UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF WATERLOO COMPUTER SCIENCE DEPARTMENT

An Analysis of 2-3 Trees and B-trees

Gaston H. Gonnet
Nivio Ziviani
Derick Wood

CS-81-21

June, 1981

An Analysis of 2-3 Trees and B-trees

Gaston H. Gonnet
Nivio Ziviani

Department of Computer Science
University of Waterloo, Waterloo
Ontario N2L 3G1, Canada

Derick Wood

Unit for Computer Science
McMaster University, Hamilton
Ontario L8S 4K1, Canada

ABSTRACT

We present a new way of describing the composition of a fringe in terms of tree collections. This enables us to answer an open problem posed by Knuth, and improve upon previous results. We give sharp bounds on the expected number of splits and on the expected depth of the deepest safe node on a particular insertion path in 2-3 trees and B-trees, and drastically improve the bounds on the expected number of nodes in 2-3 trees. Finally, we give bounds on the expected number of internal nodes, on the expected number of splits, and on the expected depth of the deepest safe node for both 2-3 trees and B-trees using an overflow technique.

Key phrases: Analysis of algorithms, 2-3 trees, B-trees, expected number of nodes, expected number of splits, storage used, overflow technique, deepest safe node.

CR *Categories:* 3.73, 3.74, 5.25

The work of the first author was supported by a Natural Sciences and Engineering Research Council of Canada Grant No. A-3353, the second by a Brazilian Coordenação do Aperfeiçoamento de Pessoal de Nivel Superior Contract No. 4799/77 and by the Universidade Federal de Minas Gerais (Brazil), and the third by a Natural Sciences and Engineering Research Council of Canada Grant No. A-7700.

June 1, 1981

An Analysis of 2-3 Trees and B-trees

Gaston H. Gonnet
Nivio Ziviani

Department of Computer Science
University of Waterloo, Waterloo
Ontario N2L 3G1, Canada

Derick Wood

Unit for Computer Science
McMaster University, Hamilton
Ontario L8S 4K1, Canada

1. Introduction

B-trees were presented by Bayer and McCreight (1972) as a dictionary structure primarily for secondary store. In a B-tree of order m each node has between $m+1$ and $2m+1$ subtrees, and all external nodes (henceforth called leaves) appear at the same level. A special class of B-trees called 2-3 trees, more appropriate for primary store, were introduced by John Hopcroft in 1970 (see Knuth, 1973, p. 468). A 2-3 tree is a B-tree of order $m=1$, as shown in Figure 1.

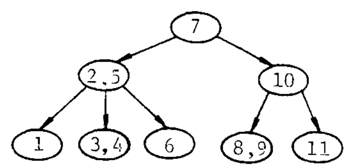


Figure 1 A 2-3 tree with 11 keys

In the intervening years B-trees have gained in popularity as regards both practice and theory, to the extend that Comer (1979a) has referred to them as ubiquitous. Comer (1979a, 1979b) described several systems which use B-trees. At the same time 2-3 trees have become equal contenders with AVL trees, often being the preferred data structure. (See the recent work of Huddleston and Mehlhorn (1980), and Guibas, McCreight, Plass and Roberts (1977).)

In spite of this interest, no analytical results were known about the performance of B-trees and 2-3 trees prior to the pioneering work of Yao (1978). Yao (1978) presented a technique of analysis now known as fringe analysis, which he used to find bounds on the expected number of nodes in a B-tree after n "random" insertions. Although his results were slightly extended by Brown (1979b), many questions of interest were left open. Let us consider some of the basic questions.

First, the expected number of nodes in a B-tree after N random insertions is certainly of interest, since each node is usually represented as a page. Thus this measure indicates how well storage is used in practice, when the assumptions of the model are met. We extended and refine the results of Yao (1978) with regard to this measure.

Second, when considering insertions, the costliest operation is surely that of splitting an overfull node, since this involves not only the creation of a new node but also an insertion into the next higher level of the tree. Knuth (see Chvatal, Klarner, and Knuth, 1972, problem 37) raised the following question related to 2-3 trees: "... how many splittings will occur on the n^{th} random insertion, on the average, ...". We present the first analysis of this measure for both 2-3 trees and B-trees, albeit only a partial one.

Third, a different insertion algorithm for B-trees, which uses a technique called overflow, was presented by Bayer and McCreight (1972, p.183) and also by Knuth (1973, pp. 477-478). In the overflow technique, instead of splitting an overfull node, we look first at its brother nodes and make a rearrangement of keys when possible. The effect of the overflow technique is to produce trees with less internal nodes on the average, which means a better storage utilization. We present an analysis of the expected number of internal nodes and the expected number of splitting operations for both B-trees and 2-3 trees using an overflow technique. Rosenberg and Snyder (1981) presented a study of B-trees with minimal number of internal nodes (and consequently optimal space utilization), and recently Eisenbarth and Mehlhorn (1980) have considered the application of fringe analysis to B-trees using overflow technique.

Fourth, consider B-trees in a concurrent environment; see Kwong and Wood (1980) for a survey of the technique used. One basic technique identified there was first used by Bayer and Schkolnick (1977), namely lock the deepest safe node on the insertion path. A node is insertion safe if it contains less than the maximum number of keys allowed. Then a safe node is the deepest one in a particular insertion path if there are no safe nodes below it. Since locking the deepest safe node effectively prevents access by other processes it is of interest to determine how deep the deepest safe node can be expected to be. Our results enble us to provide some insight into this question also.

Finally, we should point out that our results do have a basic limitation, namely we do not consider the effect of deletions. Note that both Yao (1978) and Brown (1979b) also ignored deletions. The reasons for this are twofold: firstly deletions do not preserve randomness, and secondly it is not clear how to incorporate them in the analysis.

In this work we assume that all trees are random trees. Consider a B-tree tree T with N keys and consequently $N+1$ leaves. These N keys divide all possible key values into $N+1$ intervals. An insertion into T is said to be a *random insertion* if it has an equal probability of being in any of the $N+1$ intervals defined above. A *random* $B-tree$ with N keys is a B-tree tree constructed by making N successive random insertions into an initially empty tree. Random 2-3 trees are defined in the same way as random B-trees are defined.

In Section 2 we present the basic technique used to perform the analysis of 2-3 trees and B-trees. In Section 3 we perform the analysis of 2-3 trees and derive results related the four basic questions considered above. In Section 4 we perform the analysis of B-trees and also derive results related to the four basic questions mentioned above.

2. Fringe Analysis Technique

The *fringe* of a tree consists of subtrees that are isomorphic to members of a *tree collection*, which is a finite collection of trees that satisfy an specific constraint (e.g. the collection of 2-3 trees of height k, $k>0$). Figure 2 displays the two types of trees in a 2-3 tree collection of height 1. The fringe of a 2-3 tree is obtained by deleting all nodes at a distance greater than k ($k>0$) from the leaves.

Figure 2 Tree collection of 2-3 trees of height 1

The composition of the fringe can be described in several ways. One possible way is to consider the probability that a leaf of the tree belongs to each of the members of the corresponding tree collection. In other words, the probability P is

$$P_i(N) = \frac{\underline{Expected\ number\ of\ leaves\ of\ type\ i\ in\ an\ N-key\ tree}}{N+1} \qquad [1]$$

Transitions between trees of a tree collection can be used to model the insertion process. Table 1 shows the transitions between the trees of the 2-3 tree collection shown in Figure 2, where values represent the number of leaves lost and obtained under a transition. (e.g. an insertion of a key into the type 2 tree shown in Figure 2, three leaves of the type 2 tree are lost and four leaves of the type 1 tree are obtained.) The probability of an insertion occurring in each of the subtrees of the fringe can be obtained from the steady state solution of a matrix recurrence relation in a Markov chain. This is the procedure used by Yao (1978). One main difference between the procedure presented in this paper and Yao's procedure lies on the way we have described the composition of the fringe. (Yao's description of the composition of the fringe considers the expected number of trees of type i, while we describe it in terms of leaves as in [1] above.)

$$\begin{bmatrix} -2 & 4 \\ 3 & -3 \end{bmatrix}$$

Table 1 Transition matrix corresponding to the tree collection of 2-3 trees of height 1

Let $P(N)$ be an m-component column vector containing $P_i(N)$. Then

$$P(N) = \left(I+\frac{T}{N}\right)P(N-1) \qquad [2]$$

where I is the $m \times m$ identity matrix, and T is the transition matrix. To solve $P(N)$ from [2] we define an m-component column vector $p(N)$ containing $p_i(N)$ by

$$p_i(N) = \frac{P_i(N)}{N+1}.$$

In terms of $p(N)$, [2] can be transformed to

$$p(N)(N+1) = \left(I+\frac{T}{N}\right)p(N-1)N$$

$$= (NI+T)p(N-1)$$

$$= (NI+I+T-I)p(N-1)$$

or

$$p(N) = \left(I+\frac{T-I}{N+1}\right)p(N-1) \qquad [3]$$

The matrix T has eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_m$ where $\lambda_1=1$ and $\mathrm{Re}\lambda_m \leqslant \mathrm{Re}\lambda_{m-1} \leqslant \cdots \leqslant \mathrm{Re}\lambda_2 < 1$. (The eigenvalues are considered in decreasing order of their real part.) Considering also that the matrix T is independent of N (by construction) then it is well known that $p(N)$, which is the solution of [2], converges to the solution of

$$(T-I)q(N) = 0, \quad as \ N \to \infty \qquad [4]$$

where $q(N)$ is also an m-component column vector (Knuth, 1973, pp.679-680).

