

New Layouts for the Shuffle-Exchange Graph
(Extended Abstract)

Daniel Kleitman
Frank Thomson Leighton
Margaret Lepley
Gary L. Miller

Applied Mathematics Department
Massachusetts Institute of Technology
Cambridge, Mass. 02139

1. Introduction

The shuffle-exchange graph is one of the best structures known for parallel computation. Among other things, it can be used to compute discrete fourier transforms, multiply matrices, evaluate polynomials, perform permutations and sort lists [2,4,5,7]. The algorithms needed for these operations are extremely simple and, for the most part, require no more than logarithmic time and constant space. The only exceptions are sorting lists (for which the best algorithm known requires $O(\log^2 n)$ time) and performing permutations (which requires $O(\log n)$ space per processor).

With the development of integrated circuit technology, it has become possible to place large numbers of very simple pro-

cessors on a single chip. Thus the question of how best to lay out the shuffle-exchange graph on a grid has gained practical as well as theoretical importance. Thompson was the first to address the issue. In his thesis [8], he showed that any layout of the shuffle-exchange graph requires at least $O(n^2/\log^2 n)$ area. In addition, he described a layout requiring only $O(n^2/\sqrt{\log n})$ area. Recently, Hoey and Leiserson improved the upper bound by finding an $O(n^2/\log n)$ -area layout.

In this extended abstract, we present several new layouts for the shuffle-exchange graph, including one which requires only $O(n^2/\log^2 n)$ area. The optimal layout is described and analyzed in section 3. The analysis is heavily dependent on several combinatorial results which we state in section 2 and prove in the appendix. The other layouts are described in section 4. Although these layouts are not asymptotically optimal (most require $O(n^2/\log^{3/2} n)$ area), the theory behind their development is interesting and may eventually lead to good practical layouts as well as other asymptotically optimal layouts. (Rodeh and Steinberg have independently discovered one of these layouts [6].)

Research supported in part by National Science Foundation grants MCS-05853, MCS-7719754, MCS-800756 and Office of Naval Research grant ONR-N0014-76-C-0366

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

The methods developed in this abstract appear to be quite useful in laying out more complicated networks. For example, in section 5, we show how they can be used to find an optimal $O(n^2/\log^2 n)$ -area layout for the shuffle-shift-reverse graph (the supergraph of the shuffle-exchange graph which also has shift and reverse edges). The methods are also quite suitable for practical applications. Although we do not discuss such considerations in detail in this abstract, we have found several heuristics which, when combined with the optimal layout, yield excellent practical layouts for the shuffle-exchange graph.

As it previously was not known whether or not the shuffle-exchange graph could be laid out in $O(n^2/\log^2 n)$ area, several researchers have tried to develop alternate networks which can efficiently compute discrete fourier transforms and which can be easily laid out in $O(n^2/\log^2 n)$ area. The cube-connected-cycles graph of Preparata and Vuillemin [5], is one such network. In fact, the cube-connected-cycles graph is the only network which is known to compute discrete fourier transforms in $O(\log n)$ time and to require only $O(n^2/\log^2 n)$ area. Unfortunately, each processor in this network must be capable of storing its own address (an $O(\log n)$ -bit number) and thus requires at least $O(\log n)$ space. In addition, the programming required for each processor of the cube-connected-cycles network is relatively complex and would require a great deal of area to hardwire on a chip. Neither of these drawbacks arises with the shuffle-exchange network. Thus, now that a practical $O(n^2/\log^2 n)$ -area layout for the shuffle-exchange graph has been found, it seems reasonable to use the shuffle-exchange network as the basis for designing chips to compute fast fourier transforms, evaluate polynomials and the like.

2. Preliminaries

The shuffle-exchange graph consists of $n=2^k$ nodes and $3n/2$ edges. Each node is associated with a unique k -bit binary string $a_{k-1}\dots a_0$. If $a_0=0$, the node is said to be even. Otherwise $a_0=1$ and the node is said to be odd. The value of a node is the numerical value of the associated k -bit binary string. Two nodes w and w' are linked via a shuffle edge if w' is a left or right cyclic shift of w (i.e., if $w=a_{k-1}\dots a_0$ and $w'=a_{k-2}\dots a_0 a_{k-1}$ or $w'=a_0 a_{k-1}\dots a_1$, respectively). Two nodes w and w' are linked via an exchange edge if w and w' differ only in the last bit (i.e., if $w=a_{k-1}\dots a_1 0$ and $w'=a_{k-1}\dots a_1 1$ or vice versa). For example, we have drawn the shuffle-exchange graph for $k=3$ in Figure 1. Solid lines denote shuffle edges while dashed lines denote exchange edges.

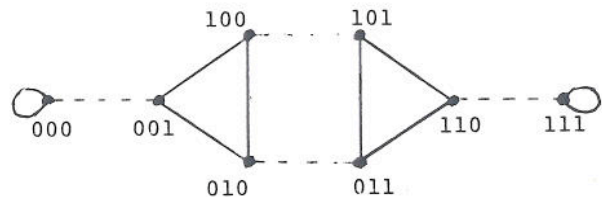


Figure 1

In this extended abstract, we will describe layouts for the shuffle exchange graph in terms of the grid model developed by Thompson [8]. In this model, processors are assumed to occupy unit area and are located only at the intersection of grid lines. Wires connect pairs of processors and are assumed to have unit width. They must follow along grid lines and are not allowed to overlap processors. Two wires can cross each other but only at the intersection of grid lines (i.e., two wires cannot overlap for any distance).

The area of the layout is defined to be the area of the smallest rectangle which contains all the wires and processors.

