1988

A Lower Bound on Embedding Tree Machines with Balanced Processor Utilization

Ajay K. Gupta

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

Report Number:
88-794

A LOWER BOUND ON EMBEDDING TREE MACHINES WITH BALANCED PROCESSOR UTILIZATION

Ajay K. Gupta
Susanne E. Hambrusch

CSD-TR-794
July 1988

A Lower Bound on Embedding Tree Machines with Balanced Processor Utilization *

Ajay K. Gupta
Susanne E. Hambrusch

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

August 30, 1988

Abstract

In this paper we show that any embedding of a $2m+1$-node complete binary tree T into an m-node complete binary tree H requires a dilation of at least 3 when every node of H is assigned one interior and one leaf node of T, except one node which is assigned one interior and two leaf nodes of T.

*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 MIP-87-15652.

1. Introduction

The problem of embedding a guest graph T into a host graph H is an interesting and well-studied graph-theoretic problem [AR, CGC, JMR, R] with applications to parallel processing and parallel computing [BCJLR, BCLR, BSM, FF, GH, KA]. In [GH] we studied embeddings for the case when both T and H are complete binary tress with n and m nodes, respectively, and $n \geq m$. When the guest graph T has more nodes than the host graph H, a node of H is assigned a number of nodes of T. Thus an embedding $< f, g >$ of T into H is a surjective mapping f from the nodes of T to the nodes of H together with a mapping g that maps every edge $e = (v, w)$ of T onto a path $g(e)$ connecting $f(v)$ and $f(w)$. We say an embedding $< f, g >$ achieves a balanced utilization if every node of H has at most $\lceil n/m \rceil$ nodes of T assigned to it. Embeddings with a balanced utilization are of practical importance since they make every node of H share an equal load. Since the leaves of a tree network may be of a different type than the interior nodes [B], we also considered in [GH] embeddings that achieve a balanced leaf and interior utilization (i.e., every node of H has at most $\lceil \frac{n+1}{2m} \rceil$ leaf and at most $\lceil \frac{n-1}{2m} \rceil$ interior nodes assigned to it). Another important cost measure in graph embeddings is the dilation which measures the maximum distance in H between any two adjacent nodes in T.

In [GH] we presented two embeddings: one with balanced utilization and a dilation of 2 and another one with a balanced leaf and interior utilization and a dilation of $2 \log \log m + 1$. Both embeddings minimize other cost measures which are not discussed in this paper. From the techniques used in these two embeddings it is apparent that achieving a balanced leaf and interior utilization is harder than just achieving a balanced utilization. However, this does not hold for all values of n and m. If $n = (m+1)^d - 1$, for some non-negative integer d, then a dilation of 2 and a balanced leaf and interior utilization are achieved by an embedding in [GH]. It appears that $n = 2m + 1$ (i.e., the two trees differ in height by one) is the "hardest" case. In this paper we show if $n = 2m + 1$, any embedding achieving a balanced leaf and interior utilization requires a dilation of at least 3.

The lower bound is obtained by assuming that a dilation of 2 is possible and considering the assignments made to the leaves of H. Note that when $n = 2m + 1$ every node of H

has one leaf and one interior node of T assigned to it, with the exception of one node which has two leaves and one interior node assigned to it. We obtain a characterization for the leaf and interior nodes assigned to every leaf of H. We then characterize the relationships between sibling leaves, leaves in a common subtree of height 3, and leaves in a common subtree of height 4 in H. These characterizations lead to contradictions with respect to the balanced leaf and interior utilization. In the next section we give the details of this lower bound.

2. Lower Bound Proof

In this section we show that any embedding of a $2m+1$-node tree T into an m-node tree H must have a dilation of at least 3 if it achieves a balanced leaf and interior utilization. We first give some definitions and notations used throughout this paper. We then give a simple argument showing that a dilation of 1 is not possible and which gives the flavor of the techniques used. We then generalize these techniques to show that a dilation of 2 is also not possible.

Let T be a $2m+1$-node complete binary tree of height k and H be an m-node complete binary tree of height $k-1$. For clarity reasons, we will refer to the nodes of T as processing elements (PEs) and to the nodes of H simply as nodes. Let $< f, g >$ be an embedding of T into H with a balanced leaf and interior utilization. In such an embedding every node of H is assigned 1 interior and 1 leaf PE of T, except one node which is assigned 1 interior and 2 leaf PEs. When leaf PE l and interior PE u are assigned to a node v of H, we denote (l, u) as the assignment of v. The path between l and u is denoted by $P(l, u)$. If the path $P(l, u)$ contains 2 PEs that are on the same level in T, we say that the path $P(l, u)$ is a *bent path.* If, in a bent path, the children of the interior PE u are leaf PEs (i.e., u is on level $k-2$ in T), we say that $P(l, u)$ is a *bpl path* (*bent path with leaves*). See Figure 1 for an example of a bpl path. If the path $P(l, u)$ is a non-bent path, we say that it is a *straight path.* In an embedding with dilation δ the PEs that are adjacent to l or u have to be assigned to nodes that are at a distance of at most δ from v. We refer to these PEs as *boundary PEs.* Precisely, PE u_1 is called a boundary PE if it is adjacent to either l or u.

If u_1 is a leaf PE, then we say it is a *leaf boundary PE.* If u_1 is an interior PE, we say it is an *interior boundary PE.* In Figure 1, u_1 and u_2 are interior boundary PEs and l_1 and l_2 are leaf boundary PEs. We now show that an embedding $< f, g >$ can not achieve a dilation of 1 if it has a balanced leaf and interior utilization.

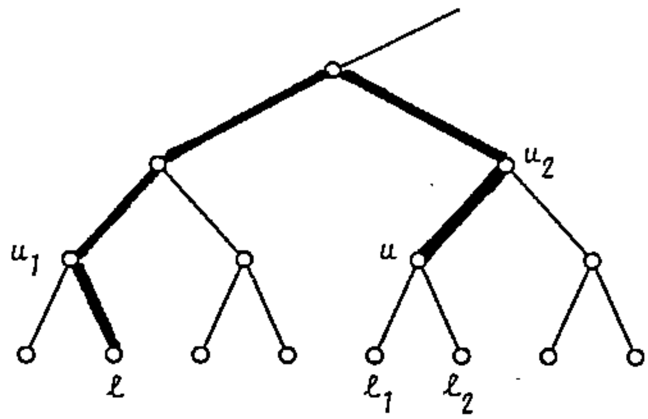


Figure 1: *A subtree of T with path $P(l, u)$ shown in bold.*

Lemma 1: *A dilation of 1 is not possible in an embedding $< f, g >$ with balanced leaf and interior utilization.*

Proof: Let (l, u) be the assignment of a leaf node v. Assume, without loss of generality, that both the parent and the sibling of v have 1 leaf PE assigned to them. If $P(l, u)$, the path from l to u, has length 2 or more, then it has at least 2 interior boundary PEs. These 2 interior PEs have to be assigned to the parent of v which is not possible in a balanced interior utilization. Thus, $P(l, u)$ must have length 1. Let (l_s, u_s) be the assignment of v_s, the sibling of v. Because of above argument, $P(l_s, u_s)$ must also be a path of length 1. $P(l, u)$ and $P(l_s, u_s)$ together have 2 leaf boundary PEs and at least 1 interior boundary PE. In order for the dilation to be 1, both leaf boundary PEs need to be assigned to the parent of v and v_s which is not possible in a balanced leaf utilization. ∎

For the remainder of this paper, let $< f, g >$ be an embedding of T into H with a dilation of 2 and a balanced leaf and interior utilization. Let (l, u) be the assignment of any leaf node v in H. Then, Lemmas 2 and 3 partially characterize $P(l, u)$, the path from l to u in T.

