Fringe analysis for parallel MacroSplit insertion algorithms in 2–3 trees * (extended abstract)

R. Baeza-Yates[1], J. Gabarró[2], and X. Messeguer[2]

[1] Departamento de Ciencias de la Computación, Universidad de Chile, Blanco Encalada 2120, Santiago, Chile. E-mail: rbaeza@dcc.uchile.cl
[2] LSI, Universitat Politècnica de Catalunya, Mòdul C5-C6, C/ Jordi Girona, 1-3, E-Barcelona 08034, Spain. E-mail: {gabarro , peypoch}@lsi.upc.es

Abstract. We extend the fringe analysis (used to study the expected behavior of balanced search trees under sequential insertions) to deal with synchronous parallel insertions on 2–3 trees. Given an insertion of k keys in a tree with n nodes, the fringe evolves following the transition matrix:

$$T_{n,k} = \left(1 + \frac{k}{n+1}\right) I + \sum_{j=0}^{k} \frac{(-1)^j}{(n+1)^j} \binom{k}{j} \begin{pmatrix} \alpha_j & -\beta_j \\ -\alpha_j & \beta_j \end{pmatrix}.$$

where the coefficients α_j and β_j take care of the precise form of the algorithm but does not depend on k or n. The derivation of this matrix uses the binomial transform recently developed by P. Poblete, J. Munro and Th. Papadakis. Due to the complexity of the preceding exact analysis, we develop also two approximations. A first one based on a simplified parallel model, and a second one based on the sequential model. These two approximated analysis prove that the parallel insertions case does not differ significantly from the sequential case, namely on the terms $O(1/n^2)$.

Keywords: Fringe analysis, Parallel algorithms, 2-3 trees, Binomial transform.

1 Introduction

One of the basic problems of managing information is the dictionary problem, where a set of keys has to be dynamically maintained. One solution to this problem are balanced search trees. One example are 2–3 trees where all leaves appear at the same depth and every node has either one key and two sons, or two keys and three sons. The exact analysis of the sequential case is still

* Partially supported by ACI-CONICYT through Catalunya-Chile Cooperation Program (DOG 2320-30.1.1997) and RITOS network (CYTED) and ESPRIT Long Term Research Project no. 20244–ALCOM IT and DGICYT under grant PB95-0787 (project KOALA) and CICIT TIC97-1475-CE.

open, but good lower and upper bounds for several complexity measures have been obtained [Yao78,EZG⁺82,BYP95] using a technique called *fringe analysis* [BY95]. This analysis studies the bottom subtrees or fringe of trees and has been applied to most search trees. We use this technique to analyze k synchronous parallel insertions in 2-3 trees for $k > 0$.

The rest of the paper is organized as follows. In section 2 we introduce the MacroSplit based synchronous parallel insertion algorithm, which is at the base of our fringe approach. In section 3, some qualitative explanations about existing insertions algorithms are given. Section 4 develops the fringe analysis giving an exact result for the transition matrix (theorem 2). The complexity of the results has forced us to address two approximations in section 5. In the first we add some assumptions to the parallel algorithm, and in the second we consider consecutive sequential insertions. Section 6 we include final remarks and future works. Finally, in the appendix we give a complete proof, using the binomial transform [PMP95], of the theorem 2.

2 MacroSplit based parallel insertion algorithm

We introduce a parallel insertion algorithm based on the idea of MacroSplit. On this algorithm an array of ordered keys $a[1\,..\,]$ is inserted into a 2-3 tree having n leaves. The MacroSplit insertions algorithm has two main successive phases.

Percolation Phase. In a top-down strategy, the set of keys to be inserted is split into several packets and these packets are routed down. Finally, these packets are attached to the leaves [PVW83,GMM96,GM97].

Reconstruction Phase. In a bottom-up phase the packets attached to the leaves are really inserted and the tree is reconstructed. This reconstruction is based in just *one unique* wave moving bottom up. First, the packets are incorporated at the bottom internal nodes of the tree. In successive steps the wave moves up, decreasing the depth one unit at each time. The evolution of this unique wave needs the usage of rules so called MacroSplit rules (see Figure 1). To define them we have several possibilities. For instance, we can take rules giving a maximum number of internal nodes holding two keys. Another possibility consists on generate a maximum number of nodes with one key.

The MacroSplit algorithm can be seen as a "height level" description of the well known parallel insertion algorithm given by W. Paul, U. Vishkin and H. Wagener in [PVW83], whose reconstruction phase has been refined (in order to avoid concurrent readings). This refinement take place splitting a MacroSplit step into several more basic steps chained together in a pipeline.

