Book Embeddings

Saïd Bettayeb
University of Houston Clear Lake
Houston, TX 77058, USA

1. Introduction

Graph embeddings play an important role in interconnection network and VLSI (Very Large Scale Integration) design. Simulation of one interconnection network by another can be represented as a graph embedding problem. Determining the number of layers required to build a VLSI chip, also called book-embedding, is another application of graph embeddings. In this chapter, we explore the latter problem. After an overview of the results on book embedding of the hypercube, we present results on book embedding of the k-ary hyperube, a variant of the hypercube. We also present recently obtained results on the book embedding of the torus graph.

Graph embeddings have been studied in the literature extensively for the important role it plays in interconnection network and VLSI (Very Large Scale Integration) design (Bernhart & Kainen, 1979; Bettayeb et al., 1989; Bettayeb & Sudborough., 1992; Bettayeb & Sudborough, 1989; Chung et., 1987; Yannakakis, 1989). Simulating an interconnection network, say A, by another, say B, is represented as a graph embedding problem where the nodes and edges of A are mapped to nodes and paths of B. A book embedding of a graph G is the mapping of the nodes of G onto the spine of a book and the edges of G onto pages so that the edges assigned to the same page do not cross. Determining the minimum number of pages required for such an embedding is the focus of this chapter. The minimum number of pages in which a graph G can be embedded is called the pagenumber of G, *pg*(G). Determining the pagenumber of an arbitrary graph has been shown to be NP-complete (Chung et al, 1987; Yannakakis, 1989). It remains NP-complete to determine if an arbitrary graph can be embedded in two pages. In a 1980 article, Garey et al (Garey et al., 1980) proved that determining the pagenumber of an arbitrary graph remains NP-complete even if we assume that the node embedding part is fixed, i.e. the layout of the nodes is given. An equally challenging task is the problem of determining the pagerwidth or geometric thickness of an arbitrary graph G which defined to be the minimum number of layers in a planar drawing of an arbitrary G.

Researchers have been drawn to study this problem and variations of this problem (Chung et al., 1987; Galil et al., 1989; Yannakakis, 1989) because of its many and diverse applications such as fault tolerant computing (Chung et al., 1987), graph drawing, and graph separators (Galil et al., 1989). Other problems remain to be solved such as the relationship of the pagenumber of a graph and other invariants. Enomoto and Miyauchi(Enomoto & Miyauchi, 1999) considered the case where edges may use more than one page.

The pagenumber of a graph has strong implications in VLSI design. The pagenumber is the minimum number of layers required to produce a VLSI chip. Another area of VLSI design that can be described in terms of book embedding is the configuring of processors in the presence of faults. Given an array of processors, some of which may be faulty, we lay them in a line. This could be either physical or logical. Running parallel to the line of processors are bundles of wires. As we scan the line of processors, we activate switches connecting the good processors to a bundle of wires and bypassing the bad processors. The bundle of wires act like a stack in that, when a processor u requests a connection to another processor, u is connected to a particular bundle and pushes the other processor connections down to a wire. When our scan reaches the processor to which processor u connects, it is popped off the bundle since the wire is no longer needed, and the other connections are returned to their original positions. The desired property in this case, is the minimization of the number of bundles required to interconnect all of the good processors in the desired layout. This is used in the Diogenes method of fault tolerant design as described by Chung, Leighton, and Rosenberg (Chung et al., 1987). If we take each bundle of wires and represent it as a page, we have a book embedding. Chung, Leighton, and Rosenberg (Chung et al., 1987) have studied the book embedding problem for a variety of graphs including trees, grids, X-trees, Benes networks, complete graphs, and the binary hypercube.

2. Definitions and Terminology

Let $G = (V, E)$ be an undirected graph. A linear layout L of the vertices of G is a mapping of V to the set $\{1, 2, 3, \dots, n\}$, where $n = |V|$. Let (x, y) and (x', y') be edges in G such that $L(x) < L(y)$ and $L(x') < L(y')$. Then (x, y) and (x', y') intersect with respect to L if $L(x) < L(x') < L(y) < L(y')$. If, in the other hand, $L(x) < L(x') < L(y') < L(y)$ or $L(x') < L(x) < L(y) < L(y')$ the edges (x, y) and (x', y') are said to nest with respect to L.

A *book embedding*, also referred to as stack embedding, is a linear layout of the vertices and an assignment of the edges to pages such that no page contains a pair of edges that intersect. The pagenumber of a graph, also referred to as book thickness, is the minimum number of pages achieved among all possible book embeddings.

When each edge has to be drawn as a straight line segment, the problem is referred to as the *geometric thickness* or *real linear thickness* (Kainen, 1990; Dillencourt et al, 2000)

Another variation studied in the literature is the so-called *queue embedding*. A queue embedding is a linear layout of the vertices of a graph and an assignment of the edges to queues such that no queue contains a pair of edges that nests (Bettayeb et al. 2010).