Lemma 2: *If the path $P(l,u)$ is a bent path, then it is a bpl path.*

Proof: Assume that $P(l,u)$ is bent path, but not a bpl path. Then, $P(l,u)$ has a total of 4 interior boundary PEs: one adjacent to l and 3 adjacent to u. These four PEs have to be assigned to either the sibling, the parent, or the grand-parent of v. This is not possible, since each node is assigned only one interior PE in the embedding $< f, g >$. ∎

Lemma 3: *If the path $P(l,u)$ is a straight path, then it has length at most 2.*

Proof: Assume that $P(l,u)$ is a straight path having length at least 3. Then $P(l,u)$ has 4 interior boundary PEs and the situation is as in Lemma 2. ∎

Throughout this paper, whenever we refer to a subtree H_i of H (resp. T_j of T) we mean the subtree of height i (resp. j) whose leaves are leaves in H (resp. T). Let H_3 be a subtree of height 3 in H whose nodes are indexed as shown in Figure 2. Assume, without loss of generality, that no node of H_3 has two leaf PEs of T assigned to it. Let (l_i, u_i) be the assignment of leaf node h_i, $0 \le i \le 3$. We will refer to the path $P(l_i, u_i)$ simply as the path P_i. We now describe a lemma that partially characterizes the assignments of sibling leaves in H_3.

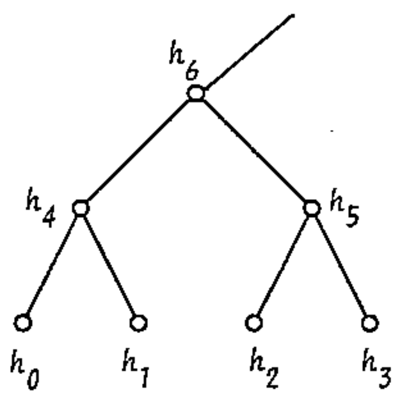


Figure 2: *Subtree H_3 and its indexing.*

Lemma 4: *Let (l_0, u_0) and (l_1, u_1) be the assignment of h_0 and h_1, respectively. Then, l_0, l_1, u_0, and u_1 come from a common subtree of height 3 in T.*

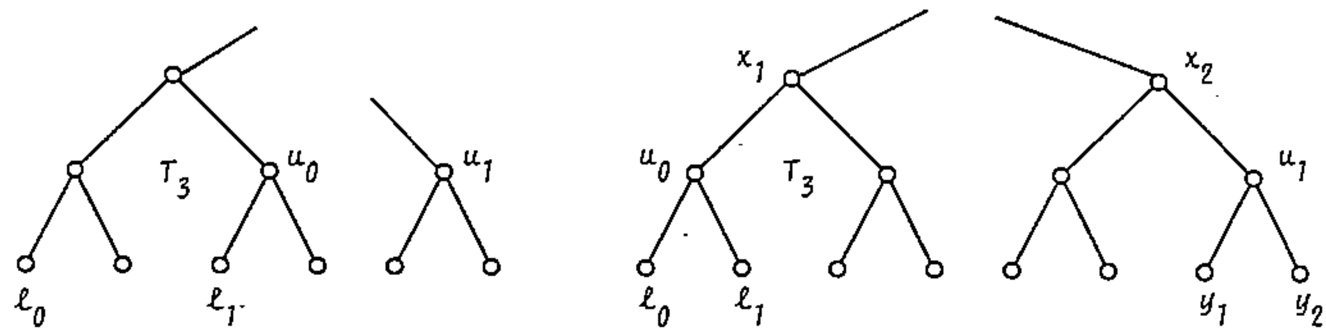
Proof: Assume that l_0, l_1, u_0, and u_1 do not come from a common subtree of height 3. Let T_r, $r \ge 4$, be the smallest subtree of T that contains l_0, l_1, u_0, and u_1. There are only two nodes, namely h_6 and h_4, at a distance of at most 2 from h_0 and h_1. Since each node of H_3 is assigned 1 leaf and 1 interior PE, a balanced leaf or interior utilization is not possible when P_0 and P_1 together have more than 2 leaf or more than 2 interior boundary PEs. The proof given below considers all possible assignments of l_0, u_0, l_1, and u_1 in T_r.

Because of Lemmas 2 and 3, each of P_0 and P_1 is either a bpl path or a straight path of length at most 2. We first show that neither P_0 nor P_1 can be a straight path of length 2. Assume without loss of generality that P_0 is a straight path of length 2. Then, if u_1 is not a child of u_0, path P_0 has 3 interior boundary PEs which is not possible in an embedding with balanced interior utilization. Hence, assume that u_1 is a child of u_0. Since l_1 can not be in a common subtree of height 3 containing l_0, u_0, and u_1, P_0 and P_1 together have 3 interior boundary PEs not yet assigned: two from P_0 and one from P_1, namely the parent of l_1. It thus follows that neither P_0 nor P_1 can be a straight path of length 2. In the remainder of this proof we consider the remaining combinations in which P_0 and P_1 can be bpl paths or straight paths of length 1. We distinguish between two cases depending on whether l_0 and l_1 come from different subtrees of height 3 in T_r or not.

Case 1: *l_0 and l_1 come from a common subtree of height 3 in T_r.*

Let T_3 be the subtree of height 3 containing l_0 and l_1. Since we assumed that l_0, u_0, l_1, and u_1 come from T_r, at least one of u_0 or u_1 must come from $T_r - T_3$, where $T_r - T_3$ denotes the subtree after the PEs from T_3 have been removed from T_r. If both u_0 and u_1 come from $T_r - T_3$, then P_0 and P_1 together have 4 leaf boundary PEs which is not possible in a balanced leaf utilization. Hence, assume that exactly one of u_0 or u_1 comes from $T_r - T_3$. Without loss of generality let u_0 come from T_3 and u_1 come from $T_r - T_3$. There are now two cases depending on whether l_0 and l_1 are siblings or not.