Lemma 2.1. Let T be an $m \times m$ real matrix with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_m$, where $\lambda_1 = 1$ and $\mathrm{Re}\lambda_m \leqslant \mathrm{Re}\lambda_{m-1} \leqslant \ldots \leqslant \mathrm{Re}\lambda_2 < 1$. If $p(N)$ is an m-component vector satisfying $p(N) = \left(I+\frac{T-I}{N+1}\right)p(N-1)$, then

$$p(N) = \alpha_1 x_1 + \sum_{j=2}^{m} \prod_{i=2}^{N} \left(\frac{i+\lambda_j}{i+1}\right) \alpha_j x_j \qquad [5]$$

or

$$p(N) = \alpha_1 x_1 + \sum_{j=2}^{m} \frac{2\Gamma(N+1+\lambda_j)}{\Gamma(N+2)\Gamma(\lambda_j+2)} \alpha_j x_j, \quad \lambda_j \neq -2, -3, -4, \ldots$$

Proof: It is known that

$$Tx_j = \lambda_j x_j$$

where λ_j and x_j are respectively an eigenvalue and an eigenvector of the matrix T. Then

$$\left(I+\frac{T-I}{N+1}\right)x_j = \lambda_j x_j$$

or

$$\frac{N}{N+1} Ix_j + \frac{Tx_j}{N+1} = \frac{N}{N+1}x_j + \frac{\lambda_j x_j}{N+1} = \frac{N+\lambda_j}{N+1}x_j.$$

Let $A^{(N)} = \left(I+\frac{T-I}{N+1}\right)$. Then

$$A^{(N)}x_j = \lambda_j^{(N)}x_j$$

and

$$\lambda_j^{(N)} = \frac{N+\lambda_j}{N+1}, \quad where \ \lambda_1^{(N)} = 1.$$

Thus

$$p(N) = \prod_{i=2}^{N} A^{(i)} p(1)$$

where

$$p(1) = \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_m x_m$$

$$p(N) = \prod_{i=2}^{N} A^{(i)} p(1) = \alpha_1 x_1 + \prod_{i=2}^{N} \lambda_2^{(i)} \alpha_2 x_2 + \cdots + \prod_{i=2}^{N} \lambda_m^{(i)} \alpha_m x_m$$

or

$$p(N) = \sum_{j=1}^{m} \prod_{i=2}^{N} \left(\frac{i+\lambda_j}{i+1} \right) \alpha_j x_j$$

$$= \alpha_1 x_1 + \sum_{j=2}^{m} \prod_{i=2}^{N} \left(\frac{i+\lambda_j}{i+1} \right) \alpha_j x_j$$

$$= \alpha_1 x_1 + \sum_{j=2}^{m} \frac{2\Gamma(N+1+\lambda_j)}{\Gamma(N+2)\Gamma(\lambda_j+2)} \alpha_j x_j. \quad \square$$

Corollary: $p(N) \approx \alpha_1 x_1 + O(N^{\lambda_2 - 1})$ [6]

Proof: [5] can be approximated by Stirling's formula

$$N! \approx \sqrt{2\pi N} \left(\frac{N}{e} \right)^N$$

and considering

$$\frac{\Gamma(N+\epsilon)}{\Gamma(N)} = \frac{(N+\epsilon)^{N+\epsilon} e^{-N-\epsilon} \sqrt{2\pi(N+\epsilon)}}{N^N e^{-N} \sqrt{2\pi N}}$$

$$= N^{\epsilon} \left(1 + \frac{\epsilon(\epsilon+1)}{2N} + O(N^{-2}) \right)$$

Thus

$$p(N) \approx \alpha_1 x_1 + O(N^{\lambda_2 - 1}). \quad \square$$

As an example consider the tree collection shown in Figure 2 for values of $N=1, 2, \ldots, 6, \ldots$. For expression [3] consider $P(1) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. (A 2-3 tree with only one key is a type 1 2-3 tree with probability 1.) For the expression [5], the eigenvalues of T are $\lambda_1 = 1$ and $\lambda_2 = -6$, with eigenvectors $\begin{bmatrix} 1 \\ 3/4 \end{bmatrix}$ and $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$ respectively, and the values for α_1 and α_2 are obtained from

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \alpha_1 \begin{bmatrix} 1 \\ 3/4 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

where $\alpha_1 = 4/7$ and $\alpha_2 = 3/7$.

3. Partial Analysis of 2-3 Trees

In a 2-3 tree every internal node contains either 1 or 2 keys. To insert a new key into a node that contains only one key, we insert it as the second key. If the node already contains two keys, we split it into two one-key nodes, and insert the middle key into the parent node. This process may propagate up if the parent node already contains two keys. When there is no node above we create a new root node to insert the middle key.

We now define certain complexity measures. Let $\overline{n}(N)$ be the expected number of nodes in a 2-3 tree after the random insertion of N keys into the initially empty tree. Let $Pr\{j\ splits\}$ be the probability that j splits occur on the $(N+1)^{th}$ random insertion into a random 2-3 tree with N keys. Let $Pr\{j\ or\ more\ splits\}$ be the probability that j or more splits occur on the $(N+1)^{th}$ random insertion into a random 2-3 tree with N keys. Let $\overline{s}(N)$ be the expected number of splits that occur in a 2-3 tree during the random insertion of N keys into the initially empty tree. Let $E[s(N)]$ be the expected number of splits that will occur on the $(N+1)^{th}$ insertion into a random 2-3 tree with N keys. Let $Pr\{dsn\ at\ j^{th}\ lowest\ level\}$ be the probability that the deepest safe node is located at the j^{th} $(j\geqslant 1)$ lowest level of a random N-key 2-3 tree. Let $Pr\{dsn\ above\ j^{th}\ lowest\ level\}$ be the probability that the deepest safe node is located above the j^{th} lowest level of a random N-key 2-3 tree.

In Sections 3.1, 3.2, and 3.3 we shall derive exact values for $Pr\{0\ splits\}$, $Pr\{1\ split\}$, $Pr\{2\ splits\}$, $Pr\{1\ or\ more\ splits\}$, $Pr\{2\ or\ more\ splits\}$, $Pr\{3\ or\ more\ splits\}$, and bounds on $\overline{s}(N)$, $E[s(N)]$, and improve Yao's previous results on $\overline{n}(N)$. In Section 3.4 we shall derive exact values for $Pr\{0\ splits\}$, $Pr\{1\ split\}$, $Pr\{2\ or\ more\ splits\}$, and bounds on $\overline{n}(N)$, $\overline{s}(N)$, and $E[s(N)]$ for an insertion algorithm that uses an overflow technique. In Section 3.5 we shall derive exact values for $Pr\{dsn\ at\ 1^{st}\ lowest\ level\}$, $Pr\{dsn\ at\ 2^{nd}\ lowest\ level\}$, $Pr\{dsn\ at\ 3^{rd}\ lowest\ level\}$, and $Pr\{dsn\ above\ 3^{rd}\ lowest\ level\}$ for the normal insertion algorithm, and $Pr\{dsn\ at\ 1^{st}\ lowest\ level\}$, $Pr\{dsn\ at\ 2^{nd}\ lowest\ level\}$, and $Pr\{dsn\ above\ 2^{nd}\ lowest\ level\}$ for the insertion algorithm using an overflow technique. In Section 3.6 we discuss the possibilities of higher order analyses.

Table 2 shows the summary of the results related to 2-3 trees using the normal insertion algorithm, and Table 3 shows the summary of the results related to 2-3 trees using the overflow technique.

3.1. First Order Analysis

The analysis of the lowest level of the 2-3 tree to estimate $\overline{n}(N)$, $Pr\{0\ splits\}$, $Pr\{1\ or\ more\ splits\}$, $\overline{s}(N)$, and $E[s(N)]$ can be carried out in the following way. The tree collection shown in Figure 2 contains two members and Table 1 shows that its corresponding transition matrix is

$$T = \begin{bmatrix} -2 & 4 \\ 3 & -3 \end{bmatrix}$$

From [4] we have $(T-I)p(N) = 0$, and therefore $p_1(\infty) = 4/7$, and $p_2(\infty) = 3/7$. Since the eigenvalues of T are 1 and -6, we observe that $p_1(N) = 4/7$ and $p_2(N) = 3/7$ for $N\geqslant 6$.

	First order analysis ($N \geqslant 6$)	Second order † analysis ($N \to \infty$)	Third order ‡ analysis ($N \to \infty$)
$\frac{\bar{n}(N)}{N}$	$[0.64+0.14/N,\ 0.86-0.14/N]$	$[0.70+0.20/N,\ 0.79-0.21/N]$	$[0.73+0.23/N,\ 0.77-0.23/N]$
Pr{0 *splits*}	4/7	4/7	4/7
Pr{1 *or more splits*}	3/7	3/7	3/7
Pr{1 *split*}	-	0.25	0.25
Pr{2 *or more splits*}	-	0.18	0.18
Pr{2 *splits*}	-	-	0.10
Pr{3 *or more splits*}	-	-	0.08
$\bar{s}(N)$	$[0.64 + 0.14/N - \lceil \log_3(N+1) \rceil /N,\ 0.86 - 0.14/N - \lfloor \log_2(N+1) \rfloor /N]$	$[0.70 + 0.20/N - \lceil \log_3(N+1) \rceil /N,\ 0.79 - 0.21/N - \lfloor \log_2(N+1) \rfloor /N]$	$[0.73 + 0.23/N - \lceil \log_3(N+1) \rceil /N,\ 0.77 - 0.23/N - \lfloor \log_2(N+1) \rfloor /N]$
$E[s(N)]$	$[0.43,\ 0.43 \lfloor \log_2(N+1) \rfloor]$	$[0.61,\ 0.25 + 0.18 \lfloor \log_2(N+1) \rfloor]$	$[0.69,\ 0.46 + 0.08 \lfloor \log_2(N+1) \rfloor]$
Pr{*dsn at* 1^{st} *l.level*}	4/7	4/7	4/7
Pr{*dsn at* 2^{nd} *l.level*}	-	0.25	0.25
Pr{*dsn at* 3^{rd} *l.level*}	-	-	0.10
Pr{*dsn above* 3^{rd} *lowest level*}	-	-	0.08

† Results are approximated to $O(N^{-6.55})$

‡ Results are approximated to $O(N^{-4.37})$

Table 2 Summary of the 2-3 tree results

	Second order analysis ($N \to \infty$) †
$\frac{\bar{n}(N)}{N}$	$[0.63+0.13/N,\ 0.71-0.29/N]$
Pr{0 *splits*}	0.61
Pr{1 *split*}	0.23
Pr{2 *or more splits*}	0.16
$\bar{s}(N)$	$[0.63+0.13/N - \lceil \log_3(N+1) \rceil /N,\ 0.71-0.29/N - \lfloor \log_2(N+1) \rfloor /N]$
$E[s(N)]$	$[0.55,\ 0.23+0.16 \lfloor \log_2(N+1) \rfloor]$
Pr{*dsn at* 1^{st} *lowest level*}	0.61
Pr{*dsn at* 2^{nd} *lowest level*}	0.23
Pr{*dsn above* 2^{nd} *lowest level*}	0.16

† Results are approximated to $O(N^{-6.81})$

Table 3 Summary of the 2-3 tree results using an overflow technique

Let $A_i(N)$ indicate the expected number of trees of type i in a random N-key 2-3 tree. Let L_i indicate the number of leaves of the type i tree. We observe that [1] can be written as

$$P_i(N) \;=\; \frac{A_i(N)L_i}{N+1}. \qquad [7]$$

Let nl indicate the number of nodes at level l of a 2-3 tree with N keys. Let nal indicate the number of nodes above the level l of a 2-3 tree.

