ON THE AREA OF BINARY TREE LAYOUTS *

R.P. BRENT
Department of Computer Science, Australian National University, Canberra, A.C.T. 2600, Australia

H.T. KUNG
Department of Computer Science, Carnegie–Mellon University, Pittsburgh, PA 15213, U.S.A.

Received 27 September 1979; revised version received 11 April 1980

Chip area, area complexity, tree, layout, lower bound

1. Introduction

The binary tree is an important interconnection pattern for chip layouts (see, e.g., [1,3,4,6]). Suppose that nodes are separated by at least unit distance and that a wire has unit width. Then the usual layout of a complete binary tree of n leaves illustrated in Fig. 1(a) takes order n log n area. In contrast, the 'H' layout illustrated in Fig. 1(b) takes only order n area [5].

Note that the 'H' layout has only $O(n^{1/2})$ leaves on the boundary. For current LSI technology, I/O ports (or pins) of a chip are typically all on the boundary of the chip. Thus, the 'H' layout enjoys a small area at the expense of a small number of possible I/O ports. We asked ourselves the question of whether or not one can design a complete tree layout of O(n) area and with all the leaves on the boundary. This note shows that it is not possible. More precisely, we show that if the boundary of the chip is a convex curve and leaves on the boundary are separated by unit distance, then order n log n area is necessary just for accommodating all the wires. The convexity assumption is not restrictive, since almost all existing chips do have convex boundaries for packaging reasons.

* This research was supported in part by the National Science Foundation under Grant MCS 78-236-76 and the Office of Naval Research under Contracts N00014-76-C-0370 and N00014-80-C-0236.

2. The layout model

We consider nodes to be points of negligible area which must be separated by at least one unit distance in the usual Euclidean metric. A *wire* is a line joining two nodes together with all points within half a unit distance from any point on the line. It is readily seen that if two nodes are separated by a distance d, then any wire joining them must occupy *area* at least d.

3. Total wire area

Let A(n) be the minimum total wire area needed for laying out a complete binary tree of n leaves, and A′(n) the minimum total wire area needed for laying out a complete binary tree of n leaves, not counting area taken by wires that correspond to edges on one

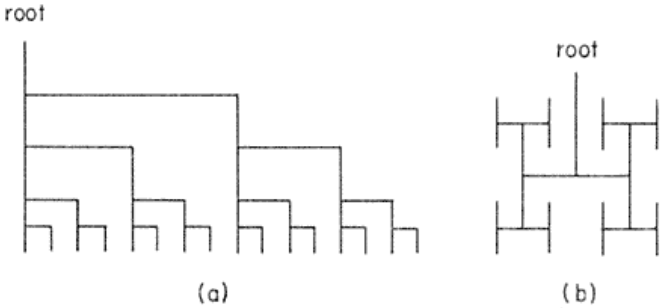
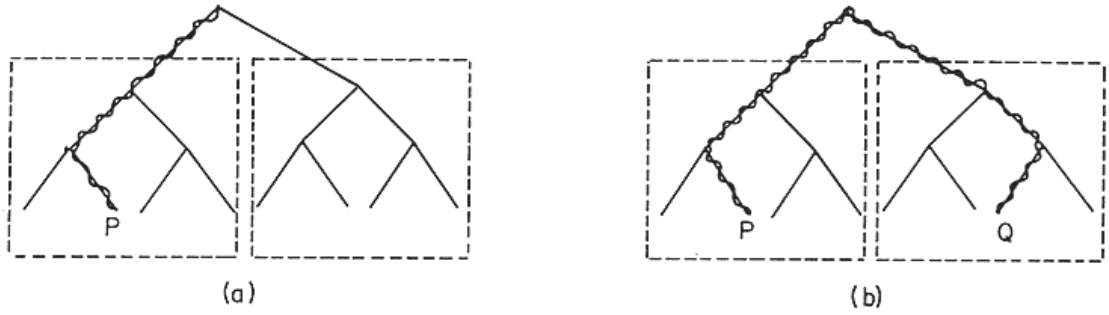


Fig. 1.



(a) (b)

Fig. 2.

designated path from the root to a leaf (one such path is marked by ~~ in Fig. 2(a)).

We shall derive lower bounds on A(n) for any chip layout. For simplicity we assume that n is a power of 2. This is not a significant restriction, for if $n \geqslant 2^k$, then $A(n) \geqslant A(2^k)$.

3.1. The degenerate case

To see the basic idea in our lower bound proof, consider first the degenerate case when all the n leaves of the tree are on some straight line segment of the boundary of the chip. We show that in this case, for n a power of 2, $n \geqslant 2$,

$$A(n) \geqslant 2A'(\tfrac{1}{2}n) + \tfrac{1}{2}n \qquad (1)$$

and

$$A'(n) \geqslant A(\tfrac{1}{2}n) + A'(\tfrac{1}{2}n). \qquad (2)$$

In any such layout, we claim that there are two leaf nodes P and Q that are at least $\frac{1}{2}n$ apart and that are in different subtrees of the root. Let P be the leaf node that is closest to one of the ends of the line segment that contains all the leaves. Choose Q to be the leaf node in the subtree of the root not containing P that is farthest from P. Then P and Q are separated by at least $\frac{1}{2}n - 1$ leaves on the line segment, and thus the wire connecting P and Q in the layout has area at least $\frac{1}{2}n$. This wire corresponds to a path in the tree, passing through the root (see Fig. 2(b)). Edges not on the path are all contained in the two subtrees, and the total length of wire needed for laying out these edges is bounded below by $2A'(\frac{1}{2}n)$ (by the definition of A'). Inequality (1) therefore follows.