If l_0 and l_1 are not siblings, then u_0 is the parent of l_0 or l_1. We depict one such situation in Figure 3(a). In this case P_0 and P_1 together have 3 leaf boundary PEs and hence balanced leaf utilization is not possible. Consider the case when l_0 and l_1 are siblings. If u_0 is not the parent of l_0 and l_1, then P_0 and P_1 have a total of 4 leaf boundary PEs which is not possible. Thus assume that u_0 is the parent of l_0 and l_1. Then P_0 and P_1 have 2 interior boundary PEs, say x_1 and x_2, and 2 leaf boundary PEs, say y_1 and y_2, as shown in Figure 3(b). These boundary PEs have to be assigned to h_6 and h_4. Without loss of generality, let x_1 and y_1 be assigned to h_6 and the other two PEs be assigned to h_4. In order to obtain a contradiction we consider the assignments to leaf nodes h_2 and h_3. Let (l_2, u_2) and (l_3, u_3) be these assignments, respectively. Because of Lemmas 2 and 3, paths P_2 and P_3 have to be either bpl paths or straight paths of length at most 2.



(a) l_0 and l_1 are not siblings. (b) l_0 and l_1 are siblings.

Figure 3: *Assignments in Case* 1 *of Lemma* 4.

We first show that neither P_2 nor P_3 can be a straight path of length 2. We can assume that l_2, u_2, l_3, and u_3 come from a common subtree of height 3, since if they do not, an argument as given earlier for P_0 and P_1 applies. Since there is only one node at a distance of 2 from the leaves in a subtree of height 3, only one of P_2 or P_3 can be a straight path of length 2. Assume, without loss of generality, that P_3 has length 2 (P_2 is either a bpl path or has length 1). Since the dilation is 2, u_3 can not coincide with either x_1 or x_2 and hence path P_3 has 2 interior boundary PEs that are distinct from x_1 and x_2. Since h_6 already has an interior PE, namely x_1, assigned to it, only h_5 is available. Two interior PEs are now required to be assigned to h_5 which is not possible in a balanced interior utilization.

Consider now the case when P_2 and P_3 are bpl paths or straight paths of length 1. PEs u_2 and u_3 together have 4 children as leaf PEs two of which may coincide with l_2 and l_3. Even when u_2 (or u_3) is a sibling of u_0 or u_1, paths P_2 and P_3 together have at least 2 leaf boundary PEs which are required to be assigned to h_5 or h_6. Since h_6 already has a leaf PE, namely y_1, assigned to it, we have a contradiction of balanced leaf utilization. This concludes Case 1 in which we assumed that l_0 and l_1 come from a common subtree of height 3.

Case 2: *l_0 and l_1 come from different subtrees of height 3 in T_r.*

Let T_3 be the subtree of height 3 containing l_0. Then, $T_r - T_3$ contains l_1. Recall that PEs u_0 and u_1 are on level $k-2$ in T. If at least one of u_0 and u_1 is not the parent of either l_0 or l_1, then P_0 and P_1 together have at least 3 leaf boundary PEs. These 3 leaf PEs have to be assigned to h_6 or h_4 which is not possible in a balanced leaf utilization. It now follows that u_0 and u_1 are parents of l_0 and l_1. In this case P_0 and P_1 together have 2 interior boundary PEs, say x_1 and x_2, and have 2 leaf boundary PEs, say y_1 and y_2. This situation is as in Figure 3(b) with l_1 and y_1 switching their positions. Thus, the argument is identical to the one given for the situation of Figure 3(b) and is therefore omitted. This concludes Case 2 and Lemma 4 follows. ∎

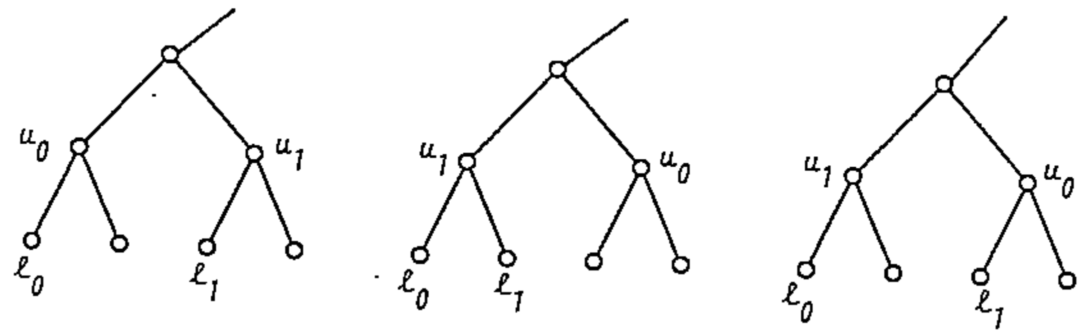
Since there is only one node at a distance of 2 from the leaves in a subtree of height 3, and because of Lemma 4 we have the following corollary.

Corollary 5: *Let (l_0, u_0) and (l_1, u_1) be the assignment of two leaf nodes in H that are siblings. Then, paths P_0 and P_1 can not both be of length 2.*

We now consider the assignments of 4 consecutive leaf nodes in a common subtree of height 3 in H. From Lemmas 2 and 3 we know that the path of the assignment of a leaf node has to be either a bpl path or a straight path of length at most 2. Because of symmetry in complete binary trees we need not consider all the possible combinations of such paths in an assignment of 4 leaf nodes. We next describe how to exploit this symmetry. We say a path (resp. a leaf of H) is of *type* b if it is a bpl path and it is of *type* 2 (resp. 1) if it is a straight path of length 2 (resp. 1). Let (l_0, u_0) and (l_1, u_1) be the assignments of two sibling leaf nodes in H. We say that paths P_0 and P_1 have a type assignment qr, when P_0 is of type q and P_1 is of type r; $q, r \in \{b, 2, 1\}$. Let P_0 and P_1 have a type assignment qr, and assume that P_0 and P_1 together have b_l leaf and b_i interior boundary PEs. If P_0 and P_1 have a type assignment rq and the number of leaf and interior boundary PEs is as before, then the two type assignments are considered identical because of symmetry. Obviously, symmetric type assignments do not need to be considered separately. We now examine symmetric situations for sibling leaves in more detail.

