1986

Optimal Three-Dimensional Layouts of Complete Binary Trees

Ajay K. Gupta

Susanne E. Hambrusch
Purdue University, seh@cs.purdue.edu

Report Number:
86-605

ON OPTIMAL 3-DIMENSIONAL LAYOUTS OF COMPLETE BINARY TREES

Ajay K. Gupta
Susanne E. Hambrusch

CSD-TR-605
June 1986
Revised Nov. 1986

On Optimal 3-dimensional Layouts of Complete Binary Trees

Ajay K. Gupta

Susanne E. Hambrusch

Department of Computer Sciences
Purdue University
West Lafayette, IN 47907, USA

Abstract: We present optimal embeddings of an n-node complete binary tree in a three-dimensional or a two-dimensional grid when k, the size of one of the dimensions of the grid, is given. For the three-dimensional case we show how to obtain, for any k in the range $[1, n/2]$, a layout of $O(n + k \log n)$ volume. The same bound is shown to hold for the two-dimensional case when k is in the range $[\log n, n/2]$. We also show that these bounds are optimal within a constant factor.

Key words: Area and volume, binary trees, graph layouts.

1. Introduction

A commonly used model for laying out VLSI circuits (e.g. [Ls80], [Th80]) is to view the circuit as a bounded degree graph G in which the nodes correspond to processing elements and the edges correspond to wires. Graph G is then embedded in a two-dimensional or three-dimensional grid subject to the following assumptions and constraints:

(1) Each node occupies unit area. Distinct nodes of the graph are embedded at distinct grid intersection points.

(2) Edges have unit width and are routed along grid lines with the restriction that no two edges overlap except possibly when crossing perpendicular to each other or when bending (i.e., to form 'knock-knees'). Also, an edge cannot be routed over a node it

This work was supported by the Office of Naval Research under Contracts N00014-84-K-0502 and N00014-86-K-0689 and by the National Science Foundation under Grant DMC-84-13496.

does not connect.

The area of a two-dimensional layout is defined as the area of the "bounding-rectangle," and it equals the product of the number of vertical tracks and the number of horizontal tracks that contain a node or wire segments of the graph G. The volume of the three-dimensional layout is defined similarly, and equals the product of the number of horizontal tracks, the number of vertical tracks and the number of tracks in the third dimension.

Within three-dimensional layouts, two models, the *One-Plane-Active* and the *All-Plane-Active model,* have recently received considerable attention [Ag85][Le83][Pr83][Ro83]. In the first model only the grid intersection points on one of the boundary planes are allowed to contain nodes, while in the second model every grid intersection point can contain a node. The three-dimensional layouts considered in this paper are for the *All-Plane-Active model.* We show how to embed an n-node complete binary tree in a three-dimensional (3-d) or a two-dimensional (2-d) grid when k, the size of one of the dimensions of the grid is given. For any given k, $1 \leq k \leq n/2$, we show how to obtain a 3-d layout of $O(n + k \log n)$ volume. For the 2-d case we present an embedding using $O(n + k \log n)$ area, when k is in the range $[\log n, n/2]$. We also show that these bounds come within a constant factor of optimality.

The 2-d layout of complete binary trees has been studied extensively: If all the leaves of an n-node complete binary tree are required to be on the boundary, then $\Omega(n \log n)$ area is necessary and sufficient [Ul84]. We refer to the layout placing all leaves on one of the longer sides of the layout as the *B-tree* layout. Furthermore, the *H-tree* layout achieves $O(n)$ area by placing $\Theta(\sqrt{n})$ leaves on the boundary. Brent and Kung have shown that the length of the shorter side of a layout of the complete binary tree has to be at least $\log n$ [Br80] and optimal $O(n)$ area 2-d layouts for $\log n \leq k \leq n/\log n$ are described in [Cz86]. Layouts minimizing the maximum edge length are studied in [Pa81] and [Cz86]. Rosenberg in [Ro83] describes $O(n)$ volume embeddings of complete binary trees in 3-d grids for $k = n^{1/3}$. The results of our paper cover the entire range of k for both 2-d and 3-d layouts. Section 2 presents the construction of the layouts and section 3 presents the lower bound proofs.

