SIAM J. COMPUT.
Vol. 15, No. 1, February 1986

002

AN AMORTIZED ANALYSIS OF INSERTIONS INTO AVL-TREES*

KURT MEHLHORN† AND ATHANASIOS TSAKALIDIS†

Abstract. We analyse the amortized behavior of AVL-trees under sequences of insertions. We show that the total rebalancing cost (=balance changes) for a sequence of n arbitrary insertions is at most $2.618n$. For random insertions the bound is improved to $2.26n$. We also show that the probability that t or more balance changes are required decreases exponentially with t.

Key words. balance search trees, insertions, AVL-trees, balance changes, amortized number, expected number

1. Introduction. As for many balanced tree schemes an insertion into an AVL-tree consists of a search down the tree followed by a rebalancing phase which works its way back to the root. Rebalancing is usually restricted to a terminal segment of the search paths. Experiments (cf. [5]) suggest that the expected length of this terminal segment is less than two, however, there is no theoretical evidence to support that claim. A first attempt of an analysis was made by C. C. Foster (see [3], cf. also Knuth [6, p. 462]). Although his analysis predicts the expected length of the terminal segment fairly well (he predicts expected length 1.85), it is not precise in a mathematical sense.

In this paper we give a *rigorous* analysis. More precisely, we prove in § 3, that the amortized length of the terminal segment is at most 2.618. That is, if we consider an arbitrary (not random) sequence of n insertions into an initially empty AVL-tree then the total length (summed over the n insertions) of the terminal segments is at most $2.618n$. We also prove that this bound is sharp by exhibiting a sequence of n insertions where the total number of balance changes is essentially $2.618n$. Our main insight is to concentrate on amortized behavior rather than expected behavior. This led us to stronger results (our results hold for arbitrary not just random sequences of insertions) and suggested to use combinatorial (not probabilistic) methods of analysis.

For sequences of random insertions we can slightly improve the bound in § 4. The expected length of the terminal segment is at least 1.47 and at most 2.26. Finally in § 5 we prove a result on the distribution of the length of the terminal path. We show that the probability that the length exceeds t decreases exponentially in t. Section 2 contains some definitions.

We close the introduction with a brief discussion of related work. Brown [2] and Mehlhorn [7], [8] studied the expected number of balanced nodes in random AVL-trees. We use their results in § 4. The amortized cost of insertions and deletions into (a, b)-trees was studied by Huddleston and Mehlhorn [4]. They prove that the amortized number of node splittings and fusings is $0(1)$ provided that $b \geqq 2a$. Finally, we want to mention that Tsakalidis [9] proves a result which is analogous to the one presented here for sequences of deletions. He shows that the total number of balance and structural changes in a sequence of deletions applied to an AVL-tree with n leaves is $1.618n$.

2. Definitions. AVL-trees were introduced by Adel'son-Vel'skii and Landis [1] in 1962. AVL-trees are binary trees in which nodes either have two sons or no sons. The

* Received by the editors August 31, 1983, and in revised form September 1, 1984.

† Fachbereich 10, Angewandte Mathematik und Informatik, Universität des Saarlandes, D-6600 Saarbrücken, West Germany.

latter nodes are called leaves. A binary search tree is AVL if the heights of the subtrees at each node differ by at most one, where the height Height (v) of node v is equal to the length of the longest path from v to a leaf.

Let $L(v)$ $[R(v)]$ be the left [right] subtree of the tree with root v. For every node v we define its height balance $hb(v)$ by

$$hb(v) = \text{Height}\,(R(v)) - \text{Height}\,(L(v)).$$

Hence the height balance can be $+1$, 0, or -1. We call a node balanced (unbalanced) if its height balance is 0 (± 1).

For every insertion we define the *critical node* as the last unbalanced node on the search path. We give the last definition more formally:

Let $v_0, v_1, \cdots, v_k$ be a path from the root v_0 to a leaf v_k of an AVL-tree. Let i be minimal such that $hb(v_i) = hb(v_{i+1}) = \cdots = hb(v_k) = 0$. Then node v_{i-1} is called the *critical node* of the path (if $i \geqq 1$) and $v_i, \cdots, v_k$ is called the *critical path.* The *length* of the critical path is $k-i$ (and this is equal to the height of node v_i). For $i=0$ no critical node is defined since this insertion causes a height increase of the tree.

We use the insertion algorithm described in Knuth [6, p. 455]. It can be summarized as follows:

At first the leaf at which the new element is to be inserted is located and this leaf is replaced by a tree with two leaves. Next the height balance of nodes on the critical path are changed from 0 to ± 1 (this corresponds to step A_6 in Knuth [6]). Finally rebalancing is completed by absorption, single rotation or double rotation at the critical node or a height increase of the tree if no critical node exists.

Let T_0 be the empty AVL-tree, i.e. the AVL-tree with 0 leaves. We consider sequences of n insertions into T_0. Let T_i, $0 \leqq i \leqq n$, be the AVL-tree obtained after the ith insertion (and completed rebalancing). We are interested in the following quantities:

x_1, the total number of balance changes $0 \to \pm 1$ in step A_6,

x_2, the total number of absorptions,

x_3, the total number of single rotations,

x_4, the total number of double rotations,

x_5, the total number of height increases.