Lemma 3.1. $\frac{nl-1}{2} \leqslant nal \leqslant nl-1$

Proof: Consider the level l as being the $N+1$ leaves of a 2-3 tree with N keys. (Each leaf represents a node.) The minimum and the maximum number of nodes above the level l is obtained when each node above level l contains 2 keys and 1 key respectively. (That is $2nal = nl-1$ and $nal = nl-1$ respectively.) □

Lemma 3.1 and expression [7] lead to the following theorem:

Theorem 3.2.

$$\left(1+\frac{1}{2}\right)\left(\frac{p_1}{L_1}+\frac{p_2}{L_2}\right)(N+1)-\frac{1}{2} \leqslant \overline{n}(N) \leqslant 2\left(\frac{p_1}{L_1}+\frac{p_2}{L_2}\right)(N+1)-1 \quad \text{for } N\geqslant 1$$

Corollary. $\frac{9}{14}+\frac{1}{7N} \leqslant \frac{\overline{n}(N)}{N} \leqslant \frac{6}{7}-\frac{1}{7N}$ for $N\geqslant 6$

The remaining results are contained in the lemmas that follow.

Lemma 3.3. $Pr\{0 \; splits\} = \frac{4}{7}$ for $N\geqslant 6$

Proof: An insertion into a type 1 tree shown in Figure 2 causes no split, and the probability that a random insertion into a random 2-3 tree falls into a type 1 tree is p_1. □

Lemma 3.4. $Pr\{1 \; or \; more \; splits\} = \frac{3}{7}$ for $N\geqslant 6$

Proof: Similar to the proof of Lemma 3.3. □

Let $\overline{h}(N)$ indicate the expected height of a random 2-3 tree with N keys.

Lemma 3.5. $\overline{s}(N) = \frac{\overline{n}(N)}{N}-\frac{\overline{h}(N)}{N}$

Proof: From the insertion algorithm we can see that each time a node split occur one new node is created, except when the node is a root, in which case two nodes are created. □

Corollary. $\lceil \log_3(N+1)\rceil \leqslant \overline{h}(N) \leqslant \lfloor \log_2(N+1)\rfloor$

Lemma 3.5 leads to the following theorem:

Theorem 3.6.

$$\frac{9}{14}+\frac{1}{7N}-\frac{\lceil \log_3(N+1)\rceil}{N} \leqslant \overline{s}(N) \leqslant \frac{6}{7}-\frac{1}{7N}-\frac{\lfloor \log_2(N+1)\rfloor}{N} \quad \text{for } N\geqslant 6$$

Lemma 3.7. $E[s(N)] \geqslant Pr\{1 \; or \; more \; splits\}$

Proof: Similar to the proof of Lemma 3.3. □

Corollary. $E[s(N)] \geqslant \frac{3}{7}$ for $N \geqslant 6$

Lemma 3.8. $E[s(N)] \leqslant Pr\{1 \text{ or more splits}\} \lfloor \log_2(N+1) \rfloor$

Proof: The upper bound on $E[s(N)]$ is equal to the number of splits/insertion in the fringe plus all splits that might occur in the nodes above the lowest level, which might be equal to the height of the tree with all nodes binary but the nodes on the path of splitting. □

Lemmas 3.7 and 3.8 lead to the following theorem:

Theorem 3.9. $\frac{3}{7} \leqslant E[s(N)] \leqslant \frac{3}{7} \lfloor \log_2(N+1) \rfloor$ for $N \geqslant 6$

It is interesting to note that the expected value for $E[s(N)]$ probably converges to the value of $\overline{s}(N)$. However, we cannot prove this; $E[s(N)]$ may oscillate between a lower bound and an upper bound, where the lower bound is the number of splits per insertion in the fringe, and the upper bound is the number of splits per insertion in the fringe plus the number of splits per insertion outside the fringe. (The upper bound is a function of $\log_2 N$.)

3.2. Second Order Analysis

The analysis for the two lowest levels of 2-3 trees leads to better bounds for $\overline{n}(N)$, $\overline{s}(N)$, $E[s(N)]$, and exact results for $Pr\{1 \text{ split}\}$, and $Pr\{2 \text{ or more splits}\}$. Yao (1978) showed that there are 12 possible trees in the tree collection of 2-3 trees of height 2, which are grouped into 7 types, as shown in Figure 3. The corresponding transition matrix is shown in Table 4.

Again using [4] we obtain

$$\begin{aligned} p_1 &= 1656/7991 \\ p_2 &= 1980/7991 \\ p_3 &= 5472/55937 \\ p_4 &= 7128/55937 \\ p_5 &= 1575/7991 \\ p_6 &= 800/7991 \\ p_7 &= 180/7991. \end{aligned} \qquad [8]$$

Since the eigenvalues of T are 1, $-5.55\pm6.25i$, -6, $8.23\pm1.37i$, and -12.44, using [3] the asymptotic values of $p(N)$ obtained from [6] are approximated to the $O(N^{-6.55})$.

Lemma 3.1 and expression [7] lead to the following theorem:

Theorem 3.10.
$$\left[\left(3+\frac{1}{2}\right) \left(\sum_{i=1}^{3} \frac{p_i}{L_i} \right) + \left(4+\frac{1}{2}\right) \left(\sum_{i=4}^{7} \frac{p_i}{L_i} \right) \right] (N+1) - \frac{1}{2}$$
$$\leqslant \overline{n}(N) \leqslant \left[4 \left(\sum_{i=1}^{3} \frac{p_i}{L_i} \right) + 5 \left(\sum_{i=4}^{7} \frac{p_i}{L_i} \right) \right] (N+1) - 1$$

Corollary.
$$\frac{78501}{111874} + \frac{11282}{55937N} + O(N^{-6.55}) \leqslant \frac{\overline{n}(N)}{N} \leqslant \frac{44343}{55937} - \frac{11594}{55937N} + O(N^{-6.55})$$

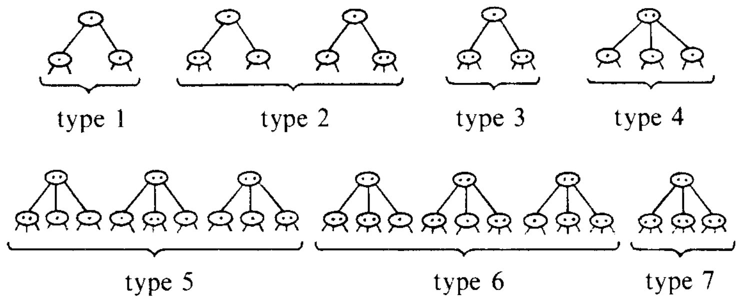


Figure 3 Tree collection of 2-3 trees of height 2

$$\begin{bmatrix} -4 & & & & 8\cdot 3/7 & 4\cdot 6/8 & 4\cdot 6/9 \\ 5 & -5 & & & & 5\cdot 6/8 & 5\cdot 6/9 \\ & 6\cdot 2/5 & -6 & & & & 6\cdot 6/9 \\ & 6\cdot 3/5 & & -6 & & & \\ & & 7 & 7 & -7 & & \\ & & & & 8\cdot 4/7 & -8 & \\ & & & & & 9\cdot 2/8 & -9 \end{bmatrix}$$

Table 4 Transition matrix corresponding to the tree collection of 2-3 trees of height 2 (values represent the number of leaves lost and obtained under a transition)

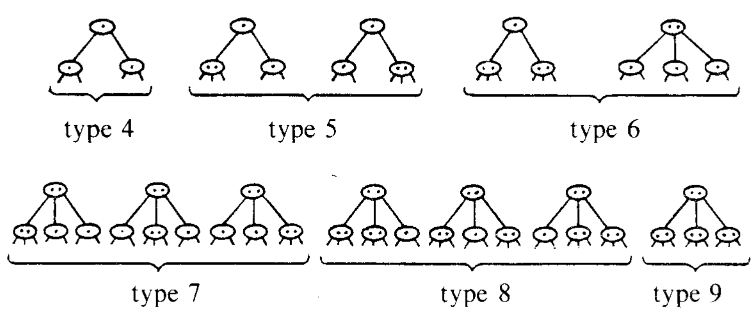


Figure 4 Tree collection of 2-3 trees of height 2 obtained by grouping type 3 and type 4 shown in Figure 3 into type 6 above

To five place decimals we have

$$0.70169+\frac{0.20169}{N}+O(N^{-6.55}) \leqslant \frac{\overline{n}(N)}{N} \leqslant 0.79273-\frac{0.20727}{N}+O(N^{-6.55}) .$$

Lemma 3.11. $Pr\{1\ split\} = \frac{13788}{55937}+O(N^{-6.55})$

Proof: An insertion into the type 2 tree shown in Figure 3 causes one split in 3/5 of the times, and an insertion into the type 3 shown in Figure 3 always causes one split. Since the probability that a random insertion into a random 2-3 tree falls into a type 2 or type 3 tree are p_2 and p_3 respectively, then $Pr\{1\ split\} = 3/5p_2+p_3$. □

Lemma 3.12. $Pr\{2\ or\ more\ splits\} = \frac{1455}{7991}+O(N^{-6.55})$

Proof: Similar to the proof of Lemma 3.11. □

Lemma 3.5 leads to the following theorem:

Theorem 3.13. $\frac{78501}{111874}+\frac{11282}{55937N}-\frac{\lceil \log_3(N+1)\rceil}{N}+O(N^{-6.55}) \leqslant \overline{s}(N) \leqslant$

$$\frac{44343}{55937}-\frac{11594}{55937N}-\frac{\lfloor \log_2(N+1)\rfloor}{N}+O(N^{-6.55})$$

To five place decimals we have

$$0.70169+\frac{0.20169}{N}-\frac{\lceil \log_3(N+1)\rceil}{N}+O(N^{-6.55}) \leqslant \overline{s}(N) \leqslant$$

$$0.79273-\frac{0.20727}{N}-\frac{\lfloor \log_2(N+1)\rfloor}{N}+O(N^{-6.55}) .$$

Lemma 3.14. $E[s(N)] \geqslant Pr\{1\ split\}+2Pr\{2\ or\ more\ splits\}$

Proof: Similar to the proof of Lemma 3.3. □

Lemma 3.15. $E[s(N)] \leqslant Pr\{1\ split\}+Pr\{2\ or\ more\ splits\}\lfloor \log_2(N+1)\rfloor$

Proof: Similar to the proof of Lemma 3.8. □

Lemmas 3.14 and 3.15 lead to the following theorem:

Theorem 3.16.

$$\frac{34158}{55937}+O(N^{-6.55}) \leqslant E[s(N)] \leqslant \frac{13788}{55937}+\frac{1455}{7991}\lfloor \log_2(N+1)\rfloor+O(N^{-6.55})$$

To five place decimals we have

$$0.61065+O(N^{-6.55}) \leqslant E[s(N)] \leqslant 0.24649+0.18208\lfloor \log_2(N+1)\rfloor+O(N^{-6.55}) .$$

3.3. Third Order Analysis

In this section we present the analysis of the three lowest levels of 2-3 trees. Brown (1979b) performed a three level analysis using a transition matrix of 978X978 elements, and obtained asymptotic values for the number of nodes with one key and the number of nodes with two keys at each of the three lowest levels. However an equivalent three level analysis can be performed on a smaller matrix by grouping trees into types, in the same way the two level matrix in the previous section was reduced from 12×12 to 7×7. If we consider combinations of the 7 types of the two level tree collection as subtrees of nodes with one and two keys then it is possible to obtain a three level tree collection with 224 types. Yet it is possible to reduce the 224 types to 147 types, as we shall see in the following.

