Random Records and Cuttings in Split Trees

Cecilia Holmgren

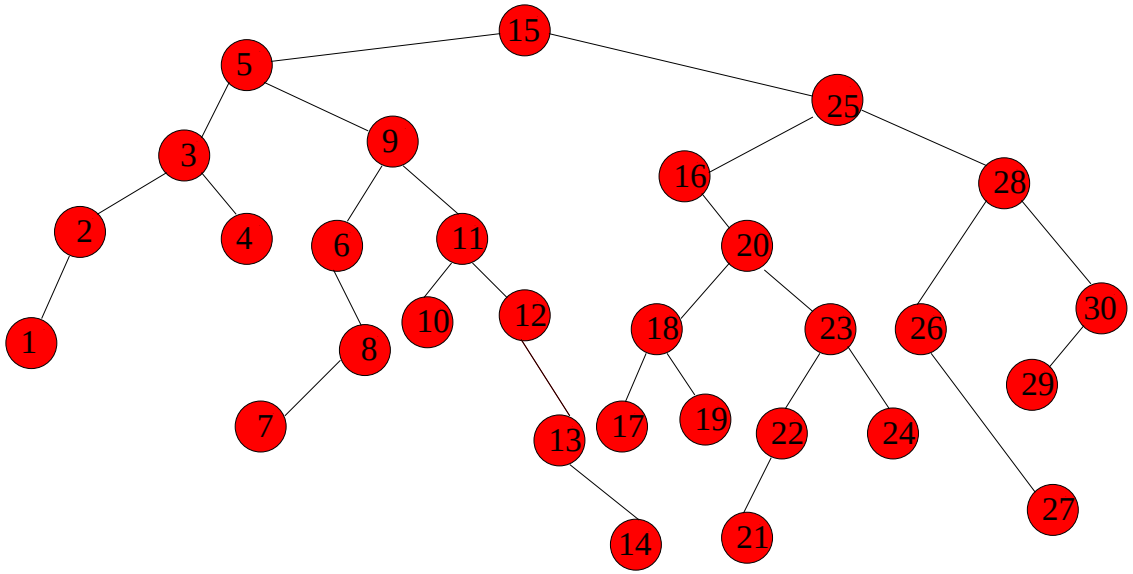
Uppsala University, Sweden

INRIA, Paris, 05 October 2009

Aim of Study

- To find the **asymptotic distribution of the number of records in random split trees**. (This number is equal in distribution to the number of cuts needed to eliminate this type of tree.)

The Binary Search Tree is an Example of a Split Tree



- Each vertex is associated with a key number, drawn from some set with n ordered numbers. Only the order relations of the keys are important. The first key is added to the root.
- Each new key is drawn from the remaining numbers and is recursively added to subtrees by comparing it with the current root's key; it is added to the left child if it is smaller and to right child if it is larger.

The Binary Search Tree (continued)

- Since the rank of the root's key is equally likely to be $\{1, 2, \ldots, n\}$, the size of its left subtree $\stackrel{d}{=} \lfloor nU \rfloor$, where U is a uniform $U(0, 1)$ r.v..
- All subtree sizes can be explained in this manner by associate each node with an independent uniform r.v. U_v. If a subtree rooted at v has size V, the size of its left subtree is $\stackrel{d}{=} \lfloor VU_v \rfloor$.
- Thus, given all U_v's the subtree size for a vertex v at depth k is close to

$$nU_1U_2 \ldots U_k,$$

where $U_i,\ i \in \{1, \ldots, k\}$ are $U(0, 1)$ r.v..

The M-ary Search Trees are Examples of Split Trees

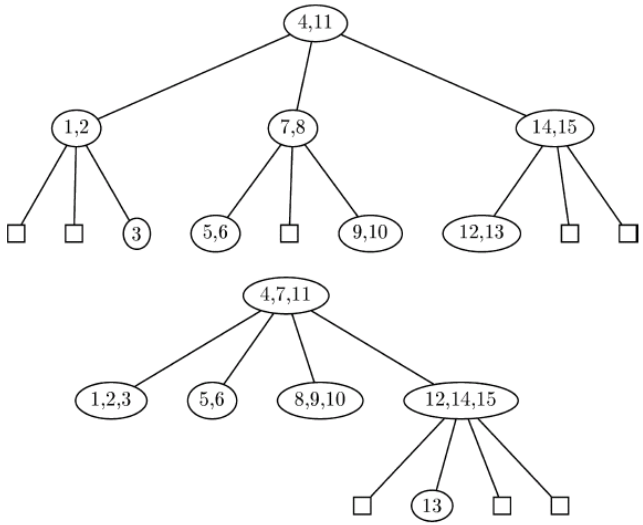


Figure: *A trinary respectively a quadrary tree generated by the keys 11, 4, 7, 15, 8, 10, 14, 9, 5, 1, 2, 12, 3, 6, 13.*