From Corollary 5 we know that two sibling leaves cannot have a type assignment

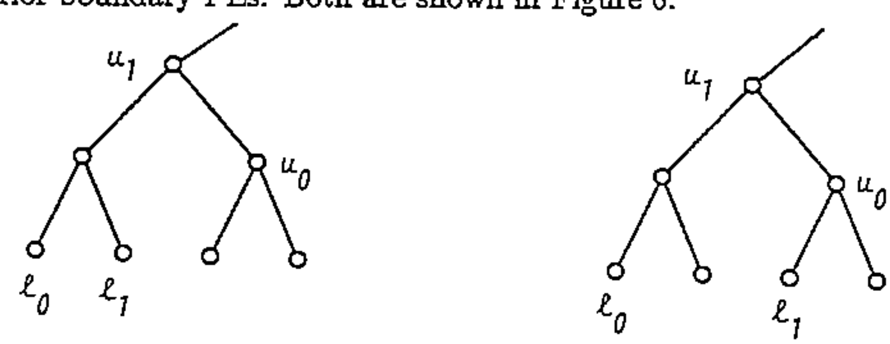
22. This leaves type assignments $11, b1, bb, b2$, and 21. Each one of $11, b1$, and bb has one interior and two leaf boundary PEs and the possible positions of l_0, u_0, l_1 and u_1 are shown in Figure $4(a) - (c)$. Note that there is some freedom in how l_0, u_0, l_1 and u_1 are chosen in T, but cases not shown in Figure 4 are all identical because of symmetry in complete binary trees. Because of symmetry in type assignments, as an example, $b1$ is identical to $1b$.



(a) Type assignment 11. (b) Type assignment $b1$. (c) Type assignment bb.

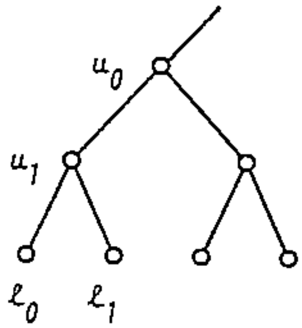
Figure 4: *Positions of PEs in T of type assignments having 1 interior and 2 leaf boundary PEs.*

Assume now that two sibling leaves in H have a type assignment $b2$. Then there are two possibilities depending on whether or not l_0 and l_1 are siblings in T. If they are, $b2$ has two interior and two leaf boundary PEs (see Figure $5(a)$); if they are not, $b2$ has two interior and one leaf boundary PEs (see Figure $5(b)$). We refer to the first possibility as $(b2)'$ and to the second one as $(b2)''$. The last type assignment to be considered is 21 for which we also have two possibilities (again depending on whether l_0 and l_1 are siblings in T). One, $(21)'$, has no leaf and two interior boundary PEs, and the second one, $(21)''$, has one leaf and two interior boundary PEs. Both are shown in Figure 6.

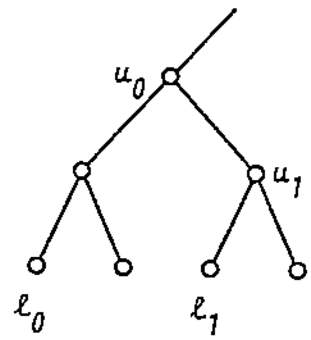


(a) Type assignment $(b2)'$. (b) Type assignment $(b2)''$.

Figure 5: *Positions of PEs in T of type assignment $b2$.*



(a) Type assignment (21)'.



(b) Type assignment (21)''.

Figure 6: *Positions of PEs in T of type assignment* 21.

We now take symmetry in type assignments one step further. Let $S = \{11,\ b1,\ bb,\ (b2)',\ (b2)'',\ (21)',\ (21)''\}$. Consider four consecutive leaves in H belonging to a common subtree of height 3. Let H_3 be such a tree. Let Q be the type assignment of the first two leaves in H_3, R be the type assignment of the other two leaves, $Q, R \in S$. Then, the requirements on the interior and leaf boundary PEs for QR are the same as for RQ.

Without taking into account any symmetry, there are a total of 12 possible type assignments to two sibling leaves in H_3, namely the type assignments 11, $b1$, $1b$, bb, $(b2)'$, $(2b)'$, $(b2)''$, $(2b)''$, $(21)'$, $(12)'$, $(21)''$, and $(12)''$. Since sibling leaf nodes can have any one of these 12 type assignments, we have a total of $12 * 12 = 144$ possible type assignments to the 4 leaf nodes of H_3. Making use of symmetric assignments in two sibling leaf nodes and in the pair of two sibling leaf nodes, there are 28 different type assignments to the leaves of H_3 to be considered. They are listed in Table 1. The next lemma will reduce this number to 18. Let A be the set of all type assignments in S that have two leaf boundary PEs; i.e., $A = \{11, b1, bb, (b2)'\}$.

Lemma 6: *If two sibling leaves in H_3 have a type assignment that is in set A, then the other two sibling leaves cannot have a type assignment which is in A.*

Proof: We show the assignments from set A in Figures $4(a)-(c)$ and $5(a)$. Every one of these has 2 leaf boundary PEs. Let Q (resp. R), $Q, R \in A$, be the type assignment of the leaves h_0, and h_1 (resp. h_2 and h_3). The PEs in Q (resp. R) must come from a common subtree of height 3, and hence the two subtrees are disjoint. Let y_1 and y_2 be the two leaf boundary PEs of type assignment Q and assume without loss of generality that y_1 is assigned to h_6 and y_2 is assigned to h_4. Since neither y_1 nor y_2 can coincide with the 2 leaf boundary PEs of R and y_1 is already assigned to h_6, h_5 must accommodate the 2 leaf boundary PEs of R. This is not possible in a balanced utilization and hence the lemma follows. ∎

1111	**11(b2)″**	b1bb	**b1(21)″**	**bb(21)′**	**(b2)′(21)′**	**(b2)″(21)″**
11b1	**11(21)′**	b1(b2)′	bbbb	**bb(21)″**	**(b2)′(21)″**	**(21)′(21)′**
11bb	**11(21)″**	**b1(b2)″**	bb(b2)′	(b2)′(b2)′	**(b2)″(b2)″**	**(21)′(21)″**
11(b2)′	b1b1	**b1(21)′**	**bb(b2)″**	**(b2)′(b2)″**	**(b2)″(21)′**	**(21)″(21)″**

Table 1: *Possible type assignments to the 4 leaves in H_3 after removing the symmetric ones (type assignments not in bold are eliminated by Lemma 6).*

From Lemma 6 it follows that the 4 leaves of H_3 can not have a type assignment which is either $1111, 11b1, 11bb, 11(b2)', b1b1, b1bb, b1(b2)', bbbb, bb(b2)'$, or $(b2)'(b2)'$. We thus are left with $28 - 10 = 18$ possible type assignments to the 4 leaves of H_3 which are shown in bold in Table 1. Theorem 7 considers these remaining type assignments and shows that all of them are not possible.

Theorem 7: *A dilation of 2 is not possible in an embedding $< f, g >$ with balanced leaf and interior utilization.*

Proof: Let H_5 be the subtree of height 5 in H whose leaves are leaves of H and whose leftmost subtree of height 3 is the subtree H_3. Throughout this proof nodes of H_5 are indexed as shown in Figure 7. We assume, without loss of generality, that no node in H_5 has 2 leaf PEs assigned to it.