3 Qualitative behavior of insertion algorithms

In the further sections we will develop the fringe analysis of the MacroSplit insertion algorithm. Based on this analysis we can try a qualitative explanation

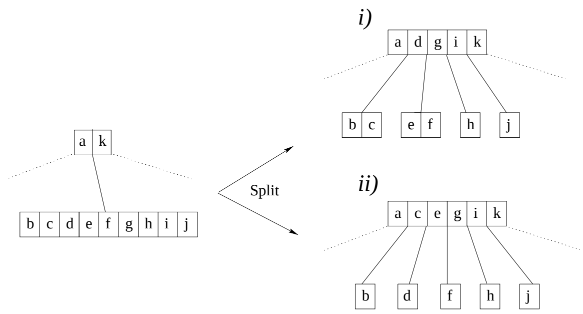

Fig. 1. We have several choices for a MacroSplits Rule. In case (i) the rule creates a maximum number of double nodes. In (ii) the rule creates the minimum number. Other intermediate strategies are also allowed.

of parallel insertion algorithms. As usual fringe analysis deals only with the distribution of the bottom insertion nodes. We will prove that in the parallel case a fraction of nodes having two leaves can be well approximated by a constant (like in the sequential case). It seems reasonable to assume that higher order fringe analysis for parallel algorithms will give close results (in the sequential case, this has been experimentally tested by R. Baeza-Yates and P. Poblete in [BYP95]).

MacroSplit algorithms. Let us assume that the 2-3 tree has n nodes and k is the number of keys to be inserted. Assume k independent of n. From the preceding remarks the expected number of levels affected by a wave is logarithmic on k. This happens because at every level it seems that a constant fraction $c \leq 1/3$ of keys will not produce further actions. The same seems to happen when $k = o(n)$.

Pipelines based algorithm. Each wave of the pipeline parallel algorithm has an expected logarithmic life time on k because the time spent at each level is constant. Then we can take advantage of this fact in the following two senses:

1. Assume that we have p processors and k keys with $p < k$. Then the first wave starts with p processors managing p keys. When the second wave starts, the first one only has a part cp of active processors because $1 - cp$ ones have inserted its key and are now free. Then the second wave starts with $1 - cp$ processors and so on. Therefore, the expected number of processors needed to insert the k keys can be reduced to $O(k/\log k)$.
2. Assume now that we have $p > k$ processors and that each wave starts with k processors. The second wave only needs ck new processors because the remainder $1 - ck$ are those left free by the first wave, and so on. Therefore, a stationary process of pipelined waves, where each of them inserts k keys, can be supported with $k = O(p)$ processes.

Much more research has to be done in order to prove mathematically the preceding assertions. To justify them let us start with a precise fringe analysis.

4 Fringe analysis for parallel insertions

The fringe of a tree is composed by the subtrees on the last level. A node with one key is designated x node, and a node with two keys is an y node. Note that bottom nodes separate leaves into $1-type$ leaves if their parents are x nodes; otherwise, $2-type$ leaves. When a new element falls in a node of type x, is transformed in a node of type y. Otherwise, a node of type y is split into two new x nodes.

Let X_t and Y_t be the random variables associated to the number of $1-type$ leaves and $2-type$ leaves respectively at the step t. We assume $X_t + Y_t = n+1$ being n the number of keys of the tree. The expected number of leaves (conditioned to the random insertion of one key) at the step t can be modeled by [EZG+82]:

$$\begin{pmatrix} E(X_{t+1} \mid 1) \\ E(Y_{t+1} \mid 1) \end{pmatrix} = T_{n,1} \begin{pmatrix} E(X_t \mid 1) \\ E(Y_t \mid 1) \end{pmatrix}$$

where $T_{n,1}$ is the transition matrix

$$T_{n,1} = \left(1 + \frac{1}{n+1}\right) I + \frac{1}{n+1} H \quad \text{being} \quad I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad H = \begin{pmatrix} -3 & 4 \\ 3 & -4 \end{pmatrix}.$$

The probability that a random chosen leaf belongs to the $i-type$ is:

$$P_i = \frac{\text{Expected number of leaves of } i-type}{\text{Number of leaves of the tree}}.$$

Then the insertion process implies the stationary values of the probability for $P_1 = 4/7$ and $P_2 = 3/7$. More details can be found in [BY95].