- In a m-ary search tree each vertex is associated with $m-1$ key numbers. The first $m-1$ drawn keys are hold by the root in increasing order creating m intervals. Then keys are added recursively to the subtrees rooted in the m children of the root decided by which of the m intervals the new key belongs to.

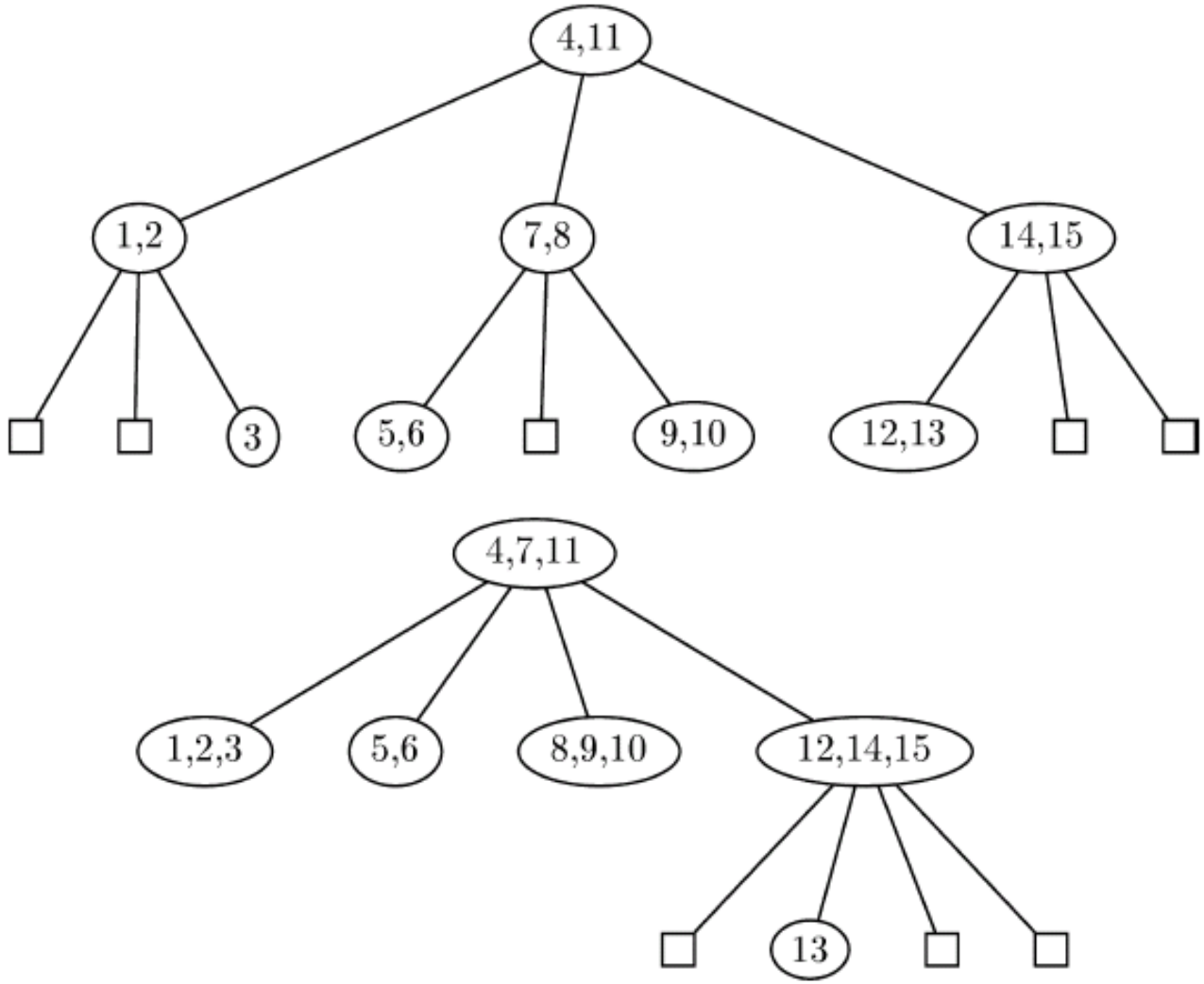


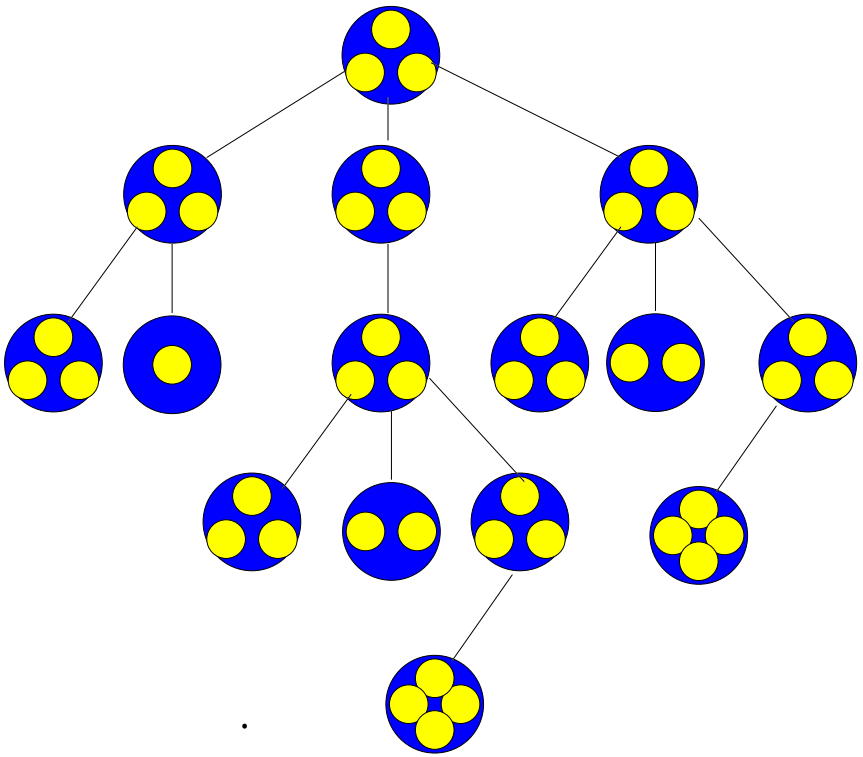
Figure: *A trinary respectively a quadrary tree generated by the keys 11, 4, 7, 15, 8, 10, 14, 9, 5, 1, 2, 12, 3, 6, 13.*

M-ary Search Trees (continued)

- Since only the order relations are important we can equally construct the m-ary search tree by drawing N i.i.d say $U(0,1)$ r.v. and add the keys recursively as in the construction of a m-ary search tree. Thus, the lengths of the intervals $V_1, \ldots, V_m$ if we cut a [0,1] interval uniformly $m-1$ times give the probabilities for going to respectively child of the root. The components $V_i,\ i \in \{1, \ldots, m\}$'s are distributed as $\min(U_1, U_2, \ldots, U_{m-1})$, where $U_1, \ldots, U_{m-1}$ are i.i.d $U(0,1)$ r.v. .
- All subtree sizes can be explained in this manner. If a subtree rooted at v holds S keys, the subtree size vector $(S_1, S_2, \ldots, S_b)$ for the children of v is multinomial $(S-m+1, V_1^v, \ldots, V_m^v)$, where $(V_1^v, \ldots, V_m^v)$ is v's splitting vector with components distributed as $\min(U_1, U_2, \ldots, U_{m-1})$.

What is a Split Tree?

(Devroye 1998)



- Branch factor b
- Cardinality N
- Vertex capacity s>0
- Independent copy of the random splitting vector $\mathcal{V}=(V_1, V_2, ..., V_b)$ is attached to each vertex.