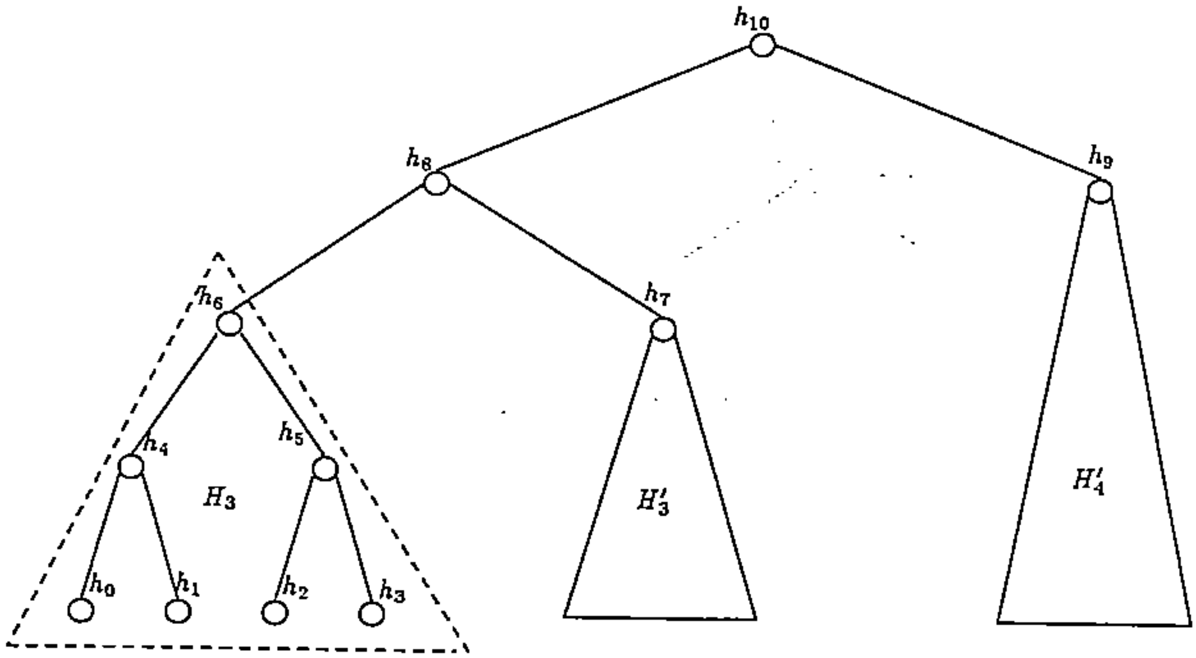


Figure 7: Subtree H_5 and its indexing.

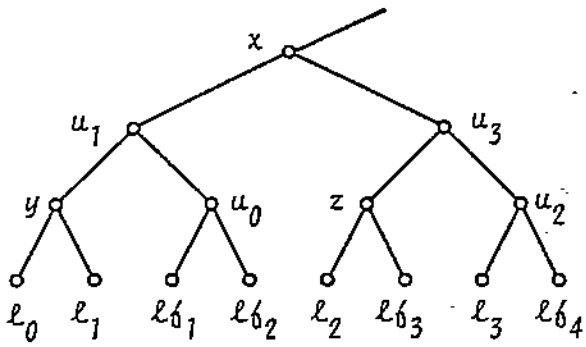
Assume that the four leaves in H_3 have a type assignment shown in bold in Table 1. We partition these 18 type assignments into two sets, set B and set C. Set B has the following 9 type assignments : $(b2)'(b2)''$, $(b2)'(21)'$, $(b2)'(21)''$, $(b2)''(b2)''$, $(b2)''(21)'$, $(b2)''(21)''$, $(21)'(21)'$, $(21)'(21)''$, and $(21)''(21)''$. We first show that a type assignment in this set requires a leaf PE to be assigned to h_8. Set C consists of the remaining 9 type assignments and we will show that a type assignment in this set requires a leaf PE to be assigned either to h_8 or to one of the 3 nodes from set $\{h_{10}, h_8, h_7\}$.

Let (l_i, u_i) be again the assignment of leaf node h_i, $0 \le i \le 3$. We first show that in any type assignment from set B, the PEs l_i and u_i, $0 \le i \le 3$, have to come from a common subtree of height 4 in T. We know from Lemma 4 that l_0, u_0, l_1, and u_1 (resp. l_2, u_2, l_3, and u_3) come from a common subtree of height 3. We also know that these two subtrees are disjoint. Let T_4 be the subtree of height 4 that contains l_0, u_0, l_1, and u_1. Assume without loss of generality that these 4 PEs are in the left subtree of T_4. We now show that l_2, u_2, l_3, and u_3 must be in the right subtree of T_4. Assume for the sake of contradiction that they are not. As before, let P_i denote the path $P(l_i, u_i)$, $0 \le i \le 3$. Paths P_0, P_1, P_2, and P_3 have a type assignment from set B. Since P_0 and P_1 have a type assignment which is either $(b2)'$, $(b2)''$, $(21)'$, or $(21)''$, paths P_0 and P_1 together have 2 interior boundary PEs, as can also be seen in Figures 5 and 6. Paths P_2 and P_3 have a type assignment which is either $(b2)''$, $(21)'$, or $(21)''$ and hence paths P_2 and P_3 together also have 2 interior boundary PEs. Since l_0, u_0, l_1 and u_1 are in the left subtree of T_4 and l_2, u_2, l_3 and u_3 are not in the right subtree of T_4, the 2 interior boundary PEs of paths P_0 and P_1 are distinct from the 2 interior boundary PEs of P_2 and P_3. Thus, a total of 4 interior PEs are required to be assigned to 3 nodes h_4, h_5, or h_6. This is not possible in a balanced interior utilization. Hence, l_2, u_2, l_3 and u_3 have to be in the right subtree of T_4.

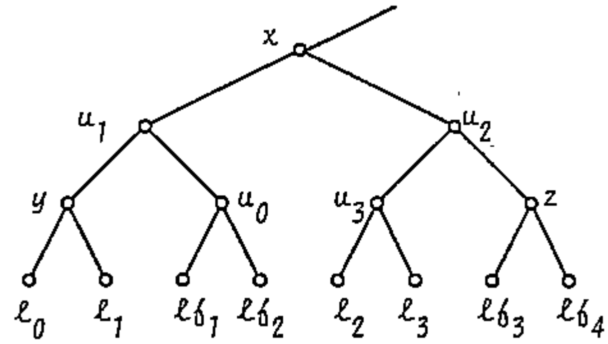
We next show that when the PEs l_i and u_i come from a common subtree of height 4 in T and when the paths P_i have a type assignment from set B, $0 \leq i \leq 3$, one leaf PE is required to be assigned to h_8. We show the positions of l_i and u_i for the type assignments from set B in Figures $8(a)-(e)$. To be consistent with the labelings, we show in Figure $8(d)$ the symmetric type assignment $(21)''(21)'$ instead of $(21)'(21)''$. Note that there is some freedom in how l_i and u_i, $0 \leq i \leq 3$, are chosen but cases not shown are all identical because of symmetry in binary trees. Paths P_0 and P_1 together have x and y as interior boundary PEs. Paths P_2 and P_3 together have x and z as interior boundary PEs. Since x is the common interior boundary PE, x is required to be assigned to h_6. This implies that the interior PE y (resp. z) has to be assigned to h_4 (resp. h_5). It is now easy to see that a total of 4 leaf PEs, labeled as lf_1, lf_2, lf_3, and lf_4 in Figure 8, have to be assigned to 4 nodes h_4, h_5, h_6, and h_8. Thus, h_8 is required to be assigned one leaf PE. This completes the description of set B.