The idea behind our approach is to group all trees with the same number of leaves into types. Thus the tree collection shown in Figure 3 is reduced from 7 types to 6 types by grouping the types 3 and 4 into one unique type, as shown in Figure 4. In this new tree collection the types are numbered sequentially from 4 to 9, where the type 4 tree has 4 leaves, the type 5 tree has 5 leaves, ..., and the type 9 tree has 9 leaves. Of course the probability related to the type 6 shown in Figure 4 is the sum of the probabilities related to the types 3 and 4 shown in Figure 3, and the probabilities of the other types remain as before. (Types 4, 5, 7, 8, and 9 shown in Figure 4 have the same probabilities as types 1, 2, 5, 6, and 7 shown in Figure 3 respectively.)

Lemma 3.17. The 6 types of the tree collection shown in Figure 4 can be used as subtrees of nodes with one or two keys in order to obtain a three level tree collection.

Proof: From the trees shown in Figure 3, the ones with the same number of leaves appear as subtrees of nodes with one or two keys having the same probability, simply because they belong to the same type. □

Lemma 3.18. The two level tree collection with 6 types shown in Figure 4 can be used to form a three level 2-3 tree collection with 147 types.

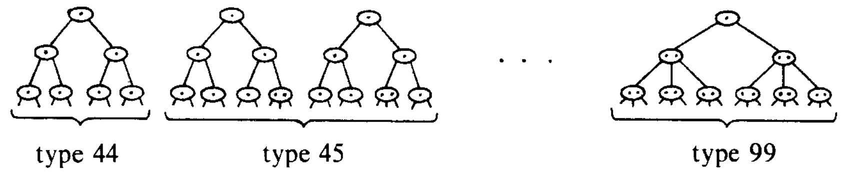
Proof: Following the notation presented in Figure 5, the 147 types of the three level tree collection are represented either as type ij ($4 \leqslant i \leqslant 9$ and $i \leqslant j \leqslant 9$) for the tree types with binary roots, or as type ijk ($4 \leqslant k \leqslant 9$, $4 \leqslant i \leqslant k$, and $4 \leqslant j \leqslant 9$) for the tree types with ternary roots. The number of tree types with binary roots is 21, and the number of tree types with ternary roots is 126, which gives a total of 147 types. □

Lemma 3.19. The transitions related to the 6 types of the tree collection shown in Figure 4 are equivalent to the transitions related to the 7 types of the tree collection shown in Figure 3 when both are used as subtrees of nodes with one or two keys in order to obtain a three level tree collection.

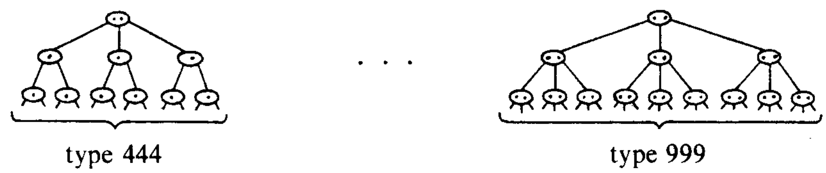
Proof: Figures 6(a) and 6(b) show the transitions related to the tree collections shown in Figure 3 and Figure 4 respectively. It is indifferent whether we use the 6 types of the tree collection shown in Figure 4 or the 7 types of the tree collection shown in Figure 3 as subtrees of nodes with one or two keys. In the case we choose the former types we have to remember that (i) the type 6 shown in Figure 6(b) is composed by types 3 and 4 shown in Figure 6(a), and (ii) from [8] that types 3 and 4 shown in Figure 6(a) occur with probabilities 5472/55937 and 7128/55937 respectively. □

Using [4] for the 147×147 transition matrix T we obtain a linear system of 147 unknowns, which was solved using an algebraic manipulation language called MAPLE, developed by Gonnet and Geddes (1981). An advantage of using such a system is that we obtain rationals instead of real numbers, avoiding computational errors. The 147 p_i's obtained contain integer numbers in the numerator and in the denominator, each one with approximately 90 digits each. Since the eigenvalues of T are 1, $-3.37 \pm 8.23i, \cdots, -30.49 \pm 2.92i$, and -32.27, the asymptotic values for $p(N)$ obtained from [6] are approximated to the $O(N^{-4.37})$.

We shall see that the analysis for the three lowest levels of 2-3 trees leads to better results for $\overline{n}(N)$, $\overline{s}(N)$, $E[s(N)]$, and exact results for $Pr\{2\ splits\}$, and

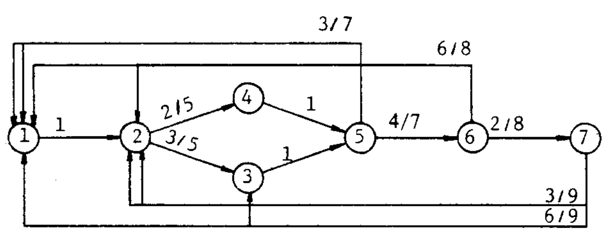


(a) Types formed by 2 height 2 subtrees under binary roots
(there are 21 types in this case)

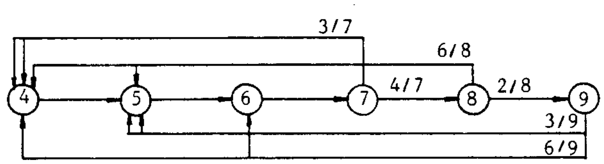


(b) Types formed by 3 height 2 subtrees under ternary roots
(there are 126 types in this case)

Figure 5 Tree collection of 2-3 trees of height 3 (type 44 is formed by two subtrees with 4 leaves each, type 45 is formed by two subtrees with 4 and 5 leaves each, etc)



(a) Transitions related to the tree collection shown in Figure 3



(b) Transitions related to the tree collection shown in Figure 4

Figure 6 Diagrams for transitions

Pr{3 *or more splits*}.

Let $nn(i)$ indicate the number of nodes of the type i tree in the tree collection shown in Figure 4.

Lemma 3.20.

$$nn(i) = 3 \qquad \text{for } 4 \leqslant i \leqslant 5$$

$$nn(6) = 3 \cdot \frac{5472}{12600} + 4 \cdot \frac{7128}{12600}$$

$$nn(i) = 4 \qquad \text{for } 7 \leqslant i \leqslant 9$$

Proof: For $i = 4, 5, 7, 8, 9$, from Figure 4 the values for $nn(i)$ are immediate. For $i = 6$, consider the two trees of type 6 shown in Figure 4. We know from [8] that the tree with 3 nodes occur with probability 5472/55937, and the tree with 4 nodes occur with probability 7128/55937. Normalizing the probabilities we obtain

$$nn(6) = 3 \cdot \frac{5472}{12600} + 4 \cdot \frac{7128}{12600} \quad \square$$

Let L_{ij} indicate the number of leaves of the type ij tree ($4 \leqslant i \leqslant 9$, $i \leqslant j \leqslant 9$) shown in Figure 5. Let L_{ijk} indicate the number of leaves of the type ijk tree ($4 \leqslant k \leqslant 9$, $4 \leqslant i \leqslant k$, $4 \leqslant j \leqslant 9$) shown in Figure 5. The proof of the following theorem is similar to the proof of theorems 3.2 and 3.10.

Theorem 3.21.

$$\left[\sum_{i=4}^{9}\sum_{j=i}^{9}\left(nn(i)+nn(j)+1+\frac{1}{2}\right)\left(\frac{p_{ij}}{L_{ij}}\right) + \sum_{k=4}^{9}\sum_{i=4}^{k}\sum_{j=4}^{9}\left(nn(i)+nn(j)+nn(k)+1+\frac{1}{2}\right)\left(\frac{p_{ijk}}{L_{ijk}}\right)\right](N+1)-\frac{1}{2}$$

$$\leqslant \bar{n}(N) \leqslant \left[\sum_{i=4}^{9}\sum_{j=i}^{9}\left(nn(i)+nn(j)+2\right)\left(\frac{p_{ij}}{L_{ij}}\right) + \sum_{k=4}^{9}\sum_{i=4}^{k}\sum_{j=4}^{9}\left(nn(i)+nn(j)+nn(k)+2\right)\left(\frac{p_{ijk}}{L_{ijk}}\right)\right](N+1)-1$$

Corollary.†

$$0.72683+\frac{0.22683}{N}+O(N^{-4.37}) \leqslant \frac{\bar{n}(N)}{N} \leqslant 0.76556-\frac{0.23444}{N}+O(N^{-4.37})$$

Lemma 3.22. $Pr\{2 \ splits\} = 0.10462+O(N^{-4.37})$

Proof: For the type 6 tree shown in Figure 4 two splits occur with probability 5472/12600, for the same reasons pointed out in the proof of Lemma 3.20. For the

† All the results of this section are presented as real numbers because the exact rationals are too long to be printed. As a curiosity, the exact lower bound on $\bar{n}(N)$ is

$$\frac{7798599314290913080528407272219562346225636732529793818193768842065373374529713557457734066}{1072960485608390776098869125251403216808988537505438482704770534002636584059387389778202 1229}$$

$= 0.72683\ 00574\ 80536 \cdots$

rest, the proof is similar to the proof of Lemma 3.11. □

Lemma 3.23. $Pr\{3 \text{ or more splits}\} = 0.07745 + O(N^{-4.37})$

Proof: Similar to the proof of Lemma 3.22. □

Lemma 3.5 leads to the following theorem:

Theorem 3.24. $$0.72683+\frac{0.22683}{N}-\frac{\lceil \log_3(N+1)\rceil}{N}+O(N^{-4.37})$$

$$\leqslant \bar{s}(N) \leqslant 0.76556-\frac{0.23444}{N}-\frac{\lfloor \log_2(N+1)\rfloor}{N}+O(N^{-4.37})$$

Lemma 3.25. $E[s(N)] \geqslant Pr\{1 \text{ split}\} + 2Pr\{2 \text{ splits}\} + 3Pr\{3 \text{ or more splits}\}$

Proof: Similar to the proof of Lemma 3.3. □

Lemma 3.26.

$$E[s(N)] \leqslant Pr\{1 \text{ split}\} + 2Pr\{2 \text{ splits}\} + Pr\{3 \text{ or more splits}\}\lfloor \log_2(N+1)\rfloor$$

Proof: Similar to the proof of Lemma 3.8. □

Lemmas 3.25 and 3.26 lead to the following theorem:

Theorem 3.27.

$$0.68810+O(N^{-4.37}) \leqslant E[s(N)] \leqslant 0.45575+0.07745 \cdots \lfloor \log_2(N+1)\rfloor+O(N^{-4.37})$$

It is important to note that the values for $\bar{n}(N)$, $\bar{s}(N)$, $E[s(N)]$, $Pr\{j \text{ splits}\}$, and $Pr\{j \text{ or more splits}\}$ for one and two level analysis can be obtained using the 147 probabilities we obtained from the three level analysis. Among other verifications, this is what we did in order to check the results of this section.