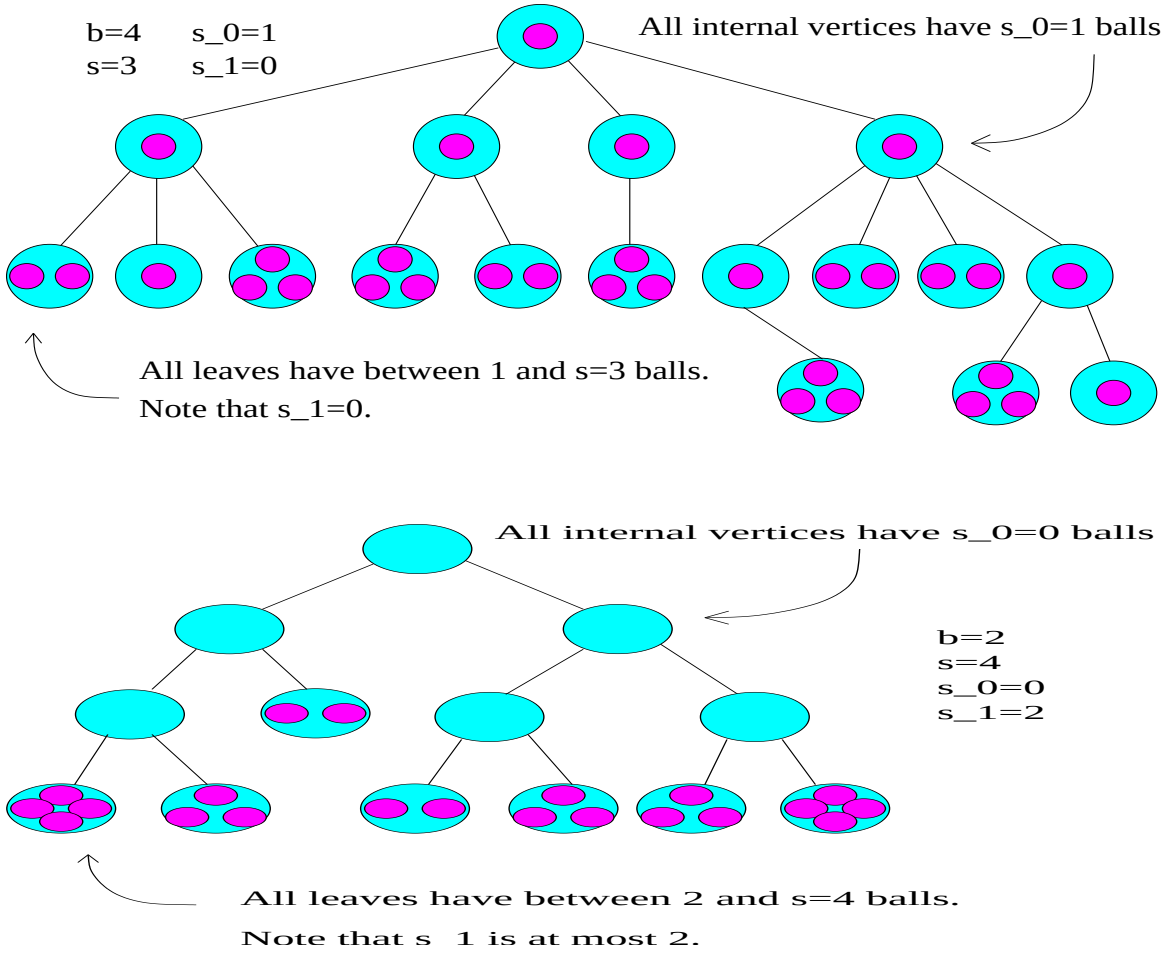
- **Binary search tree:**

  branch factor $\mathbf{b} = \mathbf{2}$,
  vertex capacity $\mathbf{s} = \mathbf{1}$,
  splitting vector $\mathcal{V} = (\mathbf{U}, \mathbf{1} - \mathbf{U})$, where $U \stackrel{d}{=} U(0,1)$.

  Keys (or balls) are added to the left child with probability U and to the right child with probability $1 - U$.

- **M'ary search trees:**

  branch factor $\mathbf{b} = \mathbf{m}$,
  vertex capacity $\mathbf{s} = \mathbf{m} - \mathbf{1}$,
  splitting vector $\mathcal{V} = (\mathbf{V_1}, \ldots, \mathbf{V_m})$, where
  $\mathbf{V_i} \stackrel{d}{=} \min(\mathbf{U_1}, \mathbf{U_2}, \ldots, \mathbf{U_{m-1}})$.