It is not difficult to show that m wires can be inserted into any layout with the addition of at most $2m$ vertical and $2m$ horizontal tracks. All of the layouts we consider in this abstract will require at least $O(n/\log n)$ vertical and $O(n/\log n)$ horizontal tracks. Thus any set of $O(n/\log n)$ nodes and edges can be inserted into a layout for the shuffle-exchange graph without increasing the total area by more than a constant factor. We will use this fact repeatedly in what follows to simplify the analysis of the layout by ignoring $O(n/\log n)$ -sized sets of nodes and edges that have undesirable properties.

The collection of all cyclic shifts of a node w is called a necklace and is denoted by $\langle w \rangle$. For example, the necklace generated by 001 is $\langle 001 \rangle = \{001, 010, 100\}$. Note that each necklace corresponds to a cycle in the shuffle-exchange graph (see Figure 1) and that shuffle edges always link nodes which are in the same necklace. If a necklace contains precisely k nodes, it is said to be full. Otherwise, the necklace contains less than k nodes and is said to be degenerate. It is a simple exercise to show that at most $O(\sqrt{n} \log n)$ nodes are contained in degenerate necklaces. Thus, by the remarks of the preceding paragraph, we do not need to consider such nodes when describing a layout for the shuffle-exchange graph. Accordingly, we henceforth consider only those nodes which are contained in full necklaces. Note that there are $O(n/\log n)$ full necklaces.

In what follows, we will be particularly interested in the size and location of the longest block of consecutive 0-bits in the k -bit binary

string associated with each node. In order that the size of this block be the same for each node within a necklace, we allow blocks to begin at the end and end at the beginning of a string. For example, the longest block of zeros in the string 01010 starts at the fifth bit and has length 2. Let $\gamma_k(t)$ denote the number of k -bit strings for which the longest block of consecutive zeros has length t . For example, $\gamma_3(2)=3$. The following combinatorial lemma provides an asymptotic bound on the growth of $\gamma_k(t)$. The proof of this lemma as well as those of lemmas 2-4 are combinatorial in nature and can be found in the appendix.

Lemma 1: For $\frac{1}{2} \log k + \log \ln k \leq t \ll k$,
 $\gamma_k(t) \sim 2^k (e^{-k2^{-(t+2)}} - e^{-k2^{-(t+1)}})$ as $k \rightarrow \infty$.

In order to illustrate the important features of the function in Lemma 1, we have sketched a graph of $2^{-k} \gamma_k(t)$ versus t in Figure 2. The maximum of $2^{-k} \gamma_k(t)$ occurs at $t = \log k - 1$ whence $2^{-k} \gamma_k(t) = \frac{\sqrt{e-1}}{e} \approx .23865$. For $t \geq \log k - 1$, $2^{-k} \gamma_k(t)$ decreases exponentially as t increases. For $t \leq \log k - 1$, $2^{-k} \gamma_k(t)$ decreases doubly exponentially as t decreases.

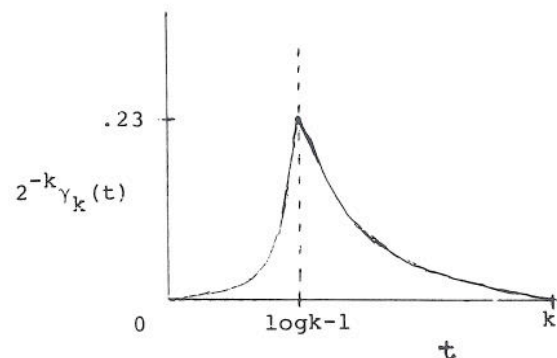


Figure 2

The following lemma bounds the size of the largest block of zeros for all but $O(n/\log n)$ nodes. Accordingly, we henceforth consider only those nodes for which the longest block of zeros has length between $\log k - \log \ln k - 1$ and $2 \log k$.

Lemma 2: The number of k -bit strings for which the largest block of zeros has length less than $\log k - \log \ln k - 1$ or length greater than $2 \log k$ is at most $O(n/\log n)$.

We will also be interested in the size of the second longest block of consecutive zeros. Usually, the size of the second longest block of zeros will be very close to the size of the longest block of zeros. We state this observation more precisely in the following lemma.

Lemma 3: The sum over all necklaces of the difference between the size of the longest and the size of the second longest block of consecutive zeros is at most $O(n/\log n)$.

Using information about the size and location of blocks of zeros within the nodes of a necklace, it is possible to distinguish one particular node of the necklace. More precisely, we define the distinguished node of a necklace to be the node containing the longest leading block of zeros. For example, 00101 is the distinguished node of $\langle 01010 \rangle$. Should two or more nodes of a necklace begin with equal and maximal length blocks of zeros, then each node in the necklace contains at least two blocks of zeros with maximal length. In such cases, we distinguish that node for which the leading block of zeros is maximal and for which the second occurrence of a maximal length block of zeros is as near as possible to the beginning of the string. For example, 01011 (not 01101) is the distinguished node of $\langle 10101 \rangle$. For some necklaces, such as $\langle 111 \rangle$ or $\langle 1010101 \rangle$, there is no uniquely

distinguished node. As we show in the following lemma, such necklaces are sufficiently rare that we need not consider them further.

Lemma 4: At most $O(n/\log n)$ nodes are contained in necklaces which fail to have a uniquely distinguished node.