We will also refer to the 5 operations (balance change, absorption, single rotation, double rotation, height increase) as $Op_1, Op_2, \cdots, Op_5$. Since every insertion terminates in an absorption, single or double rotation or height increase we clearly have

$$x_2 + x_3 + x_4 + x_5 = n.$$

Quantity x_1 is harder to estimate. Let l_i, $1 \leqq i \leqq n$, be the length of the critical path for the ith insertion. Then

$$x_1 = \sum_{i=1}^{n} l_i.$$

We will estimate x_1 for arbitrary sequences of insertions in § 3 and for random sequences in § 4. Finally let Val (T_i) be the total number of unbalanced nodes in AVL-tree T_i.

3. The total number of balance changes in arbitrary sequences of insertions. Our main tool for getting bounds on x_1, the number of balance changes $0 \to \pm 1$, is Lemma 1 below. In this lemma we relate quantities x_1, x_3, x_4, x_5, the number of insertions n and the value Val (T_n) of the final tree.

LEMMA 1. *Consider an arbitrary sequence of n insertions into an initially empty tree. Then*

$$x_1 = \text{Val}\,(T_n) + n + x_3 + x_4 - x_5.$$

Proof. We exactly estimate the rate of the increase of Val (T_i) during the transition from T_{i-1} to T_i with respect to the alternative operations.

We give the scheme how the node's balance on the critical path will be changed. In our diagrams we will always show an insertion into the left subtree of the critical node.

Case 1. Absorption.

In the figures ± 1 denotes a node with balance $+1$ or -1 and $\square$ denotes a leaf. The figure shows an insertion into the left subtree of the critical node, the other case being symmetric.

With respect to Fig. 1 the following holds:

$$\text{Val}(T_i) = \text{Val}(T_{i-1}) + (l_i - 1). \tag{1}$$

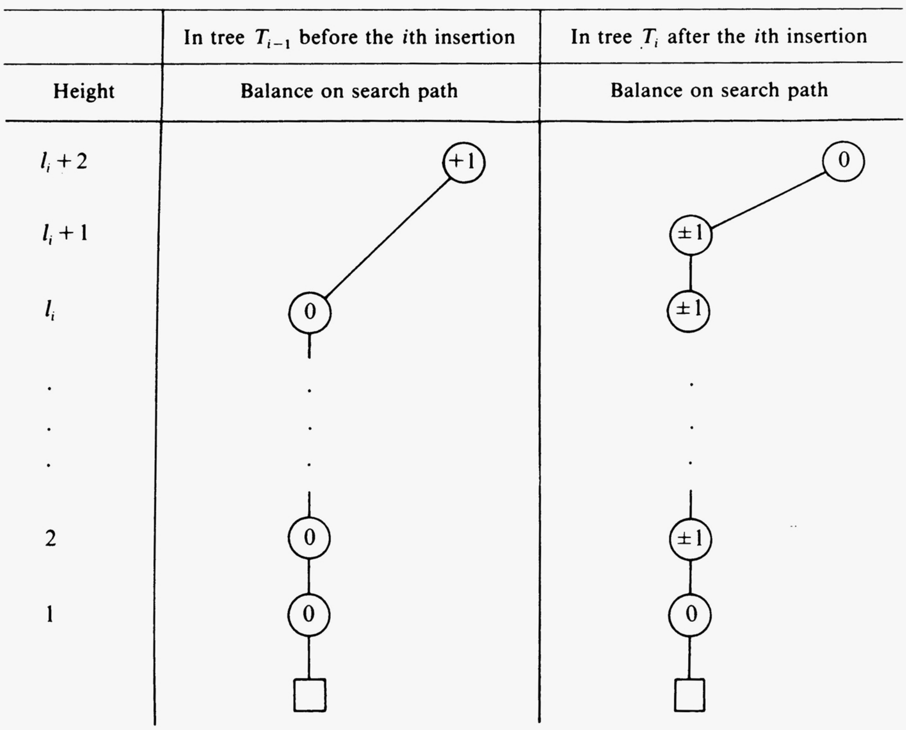


FIG. 1

Case 2. Reconstruction of the tree.

Case 2.1. Single rotation.

Fig. 2 illustrates this case and the following holds:

$$\text{Val}(T_i) = \text{Val}(T_{i-1}) + (l_i - 2). \tag{2}$$

Case 2.2. Double rotation is analogous to Case 2.1. Among the 3 top nodes of the reconstructed subtree there are exactly 2 nodes with Balance 0.

Case 3. Height increase (l_i is the height of the root), i.e. no critical node exists.

Fig. 3 illustrates this case and the following holds:

$$\text{Val}(T_i) = \text{Val}(T_{i-1}) + l_i. \tag{3}$$

We apply the recursive equations (1), (2) and (3) n times, and we get with respect to

