9-2002

On the Area of Hypercube Layouts

Ronald I. Greenberg
Rgreen@luc.edu

Lee Guan

Recommended Citation
Ronald I. Greenberg and Lee Guan. On the area of hypercube layouts. Information Processing Letters, 84(1):41--46, September 2002.

On the Area of Hypercube Layouts [1]

Ronald I. Greenberg

Dept. of Mathematical and Computer Sciences, Loyola University, 6525 N. Sheridan Rd., Chicago, IL 60626-5385, USA

Lee Guan

Acecomm Corporation, 704 Quince Orchard Rd., Gaithersburg, MD 20878, USA

Abstract

This paper precisely analyzes the wire density and required area in standard layout styles for the hypercube. It shows that the most natural, regular layout of a hypercube of N^2 nodes in the plane, in a $N \times N$ grid arrangement, uses $\lfloor 2N/3 \rfloor + 1$ horizontal wiring tracks for each row of nodes. (In the process, we see that the number of tracks per row can be reduced by 1 with a less regular design, as can also be seen from an independent argument of Bezrukov et al..) This paper also gives a simple formula for the wire density at any cut position and a full characterization of all places where the wire density is maximized (which does not occur at the bisection).

Key words: interconnection networks, hypercube, wire density, VLSI layout area, mincut linear arrangement, optimal linear arrangement, channel routing

1 Introduction

The (binary) hypercube network has been widely considered as a network for parallel computing, but its VLSI layout requires a great deal of wiring area. Studies of communications capabilities of the hypercube versus other networks (e.g., [1–5]) have varied the width of links between nodes in order to equalize the hardware costs of the networks being compared under various cost measures, some of which are closely related to VLSI layout area.

Email address: lee_guan@yahoo.com (Lee Guan).

URL: http://www.cs.luc.edu/~rig (Ronald I. Greenberg).

[1] Partially supported by NSF grant CCR-9321388.

Recall that the interconnection pattern for a hypercube of N^2 nodes can be specified by numbering the nodes from 0 to N^2-1 and requiring a link between any two nodes whose numbers expressed in binary differ in exactly one bit. When the numbers differ in the ith bit from the right, we refer to the link between the nodes as a dimension i link. (Though the links between nodes are generally considered to be bidirectional, we count them as one wire for simplicity. Results quoted in this paper must be multiplied by 2 to obtain exact correspondence with results given in Dally [2] or Ranade and Johnsson [3].)

The network cost measure used by Dally [2] is bisection width (the minimum number of wires that must be cut to divide the set of nodes into two equal halves with no connections between them). This measure may be justified by Thompson's lower bound [6,7] indicating that area is at least 1/4 of the square of the bisection width. Thompson's bound, however, does not give a precise correspondence between bisection width and area. Furthermore, as Dally notes, the maximum wire density (number of wires that must cross a cutline) does not occur at the bisection in the "normal layout" of the hypercube (nodes placed as in Figure 1). (Note that each row and column of the layout is itself a hypercube, so we can focus henceforth on the layout of an N-node hypercube in a single row.)

Ranade and Johnsson [3] consider the actual area required for the normal layout by bounding the number of horizontal tracks per row required to lay out the interconnections (following the common approach of placing vertical wires in one chip layer and horizontal wires in another). (The situation involving vertical tracks is completely analogous to that involving horizontal tracks.) They focus, however, on optimality to within an unspecified constant factor and only upper bound the number of tracks per row as $N-1$, as obtained by the assignment of wires to tracks illustrated in Figure 1.

A more sophisticated track assignment by Chen, Agrawal, and Burke [8] (with a different ordering of the nodes), yields $N - \lg N$ tracks per row. [2]

A still better measure for the number of tracks per row, utilized in [4,5], is $\lfloor 2N/3 \rfloor$. That this number represents the congestion for the natural embedding of the hypercube into a square grid also follows from an independent statement of Nakano [9] and an argument of Bezrukov et al. [10].

This paper gives a short alternative proof of the congestion result that also yields a concise formula for the wire density at every cut position and a full characterization of all positions where density is maximized. The analysis is then extended to account for the exact placement of the terminals and wires in the layout. It would be desirable to make all nodes identical, e.g., by placing the connections of each node in order of dimension (as in Figure 1); this

[2] We use $\lg x$ for $\log_2 x$, and we assume N is a power of 2.