3.4. 2-3 Trees with Overflow Technique

The overflow technique was first presented by Bayer and McCreight (1972, p.183). The idea, when applied to 2-3 trees, is the following: Assume that a key must be inserted in a node already full because it contains 2 keys; instead of splitting it, we look first at its brother node on the right. If this node has only one key, a simple rearrangement of keys makes splitting unnecessary. If the right brother node is also full (or does not exist), we can look at its left brother in essentially the same way.

The object of this section is to present a second order analysis of the 2-3 tree insertion algorithm using the overflow technique as described above, applied to the lowest level of the tree only. Figure 7 shows the two level tree collection, and Table 5 shows its corresponding transition matrix.

Using [4] we obtain

$p_1 = 1584/15949$
$p_2 = 2970/15949$
$p_3 = 3600/15949$
$p_4 = 3150/15949$
$p_5 = 2000/15949$
$p_6 = 800/15949$
$p_7 = 45/389$

Since the eigenvalues of T are 1, $-5.81\pm5.96i$, $-7.51\pm2.97i$, -8.0, and -13.37, the

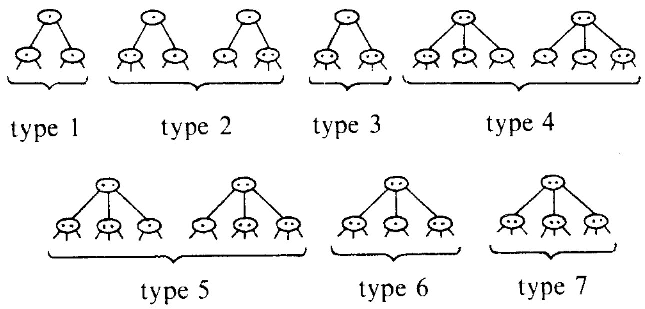


Figure 7 Tree collection of 2-3 trees of height 2 using overflow technique

−4				4·3/8		4·6/9
5	−5			5·3/8		10·3/9
	6	−6				6·6/9
		7	−7			
			8·5/7	−8		
			8·2/7		−8	
				9·5/8	9	−9

Table 5 Transition matrix corresponding to the tree collection of 2-3 trees of height 2 shown in Figure 7

asymptotic values of $p(N)$ obtained from [6] are approximated to the $O(N^{-6.81})$.

Lemma 3.1 and expression [7] lead to the following theorem:

Theorem 3.28.
$$\left[\left(3+\frac{1}{2}\right)\left(\sum_{i=1}^{3}\frac{p_i}{L_i}\right)+\left(4+\frac{1}{2}\right)\left(\sum_{i=4}^{7}\frac{p_i}{L_i}\right)\right](N+1)-\frac{1}{2}$$
$$\leqslant \overline{n}(N) \leqslant \left[4\left(\sum_{i=1}^{3}\frac{p_i}{L_i}\right)+5\left(\sum_{i=4}^{7}\frac{p_i}{L_i}\right)\right](N+1)-1$$

Corollary. $\frac{20175}{31898}+\frac{2113}{15949N}+O(N^{-6.81}) \leqslant \frac{\overline{n}(N)}{N} \leqslant \frac{11385}{15949}-\frac{4564}{15949N}+O(N^{-6.81})$

To five place decimals we have

$$0.63248+\frac{0.13248}{N}+O(N^{-6.81}) \leqslant \frac{\overline{n}(N)}{N} \leqslant 0.71384-\frac{0.28616}{N}+O(N^{-6.81}),$$

which should be compared to the

$$0.72683+\frac{0.22683}{N}+O(N^{-4.37}) \leqslant \frac{\overline{n}(N)}{N} \leqslant 0.76556-\frac{0.23444}{N}+O(N^{-4.37}),$$

which are the third order approximation of $\frac{\overline{n}(N)}{N}$ for the normal algorithm.

Lemma 3.29.

(*a*) $Pr\{0\ splits\} = \frac{9754}{15949}+O(N^{-6.81})$

(*b*) $Pr\{1\ split\} = \frac{3600}{15949}+O(N^{-6.81})$

(*c*) $Pr\{2\ or\ more\ splits\} = \frac{2595}{15949}+O(N^{-6.81})$

Proof: The proofs of (a), (b), and (c) are similar to those of Lemmas 3.3, 3.11, and 3.12, respectively. □

Lemma 3.5 leads to the following theorem:

Theorem 3.30.
$$\frac{20175}{31898}+\frac{2113}{15949N}-\frac{\lceil\log_3(N+1)\rceil}{N}+O(N^{-6.81}) \leqslant \overline{s}(N) \leqslant$$
$$\frac{11385}{15949}-\frac{4564}{15949N}-\frac{\lfloor\log_2(N+1)\rfloor}{N}+O(N^{-6.81})$$

To five place decimals we have

$$0.63248+\frac{0.13248}{N}-\frac{\lceil\log_3(N+1)\rceil}{N}+O(N^{-6.81}) \leqslant \overline{s}(N) \leqslant$$
$$0.71384-\frac{0.28616}{N}-\frac{\lfloor\log_2(N+1)\rfloor}{N}+O(N^{-6.81}),$$

which should be compared to the bounds

$$0.72683+\frac{0.22683}{N}-\frac{\lceil\log_3(N+1)\rceil}{N}+O(N^{-4.37}) \leqslant \overline{s}(N) \leqslant$$
$$0.76556-\frac{0.23444}{N}-\frac{\lfloor\log_2(N+1)\rfloor}{N}+O(N^{-4.37}),$$

which are the third order approximation of $\bar{s}(N)$ for the normal algorithm.

Lemma 3.31. $E[s(N)] \geqslant Pr\{1\ split\}+2Pr\{2\ or\ more\ splits\}$

Proof: Similar to the proof of Lemma 3.3. □

Lemma 3.32. $E[s(N)] \leqslant Pr\{1\ split\}+Pr\{2\ or\ more\ splits\}\lfloor \log_2(N+1)\rfloor$

Proof: Similar to the proof of Lemma 3.8. □

Lemmas 3.31 and 3.32 lead to the following theorem:

Theorem 3.33.

$$\frac{8790}{15949}+O(N^{-6.81}) \leqslant E[s(N)] \leqslant \frac{3600}{15949}+\frac{2595}{15949}\lfloor \log_2(N+1)\rfloor+O(N^{-6.81})$$

To five place decimals we have

$$0.55113+O(N^{-6.81}) \leqslant E[s(N)] \leqslant 0.22572+0.16270\lfloor \log_2(N+1)\rfloor+O(N^{-6.81})\ .$$

3.5. 2-3 Trees in a Concurrent Environment

A 2-3 tree node is insertion-safe if it contains only one key. When considering 2-3 trees in a concurrent environment, one possible technique to permit simultaneous access to the tree by more than one process is to lock the deepest safe node on the insertion path. (A safe node is the deepest one in a particular insertion path if there are no safe nodes below it.) The object of this section is to give a probability distribution of the depth of the deepest safe node.

3.5.1. Deepest Safe Node in 2-3 Trees with Normal Insertion Algorithm

In the following lemma we use the p's obtained in Sections 3.1, 3.2, and 3.3.

Lemma 3.34.

(*a*) $Pr\{dsn\ at\ 1^{st}\ lowest\ level\} = \frac{4}{7}$

(*b*) $Pr\{dsn\ at\ 2^{nd}\ lowest\ level\} = \frac{13788}{55937}+O(N^{-6.55})$

(*c*) $Pr\{dsn\ at\ 3^{rd}\ lowest\ level\} = 0.10462+O(N^{-4.37})$

(*d*) $Pr\{dsn\ above\ 3^{rd}\ lowest\ level\} = 0.07745+O(N^{-4.37})$

Proof: It is not difficult to see that the probability that the deepest safe node is located at j^{th} $(j \geqslant 1)$ lowest level is equal to the probability that exactly $j-1$ splits occur on the $(N+1)^{th}$ random insertion (see Lemmas 3.3, 3.11, 3.22, and 3.23 for the proof of items (*a*), (*b*), (*c*), and (*d*) respectively.) □

From Lemma 3.34 we can see that by locking the deepest safe node on the insertion path we lock at most height 3 fringe subtrees 92% of the time.

3.5.2. Deepest Safe Node in 2-3 Trees with Overflow Technique

In the following lemma we use the p's obtained in Section 3.4.

Lemma 3.35.

(*a*) $Pr\{dsn\ at\ 1^{st}\ lowest\ level\} = \frac{9754}{15949} + O(N^{-6.81})$

(*b*) $Pr\{dsn\ at\ 2^{nd}\ lowest\ level\} = \frac{3600}{15949} + O(N^{-6.81})$

(*c*) $Pr\{dsn\ above\ 2^{nd}\ lowest\ level\} = \frac{2595}{15949} + O(N^{-6.81})$

Proof: Similar to the proof of Lemma 3.34 (see Lemma 3.29 in Section 3.4 for the proof of items (*a*), (*b*), and (*c*)) □

3.6. Higher Order Analysis

Yao (1978, p. 165) predicted that an analysis for the k lowest levels would be difficult to carry out for $k=3$ and virtually impossible to carry out for $k \geqslant 4$. However, if we apply the same technique used to obtain the three level tree collection with 147 types then it might be possible to think about fourth order analysis.

In order to obtain a four level tree collection we define a 20 types three level tree collection containing trees with 8, 9, 10, ... , 27 leaves, in a way similar to the way we obtained the 6 types two level tree collection shown in Figure 4. This three level tree collection can be used to obtain a four level tree collection with 4410 types, by considering combinations of the 20 types as subtrees of nodes with one and two keys. Thus the fourth order analysis implies in the solution of a 4410×4410 linear system.