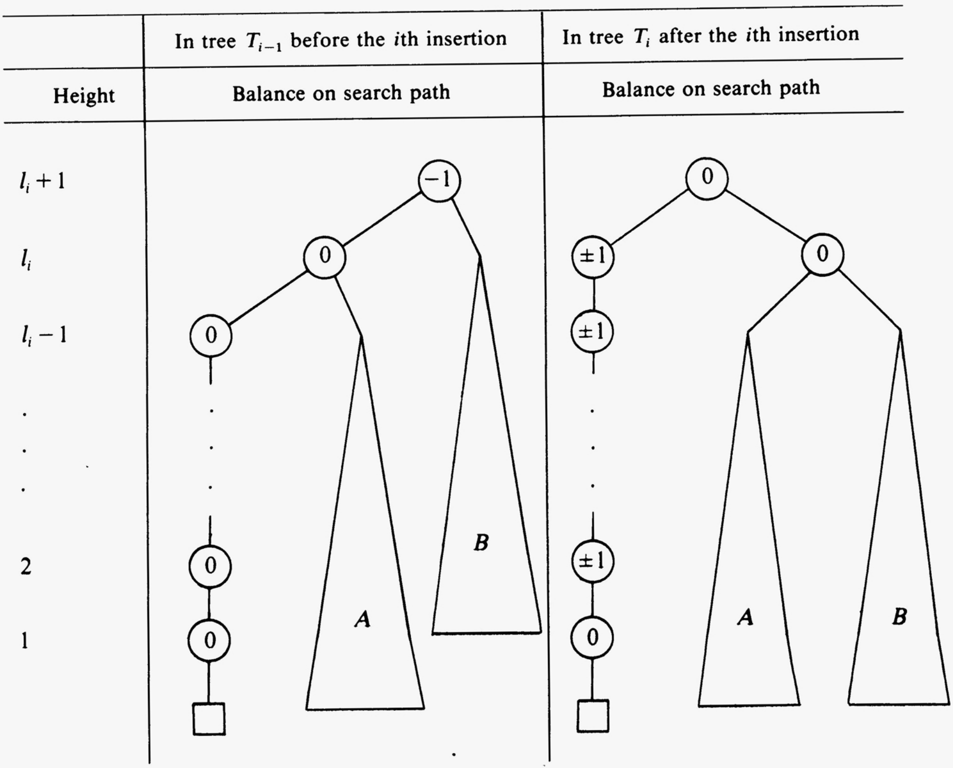

In tree T_{i-1} before the ith insertion
In tree T_i after the ith insertion
Height
Balance on search path
Balance on search path
l_i+1
l_i
l_i-1
2
1
−1
0
±1
A
B

FIG. 2

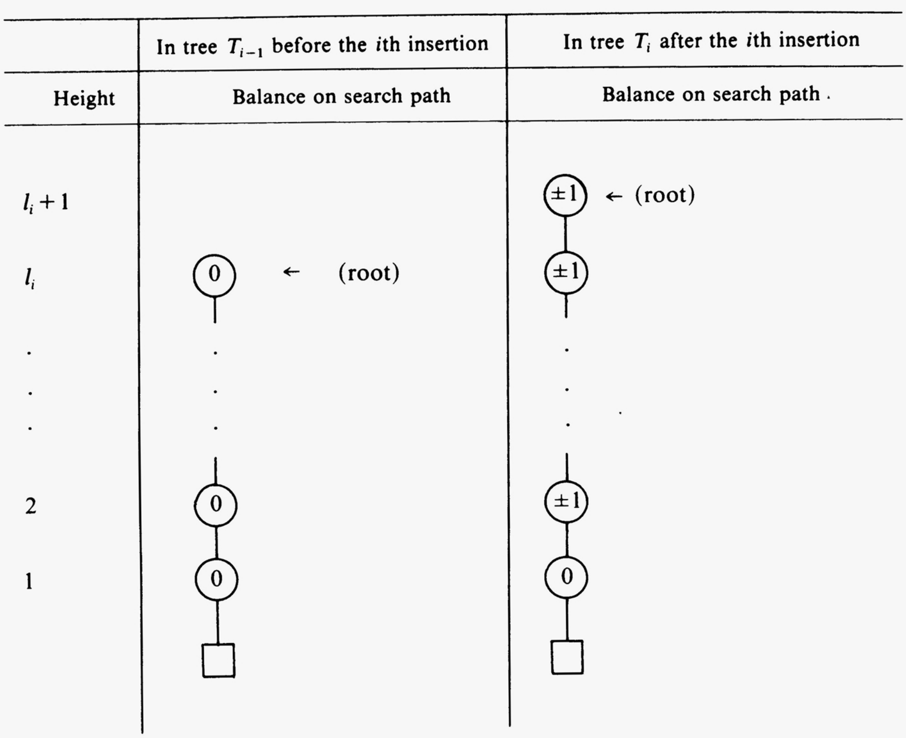

In tree T_{i-1} before the ith insertion
In tree T_i after the ith insertion
Height
Balance on search path
Balance on search path
l_i+1
l_i
2
1
0
±1
← (root)
← (root)

FIG. 3

the definitions of x_i $(1 \leqq i \leqq 5)$:

$$\mathrm{Val}\,(T_n) = \mathrm{Val}\,(T_0) + \sum_{i=1}^{n} l_i - x_2 - 2(x_3 + x_4).$$

But $\sum_{i=1}^{n} l_i = x_1$ and $\mathrm{Val}\,(T_0) = 0$ and thus

$$x_1 = \mathrm{Val}\,(T_n) + x_2 + 2(x_3 + x_4).$$

Using $x_2 + x_3 + x_4 + x_5 = n$ we obtain

$$x_1 = \mathrm{Val}\,(T_n) + n + (x_3 + x_4 - x_5). \qquad \square$$

Since $\mathrm{Val}\,(T_n) \leqq n$ and $x_3 + x_4 \leqq n$ we infer $x_1 \leqq 3n$ from Lemma 1, i.e. the amortized length of the critical path is at most 3. In Lemma 2 below we recall a better upper bound on $\mathrm{Val}\,(T_n)$ and so improve the upper bound on x_1.