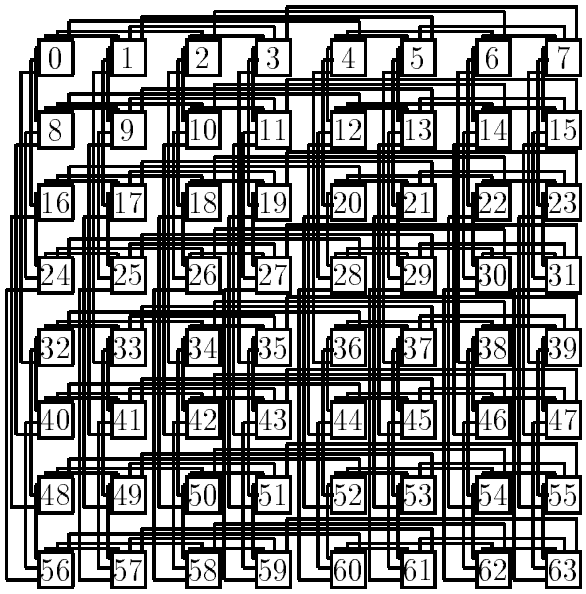

Fig. 1. The normal hypercube layout and a naive track assignment for $N^2 = 64$.

would be particularly convenient when implementing the common form of hypercube algorithm referred to as a "normal algorithm" (e.g., see [11]), in which only one dimension of communication links is used at any step, and the dimensions are used consecutively. Uniformity of nodes is also helpful for assembling the system and for replacing defective nodes. We show that such a uniform approach incurs a penalty of exactly one track per row in the VLSI layout, whereas full freedom to permute the terminals allows a layout with $\lfloor 2N/3 \rfloor$ tracks per row.

The rest of this paper is organized as follows. Section 2 introduces notation and provides background regarding the congestion result. Section 3, gives a simple formula for the wire density at each intercolumn position and a full characterization of those positions where the density is maximized. Then the analysis is extended to include the density at cutlines that run through nodes, which completes the analysis of the number of wiring tracks required. Section 4, comments on hypercube layouts in which the nodes are placed differently than in the normal scheme illustrated in Figure 1.

2 Background

As a first step towards determining the usage of wiring tracks in the normal hypercube layout, we focus on the intercolumn wire density per row. We sketch here a short proof, shown fully in [12], that the maximum intercolumn wire density per row in the normal hypercube layout is $\lfloor 2N/3 \rfloor$ and that the leftmost intercolumn position where this maximum is realized is position

$\lfloor (N+1)/3 \rfloor$. In the process we introduce notation for our main results in the next section and note important symmetry properties.

We define $f(i,k)$ to be the number of dimension k links (i.e., links spanning 2^{k-1} columns) that cross intercolumn position i in the normal layout. Using 0 to denote the position to the left of *all* the nodes, it is easy to see that the pattern for $f(0,k)$, $f(1,k)$, ..., $f(N-1,k)$ is 0, 1, 2, ..., $2^{k-1}-1$, 2^{k-1}, $2^{k-1}-1$, $2^{k-1}-2$, ..., 1, and repeat as necessary; we may express this as

$$f(i,k) = i\left(1 - 2\left(\left\lfloor \frac{i-1}{2^{k-1}} \right\rfloor \bmod 2\right)\right) \bmod 2^k \ . \tag{1}$$

Then we define $S(i,N)$ to be the total number of connections crossing intercolumn position i in the normal layout, i.e.,

$$S(i,N) = \sum_{k=1}^{\lg N} f(i,k) \ . \tag{2}$$

For the proof sketch in this section, there is also a more convenient mathematical expression for the maximum intercolumn wire density and and the leftmost position where the maximum is realized:

$$m(N) = (4N - (-1)^{\lg N} - 3)/6 \tag{3}$$
$$p(N) = (N - (-1)^{\lg N})/3 \tag{4}$$

Then the result discussed in this section is that $\max_{0<i<N} S(i,N) = m(N)$ and that $i = p(N)$ is the least i at which the maximum is achieved. The result follows from the following Lemma and two Theorems:

Lemma 1 *$S(i,N) = S(N-i,N)$ for $0 < i < N$.*

Proof sketch The result follows from showing $f(i,k) = f(N-i,k)$ for $0 < i < N$ and $1 \le k \le \lg N$, which follows from Equation 1.