We consider now that k keys are in a random parallel manner inserted into a tree of size n with X_t (respectively Y_t) leaves of $1-type$ ($2-type$). The expected values of the random variable X_{t+1} and Y_{t+1} after the insertions depends on the expected values of X_t and Y_t only. This means that the current value depends on the history of the process only through the most recent value. Therefore we deal with a Markov chain and the evolution can be analyzed through a recurrence of the conditional expectations given by

$$\begin{pmatrix} E(X_{t+1} \mid k) \\ E(Y_{t+1} \mid k) \end{pmatrix} = T_{n,k} \begin{pmatrix} E(X_t \mid k) \\ E(Y_t \mid k) \end{pmatrix}.$$

where $T_{n,k}$ is as before the transition matrix.

The transition matrix is computed by considering a uniform distribution of keys and the transformation of the bottom nodes. Let us explain this last point. Assume that k keys have been inserted, then, at most, k keys can reach a node. If the node stores more than two keys, it must be split. Table 1 shows some splits of x and y nodes; for instance, the first row shows the x node transformation into an y and the y node transformation into xx nodes under the one key insertion, and the fourth row shows how x and y nodes with new four keys can be split

k	x node	y node
1	y	xx
2	xx	xy
3	xy	xxx or yy
4	xxx or yy	xxy
5	xxy	$xxxx$ or xyy
6	$xxxx$ or xyy	$xxxy$ or yyy

Table 1. Transformation of x and y bottom node once k keys reach them

n	1	2	3	4	5	6	7	8	9	10	31	41	51
$k=1$	1	0	1	.4	.6	.5714=4/7	$\cdots$						
$k=2$	1		1		.5		.5625		5609		.5689	.5702	.5706
$k=3$	1			.4			.466			.5394	.5656	.5687	.5697

Table 2. Probability of $1-type$ leaves once k keys have been inserted repeatedly

(in some cases there are different possibilities). Note that y node transformation when k keys reach it is the same as the x node transformation when $k+1$ keys reach it.

The columns of Table 2 show the experimental evolution of the probalility values of $1-type$ leaves (the initial tree had one x node). Note that these values rapidly converge to 4/7, therefore this value seem to be an upper limit in the parallel case. The same table shows that the parallel insertion determines a leaves distribution different than those determined by sequential insertions.

We develop next the parallel insertion of two keys. We follow the same technique applied before to sequential insertions [EZG$^+$82].

4.1 Parallel insertion of two keys

Assume that we have a tree with n keys and X_t, Y_t leaves of each type with $X_t + Y_t = n+1$. We insert randomly in parallel two additional keys. Then, the expected number of leaves is given by

$$\begin{pmatrix} E(X_{t+1} \mid 2) \\ E(Y_{t+1} \mid 2) \end{pmatrix} = T_{n,2} \begin{pmatrix} E(X_t \mid 2) \\ E(Y_t \mid 2) \end{pmatrix}.$$

These two keys fall through the tree until they reach bottom nodes. As at most two keys can reach the same bottom node, we have no election in the split, *i.e.* the transformation of bottom nodes is unique (second row of table 1). Both keys can be either at the same bottom node or at different bottom nodes, and in each case bottom nodes can be of type x or y. Let $P(x,x)$ be the probability that both keys reach the same x node, $P(x_1,x_2)$ the probability to reach different x nodes and so on for the remainder probabilities $P(x,y)$ and $P(y_1,y_2)$. We denote the generic case as $P(\cdot,\cdot)$, being $(.,.)$ the generic pair of nodes accessed.

$(\cdot,\cdot)$	$P(\cdot,\cdot)$	$E(X_{t+1}\|X_t,Y_t,2,(.,.))$	$E(Y_{t+1}\|X_t,Y_t,2,(.,.))$
(x,x)	$\frac{X_t}{n+1}\frac{2}{n+1}$	X_t+2	Y_t
(x_1,x_2)	$\frac{X_t}{n+1}\frac{X_t-2}{n+1}$	X_t-4	Y_t+6
(x,y)	$2\frac{X_t}{n+1}\frac{Y_t}{n+1}$	X_t+2	Y_t
(y,y)	$\frac{Y_t}{n+1}\frac{3}{n+1}$	X_t+2	Y_t
(y_1,y_2)	$\frac{Y_t}{n+1}\frac{Y_t-3}{n+1}$	X_t+8	Y_t-6