2. Layouts of Complete Binary Trees

In this section we show how to optimally embed an n-node complete binary tree in a 3-d or a 2-d grid when k, the size of one of the dimensions of the grid is given. We assume, without loss of generality, that k is power of two. The bounds change only by a constant factor when k is not power of two. We refer to k as the length of the grid. For any given k, $1 \leq k \leq n/2$ (resp. $\log n \leq k \leq n/2$), we show how to obtain a layout of $O(n + k \log n)$ volume (resp. area). Our layout constructions and lower bound proofs make use of the fact that in a 2-d layout in which one side has length q, $O(q)$ nodes can be "pulled out" to the perimeter of the layout with only a constant factor increase in the area [Cz86].

2.1. 3-d Layout of a Complete Binary Tree

We now describe how to embed an n-node tree T in a 3-d grid of length k, $1 \leq k \leq n/2$, using $O(n + k \log n)$ volume.

We first divide the tree T into $(k+1)$ subtrees $T_0, T_1, ..., T_k$. The subtrees $T_1, T_2, ..., T_k$ are the subtrees rooted at the nodes at level $\log k$ in T and the $(k+1)^{th}$ subtree T_0 is the remaining tree formed by levels 0 through $\log k$ (see Fig 1). Note that the every T_i $(1 \leq i \leq k)$ consists of $\frac{n+1}{k-1}$ nodes, and T_0 consists of $2k - 1$ nodes.

Consider a 3-d grid G_1 of dimensions $\Theta(\sqrt{n/k}) \times k \times \Theta(\sqrt{n/k})$, which we view to consist of k planes, where each plane is of size $\Theta(\sqrt{n/k}) \times \Theta(\sqrt{n/k})$. Place the H-tree layout of T_i $(1 \leq i \leq k)$ on the i^{th} plane of G_1. This places the root r_i of T_i at the center of i^{th} plane (as shown in Fig 3).

In order to find the layout of T_0 (which has $\Theta(k)$ nodes), divide T_0 into l forests $F_1, F_2, ..., F_l$ $(l = \Theta(\log k/\sqrt{n/k}))$. Forest F_j $(1 \leq j \leq l)$ consists of $c\sqrt{n/k}$ levels of T_0, namely levels $c\ (j-1)\sqrt{n/k}$ to $c\ j\sqrt{n/k}$, for some constant c. See Fig 2. Now consider a 3-d grid G_2 of size $\Theta(\sqrt{n/k}) \times k \times l$ consisting of l planes of size $\Theta(\sqrt{n/k}) \times k$ each. Place the B-tree layout of F_j on the j^{th} plane such that the leaves of F_j lie on the right boundary if j is odd, and on the left boundary if j is even (Fig 3). Next *merge* the roots in the forest F_j with the corresponding leaves in the forest F_{j+1}. (By the *"merging"* of a leaf and a root we mean that a wire is routed from the leaf to the root and that the leaf