LEMMA 2 (Knuth). *Let T_n be an* AVL-*tree with n leaves. Then* $\mathrm{Val}\,(T_n) \leqq (\phi - 1)(n - 1)$ *where* $\phi = (1 + \sqrt{5})/2 \simeq 1.618$.

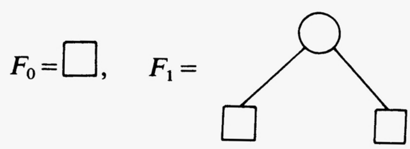
Proof. cf. [6, *exercise* 6.2.3.3]. $\square$

THEOREM 1. *The total number x_1 of balance changes in step* A6 *of the insertion algorithm in a sequence of n arbitrary insertions into an initially empty tree satisfies*

$$x_1 \leqq 2.618n.$$

Proof. Immediate from Lemma 1 and $x_3 + x_4 \leqq n$, $x_5 \geqq 0$ and $\mathrm{Val}\,(T_n) \leqq 0.618n$ (Lemma 2). $\square$

How good is the bound given in Theorem 1? Note first that the bound given in Lemma 2 is sharp: Fibonacci trees have exactly that number of unbalanced nodes. Fibonacci trees $F_0, F_1, F_2, \cdots$ are defined as follows:



and F_{h+2} consists of a root and a copy of F_h and F_{h+1} each as left and right subtree respectively. This specifies the Fibonacci trees up to the left-right symmetry. Also F_h has exactly $\mathrm{Fib}\,(h+1)$ leaves where $\mathrm{Fib}\,(0) = \mathrm{Fib}\,(1) = 1$ and $\mathrm{Fib}\,(h+2) = \mathrm{Fib}\,(h+1) + \mathrm{Fib}\,(h)$ for $h \geqq 0$.

Besides the upper bound on $\mathrm{Val}\,(T_n)$ given by Lemma 2 we use two more inequalities in the proof of Theorem 1: $x_3 + x_4 \leqq n$ and $x_5 \geqq 0$. We will next show that these bounds cannot be improved upon in general.

LEMMA 3. *There is a sequence of* $\mathrm{Fib}\,(h)$ *insertions into F_h such that*:

1) *Every insertion* (*but the last*) *is terminated by a rotation or double rotation.*
2) *The last insertion leads to a height increase of the entire tree.*
3) *The final tree is F_{h+1}.*

Proof. The claim is obviously true for $h = 0, 1$ and 2, namely we can go from F_0 to F_1 (F_1 to F_2) with one insertion which leads to an increase in height and from F_2 to F_3 with two insertions one of which leads to a rotation and one of which leads to a height increase as shown in Fig. 4.

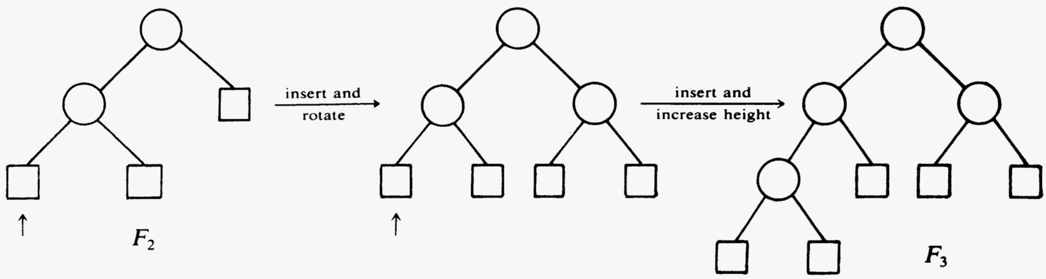
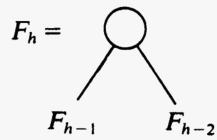


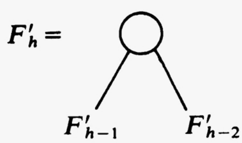
FIG. 4

For $h \geqq 3$ we proceed by induction. So consider

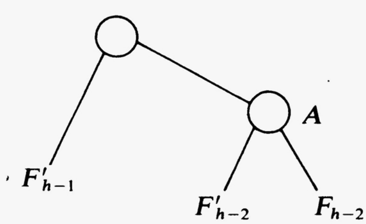


We will first use Fib $(h-1)$ insertions in order to change the left (w.l.o.g.) subtree F_{h-1} into an F_h (which we denote F'_h to distinguish it from our starting tree). Note that by induction hypothesis all but the last insertion leads to a rotation/double rotation *inside* the left subtree. The last insertion will turn the left subtree into an F'_h and increase the height of the left subtree to h, thus moving the root out of balance. Next we will rotate or double rotate at the root.

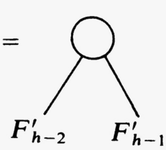
Case 1. Rotation, i.e.