A d-dimensional torus is the d-dimensional mesh with wraparound edges. The wraparound edges connect the first and last vertex in each dimension. The notation $[n_1 \times n_2 \times \dots n_d]$ denotes the d-dimensional torus where $n_i, 1 \leq i \leq d,$ is the size of the i^{th} dimension.

The binary hypercube of dimension n, denoted by $Q_2(n)$ has 2^n vertices labeled by the binary representation of integers between 0 and $2^n - 1$. Two vertices are connected by an edge if and only if their labels differ in exactly one bit position. The k-ary hypercube of dimension n, denoted by $Q_k(n)$ has k^n vertices labeled by the k-ary representation of integers between 0 and $k^n - 1$. Two vertices are connected by an edge if and only if their labels differ in exactly one position by one (modulo k).

3. Book Thickness of Graphs

In a 1979 article, Bernhart and Kainen (Bernhart & Kainen, 1979) introduced the problem of book embedding. They proved that the pagenumber of a graph G, $pg(G) \leq 2$ if and only G is a subgraph of some planar graph. They also conjectured that planar graphs have unbounded pagenumber. Heath and Istrail (Heath & Istrail, 1992) disproved this conjecture and proved that the pagenumber of graphs with genus g is O($\sqrt{g}$). The genus of a graph is defined as follows. A graph G can be embedded with no edge crossing on a compact orientable two-manifold surface on which a number of handles have been placed. A handle is used by an edge that would otherwise cross other edges. The genus of a surface is the number of handles on that surface. The genus of a graph is the minimum genus of all possible surfaces on which G can be embedded. So planar graphs have genus 0 since no handle is needed.

In (Kainen & Overbay, 2003, the authors studied the pagenuumber in terms of block-cutpoint tree. A block-cutpoint graph of a graph $G = (V, E)$ is a bipartite graph $G' = (V', E')$ where $V' = X \cup Y$. There is a vertex $x \, in \, X$ for each block of G and a vertex $y \, in \, Y$ for each cutpoint in G. The vertices x and y are connected by an edge if and only if the block represented by x contains the cutpoint y. A graph G is connected if and only if its block-cutpoint is tree. They showed that the pagenumber of a graph is the maximum of the pagenumber of its blocks. They also showed that a graph is planar if and only if it homeomorphic to a graph with pagenumber at most two.

4. Book Embedding of Planar Graphs

A graph $G = (V, E)$ has pagenumber 0 if and only if $|E| = 0$, $i.e.$ G consists of isolated vertices. Trees are shown to have pagenumber 1. A graph is said to be outerplanar if it can be drawn in the plane so that all of its vertices lie on the same face. Outerplanar graphs have pagenumber one. The following theorem is due to Kainen and Overbay (Kainen & Overbay, 2003):

<u>*Theorem:*</u> A graph G has pagenumber k with vertex ordering $v1, v2, \dots, vn$ if and only $G = G_1 \, U \, G_2 \, U \, \dots \, G_k$, where each G_i is an outerplanar graph embedded with vertex ordering $v_1, v_2, \dots, v_n$.

In (Bernhart & Kainen, 1979), the authors conjectured that planar graphs have unbounded pagenumber but this was disproved by Heath and Istrail (Heath & Istrail, 1992). Buss and Shor (Buss & Shor, 1984) gave an algorithm that embeds all planar graphs in nine pages. Heath and Istrail (Heath & Istrail, 1992)gave an improvement that achieves seven pages. Yannakakis (Yannakakis, 1989) brought the number to four and showed that indeed four pages are sufficient and necessary.

4.1 Outline of Yannakakis Algorithm

Let C be a cycle C of the graph G that contains no external chords. Denote the vertices of C by $1, 2, \dots, p$. The edge $(1, p)$ is called the long edge, and the rest of edges of C are called short edges. Let G_c be the graph obtained from C by adding its chords. If F and F′ are two inner faces of G_c where the long edge of F is a short edge of F′ then all vertices of F must appear between two consecutive vertices of F′. First, Let K be the cycle bounding the inner face of G_c that contains the long edge of C. Its short edges must be the long edges of the other inner cycles. Layout the interior of K. Then, expand recursively the inner cycles.

As pointed out in (Yannakakis, 1989), the vertices of a planar graph can be partitioned into levels. All edges connect either vertices of the same level or vertices of adjacent levels. The former are called *level-edges* and the latter *binding edges*. Level 0 consists of the vertices forming the cycle K. Laying out the interior of K is accomplished by first laying level 1 vertices and coloring the short edges of K and the binding edges between levels 0 and 1. Expand recursively the cycles formed by level 1 vertices.