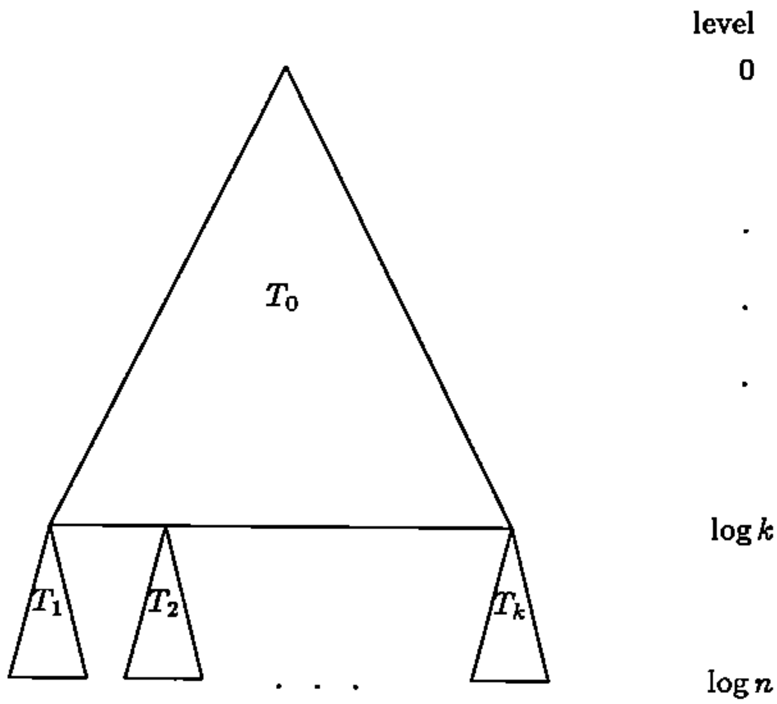

Fig 1: *Division of the tree T into $(k+1)$ subtrees*

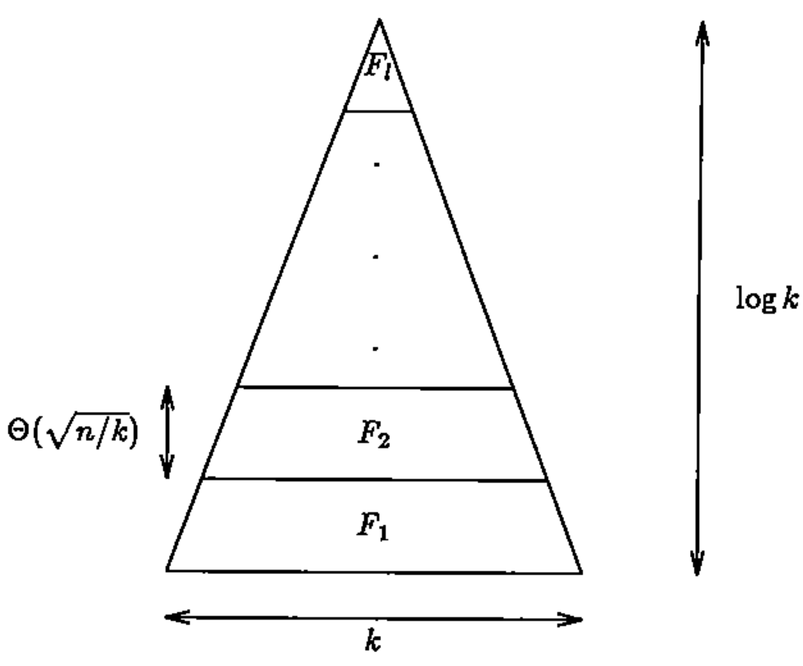

Fig 2: *Division of the tree T_0 into l subtrees*

node is deleted from the layout.) In other words, the layout of T_0 is obtained by folding the B-tree layout of T_0 in a zig-zag fashion onto the l planes of G_2.

The final step is to combine G_1 and G_2 by placing grid G_2 to the left of G_1 as shown in Fig 3. Merge the root r_i of T_i in G_1 with the i^{th} leaf of T_0 in G_2 by removing the i^{th} leaf and routing the wire incident on this leaf to r_i using the free track available from r_i to the boundary of G_1. The above construction gives the layout of T. The volume V of the final layout is:

$$V = \Theta(k * \sqrt{n/k} * (\sqrt{n/k} + \log k/\sqrt{n/k}))$$
$$= \Theta(n + k * \log k)$$
$$\leq O(n + k * \log n)$$

Depending on the value of k we have the following two cases:

Case 1: $1 \leq k \leq n/\log n \Rightarrow V = O(n)$.

Case 2: $n/\log n < k \leq n/2 \Rightarrow V = O(k \log n)$.

While for $k \leq n/\log n$ we obviously get an optimal volume, the volume increases for $k > n/\log n$. However, in section 3 we show that the bound obtained for Case 2 comes within a constant factor of optimality.

2.2. 2-d Layout of a Complete Binary Tree.