Again if we apply the same technique it is possible to obtain a five level tree collection with 148137 types, which is practically impossible to handle nowadays. Table 6 shows the sizes of the tree collections used by Yao, Brown, and in this paper, in various levels of analysis.

Analysis	Brown	Yao	Ours
First order	-	2	2
Second order	-	7	6
Third order	978	≈ 200	147
Fourth order	-	$\approx 10^6$	4410
Fifth order	-	-	148137

Table 6 Sizes of the tree collections used by Brown (1979a,p.57), Yao (1978, p.165), and in this paper

4. Partial Analysis of B-trees

According to Bayer and McCreight (1972) a *B−tree of order m* is a balanced multiway tree with the following properties: (*a*) The leaves are null nodes which all appear at the same depth. (*b*) Every node has at most $2m+1$ sons. (*c*) Every node except the root and the leaves has at least $m+1$ sons; the root is either a leaf or has at least two sons † . To insert a new key into a node that contains less than $2m$ keys we just insert it into the other keys. If the node already contains $2m$ keys, we split it into two m-keys nodes, and insert the middle key into the parent node, repeating the process again with the parent node. When there is no node above we create a new root node to insert the middle key.

Let $\bar{n}_m(N)$ be the average number of nodes in a B-tree of order m after the random insertion of N keys into the initially empty tree. Let $Pr\{j\ splits\}_m$ be the probability that j splits occur on the $(N+1)^{th}$ random insertion into a random B-tree of order m with N keys. Let $Pr\{j\ or\ more\ splits\}_m$ be the probability that j or more splits occur on the $(N+1)^{th}$ random insertion into a random B-tree of order m with N keys. Let $\bar{n}_m(N)/[N/(2m)]$ be the *storage used* by a B-tree T of order m, where $N/(2m)$ represents the number of nodes when all the nodes of T contain $2m$ keys. Let $Pr\{dsn\ at\ j^{th}\ lowest\ level\}_m$ be the probability that the deepest safe node is located at the j^{th} $(j \geq 1)$ lowest level of a random N-key B-tree of order m. Let $Pr\{dsn\ above\ j^{th}\ lowest\ level\}_m$ be the probability that the deepest safe node is located above the j^{th} lowest level of a random N-key B-tree of order m.

In section 4.1 we shall derive exact values for $Pr\{0\ splits\}_m$, $Pr\{1\ or\ more\ splits\}_m$, and bounds on $\bar{n}_m(N)$ by considering the lowest level of a random N key B-tree of order m obtained using the insertion algorithm described above. In section 4.2 we shall derive exact values for $Pr\{0\ splits\}_m$, $Pr\{1\ split\}_m$, $Pr\{1\ or\ more\ splits\}_m$, $Pr\{2\ or\ more\ splits\}_m$, and bounds on $\bar{n}_m(N)$ for an insertion algorithm for B-trees that uses an overflow technique, by considering the lowest two levels of a random N key B-tree of order m. In Section 4.3 we shall derive exact values for $Pr\{dsn\ at\ 1^{st}\ lowest\ level\}_m$ and $Pr\{dsn\ above\ 1^{st}\ lowest\ level\}_m$ for the normal insertion algorithm, and $Pr\{dsn\ at\ 1^{st}\ lowest\ level\}_m$, $Pr\{dsn\ at\ 2^{nd}\ lowest\ level\}_m$, and $Pr\{dsn\ above\ 2^{nd}\ lowest\ level\}_m$ for the insertion algorithm using an overflow technique.

Table 7 shows the summary of the results related to B-trees using the normal insertion algorithm, and Table 8 shows the summary of the results related to B-trees using an overflow technique.

† Knuth (1973, p.473) presented a slightly different definition of B-trees. In Knuth's definition every node in a B-tree of order m has at most $m-1$ keys and at least $\lceil m/2-1 \rceil$ keys. Knuth's definition considers B-trees of order $2i$, $i \geq 2$ (B-trees containing at least i keys and at most $2i-1$ keys), while the above definition does not consider such trees. However, these trees present a disadvantage: the split operation divides the node into two nodes with a different number of keys in each one, which implies that a decision about which node will contain more keys has to be taken.

	First order analysis ($N \to \infty$)
$\frac{\bar{n}_m(N)}{N}$	$\left[\frac{2m+1}{(4m^2+4m)\ln(2m+2)} + \frac{1}{2(2m+2)} + O(m^{-2}),\ \frac{1}{2m\ln(2m+2)} + \frac{1}{2(2m+2)} + O(m^{-2})\right]$
$Pr\{0\ splits\}_m$	$1 - \frac{1}{(2\ln 2)m} - (\frac{1}{8\ln 2} - \frac{1}{2})\frac{1}{(\ln 2)m^2} + O(m^{-3})$
$Pr\{1\ or\ more\ splits\}_m$	$\frac{1}{(2\ln 2)m} + (\frac{1}{8\ln 2} - \frac{1}{2})\frac{1}{(\ln 2)m^2} + O(m^{-3})$
Storage used	$\frac{1}{\ln 2} + O(m^{-1})$
$Pr\{dsn\ at\ 1^{st}\ lowest\ level\}_m$	$1 - \frac{1}{(2\ln 2)m} - (\frac{1}{8\ln 2} - \frac{1}{2})\frac{1}{(\ln 2)m^2} + O(m^{-3})$
$Pr\{dsn\ above\ 1^{st}\ lowest\ level\}_m$	$\frac{1}{(2\ln 2)m} + (\frac{1}{8\ln 2} - \frac{1}{2})\frac{1}{(\ln 2)m^2} + O(m^{-3})$

Table 7 Summary of the B-tree results

	Second order analysis ($N \to \infty$)
$\frac{\bar{n}_m(N)}{N}$	$\left[\frac{1}{2m} + (\frac{3}{8\ln 2} - \frac{1}{4})\frac{1}{m^2} + O(m^{-3}),\ \frac{1}{2m} + (\frac{3}{8\ln 2} - \frac{1}{4})\frac{1}{m^2} + O(m^{-3})\right]$
$Pr\{0\ splits\}_m$	$1 - \frac{1}{2m} - (\frac{1}{8\ln 2} - \frac{1}{4})\frac{1}{m^2} + O(m^{-3})$
$Pr\{1\ split\}_m$	$\frac{1}{2m} + (-\frac{1}{8\ln 2} - \frac{1}{4})\frac{1}{m^2} + O(m^{-3})$
$Pr\{1\ or\ more\ splits\}_m$	$\frac{1}{2m} + (\frac{1}{8\ln 2} - \frac{1}{4})\frac{1}{m^2} + O(m^{-3})$
$Pr\{2\ or\ more\ splits\}_m$	$\frac{1}{(4\ln 2)m^2} + O(m^{-3})$
Storage used	$1 + (\frac{3}{4\ln 2} - \frac{1}{2})\frac{1}{m} + O(m^{-2})$
$Pr\{dsn\ at\ 1^{st}\ lowest\ level\}_m$	$1 - \frac{1}{2m} - (\frac{1}{8\ln 2} - \frac{1}{4})\frac{1}{m^2} + O(m^{-3})$
$Pr\{dsn\ at\ 2^{nd}\ lowest\ level\}_m$	$\frac{1}{2m} + (-\frac{1}{8\ln 2} - \frac{1}{4})\frac{1}{m^2} + O(m^{-3})$
$Pr\{dsn\ above\ 2^{nd}\ lowest\ level\}_m$	$\frac{1}{(4\ln 2)m^2} + O(m^{-3})$

Table 8 Summary of the B-trees results using an overflow technique

4.1. First Order Analysis

The tree collection of B-trees of order m and height 1 contains $m+1$ types. Figure 8 shows the one level tree collection of B-trees of order $m=3$.

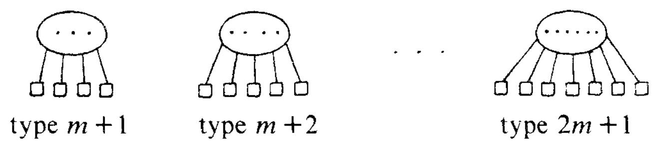


Figure 8 Tree collection of B-trees of order $m=3$ and height 1

The transition matrix T corresponding to the one level tree collection of B-trees of order m is

$$T = \begin{bmatrix} -(m+1) & & & & & 2(m+1) \\ m+2 & -(m+2) & & & & \\ & m+3 & -(m+3) & & & \\ & \circ & & \circ & & \\ & & \circ & & \circ & \\ & & & \circ & & \circ \\ & & & & 2m+1 & -(2m+1) \end{bmatrix}$$

Let H_n be the function $H_n = \sum_{i=1}^{n} \frac{1}{i}$, for $n \geqslant 1$. From [4] we have $(T-I)p(N) = 0$, and therefore

$$p_{m+1} = \frac{1}{(m+2)\left(H_{2m+2}-H_{m+1}\right)}$$

$$p_{m+2} = \frac{1}{(m+3)\left(H_{2m+2}-H_{m+1}\right)} \qquad [9]$$

$$\vdots$$

$$p_{2m+1} = \frac{1}{(2m+2)\left(H_{2m+2}-H_{m+1}\right)}.$$

Lemma 4.1. $Pr\{1 \text{ or more splits}\}_m = \dfrac{1}{(2m+2)\left(H_{2m+2}-H_{m+1}\right)}$

Proof: In the lowest level of a B-tree of order m a split occur when an insertion happens in a node with already $2m$ keys, and such nodes correspond to the type $2m+1$ of the tree collection of B-trees of order m and height 1. Thus,

$$Pr\{1 \text{ or more splits}\}_m = p_{2m+1} \quad \square$$

Lemma 4.2. $Pr\{0 \text{ splits}\}_m = 1 - \dfrac{1}{(2m+2)\left(H_{2m+2}-H_{m+1}\right)}$

Proof: Similar to the proof of lemma 4.1. □

It is well known that $H_m = \ln m + \gamma + \frac{1}{2m} - \frac{1}{12m^2} + O(m^{-4})$, where $\gamma = 0.57721...$ is Euler's constant (Knuth, 1968, p.74). Then

Corollary. $Pr\{1 \text{ or more splits}\}_m = \frac{1}{(2\ln 2)\, m} + \left[\frac{1}{8\ln 2} - \frac{1}{2}\right] \frac{1}{(\ln 2)\, m^2} + O(m^{-3})$

Let nl_m be the number of nodes at level l of an order m B-tree. Let nal_m be the number of nodes above the level l of an order m B-tree.