5. Book Embedding of the Torus and the k-ary hypercube

In (Bettayeb & Hoelzeman, 2009), we established the upper bound and lower bound on the pagenumber of the k-ary hypercube. The k-ary hypercube is a generalization of the binary hypercube. It is defined as follows. The k-ary hypercube of dimension d is an undirected graph of k^d vertices labeled by the integers 0 through k^{d-1}. Two nodes x and y are connected if and only the k-ary representations of their labels differ in exactly one position by one modulo k. We showed that the pagenumber of the n-dimensional k-ary hypercube is at least 2^{n-1} and the upper bound is $(\frac{a^n-1}{a-1})$ where a $=\left\lceil\frac{k}{2}\right\rceil$. In (Chung et al., 1987), Chung et al. showed that the binary hypercube of dimension n, $Q_2(n)$, admits an $n-1$ page embedding which is within a factor of 2 of optimal because the lower bound can easily be shown to be $\frac{n}{2}$. However, Heath, Leighton and Rosenberg (Heath et al., 1992) have shown that the pagenumber of the ternary hypercube of dimension n has a lower bound $\Omega(3^{n/9})$. It seems that when k is even the pagenumber grows linearly with the number of dimensions while it grows exponentially when k is odd. In (Bettayeb et al., 1020) we show that the pagenumber the k-ary hypercube depends on the parity of k. When k is even the pagenumber of $Q_k(n)$ grows linearly with the dimension n but when k is odd, the pagenumber grows exponentially with n. With the d-dimensional torus $[n_1 \times n_2 \times \ldots n_d]$, if all n_i, $1 \leq i \leq d$, are even the pagenumber is grows linearly with the number of dimensions d. In that paper, we also describe a layout technique with sequential corrections of the order of vertices that mitigates the problem for the case when dimensions are odd. Basically, the technique modifies the standard layout of alternating left-to-right and right-to-left segments with an amortizations of corrections that allow the reverse order to be realized without paying the penalty of all edges in the first-to-last wraparound connections to be simultaneouslu crossing. This would reduce the number of pages.

Recently, we showed that for any d-dimensional torus the pagenumber is bounded by $2^{d+1} - 3$ (Bettayeb et al., 2010). It had been shown (Bettayeb & Hoelzeman, 2009; Chung et al., 1987; Yannakakis, 1989) that the pagenumber of a graph is at least as large as the minimum number of outerplanar graphs into which it can decomposed.

For the queue embedding problem, the queue number for the torus grows linearly in the number of dimensions, regardless of the parity of the sizes of its dimensions. The queue number of the k-ary hypercube $Q_k(n)$ also grows linearly with the dimension n and does not depend on the parity of k. It is indeed $2n - 1$.

6. Conclusion

In this chapter, we presented results on book embedding and queue embedding of graphs. The upper bound for the book-embedding of the torus we achieved is $2^{d+1}-3$. It is interesting to know if this could be improved. Heath, Leighton and Rosenberg (Heath et al. 1992) showed that the ternary hypercube has a lower bound $\Omega(3^{n/9})$. It follows that the lower bound for torus is exponential in the number of dimensions when the sizes of its dimensions are odd. The authors in (Heath et al., 1992) conjectured that family of graphs with large queue number and small page (or stack) number do not exist. In (Bettayeb et al., 2010), we describe a class of graphs, namely the n-dimensional k-ary modified hypercubes which have pagenumber $O(n)$. We conjectured that the queue number for such graphs grow more rapidly than anu linear function of the dimension n.

7. References

Bernhart, F.; Kanien P.C., (1979). The Book Thickness of a Graph, *Journal of Combinatorial Theory*, Series B (27), 1979, pp. 320-331.

Bettayeb, S.; (1995). On the K-ary Hypercube. *Journal of Theoretical Computer Science* 140, 1995, pp. 333-339.

Bettayeb, S.; Heydari, H.; Morales, L.; Sudborough, I.H., (2010). Stack and Queue Layouts for Toruses and Extended Hypercubes. To appear

Bettayeb, S.; Hoelzeman, D., (2009). Upper and Lower Bounds on the Pagenumber of the Book Embedding of the k-ary Hypercube. *Journal of Digital Information Management*, 7 (1), 2009, pp. 31-35.

Bettayeb, S.; Miller, Z.; Sudborough, I.H., (1992). Embedding Grids into Hypercubes. *Journal of Computer and System Sciences*, 45 (3), 1992, pp. 340-366.

Bettayeb, S.; Sudborough, I.H., (1989). Grid Embedding into Ternary Hypercubes. *Proc. Of the ACM South Central Regional Conference*, 1989, pp. 62-64.