In this section we show that a similar layout strategy gives an optimal embedding of an n-node tree T in a 2-d grid of length k, $\log n \leq k \leq n/2$, using $O(n + k\log n)$ area. We assume, without loss of generality, that k refers to the longer side of the grid, hence we need to consider k only in the range $[\sqrt{n}, n/2]$. The result for $k \leq n/\log n$ has been known [Cz86], and it also follows from our construction which is given in terms of k.

Let $m = \lceil k^2/n \rceil$. First divide the tree T into $(m+1)$ subtrees $T_0, T_1, ..., T_m$ as described in section 2.1. The subtrees $T_1, T_2, ..., T_m$ are the subtrees rooted at the nodes at level $\lceil \log m \rceil$ in T and the $(m+1)^{th}$ subtree T_0 is the remaining subtree consisting of levels 0 through $\log m$ of T. Consider a rectangular grid G_1 of size $(m * \Theta(\sqrt{n/m})) \times \Theta(\sqrt{n/m})$. View G_1 as consisting of m square grids, each of size $\Theta(\sqrt{n/m}) \times \Theta(\sqrt{n/m})$, placed next

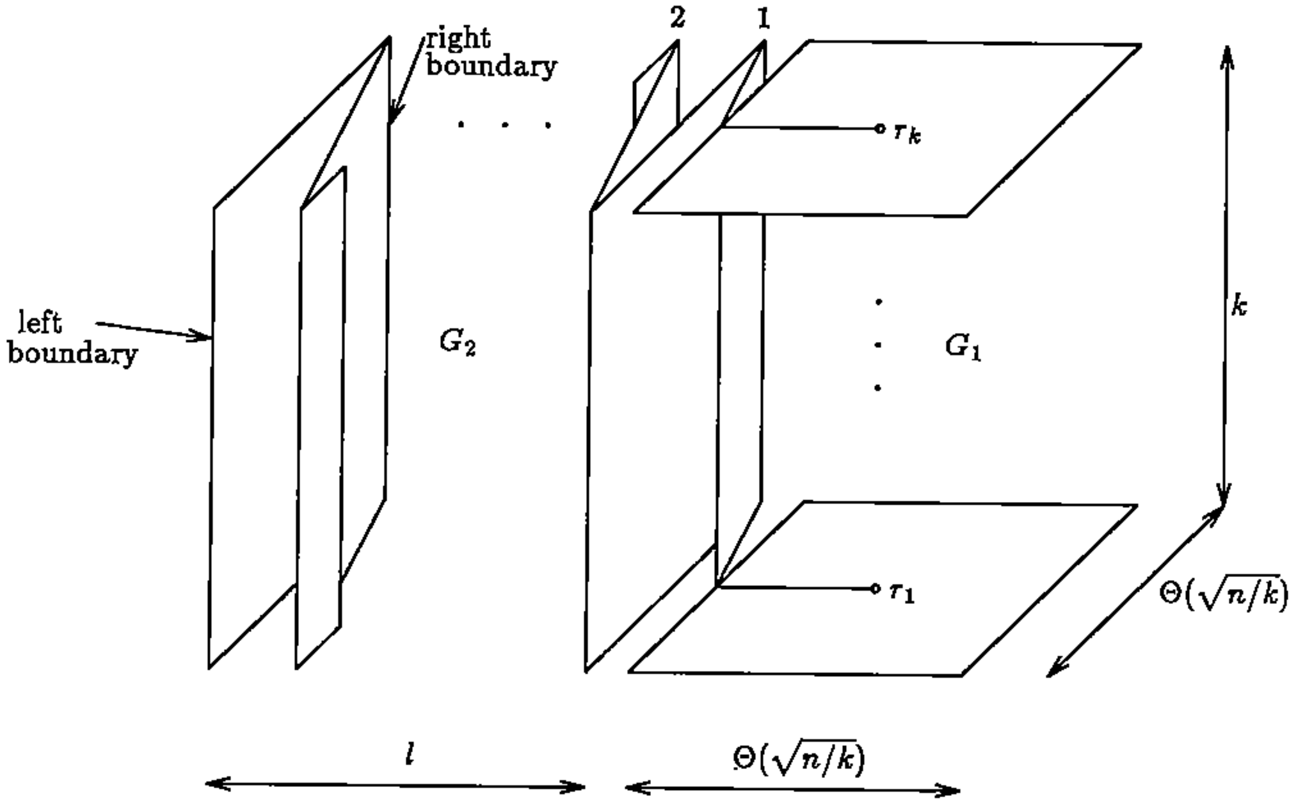

Fig 3: *Connecting layouts of T_0 and $T_1, T_2, ..., T_k$.*

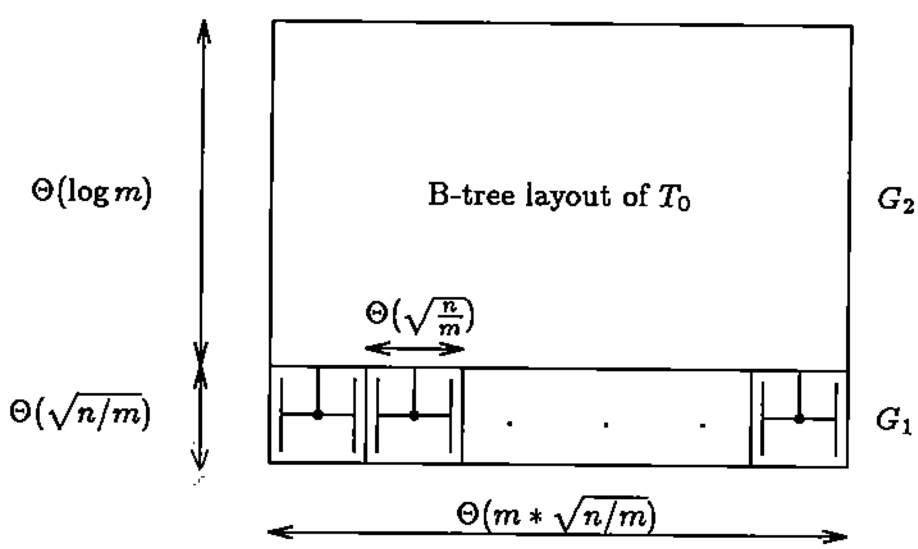