Table 3. Parallel insertion of two keys

The expected number of $1-type$ leaves is:

$$E(X_{t+1}|X_t,Y_t,2)=\sum_{(\cdot,\cdot)}P(\cdot,\cdot)\ E(X_{t+1}|X_t,Y_t,2,(.,.))$$

being $E(X_{t+1}|X_t,Y_t,2,(.,.))$ the expected number of $1-type$ leaves when two keys reach node $(\cdot,\cdot)$ conditioned to initial expected number of leaves X_t and Y_t. For instance, if both keys reach different x nodes then it holds

$$P(x_1,x_2)=\frac{X_t}{n+1}\frac{X_t-2}{n+1}.$$

The expectations of $1-type$ leaves is $E(X_{t+1}|X_t,Y_t,2,(x1,x2))=X_t-4$. Table 3 contains the other values. Moreover $E(X_{t+1}\mid 2)=E(E(X_{t+1}\mid X_t,Y_t,2))$

Lemma 1. *The transition matrix $T_{n,2}$ is*

$$\left(1+\frac{2}{n+1}\right)I+\frac{2}{n+1}H+\frac{-1}{(n+1)^2}\begin{pmatrix}-12 & 18\\ 12 & -18\end{pmatrix}\quad \textit{being}\quad H=\begin{pmatrix}-3 & 4\\ 3 & -4\end{pmatrix}$$

Proof. We compute the conditional expectation only for X_{t+1} (the Y_{t+1} term has a similar development). Then $E(X_{t+1}|X_t,Y_t,2)$ is:

$$\begin{aligned}
&\sum_{(\cdot,\cdot)}P(\cdot,\cdot)\ E(X_{t+1}|X_t,Y_t,2,(.\ ,.))\\
=&\frac{1}{(n+1)^2}\Big(2X_t(X_t+2)+X_t(X_t-2)(X_t-4)+2X_tY_t(X_t+2)\\
&\qquad+3Y_t(X_t+2)+Y_t(Y_t-3)(X_t+8)\Big)\\
=&X_t+\frac{1}{(n+1)^2}\Big(12X_t-4X_t^2+4X_tY_t+8Y_t^2-18Y_t)\Big)\\
=&\left(1-\frac{4}{n+1}+\frac{12}{(n+1)^2}\right)X_t+\left(\frac{8}{n+1}-\frac{18}{(n+1)^2}\right)Y_t
\end{aligned}$$

This concludes the proof. []

4.2 Computation of the transition matrix of the k keys insertion

Assume that we insert $k \geq 1$ additional keys on a tree with X_t leaves of $1-type$ and Y_t of $2-type$. We select one key and denote it κ. This key can reach a bottom node x or y; the first case is denoted χ case and the second one γ case. Then the expectations of $1-type$ leaves after the insertion are given by

$$E(X_{t+1}|X_t, Y_t, k) = P(\chi)\ E(X_{t+1}|X_t, Y_t, k-1, \chi) + P(\gamma)\ E(X_{t+1}|X_t, Y_t, k-1, \gamma)$$

where $E(X_{t+1}|X_t, Y_t, k-1, \chi)$ is the expected number of $1-type$ leaves once k keys have been inserted and one of them, κ, has reach an x node. Similarly for $E(X_{t+1}|X_t, Y_t, k-1, \gamma)$. Clearly

$$P(\chi) = \frac{X_t}{n+1} \quad \text{and} \quad P(\gamma) = \frac{Y_t}{n+1}.$$

If key κ reaches an x bottom node, then the probability that i keys of the remainder $k-1$ ones reach the same node is

$$b\left(i, k-1, \frac{2}{n+1}\right) = \binom{k-1}{i} \left(\frac{2}{n+1}\right)^i \left(1 - \frac{2}{n+1}\right)^{k-1-i}.$$

Then the expected values can be defined recursively as:

$$\begin{aligned}
&E(X_{t+1}|X_t, Y_t, k-1, \chi) = \\
&\qquad \sum_{i=0}^{k-1} b(i, k-1, \frac{2}{n+1}) \Big(E(X_{t+1}|X_t - 2, Y_t, k-1-i) + \mathcal{X}_{x,i+1} \Big) \\
&E(X_{t+1}|X_t, Y_t, k-1, \gamma) = \\
&\qquad \sum_{i=0}^{k-1} b(i, k-1, \frac{3}{n+1}) \Big(E(X_{t+1}|X_t, Y_t - 3, k-1-i) + \mathcal{X}_{y,i+1} \Big).
\end{aligned}$$

The term $\mathcal{X}_{x,i+1}$ is the number of $1-type$ leaves after the insertion of $i+1$ keys into an x node. In the same way, the term $\mathcal{X}_{y,i+1}$ is the number of $1-type$ leaves after the insertion of $i+1$ keys into an y node (For $2-type$ leaves we have $\mathcal{Y}_{x,i+1}$ and $\mathcal{Y}_{y,i+1}$). For instance, the second row of table 1 shows that $\mathcal{X}_{x,2} = 4$ and $\mathcal{X}_{y,2} = 2$.