We refer to the leading block of zeros of a distinguished node as the primary block of zeros. If a distinguished node has two or more maximal length blocks of zeros, then the maximal length block following the primary block is referred to as the secondary block of zeros. These definitions can be easily extended to any node contained in a necklace which has a uniquely distinguished node. For example, the primary block of zeros of 01010 starts in the fifth bit and has length two. Note that this string does not have a secondary block of zeros. As another example, we note that the secondary block of zeros of the string 11010 consists solely of the fifth bit. Note that the secondary block of zeros, if it exists, always has the same length as the primary block of zeros.

If the last bit of a node occurs in the primary block of zeros, we call that node a primary node. Similarly, if the last bit of a node occurs in the secondary block of zeros, we call that node a secondary node. Note that all primary and secondary nodes are necessarily even. For example, 10110 is a primary node, 11010 is a secondary node, and 10010 is neither primary nor secondary. Note also that, by Lemma 2, we need only consider necklaces which contain between $\log k - \log \ln k - 1$ and $2 \log k$ primary nodes. Such necklaces will also have at most $2 \log k$ secondary nodes.

In what follows we will represent each node in terms of the corresponding distinguished node. To do this, we use the notation $a_{k-1} \dots a_{i+1} \bar{a}_i a_{i-1} \dots a_0$ to

denote the node $a_{i-1} \dots a_0 a_{k-1} \dots a_i$. For example, $001\bar{0}1$ denotes the node 10010. Using this notation, a primary node has the form $0 \dots \bar{0} \dots 0w$ while a secondary node has the form $0 \dots 0w'0 \dots \bar{0} \dots 0w''$ where $0 \dots 0w$ and $0 \dots 0w'0 \dots \bar{0} \dots 0w''$ are distinguished nodes.

3. The Optimal Layout

We will present the optimal layout for the shuffle-exchange graph in two phases. First, we will describe a very simple layout which will be shown to require only $O(n^2 (\log \log n)^2 / \log^2 n)$ area. We will then modify this near-optimal layout in order to produce an optimal $O(n^2 / \log^2 n)$ -area layout, thus achieving Thompson's lower bound.

The near-optimal layout is constructed from a $\log n \times O(n/\log n)$ grid of nodes. Each column of the grid corresponds to a necklace of the shuffle-exchange graph. The nodes of each necklace are ordered from top to bottom so that the i th node is a left cyclic shift of the $(i-1)$ st node and so that the distinguished node is placed in the bottom row. The necklaces are ordered from left to right so that the values of the associated distinguished nodes form an increasing sequence. For example, we have constructed such a grid for $k=5$ in Figure 3. In the figure, we have represented each node in terms of the associated distinguished node. This representation readily illustrates the fact that the last bit of any node in the i th row corresponds to the i th bit of the associated distinguished node. Note that the necklaces $\langle 00000 \rangle$ and $\langle 11111 \rangle$ have not been included since they are degenerate.

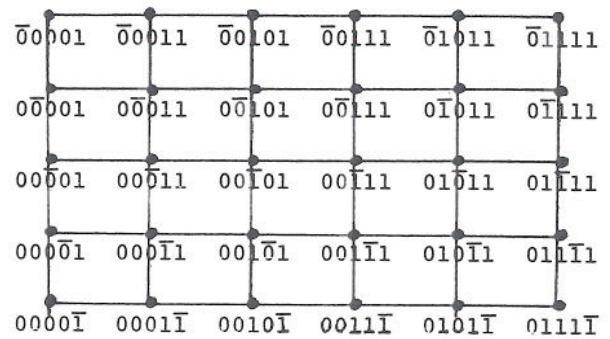


Figure 3

It is easily observed that the shuffle edges can be inserted in the grid with the addition of $O(n/\log n)$ vertical and 2 horizontal tracks. In the following, we will show that the exchange edges can be inserted with the addition of $O(n \log \log n / \log n)$ vertical and horizontal edges. In particular, we will first show that only $O(n \log \log n / \log n)$ exchange edges link nodes which are in different rows of the grid.

Such edges can be trivially inserted using only $O(n \log \log n / \log n)$ vertical and horizontal tracks. Then we will show that the edges which link nodes in the same row can be inserted with the addition of only $O(n/\log n)$ horizontal tracks. Thus the completed layout will require only $O(n^2 (\log \log n)^2 / \log^2 n)$ area.

Consider an exchange edge linking two nodes which are in different rows of the grid. In particular, assume the edge is incident to an even node in the i th row for some i . By definition, the even node can be represented as $w\bar{0}w'$ where $|w| = i-1$ and $w0w'$ is the distinguished node of $\langle w0w' \rangle$. The exchange edge is also incident to the odd node $w\bar{1}w'$. By assumption, $w\bar{1}w'$ is not located in the i th row and thus $w1w'$ is not a distinguished node. Since $w0w'$ is a distinguished node, we know that the i th bit of $w0w'$ (the bit that was changed in order to produce $w1w'$) must be in the primary or secondary block of

zeros of w_0w' . Otherwise, the primary and (if it exists) secondary blocks of zeros of wlw' would be identical in location and size to the primary and secondary blocks of w_0w' . This would imply that wlw' is also distinguished, a contradiction. Thus w_0w' must be a primary or secondary node. As was previously mentioned, we can assume that each necklace has at most $2\log k = 2\log\log n$ primary and $2\log\log n$ secondary nodes. Thus at most $4\log\log n$ nodes in any necklace are both even and incident to an exchange edge linking nodes in different rows. Since every exchange edge is incident to an even node and since there are $O(n/\log n)$ necklaces, we can conclude that there are at most $O(n\log\log n/\log n)$ exchange edges which link nodes in different rows.