Fig 4: *Joining layouts of T_0 and $T_1, T_2, ..., T_m$.*

to each other. Place the H-tree layout of T_i $(1 \leq i \leq m)$ on the i^{th} square grid (as shown in Fig 4). Next embed T_0 on a rectangular grid G_2' of size $\Theta(m) \times \Theta(\log m)$ using the B-tree layout. The longer side of G_2' is then stretched such that the distance between two adjacent leaves in T_0 is $\Theta(\sqrt{n/m})$. The stretched layout, which we call G_2, has size $(m * \Theta(\sqrt{n/m})) \times \Theta(\log m)$. Finally, place G_2 on the top of G_1 (as shown in Fig 4). Merge the leaves of T_0, which are positioned on the bottom boundary of G_2, with the roots of T_i by using the free tracks available in G_1.

Now we have a layout for T in a 2-d grid whose area A is:

$A = \Theta((m * \sqrt{n/m}) * (\sqrt{n/m} + \log m)) \;\; = \Theta(n + k * \log(k^2/n))$

Depending on the value of k we again have the following two cases:

Case 1: $\sqrt{n} \leq k \leq n/\log n \Rightarrow A = O(n)$.

Case 2: $n/\log n < k \leq n/2 \;\; \Rightarrow A = O(k \log n)$.

Note that the total number of leaves on the boundary is $(m+2)\sqrt{\frac{n+1}{2m}} - 2$, which is $\Theta(k)$ (since $m = \lceil k^2/n \rceil$).