Lemma 4.3. $\frac{nl_m - 1}{2m} \leqslant nal_m \leqslant \frac{nl_m - 1}{m}$

Proof: Consider the level l as being the $N+1$ leaves of a B-tree with N keys. (Each leaf represents a node.) The *minimum* and the *maximum* number of nodes above the level l is obtained when each node above the level l contains $2m$ and m keys respectively. (That is $2m \times nal_m = nl_m - 1$ and $m \times nal_m = nl_m - 1$ respectively.) □

Lemma 4.3 and expression [7] lead to the following theorem:

Theorem 4.4.

$$\left(1+\frac{1}{2m}\right)\left[\sum_{i=m+1}^{2m+1} \frac{p_i}{L_i}\right](N+1) - \frac{1}{2} \leqslant \bar{n}_m(N) \leqslant \left(1+\frac{1}{m}\right)\left[\sum_{i=m+1}^{2m+1} \frac{p_i}{L_i}\right](N+1) - 1$$

Corollary.

$$\left[\frac{2m+1}{(4m^2+4m)(H_{2m+2}-H_{m+1})}\right]\left(1-\frac{1}{N}\right) - \frac{1}{2N} + O(N^{\lambda_2 - 1}) \leqslant \frac{\bar{n}_m(N)}{N} \leqslant$$

$$\left[\frac{1}{2m(H_{2m+2}-H_{m+1})}\right]\left(1-\frac{1}{N}\right) - \frac{1}{N} + O(N^{\lambda_2 - 1})$$

where $\lambda_2 < 1$.

Corollary.

$$\frac{2m+1}{(4m^2+4m)\ln(2m+2)} + \frac{1}{2(2m+2)} - \frac{1}{12(2m+2)^2} + O(m^{-4}) \leqslant \frac{\bar{n}_m(N)}{N} \leqslant$$

$$\frac{1}{2m\ln(2m+2)} + \frac{1}{2(2m+2)} - \frac{1}{12(2m+2)^2} + O(m^{-4})$$

Corollary. *Storage used* $= \frac{1}{\ln 2} + O(m^{-1})$

4.2. B-trees with Overflow Technique

One possible variant of the overflow technique presented in Section 3.4 is the following: Assume that a key must be inserted in a node already full because it contains $2m$ keys; instead of splitting it, we look first at all its brother nodes. If one of its brother nodes has less than $2m$ keys then a rearrangement of keys is performed, otherwise the split is performed. In this section we present a second order analysis of the B-tree insertion algorithm using the overflow technique as described above, applied to the lowest level of the tree only.

Any tree collection of B-trees of order m using the overflow technique described above contains $(m+1)(2m+1)$ types. Figure 9 shows the transition diagram corresponding to the two level tree collection of B-trees of order $m=2$. The transition matrix T corresponding to the two level tree collection of B-trees of order m using the overflow technique described above is shown in Table 9.

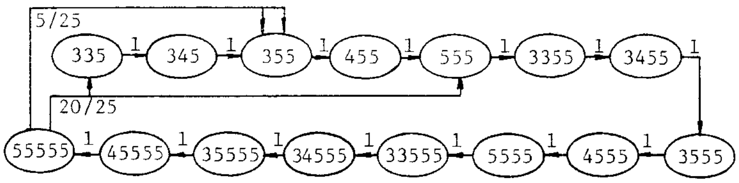


Figure 9 Transition diagram representing the two level tree collection for B-trees of order $m=2$ using overflow technique (e.g. type 335 corresponds to the height 2 type tree containing a root node with 3 descendants, the first one with 3 leaves, the second one also with 3 leaves, and the third one with 5 leaves)

$$
\begin{bmatrix}
-[(m+1)+(m+1)+(m-1)(2m+1)] & & & & 2m/(2m+1)\times[(m+1)+(m+1)+(m-1)(2m+1)] \\
\ddots & & & & \\
-[(m+1)+2m+(m-1)(2m+1)] & & & & \\
[(m+1)+m(2m+1)] & -[(m+1)+m(2m+1)] & & & 2/(2m+1)[(m+1)+m(2m+1)] \\
 & \ddots & & & \\
 & -[2m+m(2m+1)] & & & \\
 & [(m+1)\times(2m+1] & -[(m+1)\times(2m+1)] & & 2m/(2m+1)\times[(m+1)(2m+1)] \\
 & & [(m+1)+(m+1)+m(2m+1)] & & \\
 & & \ddots & & \\
 & & [2m+2m(2m+1)] & -[2m+2m(2m+1)] & \\
 & & & (2m+1)(2m+1) & -(2m+1)(2m+1)
\end{bmatrix}
$$

Table 9 Transition matrix corresponding to the tree collection of B-trees of order m and height 2 using an overflow technique

In order to obtain the vector $p(N)$ from [4], we make $p_{(2m+1)(2m+1)}=1$ † and solve for all the other $p's$. After this we normalize the $p's$ by dividing each one by their sum. Then

$$p_{(2m+1)(2m+1)} = 1$$

$$p_{(2m)+2m(2m+1)} = \frac{(2m+1)(2m+1)+1}{(2m+1)(2m+1)} \quad \left[= \frac{4m^2+4m+2}{(2m+1)(2m+1)} \right]$$

$$p_{(2m-1)+2m(2m+1)} = \frac{4m^2+4m+2}{(2m)+2m(2m+1)}$$

$$\vdots$$

$$p_{2m(2m+1)} = \frac{4m^2+4m+2}{2m(2m+1)+1}$$

$$p_{(2m)+(2m-1)(2m+1)} = \frac{4m^2+4m+2}{2m(2m+1)} \qquad [10]$$

$$\vdots$$

$$p_{(m+1)(2m+1)} = \frac{4m^2+4m+2}{(m+1)(2m+1)+1}$$

$$p_{(2m)+m(2m+1)} = \frac{1}{(m+1)(2m+1)} \left[4m^2+4m+2-\frac{2m}{2m+1}(m+1)(2m+1) \right]$$

$$= \frac{2m^2+2m+2}{(m+1)(2m+1)}$$

$$\vdots$$

$$p_{(m+1)+m(2m+1)} = \frac{2m^2+2m+2}{(m+2)+m(2m+1)}$$

$$p_{(m+1)+(2m)+(m-1)(2m+1)} = \frac{1}{(m+1)+m(2m+1)} \times$$

$$\left[2m^2+2m+2-\frac{2}{2m+1}\Big(m+1+m(2m+1)\Big) \right]$$

$$= \frac{(4m^3+2m^2+2m)/(2m+1)}{(m+1)+m(2m+1)}$$

$$\vdots$$

† $p_{(2m+1)(2m+1)}$ means $p_{(2m+1)+(2m+1)+\ldots+(2m+1)}$, where $(2m+1)$ appear $2m+1$ times. Applying this notation to the B-tree of order $m=2$ shown in Figure 9, p_{55555} is equivalent to $p_{(2m+1)(2m+1)}$, p_{335} is equivalent to $p_{(m+1)+(m+1)+(m-1)(2m+1)}$, etc.

$$p_{(m+1)+(m+2)+(m-1)(2m+1)} = \frac{(4m^3+2m^2+2m)/(2m+1)}{(m+1)+(m+3)+(m-1)(2m+1)}$$

$$p_{(m+1)+(m+1)+(m-1)(2m+1)} = \frac{(4m^3+2m^2+2m)/(2m+1)}{(m+1)+(m+2)+(m-1)(2m+1)}$$

Let S be the sum of all $p's$ above. Then

$$\begin{aligned} S = & \left(\frac{4m^3+2m^2+2m}{2m+1}\right)\left[H_{2m^2+2m+1} - H_{2m^2+m+1}\right] + \\ & (2m^2+2m+2)\left[H_{2m^2+3m+1} - H_{2m^2+2m+1}\right] + \\ & \left((2m+1)(2m+1)+1\right)\left[H_{4m^2+4m+2} - H_{2m^2+3m+1}\right] \end{aligned} \qquad [11]$$

To obtain the final probabilities all the above $p's$ have to be divided by S.

Let $\psi(Z)$ be the function $\psi(Z)=\frac{\Gamma'(Z)}{\Gamma(Z)}$ (Abramowitz and Stegun, 1972, § 6.3.1).

Lemma 4.5. $Pr\{1 \text{ or more splits}\}_m =$

$$\frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\right]\left[\psi\left(2m+2+\frac{1}{2m+1}\right)-\psi\left(m+1+\frac{1}{2m+1}\right)\right]$$

where S is as defined in [11].

Proof: $Pr\{1 \text{ or more splits}\}_m = p_{(m+1)(2m+1)}+p_{(m+2)(2m+1)}+\cdots+p_{(2m+1)(2m+1)}$

$$= \frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\sum_{i=1}^{m+1}\frac{1}{(m+i)+\frac{1}{2m+1}}\right]$$

where $$\sum_{i=1}^{m+1}\frac{1}{(m+i)+\frac{1}{2m+1}} = \psi\left(2m+2+\frac{1}{2m+1}\right)-\psi\left(m+1+\frac{1}{2m+1}\right) \quad \square$$

It is well known (Abramowitz and Stegun, 1972, § 6.3.18) that

$$\psi(m) = \ln m - \frac{1}{2m} - \frac{1}{12m^2} + O(m^{-4}) \cdot$$

Corollary. $Pr\{1 \text{ or more splits}\}_m = \frac{1}{2m} + \left[\frac{1}{8\ln 2} - \frac{1}{4}\right]\frac{1}{m^2} + O(m^{-3})$

Lemma 4.6.

$$Pr\{1 \text{ split}\}_m = \frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\right]\left[\psi\left(2m+1+\frac{1}{2m+1}\right)-\psi\left(m+1+\frac{1}{2m+1}\right)\right]$$

where S is as defined in [11].

Proof: The only difference from the proof of lemma 4.5 is that

$Pr\{1 \text{ split}\}_m = p_{(m+1)(2m+1)}+p_{(m+2)(2m+1)}+\cdots+p_{(2m)+(2m)(2m+1)}$ $\square$

Corollary. $Pr\{1 \text{ split}\}_m = \frac{1}{2m} + \left[-\frac{1}{8\ln 2} - \frac{1}{4}\right]\frac{1}{m^2} + O(m^{-3})$

Lemma 4.7. $Pr\{2 \; or \; more \; splits\}_m$ =

$$\frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\right]\left[\psi\left(2m+2+\frac{1}{2m+1}\right)-\psi\left(2m+1+\frac{1}{2m+1}\right)\right]$$

where S is as defined in [11].

Proof: $Pr\{2 \; or \; more \; splits\}_m \; = \; Pr\{1 \; or \; more \; splits\}_m - Pr\{1 \; split\}_m$ □

Corollary. $Pr\{2 \; or \; more \; splits\}_m \; = \; \frac{1}{(4\ln 2)\, m^2} + O(m^{-3})$

Lemma 4.8. $Pr\{0 \; splits\}_m$ =

$$1 - \frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\right]\left[\psi\left(2m+2+\frac{1}{2m+1}\right)-\psi\left(m+1+\frac{1}{2m+1}\right)\right]$$

where S is as defined in [11].