It remains to show that those exchange edges which do link two nodes in the same row can be inserted with the addition of at most $O(n/\log n)$ horizontal tracks. The analysis is divided into two parts. In the first part, we show that at most $O(n/\log n)$ exchange edges are contained in the first $\log k$ rows. Such edges can be easily inserted with the addition of $O(n/\log n)$ horizontal tracks. In the second part, we show that only 2^{k-i} horizontal tracks are needed to insert the exchange edges contained in row i for $i > \log k$. Since $\sum_{i > \log k} 2^{k-i} \leq 2^k/k = n/\log n$, this will be sufficient to show that at most $O(n/\log n)$ additional horizontal tracks are necessary to insert the remaining exchange edges.

Consider a necklace which has t primary nodes for some $t \leq \log k$. By definition, the nodes in the first t rows of such a necklace are all even. Thus, such a necklace can have at most $r = \log k - t$ odd nodes in the first $\log k$ rows. By Lemma 1, we know that there are

$$\frac{1}{k} \gamma_k(t) \sim \frac{2^k}{k} (e^{-k2^{-t-2}} - e^{-k2^{-t-1}})$$

such necklaces for $\frac{1}{2} \log k + \log \ln k \leq t \ll k$.

By Lemma 2, we can assume that $t \geq \log k - \log \ln k - 1$ and thus the total number of odd nodes occurring in the first $\log k$ rows is at most

$$\begin{aligned} & \sum_{t=\log k - \log \ln k - 1}^{\log k} (\log k - t) \frac{2^k}{k} (e^{-k2^{-t-2}} - e^{-k2^{-t-1}}) \\ &= \frac{2^k}{k} \sum_{r=0}^{\log \ln k + 1} r (e^{-k2^{r-2-\log k}} - e^{-k2^{r-1-\log k}}) \\ &= \frac{2^k}{k} \sum_{r=0}^{\log \ln k + 1} r (e^{-2^{r-2}} - e^{-2^{r-1}}) \\ &\leq \frac{2^k}{k} \sum_{r=0}^{\log \ln k + 2} e^{-2^{r-2}} \leq \frac{2^k}{k} \sum_{r=0}^{\infty} e^{-2^{r-2}} \end{aligned}$$

$= O(n/\log n)$. Since every exchange edge is incident to an odd node, the above bound implies that at most $O(n/\log n)$ exchange edges are contained in the first $\log k$ rows.

We next consider the number of horizontal tracks necessary to insert the exchange edges contained in the i th row for $i > \log k$. This number is identical to the maximum number of exchange edges that can overlap each other at a single point. In Figure 4, we illustrate the conditions necessary for two exchange edges to overlap. All representations are in terms of distinguished nodes.

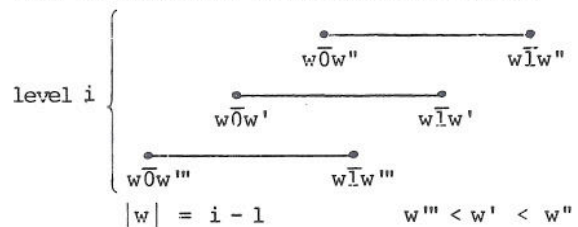


Figure 4

Note that the even end of an exchange edge is always to the left of the odd end. Also note that any node which occurs between w_0w' and w_1w' must be represented as w_0w'' where $w'' > w'$ or as w_1w''' where $w''' < w'$. In either case, the exchange edge incident to the overlapped node ex-

tends beyond the exchange edge linking $w\bar{0}w'$ to $w\bar{1}w'$. Since there are at most $2^{k-i} - 1$ nodes between $w\bar{0}w'$ and $w\bar{1}w'$, these facts imply that at most 2^{k-i} exchange edges can overlap at any point. This observation completes the argument that the near-optimal layout requires only $O(n^2(\log\log n)^2/\log^2 n)$ area.

In order to produce an optimal $O(n^2/\log^2 n)$ -area layout of the shuffle-exchange graph, we must relocate the primary and secondary nodes of each necklace. In particular, it is important that these nodes be positioned closer to and in the same row as the nodes to which they are linked via an exchange edge. In order to do this, we must break up each necklace into two or, possibly, three pieces. The basic piece of the necklace will consist of all those nodes which are neither primary nor secondary. The primary piece of the necklace will consist of the primary nodes while the secondary piece will consist of the secondary nodes, if there are any. For example, the basic piece of $\langle 01011 \rangle$ is $\{0\bar{1}011, 010\bar{1}1, 0101\bar{1}\}$, the primary piece is $\{0\bar{1}011\}$, and the secondary piece is $\{010\bar{1}1\}$.

It is also necessary to extend the notion of a distinguished node to include pieces of necklaces. The distinguished node of a basic piece will be the same as the distinguished node for the associated necklace. The distinguished node of the primary piece of a necklace is that node in the necklace which is distinguished when we ignore the primary block of zeros (i.e., when we temporarily replace the primary block of zeros in each node of the necklace with an equal-sized block of ones). Similarly, the distinguished node of the secondary piece of a necklace is that node which is distinguished when we ignore the secondary block of zeros. For example, 010110111 is the distinguished node of the basic piece of $\langle 010110111 \rangle$, 011011101 is the distinguished node of the primary

piece and 011101011 is the distinguished node of the secondary piece. Note that the distinguished nodes of the primary and secondary pieces of any necklace are odd nodes and thus are not contained in those pieces.

It is possible that some necklaces will have a distinguished node but will not have a distinguished node for the primary or secondary piece of the necklace. Fortunately, arguments such as those used to prove Lemmas 3 and 4 can be used to show that at most $O(n/\log n)$ nodes are contained in such necklaces. Thus, we can assume henceforth that every piece of every necklace has an associated distinguished node.