3. Lower Bounds.

In this section we show that the layouts given in section 2 are optimal. Lower bound proofs need only be given for k in the range of $(n/\log n, n/2]$. In the 2-d case we state the result in terms of the leaves on the boundary of the layout, since in any 2-d layout with a side of size $\Theta(k)$, k leaves of the tree can be pulled to the boundary with only a constant factor increase in the area of the layout. We start by giving the result for 2-d layouts since the result for 3-d layouts makes use of it.

Theorem 3.1. *Every 2-dimensional layout of an n-node complete binary tree requires $\Omega(k \log n)$ area, when k, $n/\log n < k \leq n/2$, leaves of the tree are required to be positioned on the boundary of the layout.*

Proof (via contradiction): Suppose there exists a layout L of an n-node tree T using area $A = o(k \log n)$ that has k leaves on the boundary. W.l.o.g. we can assume that in L all the k leaves of T are positioned on one longer side of the grid [Ul84]. Let l and w be the dimension of the rectangle R circumscribing the layout L, $l \geq w$. Since k leaves are on one longer side of R, $l \leq k$.

We transform the layout L into a layout L' of a $(2^{\lceil \log k \rceil + 1} - 1)$-node tree T'. L' will have $o(k \log k)$ area and all leaves of T' will be positioned on the boundary. Our transformation increases the area by at most a constant factor and is done as follows:

• Prune the tree T below level $\lceil \log k \rceil$ in L (by deleting nodes and wires corresponding to the edges below level $\lceil \log k \rceil$ of T). Thus obtain a layout L'' of the tree T'.

• Pull all the leaves of T' to one longer side s of L'' by inserting a new grid line in L'' for each leaf. This new grid line is used to route the wire from the leaf to the side s. This gives layout L' of the tree T'. As stated earlier, this step increases the length of L'' by at most a constant factor.

Now consider the area A of the layout L'.

$A = c_1 * l * w = o(k \log n)$, where c_1 is a constant.

$= o(k \log k)$

This is a contradiction, since any layout of T' with all the leaves positioned on the boundary requires area $\Omega(k \log k)$.

∎

Theorem 3.2: *In the All-Plane-Active model, any 3-dimensional layout of an n-node complete binary tree requires $\Omega(k \log n)$ volume, when one side of the 3-d grid is k and $n/\log n < k \leq n/2$.*

Proof (via contradiction): Assume that there exists a 3-d layout L_1 of the tree T using $o(k \log n)$ volume. Let the dimensions of L_1 be $l \times w \times k$ and w.l.o.g. assume $l \leq w$.

We first transform the given $l \times w \times k$ 3-d layout into an $lk \times lw$ 2-d layout L_2 by projecting the 3-d grid onto a 2-d grid [Le83]. Since $k > n/\log n$ we have $lk \geq lw$.

Let $n' = lk$, and let one of the longer sides of L_2 be s_0. We next show how to transform the layout L_2 into a layout L' of an $(2^{\lceil \log n' \rceil + 1} - 1)$-node tree T' that has all of its leaves positioned on one longer side of L'. The transformation will increase the area of L_2 by at most a constant factor. Depending on the value of n' we distinguish two cases.

Case 1: $n' \leq n/2$.

In this case prune the tree T below level $\lceil \log n' \rceil$ and thus obtain the tree T'. To obtain L' delete all the nodes and wires corresponding to the nodes and edges pruned. Next pull the leaves of T' to side s_0 of L_2 by intoducing a grid line for each leaf in L_2 as described in Theorem 3.1. This increases the area of L_2 only by a constant factor.

The new layout obtained from L_2, corresponds to a layout L' of the tree T' which has all the leaves of T' positioned on the boundary. Now consider the area A of L'.

$A = c * lk * lw = o(l * k \log n)$, where c is a constant.

$= o(n' \log n')$

Note that in this case $\lim_{n \to \infty} \log n / \log n' = 1$.

Case 2: $n' > n/2$.

In this case we augment tree T by subtrees $T_1, T_2, \ldots, T_{n/2}$ of height $\lceil \log n' - \log n \rceil$ each. Every subtree T_i will have as root the i^{th} leaf t_i of T. This augmentation results in the tree T' of height $\lceil \log n' \rceil$. See Fig 5.