Theorem 2 *$S(p(N),N) = m(N)$.*

Proof sketch The proof is by induction on $\lg N$ using Equations 1–4 and Lemma 1. □

Now we need only that $S(i,N) \le m(N)$, but the following theorem includes additional information to make the proof easier:

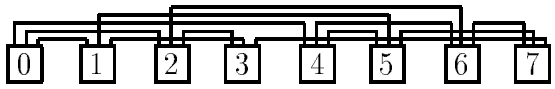

Fig. 2. Wiring a row in $m(N) = 5$ tracks for $N = 8$.

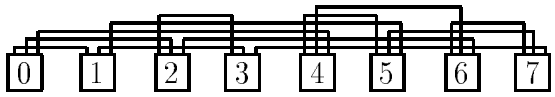

Fig. 3. Wiring a row requires $m(N)+1 = 6$ tracks for $N = 8$ when the wires leaving each node are in order of increasing dimension.

Theorem 3 *$S(i, N) \leq \min\{m(N), m(N) - (p(N) - i)\}$ for $0 < i < N$.*

Proof sketch We again use induction on $\lg N$.

We first use Equations 1–4 to show that $S(i, N) \leq m(N) - (p(N) - i)$. Then we show $S(i, N) \leq m(N)$ by considering the three cases of $i > N/2$, $i \leq p(N)$, and $p(N) < i \leq N/2$ and using Lemma 1 again. □

3 Number of wiring tracks

Though we know the maximum intercolumn wire density per row in the layout of Figure 1, we still need to determine the number of horizontal wiring tracks required to route the wires. Fortunately, an early channel routing algorithm of Hashimoto and Stevens [13], the left-edge algorithm, guarantees that the density and number of tracks are equal, since we have no vertical constraints (e.g., see [14]). To obtain a layout using exactly $m(N)$ tracks, however, we must be free to permute the locations of connections on each hypercube node so that the density (maximum number of wires crossing a vertical line) is no higher when the cutline runs through nodes than when it runs between nodes. A layout using $m(N) = 5$ tracks for one row of the 64-node hypercube is illustrated in Figure 2. (This figure uses a track assignment slightly different than the assignment produced by the left edge algorithm in order to reduce the number of wire crossings.)

If we require that each node has its connections in order of dimensions 1, 2, ... $\lg N$, we cannot achieve a routing in $m(N)$ tracks when $N > 2$; Figure 3 with 6 tracks shows the best layout of a row when $N = 8$. Even with this fixed order of connections, however, the density (and therefore the number of tracks) is just $m(N) + 1$ for $N > 2$. Our approach to obtaining this stronger result also produces a characterization of *all* locations where the density is maximized. We encapsulate these results in the following two Theorems.

Theorem 4 *The values of i in binary for which $S(i,N)$ is maximized are those obtained as follows. Starting from the leftmost bit of i and moving right, choose pairs of bits to be 01 or 10 except that when $\lg N$ is even, the last pair may be 11. When $\lg N$ is odd, the 1 remaining bit is set to 1.*

PROOF. Considering the number i represented in binary, define $b(i,j)$ to be the bit in the j-th position from the right ($1 \le j \le \lg N$), and define $e(i,j)$ to be the excess of 1's over 0's in bit positions greater than j (i.e., the number of 1's minus the number of 0's in the relevant portion of i's representation). Also, let r denote the number of consecutive 0's at the right end of i's representation. (Using the notation 0^r to represent a string of r 0's, note that with i of the form $X10^r$, $i-1$ is $X01^r$, and $-i$ is $\overline{X}10^r$, where $\overline{X}$ is the bitwise complement of X.) Starting from the definitions of $S(i,N)$ and $f(i,k)$ in Equations 2 and 1, we can express $S(i,N)$ as