Theorem 2. *The expected number of $1-type$ and $2-type$ leaves after the random insertion of k keys into a tree with X_{t+1} leaves of $1-type$ and Y_{t+1} of $2-type$ are given by*

$$\begin{pmatrix} E(X_{t+1} \mid k) \\ E(Y_{t+1} \mid k) \end{pmatrix} = T_{n,k} \begin{pmatrix} E(X_t \mid k) \\ E(Y_t \mid k) \end{pmatrix}.$$

with $T_{n,k}$ is the transition matrix

$$T_{n,k} = \left(1 + \frac{k}{n+1}\right) I + \sum_{j=0}^{k} \frac{(-1)^j}{(n+1)^j} \binom{k}{j} \begin{pmatrix} \alpha_j & -\beta_j \\ -\alpha_j & \beta_j \end{pmatrix},$$

where

$$\alpha_j = -2^{j-1} \sum_{i=0}^{j} (-1)^i \binom{j}{i} \mathcal{Y}_{x,i} \qquad \textit{and} \qquad \beta_j = -3^{j-1} \sum_{i=0}^{j} (-1)^i \binom{j}{i} \mathcal{X}_{y,i}.$$

The proof is given in the appendix.

From this transition matrix and using the fact that the probabilities can be defined as

$$\overrightarrow{P_{t+1,k}} = \left(\frac{E(X_{t+1} \mid k)}{n+1+k}, \frac{E(Y_{t+1} \mid k)}{n+1+k}\right),$$

it is possible to have a recurrence in one variable, obtaining that for constant k and asymptotically in the number of keys n, $\overrightarrow{P_{t+1}} = [4/7, 3/7] + O(k/n)$. The above seems to be true for any k of $o(n)$.

5 Approximated Analysis

Motivated by the complexity of the exact analysis of the generic case of k insertions, we present two approximated analysis. The first one approaches the distribution with a binomial. The second approximation considers k sequential insertions. The two approximations give good results for $n >> k$.

Binomial approximation. Let X_t, Y_t be the current number of leaves. Assume that r keys reach an x bottom node and $k - r$ keys and y bottom node with probability $\binom{k}{r} p^r q^{k-r}$ being $p = \frac{X_t}{n+1}$ and $q = \frac{Y_t}{n+1}$. Then, the new state is determined by

$$\begin{pmatrix} X_{t+1} \\ Y_{t+1} \end{pmatrix} = \begin{pmatrix} X_t \\ Y_t \end{pmatrix} + \binom{k}{r} p^r q^{k-r} \left[r \begin{pmatrix} -2 \\ 3 \end{pmatrix} + (k-r) \begin{pmatrix} 4 \\ 3 \end{pmatrix} \right].$$

Note that $E(p) = E(\frac{X_t}{n+1}) = p_1(n)$ and $E(q) = p_2(n)$.

Lemma 3. *It holds $E(X_{t+1}) = E(X_t) - 6kp_1(n) + 4k$ and for $n \geq 6$, $E(X_t) = \frac{4}{7}(n+1)$.*

Proof. $E(X_{t+1}) = E(E(X_{t+1} \mid X_t, Y_t)) = E\left(\sum_{r=0}^{k} (x_n - 6r + 4k) \binom{k}{r} p^r q^{k-r}\right)$. $\square$

The transition matrix in the binomial approximation is:

$$T_{n,k}^{bin} = \left(1 + \frac{k}{n+1}\right) I + \frac{k}{n+1} H \quad \text{being} \quad H = \begin{pmatrix} -3 & 4 \\ 3 & -4 \end{pmatrix}$$

Lemma 4. *It holds*

$$Var(X_{t+1}) = \Big(1 - \frac{12k}{n+1}\Big)\, Var(X_t) + \frac{6^2 k(k-1)}{(n+1)^2}\, Var(X_t) + 6^2 k p_1(n) p_2(n),$$

and for $n \geq 6$ *the asymptotic expression of* $Var(X_t)$ *is*

$$\frac{12}{13}\Big(\frac{6}{7}\Big)^2 (n+1)\Big(1 + \frac{6^2}{12}\Big(\frac{k-1}{n+1}\Big) + \frac{6^4}{12 \cdot 11}\Big(\frac{k-1}{n+1}\Big)^2 + \frac{6^4 (k-1)^2 (35k-36)}{12 \cdot 11 \cdot 10 \cdot (n+1)^3} + O(n^{-4})\Big).$$

This lemma can be proved by usual techniques. Using this approximation, the variance of the parallel insertions has the same first order term of the sequential one [BP85].