Having a leaf requirement on h_8 implies the following. Let H_4 be a subtree of height 4 in which no node has 2 leaf PEs assigned to it. If the 4 leaves in the left subtree of H_4 have type assignment Q, $Q \in B$, then the four leaves in the right subtree of H_4 can not have a type assignment in B since that would require 2 leaf PEs to be assigned to the root of H_4 which is not possible in a balanced leaf utilization.

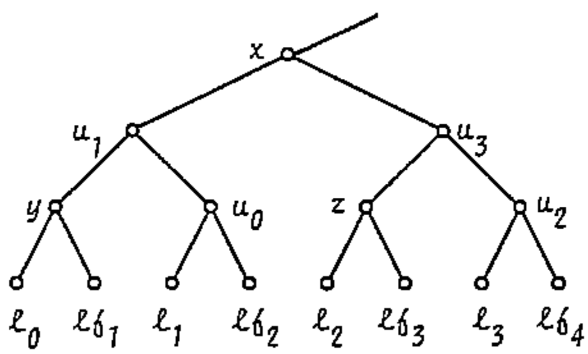
We now consider type assignments from set C. Recall that set C contains the following 9 type assignments: $11(b2)''$, $11(21)'$, $11(21)''$, $b1(b2)''$, $b1(21)'$, $b1(21)''$, $bb(b2)''$, $bb(21)'$, and $bb(21)''$. We show the positions of the PEs in T for these type assignments in Figures $9(a)-(f)$. Once again note that there is some freedom in how PEs l_i and u_i, $0 \leq i \leq 3$, are chosen but cases not shown are all identical because of symmetry in binary trees. Observe



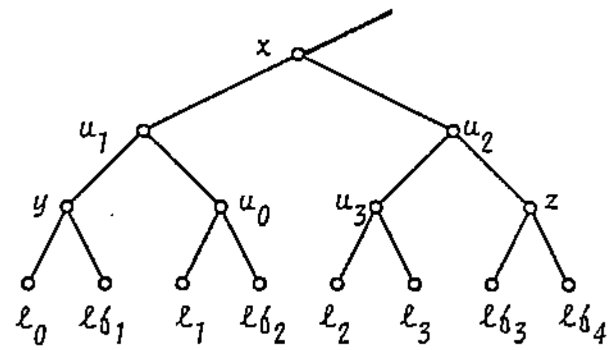
(a) Type assignment $(b2)'(b2)''$.
[for type assignment $(b2)'(21)''$ switch positions of u_2 and u_3.]



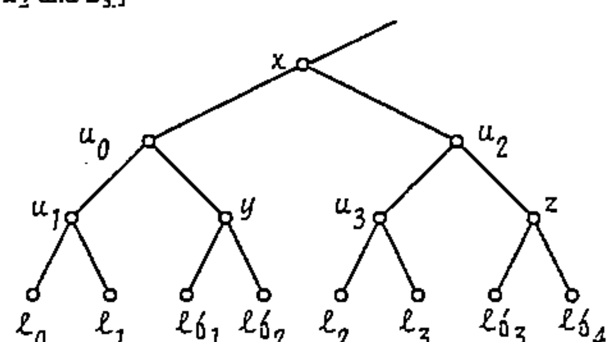
(b) Type assignment $(b2)'(21)'$.



(c) Type assignment $(b2)''(b2)''$.
[for type assignment $(b2)''(21)''$ swich positions of u_2 and u_3, and for type assignment $(21)''(21)''$ switch positions of u_0 and u_1 and of u_2 and u_3.]



(d) Type assignment $(b2)''(21)'$.
[for type assignment $(21)''(21)'$ switch positions of u_0 and u_1.]



(e) Type assignment $(21)'(21)'$.

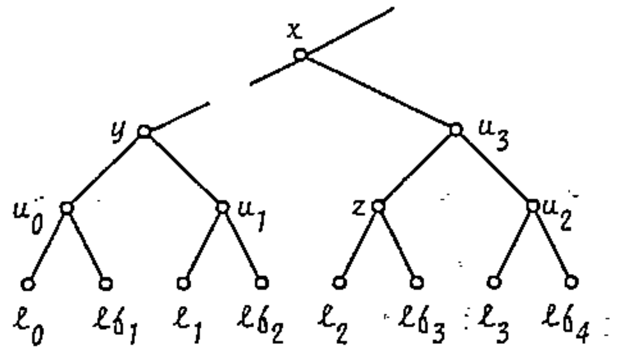
Figure 8: Positions of PEs in T for the type assignments in set B.

that for a type assignment in C the PEs l_i and u_i, $0 \leq i \leq 3$, may or may not come from a common subtree of height 4 in T. We indicate this by showing disjoint subtrees in Figure 9. We now show that any type assignment from set C requires a leaf PE to be assigned either to h_8 or to one of the 3 nodes from set $\{h_{10}, h_8, h_7\}$. Since P_0 and P_1 have a type assignment which is either 11, $b1$, or bb, paths P_0 and P_1 together have 1 interior boundary PE. Let y be this PE. Furthermore, P_0 and P_1 have 2 leaf boundary PEs which we refer to as lf_1 and lf_2. Paths P_2 and P_3 have a type assignment which is either $(b2)''$, $(21)'$, or $(21)''$ and they together have 2 interior boundary PEs. Let x and z be these PEs.

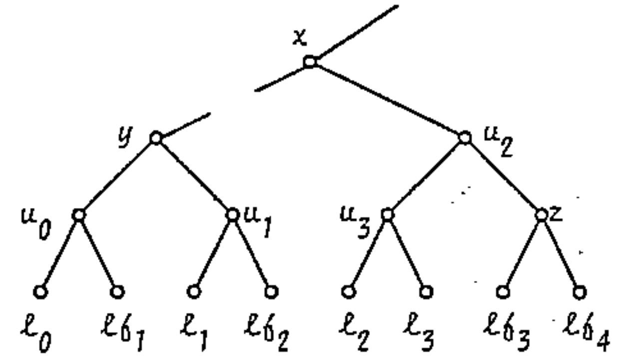
The leaf PEs lf_1 and lf_2 are required to be assigned to h_6 or h_4. Without loss of generality let lf_1 be assigned to h_6 and lf_2 be assigned to h_4 as shown in Figure 10. The interior PEs x and z have to be assigned to h_6 and h_5 and thus node h_6 is required to have an interior PE assigned to it. The interior PE y has to be assigned to either h_6 or h_4, but since h_6 already has an interior PE (either x or z) assigned to it, y has to be assigned to h_4. There are two possibilities for x and z. In the first possibility x is assigned to h_6 and z is assigned to h_5. In the second possibility x is assigned to h_5 and z is assigned to h_6. Both situations are shown in Figure 10, where the assignments of the second possibility are shown in brackets. We thus divide the type assignments from set C into two sets C' and C''. Set C' consists of all type assignments from set C in which x is assigned to h_6. Set C'' consists of all type assignments from set C in which x is assigned to h_5. We now consider sets C' and C'' in more detail and show that a type assignment in C' requires a leaf PE to be assigned to h_8 and that a type assignment in C'' requires a leaf PE to be assigned to one of the nodes from set $\{h_{10}, h_8, h_7\}$.

