely to be quite complicated.

5. Cost tradeoffs. In this section, we point out a rather interesting anomaly that could be important in the context of our study. We describe here two families of graphs that engender pagenumber-pagewidth *tradeoffs*. Each of these families can be laid out using some number p pages—but only if the widths of the pages are allowed to grow proportionally to the size of the graph being laid out. However, if one uses just one additional page, then the widths of the pages can be kept bounded by a constant.

Both of the graph families have the following form. The *depth-k K_n-cylinder* $C(k,n)$ is the graph whose vertex-set is the union of the k sets

$$V_{i,n}=\{v_{i,1},v_{i,2},\cdots,v_{i,n}\},\qquad 1\leqq i\leqq k,$$

and whose edges (a) connect each set $V_{i,n}$ into an n-clique, and (b) connect each vertex $v_{i,j}$ to vertex $v_{i+1,j}$, $1\leqq i<k$, $1\leqq j\leqq n$.

The anomalies of interest appear in the first two parts of the next result. The third part of the result indicates the failure of the obvious generalization of the first two parts.

PROPOSITION 5.1. (1a) *Any* 1*-page layout of* $C(k,2)$ *has pagewidth at least* $k/2$. (1b) *There are* 2*-page layouts of* $C(k,2)$ *having pagewidth* 2.

(2a) *Any* 2*-page layout of* $C(k,3)$ *has pagewidth at least* $k/2$. (2b) *There are* 3*-page layouts of* $C(k,3)$ *having pagewidth* 4.

(3) *There are* 3*-page layouts of* $C(k,4)$ *having pagewidth* 4.

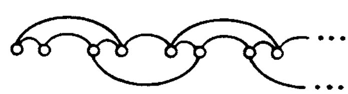
Proof sketch. The fact that $C(k,2)$ is outerplanar guarantees that it is 1-page embeddable. The fact that $C(k,3)$ is planar and subhamiltonian (a hamiltonian cycle can be traced by going from $v_{1,1}$ to $v_{1,2}$ to $v_{2,1}$ to $v_{2,2}$, and so on until one has reached $v_{2,n}$; at that point one goes to $v_{n,3}$, thence to $v_{n-1,3}$, and so forth, to $v_{1,3}$) guarantees that it is 2-page embeddable. Proving the lower bounds on the pagewidths of the resulting layouts proceeds by showing that at least half of the constituent n-cliques must be nested in any minimal-page layout. This is easily verified directly in the case of $C(k, 2)$: any (not necessarily contiguous) sequence of the form

$$v_{a,1} \cdots v_{b,2} \cdots v_{c,1} \cdots v_{d,2}$$

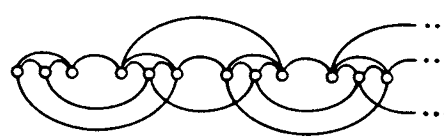
(or its reversal), where $\{a, b\} = \{1, 2\}$ precludes an embedding using just one page. (This verification is a special case of Syslo's result [23] that every biconnected outerplanar graph has a unique outerplanar embedding.) In the case of $C(k, 3)$, a direct verification is a bit more difficult; but the result follows immediately from Whitney's proof [27] that every triconnected planar graph has a unique planar embedding.

The existence of the claimed small-pagewidth layouts can be verified by the reader from the illustrative layouts depicted in Fig. 6. □

(a)



(b)



(c)

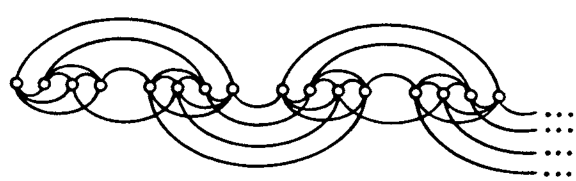


FIG. 6. *A small-width layout for* (a) $C(4,2)$, (b) $C(4,3)$, (c) $C(4,4)$.

It would be interesting to know whether or not there exist pagewidth-pagenumber tradeoffs analogous to those of Proposition 5.1 for every number of pages; i.e., can using one more page decrease pagewidth unboundedly?

Acknowledgments. It is a pleasure to thank Lenny Heath, Ravi Kannan, and Gary Miller for helpful conversations leading to several key insights.

REFERENCES

[1] P. B. ARNOLD (1982), *Complexity results for single-row routing*, Harvard Univ. Ctr. Res. Comput. Tech. Report TR-22-82.

[2] V. E. BENES (1964), *Optimal rearrangeable multistage connecting networks*, Bell Syst. Tech. J., 43, pp. 1641-1656.