The layouts of trees $T_1, T_2, \ldots, T_{n/2}$ are added to the existing layout L_2 to get L' as follows.

- Pull the leaves t_i of T to the side s_0 of L_2 as described in Theorem 3.1. Let m_i be the number of additional grid lines required such that distance between t_i and t_{i+1} is at least $\lceil n'/n \rceil$. Introduce m_i new horizontal grid lines in L_2 for t_i below the horizontal grid line on which t_i lies. This gives layout L_2' of size $(3n'/2 + 1) \times lw$ (in the worst case).
- Place the B-tree layouts of $T_1, T_2, \ldots, T_{n/2}$ in this order on a 2-d grid of size $(3n'/2 + 1) \times \lceil \log n' - \log n \rceil$ such that the root r_i of T_i is positioned on the corresponding horizontal grid line on which t_i lies. This gives a layout L_3 in which the leaves of $T_i' s$ lie on the longer side s_2 and the $r_i' s$ lie on the opposite side s_1. Next join layouts L_2' and L_3 at the side s_0 of L_2' and the side s_1 of L_3 and merge the $t_i' s$ and $r_i' s$ (as shown in Fig 6).

The new layout so obtained corresponds to a layout L' of the tree T' in which all the leaves of T' are positioned on the boundary. Now consider the area A of L'.

$A = c_1(3lk - n)/2 * lw + c_2(3lk/2+1) * \lceil \log n' - \log n \rceil$, where c_1 and c_2 are constants.

$= o(lk \log n) + o(lk \log lk)$

$= o(n' \log n')$, since $lk = n'$ and $lk > n/2$.

Now observe that in both Case 1 and Case 2 the area of the 2-d layout L' of tree T', with all the leaves positioned on the boundary of the layout, is $o(n' \log n')$, which is a contradiction. ∎