$$
\begin{aligned}
&\sum_{k=1}^{\lg N} i(1-2b(i-1,k)) \bmod 2^k \\
&\quad = \sum_{k=1}^{\lg N}\sum_{j=1}^{k} b(i(1-2b(i-1,k)),j)\cdot 2^{j-1} \\
&\quad = \sum_{j=1}^{\lg N}\sum_{k=j}^{\lg N} 2^{j-1} \begin{cases} b(i,j) & \text{if } b(i-1,k)=0 \\ b(-i,j) & \text{if } b(i-1,k)=1 \end{cases} \\
&\quad = \sum_{j=1}^{\lg N} 2^{j-1}\left[\frac{b(-i,j)+b(i,j)}{2}(\lg N - j + 1) + \frac{b(-i,j)-b(i,j)}{2}e(i-1,j-1)\right] \\
&\quad = \sum_{j=1}^{r} 0 + 2^r(\lg N - r) + \sum_{j=r+2}^{\lg N} 2^{j-2}\left[\lg N - j + 1 + (1-2b(i,j))e(i-1,j-1)\right] \\
&\quad = 2^r(\lg N - r) + \lg N \sum_{j=r+2}^{\lg N} 2^{j-2} - \sum_{j=r+2}^{\lg N} j2^{j-2} + \sum_{j=r+2}^{\lg N} 2^{j-2}(1-2b(i,j))e(i,j) \\
&\quad = \tfrac{1}{2}N + \sum_{j=r+2}^{\lg N} 2^{j-2}(1-2b(i,j))e(i,j) \\
&\quad = \tfrac{1}{2}N - \sum_{j=1}^{r+1} 2^{j-2}(1-2b(i,j))e(i,j) + \sum_{j=1}^{\lg N} 2^{j-2}(1-2b(i,j))e(i,j) \\
&\quad = \tfrac{1}{2}N - \sum_{j=1}^{r} 2^{j-2}(e(i,0)+j) + 2^{r-1}(e(i,0)+r-1) + \sum_{j=1}^{\lg N} 2^{j-2}(1-2b(i,j))e(i,j) \\
&\quad = \tfrac{1}{2}(e(i,0)+N-1) + \sum_{j=1}^{\lg N-1} 2^{j-2}\cdot \begin{cases} e(i,j) & \text{if } b(i,j)=0 \\ -e(i,j) & \text{if } b(i,j)=1 \end{cases} . \qquad (5)
\end{aligned}
$$

From this expression, we can see that $S(i,N)$ is maximized by setting pairs of

bits greedily from the left end of i's representation, except for a slight variation when j becomes small, as in the theorem statement. (It is also easy to check that this maximum equals $m(N)$ of Equation 3.) $\Box$

Now we proceed to analyze the maximum density in a row of the layout when it is required that each node has its connections in order of dimensions 1, 2, ... $\lg N$. We define $T(i,p,N)$ to be the number of wires crossing a cutline just to the right of the p-th terminal position on a node in column $i-1$ for $1 \leq p \leq \lg N$ (so $T(i, \lg N, N) = S(i,N)$).

Theorem 5 *For $N > 2$, the maximum value of $T(i,p,N)$ over all i and p is $m(N)+1$ and is realized at an i for which $S(i,N) = m(N)$.*

PROOF. We can express $T(i,p,N)$ in terms of $S(i,N)$ by using the notation defined at the beginning of the proof of Theorem 4; specifically, $T(i,p,N) = S(i,N) + e(i-1,p)$. The term $e(i-1,p)$ can be reexpressed in terms of $e(i,p)$ based on the value of r defined above. For $p > r$, we have $e(i-1,p) = e(i,p)$. For $p \leq r$, we have $e(i-1,p) = e(i,p) + 2(r-p-1)$.

When $r = 0$, we know $p > r$, and we see that the strategy for choosing i described in Theorem 4 remains optimal, since the $e(i,p)$ term is small compared to 2^j for most values of j in Equation 5. With such an i, the largest $e(i,p)$ we can achieve is 1 (if at least one of the pairs of bits under the strategy of Theorem 4 is 10 or 11).

When $r = 1$, the situation is essentially the same as for $r = 0$, except that we must choose $p > 1$ to maximize $e(i-1,p)$. We still must choose an i that maximizes $S(i,N)$, and $e(i-1,p)$ will be at most 1.

Choosing $r \geq 2$ contradicts choosing i to maximize $S(i,N)$, and the deficit in the value of $S(i,N)$ cannot be recouped through the term $e(i-1,p)$. (For $r = 2$, $e(i-1,p)$ cannot exceed $e(i,p)$, while increasing values of r cause increasing deterioration in the value of $S(i,N)$.) $\Box$

Note that this result is not an idiosyncrasy of the particular ordering chosen for the terminals on each node. Rather, because of the symmetry in the layout, it is apparent than any ordering that is the same for all nodes leads to $m(N)+1$ tracks; an ordering that reduces $T(i,p,N)$ where it exceeds $m(N)$ will make a corresponding increase from $m(N)$ to $m(N)+1$ in another position.

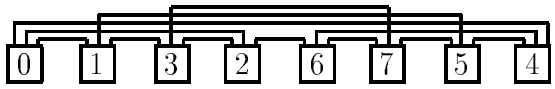

Fig. 4. The top row of a gray code derived layout for $N = 8$.