As before, the layout is constructed from a $\log n \times O(n/\log n)$ grid. Each column of the grid corresponds to a piece of a necklace. The nodes of each piece are arranged within a column so that a node of the form $a_{k-1}\dots\bar{a}_{k-i}\dots a_0$ (where $a_{k-1}\dots a_0$ is the distinguished node of the associated piece) is placed in the i th row of the grid. Note that nodes in the basic piece of any necklace (these include all odd nodes) are in the same row as they were in the near-optimal layout. The columns are ordered from left to right so that the values of the distinguished nodes of the

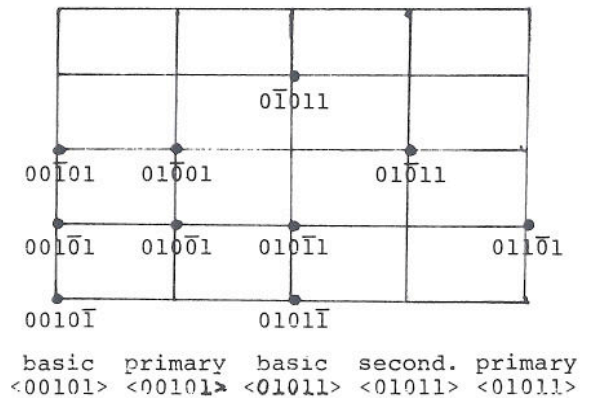


Figure 5

associated pieces will form a nondecreasing sequence. For example, we have constructed such a grid for $k = 5$ in Figure 5. Note that the necklaces $\langle 00001 \rangle$, $\langle 00011 \rangle$, $\langle 00111 \rangle$ and $\langle 01111 \rangle$ have not been included since their associated primary pieces do not have distinguished nodes.

We now prove our main result.

Theorem 1: The shuffle-exchange graph can be laid out in $O(n^2/\log^2 n)$ area. In particular, the layout just described requires only $O(n^2/\log^2 n)$ area.

Proof: As each necklace has been broken up into at most four contiguous pieces (the basic piece may have been broken up into two contiguous pieces), the shuffle edges can be inserted with the addition of just $O(n/\log n)$ vertical and horizontal tracks.

As before, we divide the analysis of the exchange edges into two cases. We first show that at most $O(n/\log n)$ exchange edges link nodes which are in different rows. Thus, these edges can be inserted with the addition of at most $O(n/\log n)$ horizontal and vertical tracks. We then show that those exchange edges which link two nodes in the same row can be inserted with the addition of just $O(n/\log n)$ horizontal tracks. The arguments will be nearly identical to those used in the analysis of the near-optimal layout.

Consider an exchange edge linking two nodes which are in different rows of the grid. From before, we know that the even node incident to the edge is either a primary or secondary node. Assume for the purposes of contradiction that the even node is a secondary node. Then this node can be represented as $w\bar{0}w'$ where $w0w'$ is the distinguished node of the secondary piece of $\langle w0w' \rangle$ and $|w| = i-1$ for some i . By definition, $w\bar{0}w'$ is located in the i th row and is linked to $w\bar{1}w'$ via the exchange edge. Since $w\bar{1}w'$ is odd, it is contained in the basic

piece of $\langle w\bar{1}w' \rangle$. By assumption, $w\bar{1}w'$ is not also in the i th row and thus $w\bar{1}w'$ cannot be the distinguished node of $\langle w\bar{1}w' \rangle$. Since the lengths of the two blocks of zeros in $w\bar{1}w'$ created by switching the i th bit from 0 to 1 are less than the length of the primary block of zeros (in fact, the sum of their lengths is precisely one less than the length of the primary block), $w\bar{1}w'$ will be the distinguished node of $\langle w\bar{1}w' \rangle$ precisely when $w0w'$ is the node distinguished in $\langle w0w' \rangle$ when the secondary block of zeros is ignored. By definition, this is the case precisely when $w0w'$ is the distinguished node of the secondary piece of $\langle w0w' \rangle$. By assumption, $w0w'$ is the distinguished node of the secondary piece of $\langle w0w' \rangle$ and thus we can conclude that $w\bar{1}w'$ is the distinguished node of $\langle w\bar{1}w' \rangle$, a contradiction.

Consider a primary node which is incident to an exchange edge linking two nodes in different rows. By the preceding arguments, this node must be of the

form $w\overbrace{10\dots 0}^{t_1}\overbrace{00\dots 0}^{t_2}1w'$ where $w10\dots 01w'$ is the distinguished node of the primary piece of $\langle w10\dots 01w' \rangle$ and either t_1 or t_2 is larger than or equal to the length of the longest block of zeros in $w11w'$.

Otherwise, $w\overbrace{10\dots 0}^{t_1}1\overbrace{0\dots 0}^{t_2}1w'$ will be the distinguished node of $\langle w\overbrace{10\dots 0}^{t_1}1\overbrace{0\dots 0}^{t_2}1w' \rangle$ and thus $w\overbrace{10\dots 0}^{t_1}1\overbrace{0\dots 0}^{t_2}1w'$ will be on the same row as $w\overbrace{10\dots 0}^{t_1}\overbrace{00\dots 0}^{t_2}1w'$. Thus, each necklace contains at most $2r$ such primary nodes where r is the difference between the longest and second longest block of zeros in any string of the necklace. By Lemma 3, we know that there are at most $O(n/\log n)$ such primary nodes in the entire shuffle-exchange graph. Thus, at most $O(n/\log n)$ exchange

edges link nodes which are in different rows.