Balls are added one at a time. Each new ball starts in the root and is recursively added to subtrees, using the probabilities given by the splitting vectors $\mathcal{V}_v = (V_1, \ldots, V_b)$. V_i is the probability for adding the ball to the i:th child. The ball stops when it reaches a leaf. When a leaf gets $s+1$ balls it splits and sends balls to its children.

Examples of Split Trees and Common Properties

- The class of split trees includes many important random trees such as **binary search trees**, **m-ary search trees**, **quadtrees**, **median of $(2k+1)$-trees**, **simplex trees**, **tries** and **digital search trees**.
- The maximal depth (or height) of split trees is $O(\log n)$.
- Split trees have similar properties to the deterministic complete binary tree, with maximal depth $\lfloor \log_2 n \rfloor$ and most vertices close to this depth.
- In split trees most vertices are close to the depth $\mu^{-1} \ln n$, for some constant μ depending on the split tree.
  In the specific case of the binary search tree this depth is $2 \ln n$.

What is a Record in a Rooted Tree?

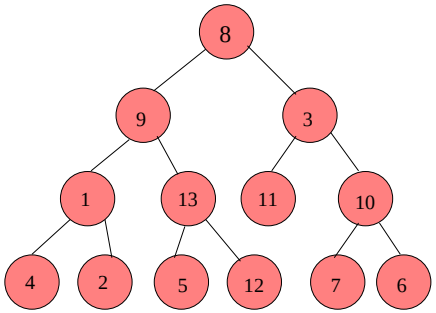
- Given a rooted tree T, let each vertex v have a random value λ_v attached to it. Assume that these values are i.i.d. with a continuous distribution.
- A value λ_v is a **record** if it is the smallest value in the path from the root to v. Let $X(T)$ denote the (random) number of records.

What is a Cutting in a Rooted Tree?

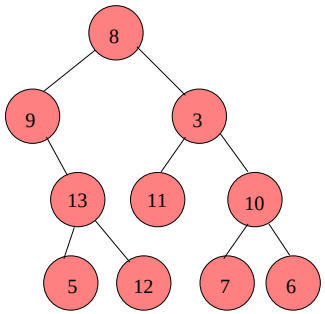
- Choose one vertex at random.
- Cut in this vertex so that the tree separates into two parts, and keep only the part containing the root.
- Continue recursively until the root is cut.

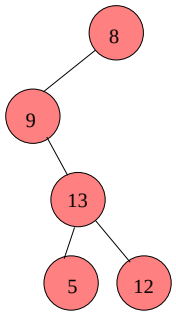
Records and Cuttings in Rooted Trees

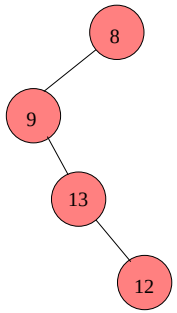
- The number of records $X(T)$ is equal in distribution to the number of cuts. (Janson 2004)
  Think! A vertex v is cut at some time iff λ_v is a record.



First generate the values λ_v and then cut the tree, each time choosing the vertex with the smallest value λ_v of the remaining ones.







Aim of Study

- To find the asymptotic distribution of the number of records (or equivalently the number of cuts) in random split trees.

Background

- Cutting down trees first introduced by Meir and Moon (1970).
- Janson uses a probabilistic approach considering records instead of cuts in the tree finding the distribution (after normalization) of $X_v(T)$ = number of records (=number of cuts). He finds the asymptotic distributions for conditioned Galton-Watson trees (2004), e.g labelled trees and random binary trees and for a fixed complete binary tree (2004).
- Drmota, Iksanov, Moehle and Roesler, recently used analytic methods to prove asymptotic distributions of the number of cuts in the random recursive tree.

The Main Theorem

Let T_N be a split tree with N balls, and **let $\mathbf{X}(\mathbf{T_N})$ be the number of records (or cuts) in $\mathbf{T_N}$.**

Theorem

Suppose that $N \to \infty$. Then

$$\left(X(T_N) - \frac{\alpha N}{\mu^{-1} \ln N} - \frac{\alpha N \ln\ln N}{\mu^{-1} \ln^2 N} \right) \Big/ \frac{\alpha N}{\mu^{-2} \ln^2 N} \xrightarrow{d} -W, \quad (1)$$

where μ and α are constants and W has an infinitely divisible distribution more precisely a **weakly 1-stable distribution** with characteristic function

$$\mathbf{E}\left(e^{itW}\right) = \exp\left(-\frac{\mu^{-1}}{2}\pi|t| + it(C) - i|t|\mu^{-1} \ln|t| \right), \quad (2)$$

where C is a constant.