Sequential approximation. Like the transition matrix can be written as

$$T_{n,1} = \left(1 + \frac{1}{n+1}\right) I + \frac{1}{n+1}\begin{pmatrix} -3 & 4 \\ 3 & -4 \end{pmatrix} = \frac{n+2}{n+1}\left(I + \frac{1}{n+2} H\right),$$

the transition matrix to insert sequentially k keys (one after another), defined by the composition $T_{seq}(n,k) = T_{n+k-1,1} \cdots T_{n,1}$, is equal to

$$T_{seq}(n,k) = \frac{n+k+1}{n+1} \cdot \left(I + \frac{1}{n+k+1} H\right)\left(I + \frac{1}{n+k} H\right) \cdots \left(I + \frac{1}{n+2} H\right).$$

Lemma 5. *The transition matrix* $T_{seq}(n,k)$ *is (with* $c_{k,1} = k$*):*

$$\left(1 + \frac{k}{n+1}\right) I + \frac{c_{k,1}}{n+1} H + \frac{c_{k,2}}{(n+k)(n+1)} H + \frac{c_{k,3}}{(n+k)(n+k-1)(n+1)} H + \cdots$$

This expression allow us to guess the form of the transition matrix for the parallel case

$$T_{n,k} = T_{seq}(n,k) + I \cdot O\left(\frac{1}{n^2}\right).$$

Therefore, parallelizing the insertions only changes second order terms with respect to the sequential case.

6 Final Remarks and Future Work

Our results show that the parallel insertion of a constant number of keys does not differ significantly from the sequential case. This result is intuitive, although we have seen that was not easy to prove. We have analyzed a parallel and sequential approximations, and the two cases differs from the exact analysis in the second order term, being equal the first terms.

Our analysis can be also applied to AVL trees and other balanced search trees with minor changes (that is, the analysis of the fringe). Further work implies the use of our results to do a better performance study of distributed parallel algorithms, as shown in section 3.

References

[BP85] A. Bagchi and A. K. Pal. Asymptotic normality in the generalized Polya-Eggenberger urn model, with an application to computer data structures. *Siam J. Alg. Disc. Meth.*, 6(3):394–405, 1985.

[BY95] R.A. Baeza-Yates. Fringe analysis revisited. *ACM Computing Surveys*, 27(1):109–119, 1995.

[BYP95] R.A. Baeza-Yates and Poblete P.V. Higher-order analysis of 2–3 trees. *International Journal of Foundations of Computer Science*, 6(1):1–10, 1995.

[EZG+82] B. Eisenbarth, N. Ziviani, G.H. Gonnet, K. Mehlhorn, and D. Wood. The theory of fringe analysis and its application to 2–3 trees and B-trees. *Information and Control*, 55(1-3):125–174, 1982.

[GM97] J. Gabarró and X. Messeguer. A unified approach to concurrent and parallel algorithms on balanced data structures. In IEEE, editor, *Proc. of XVII International Conference of the Chilean Computer Society*, 1997.

[GMM96] J. Gabarró, C. Martínez, and X. Messeguer. A design of a parallel dictionary using skip lists. *Theoretical Computer Science*, (158):1–33, 1996.

[PMP95] P.V. Poblete, J.I. Munro, and T. Papadakis. The binomial transform and its application to the analysis of skip lists. In *ESA 95*, pages 1–10. Springer Verlag, 1995.

[PVW83] W. Paul, U. Vishkin, and H. Wagener. Parallel dictionaries on 2–3 trees. In J. Díaz, editor, *Proc. 10th International Colloquium on Automata, Programming and Languages, LNCS 154*, pages 597–609. Springer-Verlag, 1983. Also appeared as "Parallel computation on 2–3 trees" in RAIRO Informatique Théorique, pages 397-404, 1983.

[Yao78] A. Yao. On random 2–3 trees. *Acta Informatica*, 9(159):159–170, 1978.