It remains to consider those exchange edges linking nodes which are in the same row of the grid. The analysis of these edges is nearly identical to that for the near-optimal layout. In particular, there are still only $O(n/\log n)$ odd nodes in the first $\log k$ rows and thus the $O(n/\log n)$ exchange edges contained in the first $\log k$ rows can be inserted with the addition of only $O(n/\log n)$ horizontal tracks. As before, two exchange edges can overlap on the i th row only if the first i bits of the associated odd nodes are identical. Thus at most 2^{k-i} tracks are needed to insert all of the exchange edges contained in the i th row for $i > \log k$. Since $\sum_{i > \log k} 2^{k-i} \leq 2^k/k$, we can conclude that at most $O(n/\log n)$ additional horizontal tracks are needed to insert all such exchange edges. This concludes the proof that the layout requires only $O(n/\log n)$ vertical and horizontal tracks and thus only $O(n^2/\log^2 n)$ area. \square

The methods developed in this section can be used to find several other optimal layouts for the shuffle-exchange graph. The key variant is the method used to distinguish a node. The method must be, for the most part, impervious to small alterations in the necklace. The method used in this abstract satisfies this constraint. Only by changing a bit in the primary or secondary block of zeros can we globally change the distinguished node. Another possible method is to distinguish that node of a necklace which has the minimal value. Although the proof is substantially more complicated, such a method of distinguishing nodes also leads to an optimal layout for the shuffle-exchange graph.

4. Other Layouts

The layouts for the shuffle-exchange graph considered in this section are based on Hoey and Leiserson's complex plane diagram [1]. These layouts are not asymptotically optimal. In fact, the best area bound known for such a layout is $O(n^2/\log^{3/2} n)$. On the other hand, some of these layouts compare quite favorably to the known optimal layouts for small values of k , say $k=7$. It is important to point out that the size of the shuffle-exchange networks must grow exponentially with k and thus, at least for VLSI applications, very few values of k will ever be achievable networks.

Hoey and Leiserson's diagram is the embedding of the shuffle-exchange graph in the complex plane produced by mapping each node $a_{k-1} \dots a_0$ to the point $a_{k-1} \omega^{k-1} + \dots + a_1 \omega + a_0$ where $\omega = e^{2\pi i/k}$, the k th primitive root of unity. For example, the diagram for $k=5$ is shown in Figure 6. For convenience, the nodes are referenced by their value.

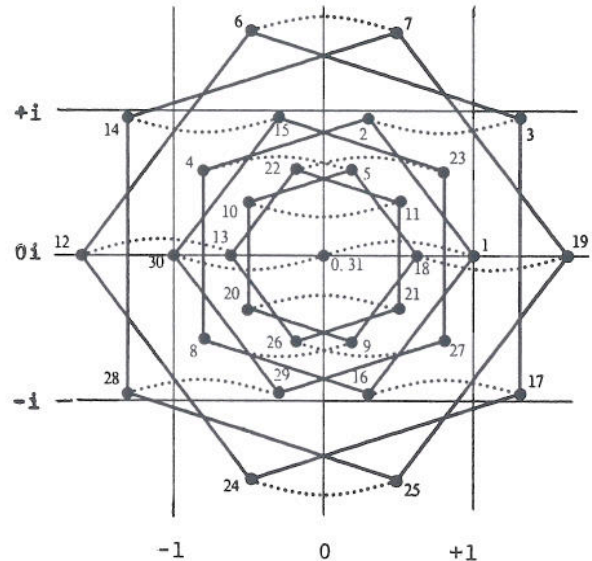


Figure 6
(taken from [1])

It is not difficult to show that the nodes of each necklace are mapped onto a circle centered at the origin. Further, the nodes are spaced around the circle so that the traversal of a shuffle edge corresponds to a rotation of $2\pi/k$ radians in the complex plane. In what follows, we will only consider nodes which are not mapped to the origin. Since less than $O(n/\log n)$ nodes are mapped to the origin [2], these nodes and the edges incident to them can be inserted later without increasing the total area by more than a constant factor.

It is also easy to show that exchange edges are horizontal and have length one in the complex plane diagram. In some cases, two or more exchange edges are contained in a single horizontal line. Such lines are called levels. More precisely, a level is a horizontal line in the complex plane containing one or more nodes of the embedded graph. For example, there are 9 levels in the complex plane diagram shown in Figure 6. In general, there are at most $3^{\lfloor (k-1)/2 \rfloor}$ levels [2].

In order to lay out the shuffle-exchange graph, we first form a grid composed of levels and necklaces. Each row of the grid corresponds to a level of the complex plane diagram. The columns are divided into consecutive column pairs, each pair corresponding to a necklace. In particular, the leftmost column of a column pair corresponds to that part of the necklace contained in the left half of the complex plane while the rightmost column corresponds to that part of the necklace contained in the right half of the complex plane. We assume that the rows are ordered top-to-bottom to be consistent with the natural ordering of levels in the complex plane but (for the time being) place no restriction on the left-to-right ordering of the necklaces. Each node is then placed at the intersec-

tion of the level and part of the necklace (left half or right half) in which it occurs.

It is now a simple matter to insert the edges. All of the shuffle edges can be inserted with the addition of $O(n/\log n)$ vertical tracks and 2 horizontal tracks. Since each exchange edge links two nodes which are in the same row, at most $O(n)$ horizontal edges are needed to insert the exchange edges. For example, we have drawn such a layout for $k=5$ in Figure 7.