Infinitely Divisible Distributions

A distribution of a random variable Z is infinitely divisible if for each n, there exist i.i.d random variables $Z_{n,k}, \quad 1 \leq k \leq n$, such that

$$Z \stackrel{d}{=} \sum_{k=1}^{n} Z_{n,k}, \qquad \forall\ n,$$

or equivalently

$$\mathbf{E}\left(e^{itZ}\right) = \left(\mathbf{E}\left(e^{itZ_{n,1}}\right)\right)^{n}, \qquad \forall\ n.$$

α-Stable Distributions

The stable distributions belong to the class of infinitely divisible distributions.
A distribution of a random variable Z is α-stable for $\alpha \in (0, 2]$ if for a sequence of i.i.d random variables $Z_k,\ k \geq 1$ distributed as Z there exists constants c_n such that

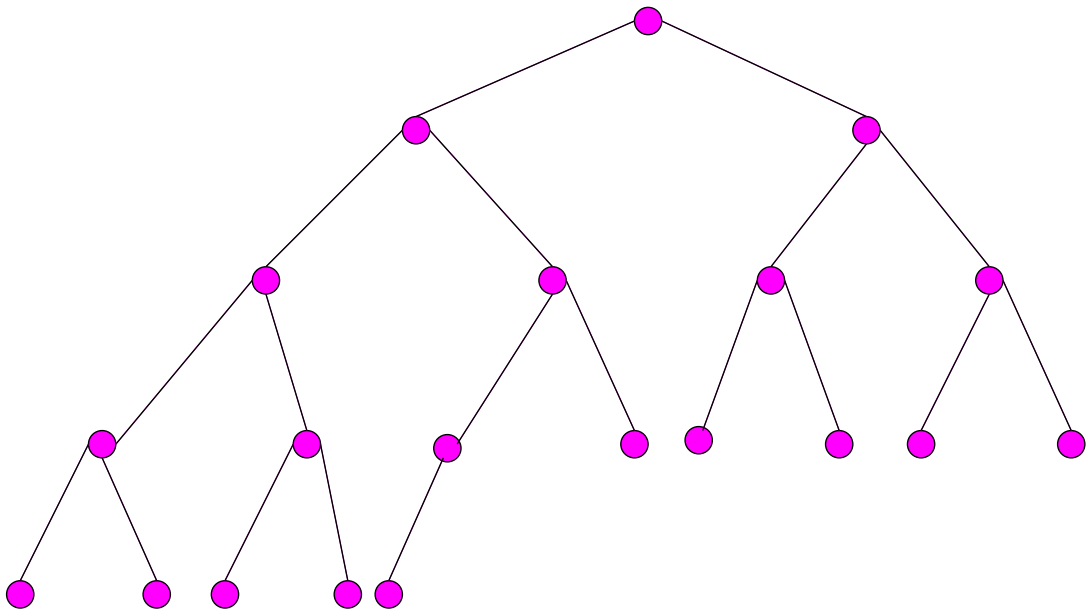
$$\sum_{k=1}^{n} Z_k \stackrel{d}{=} n^{\frac{1}{\alpha}} Z + c_n,$$

for all n. The distribution is strictly stable if for all n, $c_n = 0$ and weakly stable otherwise.

Method of Proof of the Main Theorem

- To express the number of records $X(T)$ by a sum of i.i.d. r.v. derived from λ_v and then apply a classical limit theorem for convergence of a sum of triangular null arrays to infinitely divisible distributions. *This method was first used by Janson for finding the distribution of the number of records in the deterministic complete binary tree. For the Galton Watson trees the method of moments was used but this method is not possible to use for trees of logaritmic height!*
- To extend the Janson method so that it can be used for the more complex random binary search tree.
- To generalize the proofs for the binary search tree and show that this method can be used also for all other types of split trees.