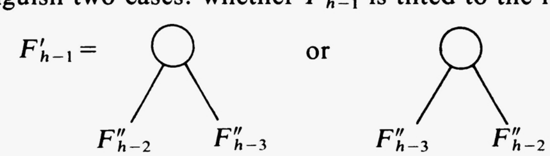
Rotation yields



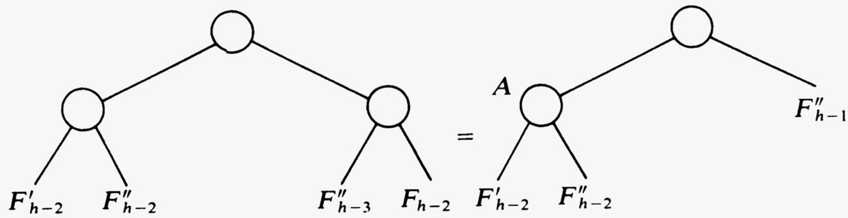
Case 2. Double rotation, i.e.



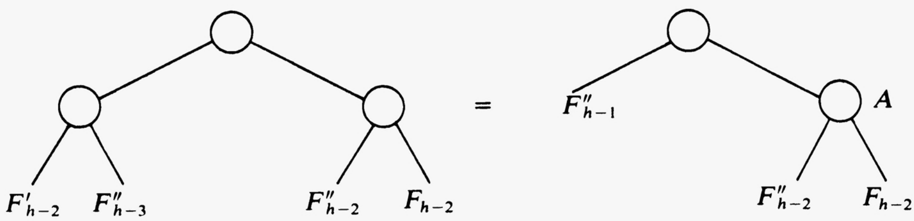
We have to distinguish two cases: whether F'_{h-1} is tilted to the left or right, i.e.



In the first case double rotation yields



In the second case double rotation yields



In either case we obtain up to left right symmetry the same tree: a balanced root whose subtrees are a F_{h-1} and a balanced node A with two copies of F_{h-2}.

Next we perform Fib $(h-2)$ insertions into F_{h-2}. All but the last insertion will lead to rotations/double rotations within that tree without increasing the height. The last insertion creates a F_{h-1} and moves node A and the root out of balance thus increasing the total height of the tree. Altogether we created an F_{h+1} out of F_h by

$$\mathrm{Fib}\,(h-1)+\mathrm{Fib}\,(h-2)=\mathrm{Fib}\,(h)$$

insertions; all insertions (but the last) lead to rotations or double rotations and the last insertion increased the height. □

THEOREM 2. *There are infinitely many n and sequences of n insertions such that the total number of balance changes in step* A6 *of the insertion algorithm satisfies*

$$x_1 \geqq 2.618n - O(\log n).$$

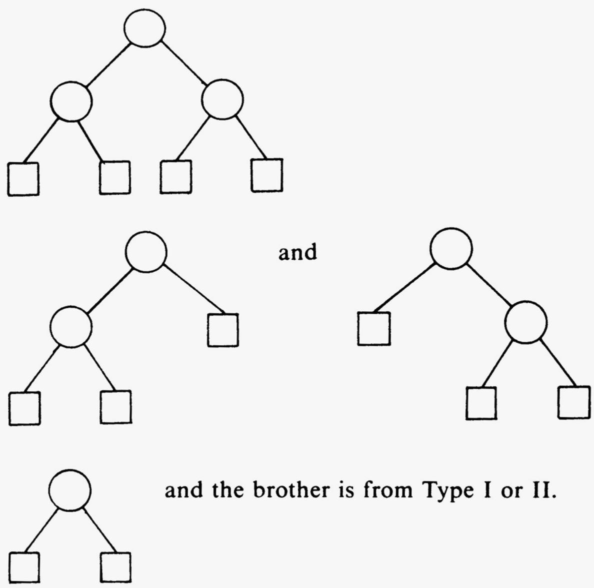
Proof. Let $n = \mathrm{Fib}\,(h+1)$ for some h. Start with the empty tree and build $F_0, F_1, \cdots, F_h$ as described in Lemma 3. Then $x_2 = 0$ since absorption never occurs and hence $x_3 + x_4 + x_5 = n$. Also $x_5 = h = O(\log n)$ and $\mathrm{Val}\,(T_n) = (\phi - 1)(n-1)$. □

Theorems 1 and 2 together give complete information about the amortized cost of step A_6 of the insertion algorithm in the worst case. The amortized length of the critical path is 2.618 in the worst case and thus very small. We want to stress again that this bound holds for *arbitrary* sequences of insertions. Experimental data (Foster [3], Knuth [6], Karlton et al. [5]) is only available for random sequences of insertions. It suggests that the expected length of the critical path under random insertions is about 1.8 and hence only slightly less than amortized length.

4. The total number of balance changes under *n* random insertions.

Randomness assumption. Consider a tree T with $n-1$ nodes in it and hence with n leaves. These $n-1$ keys divide all possible key values into n intervals. The insertion of a new key k into T is said to be a random insertion if k has equal probabilities for being in any one of the n intervals defined above. In other words, each leaf has equal probability of being split into two.

Mehlhorn [7] has analysed the fringe of an AVL-tree under n random insertions refining work of Brown [2]. The fringe consists of the 3 types of subtrees shown in Fig. 5.



Type I.

Type II. and

Type III. and the brother is from Type I or II.

FIG. 5

We quote two results from [7].