Set C' : In Figures 9(a), (c), and (e) paths P_2 and P_3 together have 1 leaf boundary PE lf_4 which has to be assigned to either h_6 or h_5. Since h_6 already has leaf PE lf_1 assigned to it, leaf PE lf_4 is assigned to h_5. In set C' interior PE z is assigned to h_5. Since z has leaf PE lf_3 as its child and since every node at a distance of at most 2 from h_5, except h_8, already has a leaf PE assigned to it, lf_3 has to be assigned to h_8. In Figures 9(b), (d), and (f) one of lf_3 or lf_4, say lf_4, has to be assigned to h_5, and the other leaf, say lf_3, has to be assigned to h_8.

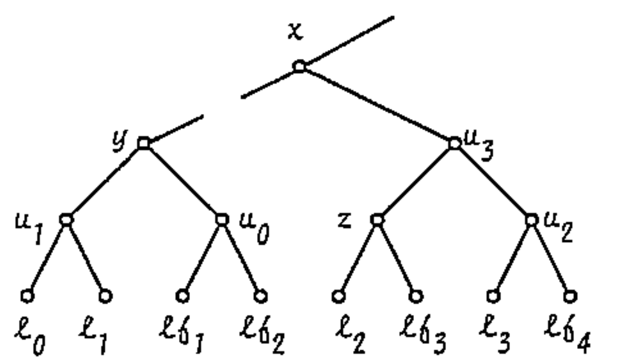
Set C'' : Since h_6 already has a leaf PE, namely lf_1, assigned to it, leaf boundary PE



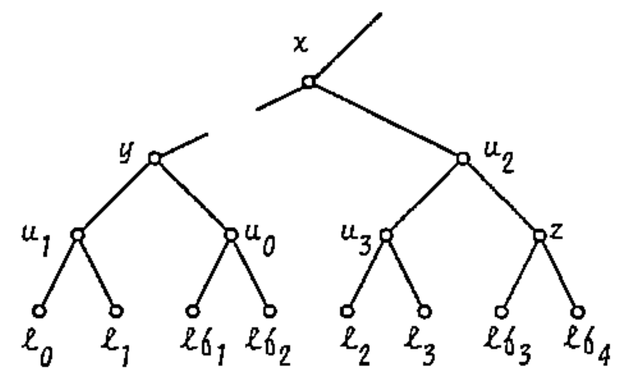
(a) Type assignment $11(b2)''$.
[for type assignment $11(21)''$ switch positions of u_2 and u_3.]



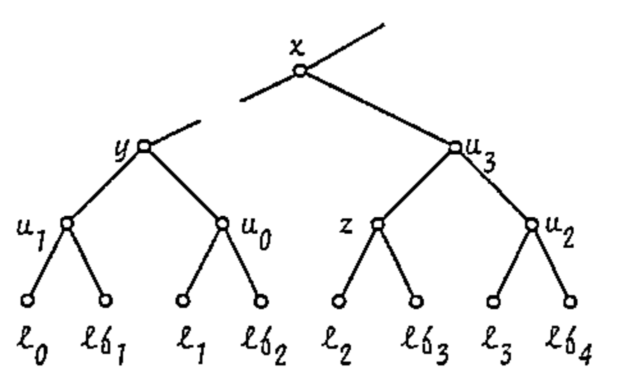
(b) Type assignment $11(21)'$.



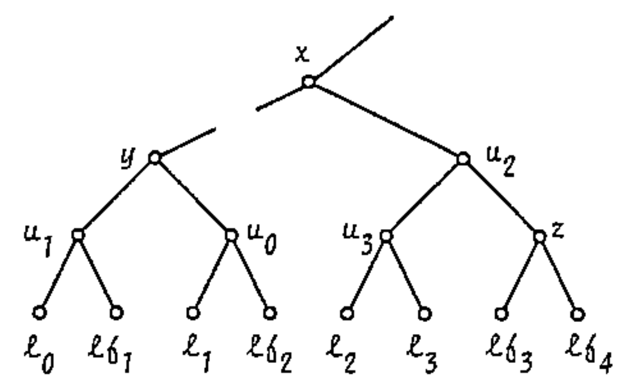
(c) Type assignment $b1(b2)''$.
[for type assignment $b1(21)''$ switch positions of u_2 and u_3.]



(d) Type assignment $b1(21)'$.



(e) Type assignment $bb(b2)''$.
[for type assignment $bb(21)''$ switch positions of u_2 and u_3.]



(f) Type assignment $bb(21)'$.

Figure 9: Positions of PEs in T for the type assignments in set C.

lf_4 in Figures 9(a), (c), and (e) has to be assigned to h_5. In Figures 9(b), (d), and (f) the 2 leaf PEs lf_3 and lf_4 have to be assigned to 2 nodes from the set $\{h_{10}, h_8, h_7, h_5\}$. We show that either lf_3 or lf_4 has to be assigned to h_5. Assume the contrary, i.e., h_5 is assigned a leaf PE lf that is neither lf_3 nor lf_4. But now the parent of lf, which is distinct from u_2, u_3, x, y, z, and the parent of x, has to be assigned to a node at a distance of at most 2 from h_5. Since every one of these nodes already has an interior PE assigned to it, h_5 has to be assigned either lf_3 or lf_4. Say that h_5 is assigned lf_4. It now follows easily that lf_3 has to be assigned to either h_{10}, h_8, or h_7. This completes the description of set C''.