[3] F. BERNHART AND P. C. KAINEN (1979), *The book thickness of a graph*, J. Combin. Theory, Ser. B, 27, pp. 320-331.

[4] S. N. BHATT AND F. T. LEIGHTON (1984), *A framework for solving* VLSI *graph layout problems*, J. Comput. Syst. Sci., 28, pp. 300-343.

[5] R. P. BRENT AND H. T. KUNG (1980), *On the area of binary tree layouts*, Inform. Process. Lett., 11, pp. 44–46.

[6] J. BUSS AND P. SHOR (1984), *On the pagenumber of planar graphs*, 16th Annual ACM Symp. on Theory of Computing, pp. 98–100.

[7] F. R. K. CHUNG, F. T. LEIGHTON AND A. L. ROSENBERG (1983), DIOGENES—*A methodology for designing fault-tolerant processor arrays*, 13th Internat. Conf. on Fault-Tolerant Computing, pp. 26–32.

[8] D. DOLEV AND H. TRICKEY (1982), *Embedding a tree on a line*, IBM Report RJ-3368.

[9] P. ERDÖS AND E. SZEKERES (1935), *A combinatorial problem in geometry*, Compositio Math., 2, pp. 463–470.

[10] S. EVEN AND A. ITAI (1971), *Queues, stacks, and graphs*, in Theory of Machines and Computations, Z. Kohavi and A. Paz, eds., Academic Press, New York, pp. 71–86.

[11] Z. GALIL, R. KANNAN AND E. SZEMEREDI (1986), *On nontrivial separators for k-page graphs and simulations by nondeterministic one-tape Turing machines*, 18th Annual ACM Symp. on Theory of Computing.

[12] R. A. GAMES (1986), *Optimal book embeddings of the FFT butterfly, Benes, and barrel shifter networks*, Algorithmica, to appear.

[13] M. R. GAREY, D. S. JOHNSON, G. L. MILLER AND C. H. PAPADIMITRIOU (1980), *The complexity of coloring circular arcs and chords*, this Journal, 1, pp. 216–227.

[14] J. M. HAMMERSLEY (1972), *A few seedlings of research*, 6th Berkeley Symp. on Math. Stat. and Prob., Vol. 1, pp. 345–394.

[15] L. S. HEATH (1984), *Embedding planar graphs in seven pages*, 25th IEEE Symp. on Foundations of Computer Science, pp. 74–83.

[16] ——— (1985), *Algorithms for embedding graphs in books*, Ph.D. dissertation, Univ. North Carolina, Chapel Hill, NC.

[17] J.-W. HONG AND A. L. ROSENBERG (1982), *Graphs that are almost binary trees*, SIAM J. Comput., 11, pp. 227–242.

[18] D. J. MUDER (1985), *Book embeddings of regular complete bipartite graphs*, Typescript, The MITRE Corp.

[19] J. PETERSEN (1891), *Die Theorie der regulaeren Graphen*, Acta Math., 15, pp. 193–220.

[20] R. RAGHAVAN AND S. SAHNI (1983), *Single row routing*, IEEE Trans. Comput., C-32, pp. 209–220.

[21] A. L. ROSENBERG (1983), *The Diogenes approach to testable fault-tolerant arrays of processors*, IEEE Trans. Comput., C-32, pp. 902–910.

[22] H. C. SO (1974), *Some theoretical results on the routing of multilayer printed-wiring boards*, 1974 IEEE Intl. Symp. on Circuits and Systems, pp. 296–303.

[23] M. M. SYSLO (1979), *Characterizations of outerplanar graphs*, Discrete Math., 26, pp. 47–53.

[24] R. E. TARJAN (1972), *Sorting using networks of queues and stacks*, J. Assoc. Comput. Mach., 19, pp. 341–346.

[25] C. D. THOMPSON (1980), *A Complexity Theory for* VLSI, Ph.D. thesis, Carnegie-Mellon Univ., Pittsburgh, PA.

[26] V. G. VIZING (1964), *On an estimate of the chromatic class of a p-graph*, Diskret. Analiz., 3, pp. 25–30. (In Russian.)

[27] H. WHITNEY (1933), *A set of topological invariants for graphs*, Amer. J. Math., 55, pp. 221–235.

[28] A. WIGDERSON (1982), *The complexity of the hamiltonian circuit problem for maximal planar graphs*, Princeton Univ. EECS Dept. Report 298.

[29] M. YANNAKAKIS (1986), *Four pages are necessary and sufficient for planar graphs*, 18th Annual ACM Symp. on Theory of Computing.

[30] D. WEST (1985), Personal communication.

[31] L. S. HEATH (1985), Personal communication.