THEOREM 3 [7]. *Let $a_i(n)$, for $1 \leqq i \leqq 3$, be the respective number of the above subtree of type i in a random* AVL-*tree T with n leaves. Then*

$$\begin{pmatrix} a_1(n) \\ a_2(n) \\ a_3(n) \end{pmatrix} = \frac{1}{35-28p(n)} \begin{pmatrix} 3 \\ 5-4p(n) \\ 4-8p(n) \end{pmatrix} n + \begin{pmatrix} o(n) \\ o(n) \\ o(n) \end{pmatrix}$$

with $0 \leqq p(n) \leqq \frac{4}{11}$ and $p(n)$ is the probability that the brother of a type I *subtree is of type* III *in a random* AVL-*tree with n leaves.*

THEOREM 4 [7]. *Let $\overline{\mathrm{Val}}\,(n)$ be the expected number of unbalanced nodes in a random* AVL-*tree with n leaves. Then*

$$(\tfrac{17}{91})n - o(n) \leqq \overline{\mathrm{Val}}\,(n) \leqq (\tfrac{17}{35})n + o(n).$$

Proof. In [7] it is shown that for the average number $\bar{B}(n)$ of balanced nodes in a random AVL-tree with n leaves the following holds:

$$(\tfrac{18}{35})n + o(n) \leqq \bar{B}(n) \leqq (\tfrac{74}{91})n + o(n).$$

Since $\overline{\mathrm{Val}}\,(n) + \bar{B}(n) = n$ we get the bounds of $\overline{\mathrm{Val}}\,(n)$: $\square$

THEOREM 5. *The expected number $\bar{x}_1$ of balance changes in step* A6 *of the insertion algorithm in a sequence of n random insertions into an initially empty* AVL-*tree satisfies*

$$1.47n - o(n) \leqq \bar{x}_i \leqq 2.26n + o(n).$$

Proof. Let $p_j(i)$ denote the probability of the execution of operation Op_j (see § 2) during the $(i+1)$th insertion into a random AVL-tree with i leaves $(2 \leqq j \leqq 5)$. Then according to Theorem 3 we have

$$p_2(i) \geqq \text{Probability of the absorptions on the fringe}$$
$$= \frac{a_2(i)+2a_3(i)}{i} \geqq \min_{0\leqq p\leqq 4/11} \left\{\frac{5-4p+2(4-8p)}{35-28p}\right\} - o(1) = 0.230 - o(1).$$

Also

$$p_3(i)+p_4(i) \geqq \text{probability of single and double rotation on the fringe}$$
$$\geqq (2a_2(i)+4a_1(i)\cdot p(i))/i - o(1)$$
$$\geqq \min_{0\leqq p\leqq 4/11} \left\{\frac{2(5-4p)+4\cdot 3\cdot p)}{35-28p}\right\} - o(1) = 0.285 - o(1).$$

This can be seen as follows: a rotation will always happen when the insertion is in one of the two deep leaves of a Type II tree (probability $2\cdot a_2(i)/i$) and if insertion is in any leaf of a type I tree whose brother is a Type III tree (probability $4a_1(i)\cdot p(i)/i$).

Furthermore, $p_2(i)+p_3(i)+p_4(i)+p_5(i)=1$ and hence

$$0.285 - o(1) \leqq p_3(i)+p_4(i) \leqq 0.770 - p_5(i) + o(1),$$
$$= 0.770 + o(1),$$

since $p_5(i) = o(1)$.

For the $(i+1)th$ insertion we define the random variable y_i as follows $(0 \leqq i \leqq n-1)$

$$y_1 = \begin{cases} 1 & \text{if the } (i+1)\text{th random insertion into a random AVL-tree with} \\ & i \text{ leaves causes a single or double rotation,} \\ 0 & \text{otherwise.} \end{cases}$$

The expectation $E(y_i)$ of random variables y_i is equal to $p_3(i)+p_4(i)$ and hence

$$0.285 - o(1) \leqq E(y_i) \leqq 0.770 + o(1).$$

Let $\bar{x}_{3,4}$ denote the expected number of single or double rotations in a sequence of n random insertions, and let $\bar{x}_5$ be the expected number of height increases. Then

$$\bar{x}_{3,4} = \sum_{i=0}^{n-1} E(y_i)$$

and hence

$$0.285n - o(n) \leqq \bar{x}_{3,4} \leqq 0.770n + o(n).$$

Also $\log n \leqq \bar{x}_5 \leqq 1.44 \log n$ since an AVL-tree with n leaves has height between $\log n$ and $1.44 \log n$.

Substituting into $\bar{x}_1 = \overline{\text{Val}}\,(n) + n + \bar{x}_{3,4} - \bar{x}_5$ and using Theorem 4 we get

$$1.47n - o(n) \leqq \bar{x}_1 \leqq 2.26n + o(n). \qquad \square$$

As mentioned above, there is considerable experimental evidence for the belief that $\bar{x}_1$ should be about $1.78n$ a figure which is near the average of the lower and upper bound given in Theorem 3.

5. On the distribution of the balance changes on the levels of the tree. In this section we study the distribution of the balance changes $0 \to \pm 1$ to the different levels of an AVL-tree. We need to refine the definition of Val (T) in 3 with respect to the heights.

DEFINITION. $\mathrm{val}_t(T)$ is the number of nodes with balance ± 1 and height t in the AVL-tree T.