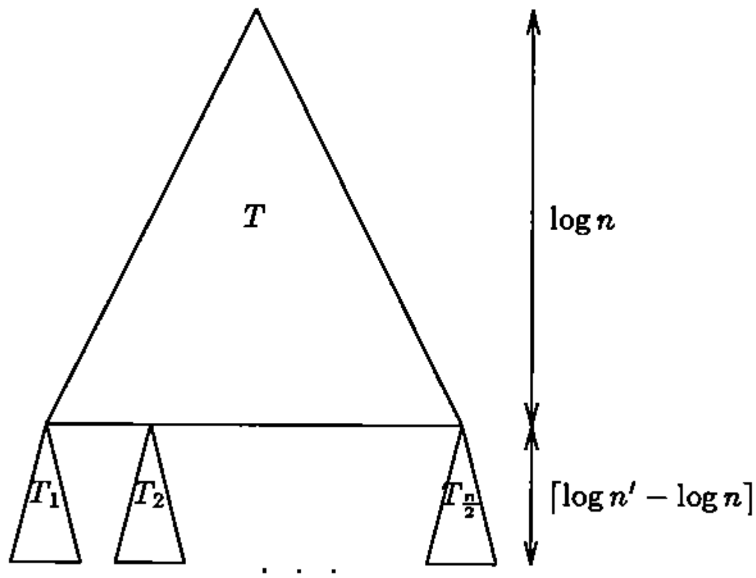

Fig 5: *Augmentation of the tree T*

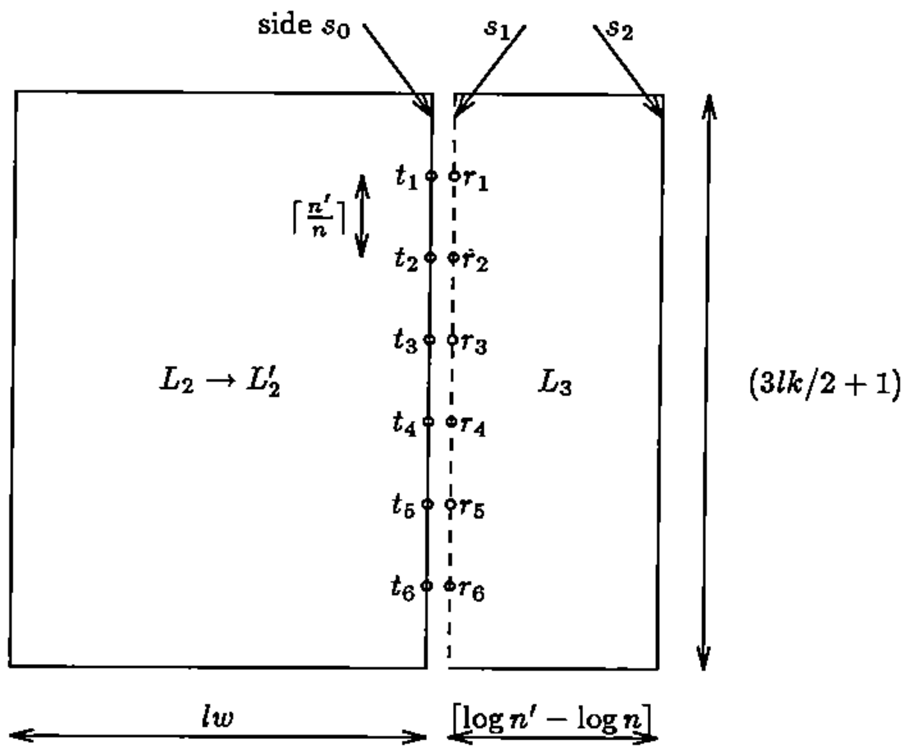

Fig 6: *Joining layouts L_2' and L_3*

References.

[Ag85] A. Aggarwal, M. Klawe, D. Lichtenstein, N. Linial and A. Wigderson, *"Multi-Layer Grid Embeddings,"* *Twenty-Seventh Annual Symposium on Foundations of Computer Science*, 1985, pp. 186-196.

[Br80] R. P. Brent and H. T. Kung, *"On The Area of Binary Tree Layouts,"* Information *Processing Letters*, 1980, pp. 46-48.

[Cz86] P. Czerwinski and V. Ramachandran, *"Optimal VLSI Graph Embeddings in Variable Aspect Ratio Rectangles,"* To appear in *Algorithmica*. 1986.

[Ls80] C. E. Leiserson, *"Area-Efficient Graph Layouts (for VLSI),"* Twenty-Second Annual *Symposium on Foundations of Computer Science*, Oct. 1980, pp. 270-281.

[Le83] F. T. Leighton and A. L. Rosenberg, *"Automatic Generation of Three-Dimensional Circuit Layouts," Proceedings IEEE International Conference on Computer Design*, 1983.

[Le86] F. T. Leighton and A. L. Rosenberg, *"Three- Dimensional Circuit Layouts," SIAM Journal of Computing*, Vol 15, No 3, August 1986, pp 793-813.

[Pa81] M. S. Paterson, W. L. Ruzzo and L. Snyder, *"Bounds on Minimax Edge Length for Complete Binary Trees," Proceedings of the* 13^{th} *Annual ACM Symposium on Theory of Computing*, May 1981, pp. 293-299.

[Pr83] F. P. Preparata, *"Optimal Three-Dimensional VLSI Layouts," Math. Systems Theory* 16, 1983, pp. 1-8.

[Ro83] A. L. Rosenberg, *"Three-Dimensional VLSI: A Case Study, " Journal Of the Association of Computing Machinary*, Vol. 30, No. 3, July 1983, pp. 397-416.

[Th80] C. D. Thompson, *"A Complexity Theory for VLSI,"* Ph.D. Thesis, Carnegie-Mellon University, 1980.

[Ul84] J. D. Ullman, *"Computational Aspects of VLSI,"* Computer Science Press, 1984.

[Va81] L. G. Valiant, *"Universality Considerations in VLSI Circuits," IEEE* Transaction on *Computers*, Vol. C-30, no. 2, 1981, pp. 135-140.