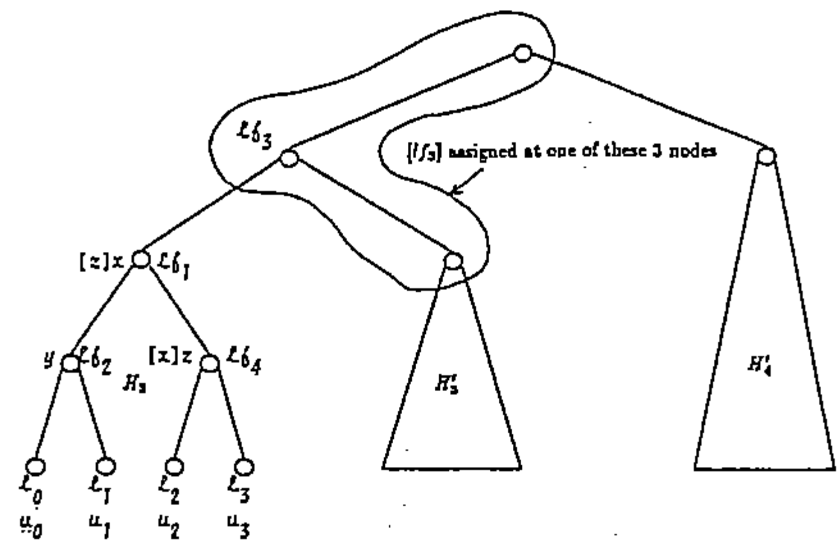


Figure 10: Subtree H_5 showing the assignments of PEs to nodes as in set C' [C''].

In order to complete proof, we now consider the assignments of 8 consecutive leaf nodes in a common subtree of height 4 in H and then consider the assignments of 16 consecutive leaf nodes in a common subtree of height 5. Let H_4 be the left subtree of H_5. Recall that H_3 (resp. H'_3) is the left (resp. right) subtree of H_4 as shown in Figure 7. Furthermore, we assume that no node in H_4 has 2 leaf PEs assigned to it. Let Q be the type assignment of the 4 leaf nodes in H_3. We know that Q has to be in set B, C', or C''. Let R be the type assignment of the 4 leaf nodes in H'_3. Then, R has to be in the set B, C', or C''. We already showed that when Q is in the set B, then R can not be in the set B. This leaves R to be in the set C' or C'', and Q to be in the set B, C', or C'' (not considering the symmetric type assignments).

First consider the case when $R \in C'$ and $Q \in B$ or C'. Type assignment R requires

1 leaf PE, say lf, to be assigned to h_8. Type assignment Q also requires 1 leaf PE, say lf', to be assigned to h_8. Since lf' can not coincide with lf, 2 leaf PEs are required to be assigned to h_8 which is not possible in balanced leaf utilization.

Next consider the remaining 3 combinations, namely $R \in C''$ and $Q \in B$, C', or C''. We show that a leaf PE is required to be assigned to h_{10}. Since $R \in C''$, one of the nodes from set $\{h_{10}, h_8, h_6\}$ is required to be assigned a leaf PE. Let lf be this PE. But since h_6 already has a leaf PE assigned from type assignment Q, lf has to be assigned to either h_{10} or h_8. When $Q \in B$ or $Q \in C'$, type assignment Q requires a leaf PE, which is distinct from lf, to be assigned to h_8 and hence leaf PE lf has to be assigned to h_{10}. When $Q \in C''$, type assignment Q requires a leaf PE, say lf' that is distinct from lf, to be assigned to one of the nodes from set $\{h_{10}, h_8, h_7\}$. But since h_7, the root of H_3', already has a leaf PE assigned from type assignment R, lf' has to be assigned to either h_{10} or h_8. Thus, we have 2 leaf PEs lf and lf' that have to be assigned to h_{10} and h_8. Without loss of generality let lf be assigned to h_{10} and thus lf' is assigned to h_8.

In order to get a contradiction we consider the assignments to the 8 leaf nodes in H_4', the right subtree of H_5. Let Q' (resp. R') be the type assignment of leaf nodes in the left (resp. right) subtree of H_4'. From our previous discussion it follows that the only possible assignment for the leaves of H_4' is $R' \in C''$ and $Q' \in B$, C', or C''. In each of these 3 cases a leaf PE , say lf'', is required to be assigned to h_{10}. Since h_{10} already has a leaf PE, namely lf, assigned to it and since lf'' is distinct from lf, we have a requirement of 2 leaf PEs on h_{10}. This is not possible since we assumed that only 1 leaf PE is assigned to h_{10}. Theorem 7 now follows. ∎

3. Conclusions

We have shown that any embedding of a $2m+1$-PE complete binary tree T into an m-node complete binary tree H with a balanced leaf and interior utilization requires a dilation of at least 3. The best known upper bound on the dilation for such an embedding is $2\log\log m + 1$ [GH] and we conjecture that this is optimal within a constant factor. Note that if we require every node of H to be assigned 2 arbitrary PEs of T (and one node to be assigned 3 PEs), then it is easy to achieve a dilation of 1. We consider it unlikely that the techniques used in this paper generalize so that the gap between 3 and $2\log\log m + 1$ can be closed. The main reasons appear to be the inability to easily classify the paths $P(l, u)$ and the resulting exponential growth in the number of cases to be considered.

4. References

[AR] R. Aleliunas, A. Rosenberg, "On Embedding Rectangular Grids into Square Grids," *IEEE Trans. on Comp.*, Vol. V-31, pp. 907-913, 1982.

[B] S. A. Browning, "The Tree Machine: A Highly Concurrent Computing Environment," *Ph.D. Thesis*, Computer Science Dept., Caltech., 1980.

[BCJLR] S. Bhatt, F. Chung, H. Jai-Wei, F. T. Leighton, and A. L. Rosenberg, "Optimal Simulations by Butterfly Networks," *Proc. of* 23^{rd} *STOC*, pp. 192-204, 1988.

[BCLR] S. Bhatt, F. Chung, F. T. Leighton, A. Rosenberg, "Optimal Simulations of Tree Machines," *Proc. of* 27^{th} *FOCS*, pp. 274-282, 1986.

[BSM] S. Bettayeb, I. H. Sudborough, and Z. miller, "Embedding Grids into Hypercubes," *Proc. of* 3^{rd} *Aegean Workshop on Computing*, 1988.

[CGC] F. R. K. Chung, R. L. Graham, and D. Coppersmith, "On Graphs Containing All Small Trees," *The Theory and Applications of Graphs*, (G. Chartand, Ed.), pp. 255-264, Wiley, 1981.

[FF] J. P. Fishburn and R. A. Finkel, "Quotient Networks," *IEEE Transactions on Computers*, Vol. C-31, No. 4, April 1982, pp. 288-295.

[GH] A. K. Gupta and S. E. Hambrusch, "Embedding Large Tree Machines into Small Ones," *Proceedings of the* 5^{th} *MIT Conference on Advanced Research in VLSI*, pp. 179-199, March 1988.

[JMR] H. Jai-Wei, K. Mehlhorn, A. Rosenberg, "Cost Trade-offs in Graph Embeddings, with Applications," *JACM*, pp. 709-728, 1983.

[KA] S. R. Kosaraju, M. Atallah, "Optimal Simulations between Mesh Connected Array of Processors," *Journal of the Association of Computing Machinery*, Vol. 35, No. 3, pp. 635-650, July 1988.

[R] A. L. Rosenberg, "Preserving Proximity in Arrays," *SIAM J. on Computing*, pp. 443-460, 1979.