Buss, J.F.; Shor, P.W., (1984). On the Pagenumber of Planar Graphs. Proc. Of the 16th Annual Symposium on Theory of Computing, 1984, pp. 98-100.

Chung, F.R.K; Leightonm F.T; Rosenberg, A.L. (1983). DIOGENES: A Methodology for Designing Fault Tolerant Processor Arrays. *13th Conf. on Fault Tolerant Computing, 1983, pp. 26-32.*

Chung, F.R.K; Leightonm F.T; Rosenberg, A.L. (1987). Embedding Graphs in Books: A Layout Problem with Application to VLSI Design. *SIAM Journal of Algebraic Discrete Methods*, 8 (1), 1987, pp. 33-58.

Dean, A.M.; Hutchinson, J.P. (1991). Relations among Embedding Parameters for Graphs. *Graph Theory, Combinatorics, and Applications*, vol. 1, Wiley Interscience Publ., New York, 1991, pp. 287-296.

Dean, A.M.; Hutchinson, J.P. ; Sheinerman, E.R. (1991). On the Thickness and Arboricity of a Graph. *Journal of Combinatorial Theory Series B,* vol. 52, 1991, pp. 147-151.

Dillencourt, M.B.; Epstein, D.; Hirschberg, D.S. (2000). Geometric Thicknes of Complete Graphs. *Journal of Graph Algorithms and Applications.* vol. 4, 2000, pp. 5-17.

Enomoto, H.; Miyauchi, M.S., (1999). Embedding Graphs into a Three Page Book with O(M log N) crossings of edges over the spine. *SIAM Journal of Discrete Math.* 12, 1999, pp. 337-341.

Heath, L.S; Istrail, S. (1992). Comparing Queues and Stacks As Mechanisms for Laying out Graphs. *Journal of the Association for computing Machinery*, 39, 1992, pp. 479-501.

Heath, L.S; Leighton, F.T.; Rosenberg, A.L. (1992). Comparing Queues and Stacks As Mechanisms for Laying out Graphs. *SIAM Journal of Discrete Mathematics*, 5 (3), 1992, pp. 398-412.

Kainen, P.C.; Overbay, S. (2003). Book Embeddings of Graphs and a Theorem of Whitney. *Technical Report GUGU-2/25/03, http:// www.georgetown.edu/faculty/kainen/pbip3.pdf* .

Kainen, P.C. 1990. The Book Thickness of a Graph. *Congr. Numer*., 71, 1990, pp. 127-132.

Yannakakis, M. (1989) Embedding Planar Graphs in four Pages., Journal of Computer and System Sciences, 38 (1), 1989, pp. 36-67.

VLSI

Edited by Zhongfeng Wang

ISBN 978-953-307-049-0
Hard cover, 456 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

The process of Integrated Circuits (IC) started its era of VLSI (Very Large Scale Integration) in 1970’s when thousands of transistors were integrated into one single chip. Nowadays we are able to integrate more than a billion transistors on a single chip. However, the term “VLSI” is still being used, though there was some effort to coin a new term ULSI (Ultra-Large Scale Integration) for fine distinctions many years ago. VLSI technology has brought tremendous benefits to our everyday life since its occurrence. VLSI circuits are used everywhere, real applications include microprocessors in a personal computer or workstation, chips in a graphic card, digital camera or camcorder, chips in a cell phone or a portable computing device, and embedded processors in an automobile, et al. VLSI covers many phases of design and fabrication of integrated circuits. For a commercial chip design, it involves system definition, VLSI architecture design and optimization, RTL (register transfer language) coding, (pre- and post-synthesis) simulation and verification, synthesis, place and route, timing analyses and timing closure, and multi-step semiconductor device fabrication including wafer processing, die preparation, IC packaging and testing, et al. As the process technology scales down, hundreds or even thousands of millions of transistors are integrated into one single chip. Hence, more and more complicated systems can be integrated into a single chip, the so-called System-on-chip (SoC), which brings to VLSI engineers ever increasingly challenges to master techniques in various phases of VLSI design. For modern SoC design, practical applications are usually speed hungry. For instance, Ethernet standard has evolved from 10Mbps to 10Gbps. Now the specification for 100Mbps Ethernet is on the way. On the other hand, with the popularity of wireless and portable computing devices, low power consumption has become extremely critical. To meet these contradicting requirements, VLSI designers have to perform optimizations at all levels of design. This book is intended to cover a wide range of VLSI design topics. The book can be roughly partitioned into four parts. Part I is mainly focused on algorithmic level and architectural level VLSI design and optimization for image and video signal processing systems. Part II addresses VLSI design optimizations for cryptography and error correction coding. Part III discusses general SoC design techniques as well as other application-specific VLSI design optimizations. The last part will cover generic nano-scale circuit-level design techniques.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821