Example. In the example in Fig. 6

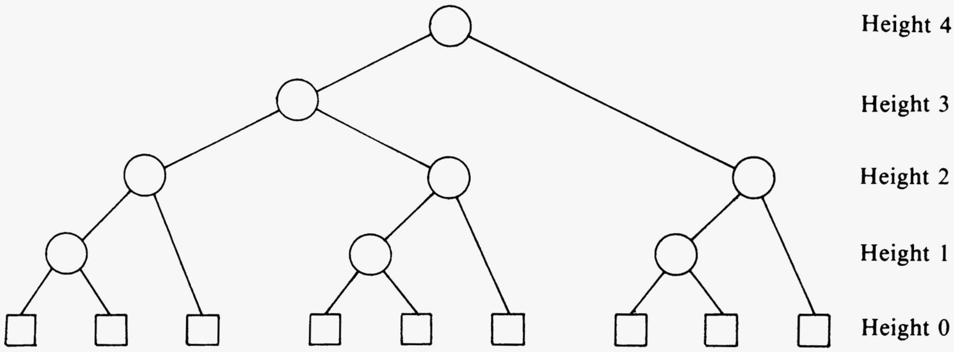


FIG. 6

we have $\mathrm{val}_1(T)=0$, $\mathrm{val}_2(T)=3$, $\mathrm{val}_3(T)=0$ and $\mathrm{val}_4(T)=1$. Note that nodes of height 1 are always balanced and hence $\mathrm{val}_1(T)=0$ for all trees T.

Next we need to refine the analysis of § 3. There we studied the effect of an insertion on $\mathrm{Val}(T)$; in this section we investigate this effect separately for each height t. As in § 3 we consider a sequence of n insertions into the initially empty tree T_0; let T_i be the tree after the ith insertion.

LEMMA 4. *Let h be the height of the critical node for the insertion into T_{i-1} if such a node exists and let h be the height of T_{i-1} otherwise. Then*

$$\text{1)}\qquad \mathrm{val}_t(T_i)=\mathrm{val}_t(T_{i-1})+\begin{cases} 0 & \text{if } t=1 \text{ or } t>h,\\ 1 & \text{if } 2\leqq t\leqq h-1,\\ -1 & \text{if } t=h,\end{cases}$$

if a critical node exists;

$$\text{2)}\qquad \mathrm{val}_t(T_i)=\mathrm{val}_t(T_{i-1})+\begin{cases} 0 & \text{if } t=1,\\ 1 & \text{if } 2\leqq t\leqq h+1\end{cases}$$

if no critical node exists.

Proof. By inspection of the figures in the proof of Lemma 1. □

It is interesting to observe that absorption, rotation or double rotation have exactly the same effect on $\mathrm{val}_t(T)$; this fact makes our analysis possible. Lemma 4 gives the effect of a single insertion on $\mathrm{val}_t(T)$, $t\geqq 1$. In Lemma 5 below we study the cumulative effect of all n insertions.

DEFINITION. Let C_t, $t\geqq 2$, be the number of insertions (among our sequence of n insertions) such that the critical node exists and has height $\geqq t$ and let I_t, $t\geqq 2$, be the number of insertions such that the critical node does not exist and the tree inserted into has height $\geqq t-2$.

We have the following Lemmas 5 and 6. In Lemma 6 we derive an upper bound on $\mathrm{val}_t(T_n)$ and in Lemma 5 we derive bounds on C_t and I_t in terms of $\mathrm{val}_i(T_n)$, $i\leqq t$. We then use Lemmas 5 and 6 to prove the main result of this section, Theorem 6.

LEMMA 5. a) $C_{t+1}=\sum_{i=2}^{t}(\mathrm{val}_i(T_n)-I_{i+1})/2^{t+1-i}+C_2/2^{t-1}$ *for* $t\geqq 2$,

b) $I_t=\mathrm{height}(T_n)+2-t$ *for* $2\leqq t\leqq \mathrm{height}(T_n)+1$,

c) $C_2+I_2=n$.

Proof. a) The proof of part a) is based on the following claim.

CLAIM. $\text{val}_t(T_n) = 2C_{t+1} + I_{t+1} - C_t$ *for* $t \geqq 2$.

Proof. Certainly $\text{val}_t(T_0) = 0$ for $t \geqq 2$. Next note that val_t is increased by one for every insertion such that either the critical node exists and has height $\geqq t+1$ or the critical node does not exist and the tree inserted into has height $\geqq t-1$ (there are exactly $C_{t+1} + I_{t+1}$ insertions of this type) and that val_t is decreased by one for every insertion whose critical node exists and has height t (there are exactly $C_t - C_{t+1}$ insertions of this type).

Thus

$$\text{val}_t(T_n) = \text{val}_t(T_0) + (C_{t+1} + I_{t+1}) - (C_t - C_{t+1}) \quad \text{for } t \geqq 2. \qquad \square$$

The claim above can be rewritten as

$$C_{t+1} = (\text{val}_t(T_n) - I_{t+1})/2 + C_t/2.$$

Part a) is now easily shown by induction.

b) For l, $0 \leqq l \leqq \text{height}(T_n) - 1$, there is exactly one insertion which increases the height from l to $l+1$. Thus $I_t = \text{height}(T_n) + 2 - t$.

c) The critical node always has height at least 2 and the tree inserted into has height at least 0. $\square$

LEMMA 6. $\text{val}_t(T_n) \leqq n/(1.618)^t$ *for* $t \geqq 1$.

Proof. We use the following notations: $m_1(t)$ is the number of unbalanced nodes of height t in T_n, and $m_2(t)$ is the number of balanced nodes of height t in T_n. Then $m_1(1) = 0$ and $m_1(t) + m_2(t)$ is the number of nodes of height t, $t \geqq 1$. Also $\text{val}_t(T_n) = m_1(t)$.