Proof: $Pr\{0 \; splits\}_m \; = \; 1 - Pr\{1 \; or \; more \; splits\}_m$. □

Corollary. $Pr\{0 \; splits\}_m \; = \; 1 - \frac{1}{2m} - \left[\frac{1}{8\ln 2} - \frac{1}{4}\right]\frac{1}{m^2} + O(m^{-3})$

Lemma 4.3 and expression [7] lead to the following theorem:

Theorem 4.9. $A(2m)(N+1) - \frac{1}{2} \; \leqslant \; \bar{n}_m(N) \; \leqslant \; A(m)(N+1) - 1$

where

$$A(\xi) \; = \; \frac{1}{S}\left\{\left(m+2+\frac{1}{\xi}\right)\left[\frac{p(m+1)+(m+1)+(m-1)(2m+1)}{(m+1)+(m+1)+(m-1)(2m+1)} + \frac{p(m+1)+(m+2)+(m-1)(2m+1)}{(m+1)+(m+2)+(m-1)(2m+1)} + \cdots + \frac{p(m+1)(2m+1)}{(m+1)(2m+1)}\right] + \right.$$

$$\left(m+3+\frac{1}{\xi}\right)\left[\frac{p(m+1)+(m+1)+m(2m+1)}{(m+1)+(m+1)+m(2m+1)} + \frac{p(m+1)+(m+2)+m(2m+1)}{(m+1)+(m+2)+m(2m+1)} + \cdots + \frac{p(m+2)(2m+1)}{(m+2)(2m+1)}\right] + \cdots + \left(2m+2+\frac{1}{\xi}\right)\left[\frac{p(m+1)+(m+1)+(2m-1)(2m+1)}{(m+1)+(m+1)+(2m-1)(2m+1)} + \right.$$

$$\left.\left.\frac{p(m+1)+(m+2)+(2m-1)(2m+1)}{(m+1)+(m+2)+(2m-1)(2m+1)} + \cdots + \frac{p(2m+1)(2m+1)}{(2m+1)(2m+1)}\right]\right\}$$

where S is as defined in [11].

Substituting [10] in the expression of theorem 4.9 gives:

Corollary.

$$B(2m)\left(1-\frac{1}{N}\right) - \frac{1}{2N} + O(N^{\lambda_2 - 1}) \leqslant \frac{\bar{n}_m(N)}{N} \leqslant$$

$$B(m)\left(1-\frac{1}{N}\right) - \frac{1}{N} + O(N^{\lambda_2 - 1}) \quad , \lambda_2 < 1$$

where

$$B(\xi) = \frac{1}{S}\left\{\left(m+2+\frac{1}{\xi}\right)\left[\left(\frac{4m^3+2m^2+2m}{2m+1}\right)\times\right.\right.$$

$$\left(\frac{1}{(m+1)+(m+1)+(m-1)(2m+1)}-\frac{1}{(m+1)+m(2m+1)}\right)+$$

$$\left(2m^2+2m+2\right)\left(\frac{1}{(m+1)+m(2m+1)}-\frac{1}{(m+1)(2m+1)}\right)+$$

$$\left.\left(4m^2+4m+2\right)\left(\frac{1}{(m+1)(2m+1)}\cdot\frac{1}{(m+1)(2m+1)+1}\right)\right]+$$

$$\left(4m^2+4m+2\right)\left[\frac{m+3+\frac{1}{\xi}}{(m+1)+(m+1)+m(2m+1)}-\frac{2m+2+\frac{1}{\xi}}{(2m+1)(2m+1)+1}+\right.$$

$$\left.\left.\frac{\psi\left(2m+1+\frac{1}{2m+1}\right)-\psi\left(m+2+\frac{1}{2m+1}\right)}{2m+1}\right]\right\}$$

where S is as defined in [11].

Corollary.

$$\frac{1}{2m}+\left[\frac{3}{8\ln 2}-\frac{1}{4}\right]\frac{1}{m^2}+\left[-\frac{9}{32\ln 2}+\frac{1}{8}\right]\frac{1}{m^3}+O(m^{-4}) \leqslant \frac{\overline{n}_m(N)}{N} \leqslant$$

$$\frac{1}{2m}+\left[\frac{3}{8\ln 2}-\frac{1}{4}\right]\frac{1}{m^2}+\left[-\frac{5}{32\ln 2}+\frac{1}{8}\right]\frac{1}{m^3}+O(m^{-4})$$

Corollary. *Storage used* $= 1+\left(\frac{3}{4\ln 2}-\frac{1}{2}\right)\frac{1}{m}+O(m^{-2})$

4.3. B-trees in a Concurrent Environment

A node of a B-tree of order m is insertion safe if it contains less than $2m$ keys. A safe node is the deepest one in a particular insertion path if there are no safe nodes below it. The object of this section is to derive probabilities related to the depth of the deepest safe node.

4.3.1. Deepest Safe Node in B-trees with Normal Insertion Algorithm

Lemma 4.10.

(*a*) $Pr\{dsn \text{ at } 1^{st} \text{ lowest level}\} = 1-\frac{1}{(2m+2)\left[H_{2m+2}-H_{m+1}\right]}$

(*b*) $Pr\{dsn \text{ above } 1^{st} \text{ lowest level}\} = \frac{1}{(2m+2)\left[H_{2m+2}-H_{m+1}\right]}$

Proof: Similar to the proof of lemma 3.34. □

Corollary.

$$(a)\quad Pr\{dsn\ at\ 1^{st}\ lowest\ level\} = 1-\frac{1}{(2\ln 2)\,m}-\left[\frac{1}{8\ln 2}-\frac{1}{2}\right]\frac{1}{(\ln 2)\,m^2}+O(m^{-3})$$

$$(b)\quad Pr\{dsn\ above\ 1^{st}\ lowest\ level\} = \frac{1}{(2\ln 2)\,m}+\left[\frac{1}{8\ln 2}-\frac{1}{2}\right]\frac{1}{(\ln 2)\,m^2}+O(m^{-3})$$

This analysis shows that complicated solutions for the use of B-trees in a concurrent environment are of merely academic interest, since the solution analysed in this paper will lock height 1 fringe subtrees most of the time.

4.3.2. Deepest Safe Node in B-trees with Overflow Technique

Lemma 4.11.

(*a*) $Pr\{dsn\ at\ 1^{st}\ lowest\ level\}$ =

$$1-\frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\right]\left[\psi\left(2m+2+\frac{1}{2m+1}\right)-\psi\left(m+1+\frac{1}{2m+1}\right)\right]$$

(*b*) $Pr\{dsn\ at\ 2^{nd}\ lowestlevel\}$ =

$$\frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\right]\left[\psi\left(2m+1+\frac{1}{2m+1}\right)-\psi\left(m+1+\frac{1}{2m+1}\right)\right]$$

(*c*) $Pr\{dsn\ above\ 2^{nd}\ lowest\ level\}$ =

$$\frac{1}{S}\left[\frac{(2m+1)(2m+1)+1}{2m+1}\right]\left[\psi\left(2m+2+\frac{1}{2m+1}\right)-\psi\left(2m+1+\frac{1}{2m+1}\right)\right]$$

where S is as defined in [11].

Proof: Similar to the proof of Lemma 3.34. □

Corollary.

$$(a)\quad Pr\{dsn\ at\ 1^{st}\ lowest\ level\} = 1-\frac{1}{2m}-\left[\frac{1}{8\ln 2}-\frac{1}{4}\right]\frac{1}{m^2}+O(m^{-3})$$

$$(b)\quad Pr\{dsn\ at\ 2^{nd}\ lowest\ level\} = \frac{1}{2m}+\left[-\frac{1}{8\ln 2}-\frac{1}{4}\right]\frac{1}{m^2}+O(m^{-3})$$

$$(c)\quad Pr\{dsn\ above\ 2^{nd}\ lowest\ level\} = \frac{1}{(4\ln 2)\,m^2}+O(m^{-3})$$

References

Abramowitz, M. and Stegun, I.A. *Handbook of Mathematical Functions*, (New York: Dover, 1972).

Bayer, R. and McCreight, E. ."Organization and Maintenance of Large Ordered Indexes", *Acta Informatica 1*, 3 (1972), 173-189.

Bayer, R. and Schkolnick, M. "Concurrency of Operations on B-trees", *Acta Informatica 9*, 1 (1977), 1-21.

Brown, M. "A Partial Analysis of Random Height-Balanced Trees", *SIAM Journal of Computing 8*, 1 (Feb. 1979a), 33-41.

Brown, M. "Some Observations on Random 2-3 Trees", *Information Processing Letters 9*, 2 (Aug. 1979b), 57-59.

Chvatal, V., Klarner, D.A. and Knuth, D.E. "Selected Combinatorial Research Problems", *Technical Report STAN-CS-72-292*, Computer Science Department, Stanford University, 1972.

Comer, D. "The Ubiquitous B-tree", *Computing Surveys 11*, 11 (June 1979a), 121-137.

Comer, D. "The Tree Branches", *Computing Surveys 11*, 4 (Dec. 1979b), 412.

Eisenbarth, B. and Mehlhorn, K. "Allgemeine Fringe-analysis und ihre Anwendung zur Untersuchung der Overflowtechnik bei B-Bäumen", Universität des Saarlandes, Saarbrucken, West Germany, Oct. 1980.

Gonnet, G.H. and Geddes, K.O. "MAPLE User's Manual", Department of Computer Science, University of Waterloo, Waterloo, Canada, 1981.

Guibas, L.J., McCreight, E., Plass, M. and Roberts, J. "A New Representation for Linear Lists", *9th ACM Symposium on Theory of Computing*, (1977), 49-60.

Huddleston, S. and Mehlhorn, K. "A New Data Structure for Representing Sorted Lists", Universität des Saarlandes, Saarbrucken, West Germany, Dec. 1980.

Knuth, D.E. *The Art of Computer Programming*, Vol.1 (Reading, Mass.: Addison-Wesley, 1968).

Knuth, D.E. *The Art of Computer Programming*, Vol.3 (Reading, Mass.: Addison-Wesley, 1973).

Kwong, Y.S. and Wood, D. "Approaches to Concurrency in B-trees", *Lecture Notes in Computer Science 88*, Springer-Verlag (1980), 402-413.

Rosenberg, A.L. and Snyder, L. "Time- and Space-Optimality in B-trees", *ACM Transactions on Database Systems 6*, 1 (1981), 174-193.

Yao, A. "On Random 2-3 Trees", *Acta Informatica 9*, (1978), 159-170.