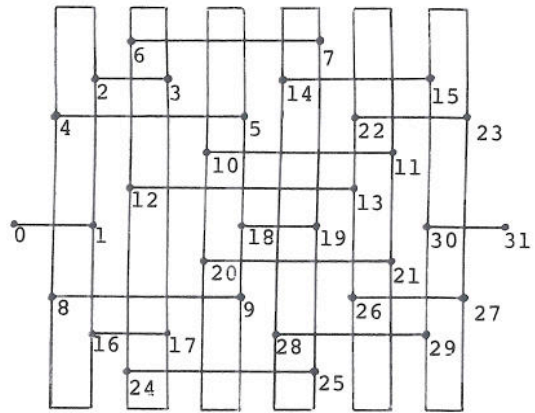


Figure 7

By rearranging the necklaces in the grid, we can increase the average number of exchange edges inserted in each track and thus decrease the number of horizontal tracks necessary to insert all of the exchange edges. For example, the arrangement of the necklaces shown in Figure 7 is optimal in this respect. Only the level corresponding to the real line requires more than one track to insert the associated exchange edges.

The necklaces in this example are ordered according to the number of 1-bits contained in any string of the necklace. In general, such an ordering is substantially better than a random one. In fact, when the necklaces are ordered in this fashion, only $O(n/\sqrt{\log n})$ horizontal tracks

are needed to insert all of the exchange edges [2]. As Rodeh and Steinberg have independently observed in [6], such a layout requires only $O(n^2/\log^{3/2}n)$ area.

There are other orderings which produce layouts with similar area bounds. For example, if the necklaces are ordered by their radii in the complex plane diagram, the induced layout requires just $O(n^2/\log^{3/2}n)$ area [2]. We summarize these results in the following theorem. The proof may be found in [2].

Theorem 2: If the necklaces are ordered by the number of 1-bits contained in any string of the necklace or are ordered by the radius of the necklace in the complex plane diagram, then the induced layout for the shuffle-exchange graph will require only $O(n^2/\log^{3/2}n)$ area.

The bisection width of a graph is the cardinality of the smallest set of edges whose removal disconnects the graph into two equal-sized subgraphs. Thompson showed that the shuffle-exchange graph has an $O(n/\log n)$ bisection width [8]. If we restrict our attention to layouts where the necklaces are placed as vertical loops (as in the complex plane diagram), then we will be interested in bisections which only contain exchange edges. Such bisections are simply bisections of what we call the necklace graph. The necklace graph is constructed from the shuffle-exchange graph by identifying vertices in the same necklace. Since any bisection of the necklace graph is also a bisection of the shuffle-exchange graph, we know that the bisection width of the necklace graph has size at least $O(n/\log n)$. The near optimal layout for the shuffle-exchange graph described in section 3 provides an $\Theta(n \log \log n / \log n)$ bisection width for the necklace graph, the best known. For comparison, we note that the orderings of the necklaces described in Theorem 2 lead to $\Theta(n/\sqrt{\log n})$ bisections.

In order to find an optimal layout for the shuffle-exchange graph which preserves the necklace structure (e.g., a layout based on the complex plane diagram), it is necessary to find an $O(n/\log n)$ bisection of the necklace graph. The converse is not true (i.e., finding an $O(n/\log n)$ bisection of the necklace graph does not necessarily lead to an optimal layout). In particular, we do not know whether or not the ordering of the necklaces defined in section 3 (which has an $O(n \log \log n / \log n)$ bisection width) leads to an $O(n^2 \log \log n / \log^2 n)$ complex plane layout.

An affirmative solution to this problem might have important practical applications. For example, when the necklaces are ordered as in section 3 and the levels are modified slightly, only 29 horizontal tracks are needed to insert all of exchange edges for $k = 7$ [2].

5. More Complex Networks

For some applications, it is useful to consider a network which has more than just shuffle and exchange edges. In particular, we will want to consider networks which also have shift edges and reverse edges. Shift edges link the i th node to the $(i+1)$ st node for all odd i . When combined with the exchange edges, the resulting network will have links between the i th and $(i+1)$ st nodes for all i . The inclusion of such edges facilitates the computation of discrete fourier transforms at sequential intervals of a continuous signal. In such applications, the data point of each processor is shifted to an adjacent processor (and a new data point is entered into the network) after each computation of a discrete fourier transform. The graph consisting of shuffle, exchange and shift edges is known as the shuffle-shift graph.

Reverse edges link pairs of nodes that are associated with binary strings which are reverses of each other. For example, $a_{k-1} \dots a_0$ is linked to $a_0 \dots a_{k-1}$ via a reverse edge. Since the algorithm which computes discrete fourier transforms on the shuffle-exchange network leaves the solution for each node $a_{k-1} \dots a_0$ in node $a_0 \dots a_{k-1}$, reverse edges provide a fast and convenient way of straightening out the solution. We define the shuffle-shift-reverse graph to be the graph consisting of all shuffle, exchange, shift and reverse edges.

Using the methods developed in section 3, it is not difficult to show that the shuffle-shift graph can be laid out using only $O(n^2/\log^2 n)$ area. As before, the necklaces are broken into two or three pieces and placed in a grid according to the value of the associated distinguished node. At most $O(n/\log n)$ shift edges link nodes which are in different rows of the grid. Of those edges which link nodes in the same row, at most $O(n/\log n)$ are contained in the first $\log k$ rows. For $i > \log k$, at most 2^{k-i} shift edges overlap at any point of the i th row. To insure that both the exchange and shift edges contained in the i th row can be inserted simultaneously using only $O(2^{k-i})$ horizontal tracks, it is necessary to consider them as maximal length chains of alternating shift and exchange edges. Since each node is incident to precisely one exchange edge and one shift edge, these chains are well defined. Further, no more than $2 \cdot 2^{k-i}$ chains can overlap at any point of the i th row. Otherwise, either 2^{k-i} exchange edges or 2^{k-i} shift edges would overlap at some point, a contradiction. Thus both the exchange and shift edges contained in the i th row can be inserted simultaneously using only $O(2^{k-i})$ horizontal tracks. By the arguments of section 3, this means that the shuffle-shift graph can be laid out in $O(n^2/\log^2 n)$ area, the least pos-

sible.