A Proof of the theorem

Before to address the proof of the main theorem we introduce some lemmas and functions. First, an easy lemma

Lemma 6. *For any integer $r > 0$ it holds*

$$\sum_{i=0}^{k-1} b\left(i, k-1, \frac{r}{v+w}\right)\left(1 + \frac{k-1-i}{n+1+r}\right) = 1 + \frac{k-1}{n+1}.$$

Second, we recall from [PMP95] the *binomial transform* $\mathcal{B}$. Let $(F_i)_{n\geq 0}$ be a sequence of real numbers, then the binomial transform of F_i is

$$\mathcal{B}_k F_i = \sum_{i=0}^{k} (-1)^i \binom{k}{i} F_i.$$

For properties of $\mathcal{B}$ refer the reader to [PMP95]. We introduce a prticular *double binomial transform*. Let p be a real number and j a positive integer, we define

$$\mathcal{B}\mathcal{B}_k\left[j, p, F_i\right] = \mathcal{B}_k\left(p^j \mathcal{B}_j F_i\right)$$

Lemma 7. *It holds*

$$
\begin{aligned}
(i) \quad & \sum_{\ell=0}^{k} b(\ell,k,p)\, F_\ell = \mathcal{BB}_k\left[\ell,p,F_i\right] \\
(ii) \quad & \mathcal{BB}_k\left[\ell,p_1,\mathcal{BB}_\ell\left[j,p_2,F_i\right]\right] = \mathcal{BB}_k\left[\ell,p_1p_2,F_i\right] \\
(iii) \quad & \sum_{\ell=0}^{k} b(\ell,k,p_1)\, \mathcal{BB}_{k-\ell}\left[j,p_2,F_i\right] = \mathcal{BB}_k\left[\ell,(1-p_1)p_2,F_i\right] \\
(iv) \quad & \mathcal{BB}_k\left[j+1,p,F_i\right] = \mathcal{BB}_k\left[j,p,F_i\right] - \mathcal{BB}_{k+1}\left[j,p,F_i\right] \\
(v) \quad & \mathcal{BB}_k\left[j,p,F_{i+1}\right] = \mathcal{BB}_k\left[j,p,F_i\right] - \frac{1}{p}\mathcal{BB}_k\left[j+1,p,F_i\right]
\end{aligned}
$$

Proof. (i) We apply the property of the binomial transform $\sum_{\ell=0}^{k} b(\ell,k,p)F_\ell = \mathcal{B}_k(p^\ell \mathcal{B}_\ell F_i)$. (ii) We apply the property of the binomial transform which simplifies the composition of two binomial transforms to the identity. (iii) The addition is equal to

$$
\sum_{\ell=0}^{k} b(\ell,k,1-p_1)\, \mathcal{BB}_\ell\left[j,p_2,F_i\right] = \mathcal{BB}_k\left[\ell,1-p_1,\mathcal{BB}\left[j,p_2,F_i\right]\right]
$$

by (i), by (ii) we obtain the desired result. (iv) and (v) By applying the property of binomial transform $\mathcal{B}_\ell F_{i+1} = \mathcal{B}_\ell F_i - \mathcal{B}_{\ell+1}F_i$. □

We address now the proof of the main theorem. Note that the coefficients α and β can be viewed as $\alpha_j = -2^{j-1}\mathcal{B}_j\mathcal{Y}_{x,i}$ and $\beta_j = -3^{j-1}\mathcal{B}_j\mathcal{X}_{y,i}$. Then the first row of the transition matrix becomes

$$
\begin{aligned}
\left(T_{n,k}^{1,1}\; T_{n,k}^{1,2}\right) &= \left(1+\tfrac{k}{n+1} - \tfrac{1}{2}\mathcal{B}_k\left[\left(\tfrac{2}{n+1}\right)^j \mathcal{B}_j\mathcal{Y}_{x,i}\right]\; \tfrac{1}{3}\mathcal{B}_k\left[\left(\tfrac{3}{n+1}\right)^j \mathcal{B}_j\mathcal{X}_{y,i}\right]\right) \\
&= \left(1+\tfrac{1}{n+1} - \tfrac{1}{2}\mathcal{BB}_k\left[j,\tfrac{2}{n+1},\mathcal{Y}_{x,i}\right]\; \tfrac{1}{3}\mathcal{BB}_k\left[j,\tfrac{3}{n+1},\mathcal{X}_{y,i}\right]\right)
\end{aligned}
$$

Proof. We prove the theorem by induction on k. For $k=1$ we have

$$
\begin{aligned}
E(X_{t+1}|X_t,Y_t,1) &= \left(T_{n,1}^{1,1}\; T_{n,1}^{1,2}\right)\begin{pmatrix} X_t \\ Y_t \end{pmatrix} \\
&= \left(1+\frac{1}{n+1} - \frac{1}{2}\mathcal{BB}_1\left[j,\frac{2}{n+1},\mathcal{Y}_{x,i}\right]\right)\cdot X_t + \frac{1}{3}\mathcal{BB}_1\left[j,\frac{3}{n+1},\mathcal{X}_{y,i}\right]\cdot Y_t.
\end{aligned}
$$

As the values of the double transform are $6/(n+1)$ and $12/(n+1)$ respectively, the theorem holds.