To establish (2) we use Fig. 2(a). The result follows by noting that the total wire area for laying out the whole tree is at least that for laying out the two subtrees. Using (1) and (2) together with $A(2) \geqslant 2$ and $A'(2) \geqslant 1$, we can show that for $k \geqslant 0$,

$$A(2^k) \geqslant A'(2^k) \geqslant \tfrac{1}{6}k2^k.$$

Thus when n is a power of 2,

$$A(n) \geqslant \tfrac{1}{6}n \log n.$$

(Assume throughout that logarithms are to base 2.) Note that the above result is derived without using the assumption that the chip boundary is convex. The proof holds whether the leaves of the left and right subtrees in Fig. 2(a) and 2(b) are interleaved or not in the layout.

3.2. The general case

We now consider the general case when the n leaves are not necessarily on a straight line segment of the chip boundary. Since the chip is bounded by a convex curve, it can be circumscribed by a triangle whose area is at most twice the chip area (see, for example, [2]). Inside the triangle we can extend wires on the boundary of the chip outwards so that all leaves are now on the boundary of the triangle and they do not overlap. Note that if a straight line segment of the boundary contains k leaves, then (because the leaves are separated by unit distance) it has length at least k. Using this fact, we claim that we can choose leaf nodes P and Q, one from each subtree of the root, such that the distance between P and Q is at least $\frac{1}{12}n$. Let P be any leaf on the boundary of the triangle. Imagine that starting from P we walk along the boundary in two directions, one clockwise and one counter clockwise (see Fig. 3). The walk in each direction terminates as

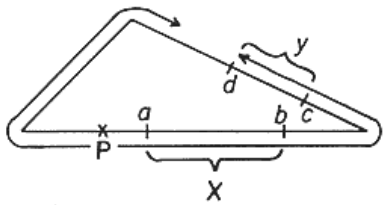


Fig. 3.

soon as it passes $\frac{1}{4}n$ leaves of the subtree of the root that does not contain P. At least one of the walking paths includes no more than one corner of the triangle. On this path one can define two closed intervals, one before and one after the corner, so that they are the smallest intervals containing all the $\frac{1}{4}n$ leaves of the subtree of the root that does not contain P. (The second interval may be empty if the path contains no corner.) Let the intervals be [a, b] and [c, d], with lengths x and y, respectively (see Fig. 3). Then the maximum distance D between P and any of the points b, c, and d satisfies

$$D \geqslant \max(x + 1, \tfrac{1}{2}y).$$

This implies that $D \geqslant \frac{1}{12}n$, since $x + 1 + y \geqslant \frac{1}{4}n$. Thus if we choose Q to be one of the points b, c, and d, that is farthest from P, then Q has distance at least $\frac{1}{12}n$ from P and does not belong to the subtree of the root containing P. Therefore, corresponding to (1) and (2), for the triangle with extended wires we have:

$$A(n) \geqslant 2A'(\tfrac{1}{2}n) + \tfrac{1}{12}n$$

and

$$A'(n) \geqslant A(\tfrac{1}{2}n) + A'(\tfrac{1}{2}n),$$

which imply that

$$A(n) \geqslant \tfrac{1}{36}n \log n, \qquad (3)$$

assuming as before that n is a power of 2.

4. The chip area

Assume that a chip consists of $c \geqslant 1$ layers, so at most c wires can overlap at any point. Then by (3) the circumscribed triangle has area at least $(n \log n)/36c$ and thus the chip has area at least $(n \log n)/72c$, when n is power of 2. If n is not a power of 2 the result holds with a slightly smaller constant. This, together with the upper bound achieved by the usual layout illustrated in Fig. 1(a), implies that

Theorem 1. For laying out a complete binary tree with n leaves on a chip, if all leaves are on the boundary of the chip, separated by unit distance, and the boundary is convex, then the minimum area required is of order exactly n log n.

Similar proofs and results hold for general t-ary trees for $t \geqslant 2$. Under the same assumptions one can show that the total chip area of a complete t-ary tree layout is bounded below by $\{(t-1)^2/[12ct(t+1) \times \log t]\}\, n \log n$.

Acknowledgment

We would like to thank Charles Leiserson and the referee for their comments on an early version of this paper.

References

[1] J.L. Bentley and H.T. Kung, A tree machine for searching problems, Proc. IEEE 1979 International Conference on Parallel Processing (1979) 257–266.
[2] H.G. Eggleston, Problems in Euclidean Space: Application of Convexity (Pergamon Press, New York, 1957).
[3] H.T. Kung, Let's design algorithms for VLSI systems, Proc. Conference on Very Large Scale Integration: Architecture, Design, Fabrication, California Institute of Technology (1979) 65–90.
[4] C.A. Mead and L.A. Conway, Introduction to VLSI Systems (Addison-Wesley, Reading, MA, 1980).
[5] C.A. Mead and M. Rem, Cost and performance of VLSI computing structures, IEEE J. Solid State Circuits 14 (2) (1979) 455–462.
[6] S.W. Song, A highly concurrent tree machine for database applications, Carnegie–Mellon University, Department of Computer Science (1980).