It is also possible to lay out the shuffle-shift-reverse graph in $O(n^2/\log^2 n)$ area, although we do not include the details here. We do mention the two key ideas involved in the layout, however. The first is to place together necklaces which are reverses of each other. This serves to shorten the horizontal distance of reverse edges. The second idea is to fold each part of the basic piece of each necklace in half (roughly speaking) so that all but $O(n/\log n)$ reverse edges link nodes which are in the same row. Using methods similar to those developed in section 3, it is then possible to show that the shuffle, exchange and shift edges can be inserted using only $O(n/\log n)$ additional horizontal and vertical tracks.

Since any permutation can be performed on the shuffle-exchange network in $O(\log n)$ time and $O(\log n)$ space per processor, the inclusion of other kinds of edges into the shuffle-exchange graph can save at most a multiplicative factor of $\log n$ time and space. For some applications, the savings in time and space may well justify the cost of inserting the edges. It is likely that many different kinds of edges can be added to the shuffle-exchange network without increasing the area of the layout by more than a constant factor. We have already mentioned that shift edges and reverse edges can be so added. In addition, it appears that transpose edges (those linking node i to node $n-1-i$) and cube-connected-cycles edges (those linking nodes which are adjacent in the cube-connected-cycles graph), can be inserted into the shuffle-shift-reverse graph without increasing the area by more than a constant factor. We are currently working toward a characterization of those edges which can be inserted without increasing the area by more than a constant factor.

Acknowledgments

In acknowledgment, we would like to thank the following people for their helpful remarks and suggestions: H. Chernoff, P. Elias, D. Hoey, C. Leiserson, R. Rivest, and M. Rodeh.

References

- [1] D. Hoey and C.E. Leiserson, "A Layout for the Shuffle-Exchange Network," Proceeding of the 1980 International Conference on Parallel Processing.
- [2] D. Kleitman, F.T. Leighton, G.L. Miller and M. Lopley are in the process of writing several MIT tech reports on layouts for the shuffle-exchange graph.
- [3] C.E. Leiserson, "Area-efficient graph layouts (for VLSI)," 21st Annual Symposium on Foundations of Computer Science, IEEE Computer Society. (October 1980).
- [4] D.S. Parker, "Notes on Shuffle-Exchange Type Switching Networks," IEEE Transactions on Computers, C-29, 3 (March 1980), pp. 213-222.
- [5] F.P. Preparata and J. Vuillemin, "The cube-connected-cycles: a versatile network for parallel computation. Technical Report 356, Institut de Recherche d'Informatique et d'Automatique (June 1979).
- [6] M. Rodch and D. Steinberg, personal communication of a result submitted to IEEE Transactions on Computers.
- [7] H.S. Stone, "Parallel processing with the perfect shuffle," IEEE Transactions on Computers, C-20, 2 (February 1971), pp. 153-161.
- [8] C.D. Thompson, A Complexity Theory for VLSI, Ph.D. Thesis, Carnegie-Mellon University Computer Science Department (1980).

Appendix

Let $\bar{\gamma}_k(t)$ denote the number of k -bit strings which do not contain $t-1$ consecutive zeros. Except for the string of all zeros (which we ignore), these are precisely the strings which do not contain the substring $v_t = \overbrace{10\dots 0}^t$. The proofs of Lemmas 1-4 depend on the following combinatorial result.

Theorem A: For large t and k ,

$$\bar{\gamma}_k(t) = 2^k e^{-k2^{-t}} e^{O(t2^{-t}, kt2^{-2t})}.$$

Proof: We first count the number $\bar{\gamma}'_k(t)$ of k -bit strings which do not contain an occurrence of v_t between the beginning and end of the string (i.e., for the time being, we ignore occurrences of v_t which begin at the end and end at the beginning of a string).

Fix t and let f_i denote the number of i -bit strings ending with v_t but which do not contain any other occurrences of v_t in the string. Set $F(x) = \sum_{i=0}^{\infty} f_i x^i$. Note

that $\bar{\gamma}'_k(t)$ is the $(k+t)$ th coefficient of $F(x)$. Let $f_i^{(j)}$ denote the number of i -bit strings ending in v_t which contain precisely j occurrences of v_t and set

$$F^{(j)}(x) = \sum_{i=0}^{\infty} f_i^{(j)} x^i. \text{ Since occurrences}$$

of v_t cannot overlap, it is not difficult to show that $F^{(j)}(x)$ is $F(x)^j$ for all $j \geq 1$.

Let g_i be the number of i -bit strings which end in v_t and set $G(x) =$

$$\sum_{i=0}^{\infty} g_i x^i. \text{ It is easily seen that } G(x) = \frac{x^t}{1-2x}. \text{ Also note that } G(x) = \sum_{j=1}^{\infty} F^{(j)}(x) = \sum_{j=1}^{\infty} F(x)^j = \frac{1}{1-F(x)} - 1. \text{ Thus}$$

$$F(x) = \frac{G(x)}{1+G(x)} = \frac{x^t}{1-2x+x^t}. \text{ Thus } \bar{\gamma}'_k(t) \text{ is}$$