We assume that the theorem holds for values smaller than k, then the recurrence becomes

$$E(X_{t+1}|X_t, Y_t, k) = \frac{X_t}{n+1}\sum_{i=0}^{k-1} b\left(i, k-1, \frac{2}{n+1}\right)\left(\left\{1 + \frac{k-1-i}{n+1-2}\right.\right.$$
$$\left.\left. -\frac{1}{2}\mathcal{BB}_{k-1-i}\left[j, \frac{2}{n+1-2}, \mathcal{Y}_{x,i}\right]\right\}(X_t - 2) + \frac{1}{3}\mathcal{BB}_{k-1-i}\left[j, \frac{3}{n+1-2}, \mathcal{X}_{y,i}\right]Y_t + \mathcal{X}_{x,i+1}\right)$$
$$+\frac{Y_t}{n+1}\sum_{i=0}^{k-1} b\left(i, k-1, \frac{3}{n+1}\right)\left(\left\{1 + \frac{k-1-i}{n+1-3} - \frac{1}{2}\mathcal{BB}_{k-1-i}\left[j, \frac{2}{n+1-3}, \mathcal{Y}_{x,i}\right]\right\}X_t\right.$$
$$\left. + \frac{1}{3}\mathcal{BB}_{k-1-i}\left[j, \frac{3}{n+1-3}, \mathcal{X}_{y,i}\right](Y_t - 3) + \mathcal{X}_{y,i+1}\right)$$

Note that $\mathcal{X}_{x,i+1} = i + 3 - \mathcal{Y}_{x,i+1}$. By applying lemma 6 on the first term, lemma 7 (v) on double binomial terms and lemma 7 (i) on $\mathcal{X}_{y,i+1}$ and $\mathcal{Y}_{x,i+1}$ terms, we obtain

$$\frac{X_t}{n+1}\left(\left\{1 + \frac{k-1}{n+1} - \frac{1}{2}\mathcal{BB}_{k-1}\left[j, \frac{2}{n+1}, \mathcal{Y}_{x,i}\right]\right\}(X_t - 2)\right.$$
$$\left. +\frac{1}{3}\mathcal{BB}_{k-1}\left[j, \frac{3}{n+1}, \mathcal{X}_{y,i}\right]Y_t + \frac{2(k-1)}{n+1} + 3 - \mathcal{BB}_{k-1}\left[j, \frac{2}{n+1}, \mathcal{Y}_{x,i+1}\right]\right)$$
$$+\frac{Y_t}{n+1}\left(\left\{1 + \frac{k-1}{n+1} - \frac{1}{2}\mathcal{BB}_{k-1}\left[j, \frac{2}{n+1}, \mathcal{Y}_{x,i}\right]\right\}X_t\right.$$
$$\left. + \frac{1}{3}\mathcal{BB}_{k-1}\left[j, \frac{3}{n+1}, \mathcal{X}_{y,i}\right](Y_t - 3) + \mathcal{BB}_{k-1}\left[j, \frac{3}{n+1}, \mathcal{X}_{y,i+1}\right]\right)$$

We separate the terms of X_t and Y_t

$$\left\{1 + \frac{k-1}{n+1} - \frac{1}{2}\mathcal{BB}_{k-1}\left[j, \frac{2}{n+1}, \mathcal{Y}_{x,i}\right]\right\}X_t$$
$$\frac{X_t}{n+1}\left\{1 + \mathcal{BB}_{k-1}\left[j, \frac{2}{n+1}, \mathcal{Y}_{x,i}\right] - \mathcal{BB}_{k-1}\left[j, \frac{2}{n+1}, \mathcal{Y}_{x,i+1}\right]\right\}$$
$$+\frac{1}{3}\mathcal{BB}_{k-1}\left[j, \frac{3}{n+1}, \mathcal{X}_{y,i}\right]Y_t + \frac{Y_t}{n+1}\left\{-\mathcal{BB}_{k-1}\left[j, \frac{3}{n+1}, \mathcal{X}_{y,i}\right] + \mathcal{BB}_{k-1}\left[j, \frac{3}{n+1}, \mathcal{X}_{y,i+1}\right]\right\}$$

Finally, by applying lemma 7 (v) we obtain

$$\left(1 + \frac{k}{n+1} - \frac{1}{2}\mathcal{BB}_k\left[j, \frac{2}{n+1}, \mathcal{Y}_{x,i}\right]\right)X_t + \frac{1}{3}\mathcal{BB}_k\left[j, \frac{3}{n+1}, \mathcal{X}_{y,i}\right]Y_t.$$