4 Alternative layouts

Another frequently considered method of mapping hypercube nodes to a regular grid is to use a gray code derived layout. The numbering of nodes in the top row of a gray code layout for a 64-node hypercube is illustrated in Figure 4. (Here we have not required the terminals on each node to be in dimension order.) Ranade and Johnsson [3] noted that the area and maximum wire length for the normal layout and the gray code layout are the same up to a constant factor. In fact, the arguments of Sections 2 and 3 can be extended to show that the maximum wire density and number of wiring tracks required per row is exactly the same for the gray code layout as for the normal layout, including a one track penalty when the nodes are identical. It is also easy to show that the total (horizontal) wire length per row is the same (in terms of the number of columns spanned). The maximum (horizontal) wire length in a row of the normal layout, however, is essentially half as large as for the gray code layout.

The results of Harper [15,16], Nakano [9], and Bezrukov et al. [10] show that the normal layout minimizes total wire length and intercolumn wire density, while a different layout minimizes maximum wire length. Bezrukov et al. also consider two new cost measures for embeddings of hypercubes into grids based on the frequent use of normal algorithms [17].

References

[1] S. Abraham, K. Padmanabhan, Performance of multicomputer networks under pin-out constraints, Journal of Parallel and Distributed Computing (1991) 237–248.

[2] W. J. Dally, Performance analysis of k-ary n-cube interconnection networks, IEEE Trans. Computers 39 (6) (1990) 775–785.

[3] A. G. Ranade, S. L. Johnsson, The communication efficiency of meshes, boolean cubes and cube connected cycles for wafer scale integration, in: Proceedings of the 1987 International Conference on Parallel Processing, 1987, pp. 479–482.

[4] R. I. Greenberg, L. Guan, An empirical comparison of area-universal and other parallel computing networks, in: Proceedings of the ISCA 9th International Conference on Parallel and Distributed Computing Systems, 1996, pp. 260–267.

[5] R. I. Greenberg, L. Guan, An empirical comparison of networks and routing strategies for parallel computation, in: Proceedings of the Eighth IASTED International Conference Parallel and Distributed Computing and Systems, Chicago, 1996, pp. 265–269.

[6] C. D. Thompson, Area-time complexity for VLSI, in: Proceedings of the 11th ACM Symposium on Theory of Computing, ACM Press, 1979, pp. 81–88.

[7] C. D. Thompson, A complexity theory for VLSI, Ph.D. thesis, Department of Computer Science, Carnegie-Mellon University (1980).

[8] C. Chen, D. P. Agrawal, J. R. Burke, dBCube: A new class of hierarchical multiprocessor interconnection networks with area efficient layout, IEEE Trans. Parallel and Distributed Systems 4 (12) (1993) 1332–1344.

[9] K. Nakano, Linear layouts of generalized hypercubes, in: Proceedings of the 19th International Workshop on Graph-Theoretic Concepts in Computer Science (WG '93), Springer-Verlag, 1994, pp. 364–365.

[10] S. L. Bezrukov, J. D. Chavez, L. H. Harper, M. Röttger, U.-P. Schroeder, The congestion of n-cube layout on a rectangular grid, Discrete Mathematics 213 (2000) 13–19.

[11] F. T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays · Trees · Hypercubes, Morgan Kaufmann, 1992.

[12] R. I. Greenberg, L. Guan, On the area of hypercube layouts, Eprint cs-dc/0105034, Comp. Sci. Res. Repository, http://arXiv.org/abs/cs.DC/0105034 (2001).

[13] A. Hashimoto, J. Stevens, Wire routing by optimizing channel assignment within large apertures, in: Proceedings of the 8th ACM/IEEE Design Automation Conference, IEEE Computer Society Press, 1971, pp. 155–169.

[14] T. Lengauer, Combinatorial Algorithms for Integrated Circuit Layout, John Wiley, 1990.

[15] L. H. Harper, Optimal assignments of numbers to vertices, Journal of the Society for Industrial and Applied Mathematics 12 (1) (1964) 131–135.

[16] L. H. Harper, Optimal numberings and isoperimetric problems on graphs, Journal of Combinatorial Theory 1 (1966) 385–393.

[17] S. L. Bezrukov, J. D. Chavez, L. H. Harper, M. Röttger, U.-P. Schroeder, Embedding of hypercubes into grids, in: MFCS '98, Springer-Verlag, 1998, pp. 693–701, lecture Notes in Computer Science 1450.