CLAIM 1. $2m_2(t) + m_1(t+1) + m_1(t) = m_1(t-1) + m_2(t-1)$ *for* $2 \leqq t \leqq \text{height}(T_n)$.

Proof. The right-hand side is the number of nodes of height $t-1$, the left-hand side counts the number of edges terminating at height $t-1$. There are two such edges for every balanced node of height t and one such edge for every unbalanced nodes of height t or $t+1$. $\square$

If we define $m_1(0) = 0$, $m_2(0) = n$ then the claim above is also true for $t = 1$.

Claim 1 yields

$$m_1(t+1) + Cm_1(t) + Cm_2(t) \leqq \frac{1}{C}[m_1(t) + Cm_1(t-1) + Cm_2(t-1)]$$

for the constant $C \simeq 1.618 < 2$ such that $C - 1/C = 1$. Hence

$$Cm_1(t) \leqq \frac{1}{C^t}[m_1(1) + Cm_1(0) + Cm_2(0)] = \frac{n}{C^{t-1}}$$

and

$$\text{val}_t(T_n) = m_1(t) \leqq \frac{n}{C^t} = \frac{n}{(1.618)^t}. \qquad \square$$

THEOREM 6. *Consider a sequence of n arbitrary insertions into an initially empty* AVL-*tree. Then*

$$C_t \leqq 6.85 \frac{n}{(1.618)^t}$$

and

$$I_t \leqq 1.44 \log n + 2 - t \quad \textit{for } t \geqq 2.$$

Proof. The bound on I_t is immediate from Lemma 5b and the fact that an AVL-tree with n leaves has height at most $1.44 \log n$. The bound on C_t follows from Lemma

5a, c and Lemma 6, namely from Lemma 5a we get

$$C_t = \sum_{i=2}^{t-1} (\mathrm{val}_i\,(T_n) - I_{t+1})/2^{t-i} + C_2/2^{t-2}$$

$$\leqq n/2^{t-2} + \sum_{i=2}^{t-1} \mathrm{val}_i\,(T_n)/2^{t-i} \quad \text{since } C_2 \leqq n \text{ by Lemma 5c and } I_{t+1} \geqq 0$$

$$\leqq n/2^{t-2} + \sum_{i=2}^{t-1} (n2^i)/2^t(1.618)^i \quad \text{because of Lemma 6}$$

$$\leqq 6.85n/(1.618)^t \quad \text{by straightforward estimation.}$$ □

Theorems 1 and 6 give us good information on the amortized length of the critical path:

1) The amortized length is bounded by 2.618.

2) The number of insertions which have a critical path of length exceeding t decreases exponentially in t.

6. Conclusions. We have shown that the total number of balance changes required to process a sequence of n arbitrary insertions into an initially empty AVL-tree is at most $2.618n$. Moreover, the number of insertions in such a sequence which have a critical path of length exceeding t decreases exponentially in t. Finally, for sequences of n random insertions the expected total number of balance changes lies between $1.47n$ and $2.26n$.

Recently, Tsakalidis [9] has shown that the total number of rebalancing operations (=balance and structural changes) in processing a sequence of n deletions from an AVL-tree with n leaves is bounded by $1.618n$. Experiments by Karlton et al. [5] suggest that the expected number of rebalancing operations is $1.126n$ for a sequence of n random deletions.

For mixed sequences of insertions and deletions the amortized rebalancing cost is *not* constant. The reader should have no difficulty in finding a sequence of n insertions and deletions which produces $O(n \log n)$ balance changes. However, for *random* sequences of insertions and deletions the expected number of rebalancing operations might be $O(n)$ as experiments (cf. [5]) suggest. We leave the proof of this experimental fact as a major challenge to the reader.

REFERENCES

[1] G. M. ADEL'SON-VEL'SKII AND E. M. LANDIS, *An algorithm for the organisation of information*, Dokl. Akad. Nauk SSSR, 146 (1962), pp. 263-266 (in Russian); English Translation in Soviet. Math., 3, pp. 1259-1262.

[2] M. R. BROWN, *A partial analysis of random height-balanced trees*, this Journal, 8 (1979), pp. 33-41.

[3] C. C. FOSTER, *Information storage and retrieval using* AVL-*trees*, ACM 20th National Conference, 1965, pp. 192-205.

[4] S. HUDDLESTON AND K. MEHLHORN, *A new data structure for representing sorted lists*, Acta Informatica, 17 (1982), pp. 157-184.

[5] P. L. KARLTON, S. H. FULLER, R. E. SCROGGS AND E. B. KAEHLER, *Performance of height-balanced trees*, Comm. ACM, 19 (1976), pp. 23-28.

[6] D. E. KNUTH, *The Art of Computer Programming, Vol.* 3, *Sorting and Searching*, Addison-Wesley, Reading, MA, 1973.

[7] K. MEHLHORN, *A partial analysis of height-balanced trees*, Technical Report A 79/13, Univ. des Saarlandes, 1979.

[8] ———, *A partial analysis of height-balanced trees under random insertions and deletions*, this Journal, 11 (1982), pp. 748-760.

[9] A. K. TSAKALIDIS, *Rebalancing operations for deletions in* AVL-*trees*, RAIRO Inform. théorique, in press.