Complete Binary Tree: Most Nodes Close to the Top Level of Depth $\log_2 n$



In Split Trees Most Nodes Close to Depth $\mathcal{O}(\ln \mathbf{n})$

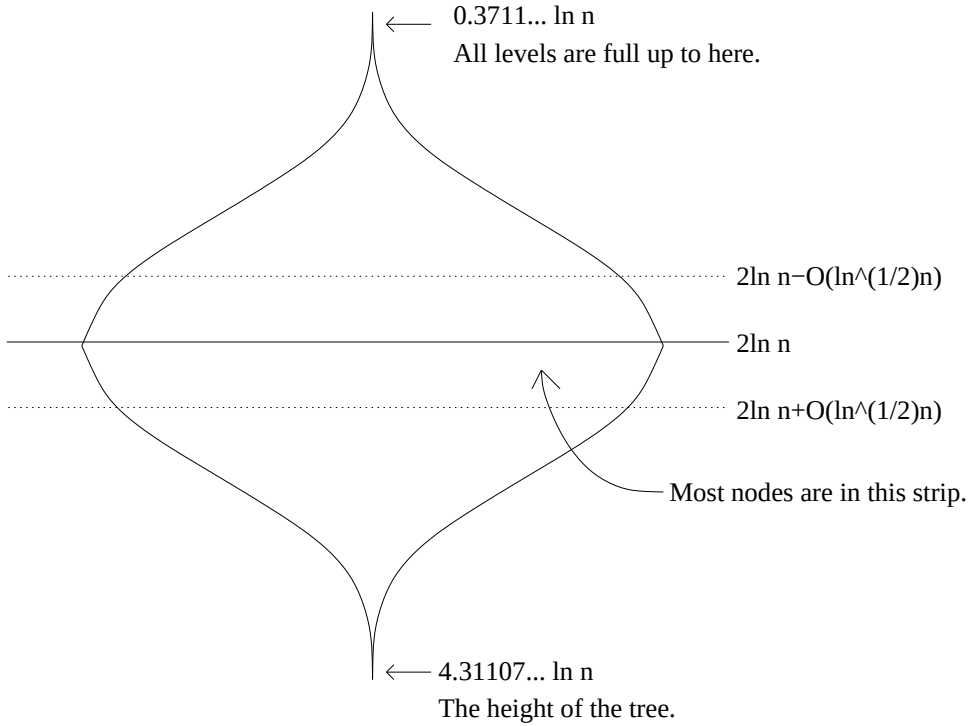


Figure: *This figure shows an example of the binary search tree where most nodes are close to* $2 \ln n$.

Subtree Sizes

- In a split tree with N balls, given the root's splitting vector $\mathcal{V}_\sigma = (V_1, \ldots, V_b)$, the numbers of balls in the subtrees rooted at the root's children are close to $NV_1, \ldots, NV_b$.
- Let N_v be the number of balls in the subtree rooted at the vertex v.
  Given all splitting vectors in the tree, N_v for v at depth k is close to

$$NW_1 W_2 \ldots W_k, \tag{3}$$

  where W_r, $r \in \{1, \ldots, k\}$ are i.i.d. r.v distributed as V_i.
  The W_r's are given by the splitting vectors associated with the vertices in the unique path from v to the root.
- The N_v's are not independent for different vertices!

"Good" and "Bad" Vertices in Split Trees

- There is a central limit theorem for the depth of vertices so that "most" vertices lie at $\mu^{-1} \ln N + \mathcal{O}\left(\sqrt{\ln N}\right)$. Devroye (1998)
- Let $h(v)$ denote the depth of a vertex v in the split tree T_N. A vertex v is called **good** if

$$\mu^{-1} \ln N - \ln^{0.6} N \leq h(v) \leq \mu^{-1} \ln N + \ln^{0.6} N,$$

and **bad** otherwise. Recall that the subtree sizes can be expressed by r.v.'s that depend on the splitting vectors. I use this fact to apply large deviations and show that the bad vertices are bounded by a small error term and can thus be ignored.

Advantage to Considering Records in Subtrees

- Consider the subtrees T_i, $1 \leq i \leq b^L$ rooted at $L = C \log\log n$.
- Let Λ_i be the smallest value of the λ_v's from the vertex i to the root of T_N. Given T_N and the λ_v's below level L,

$$X(T_N) \approx \sum_{i=1}^{b^L} X(T_i)_{\Lambda_i}.$$

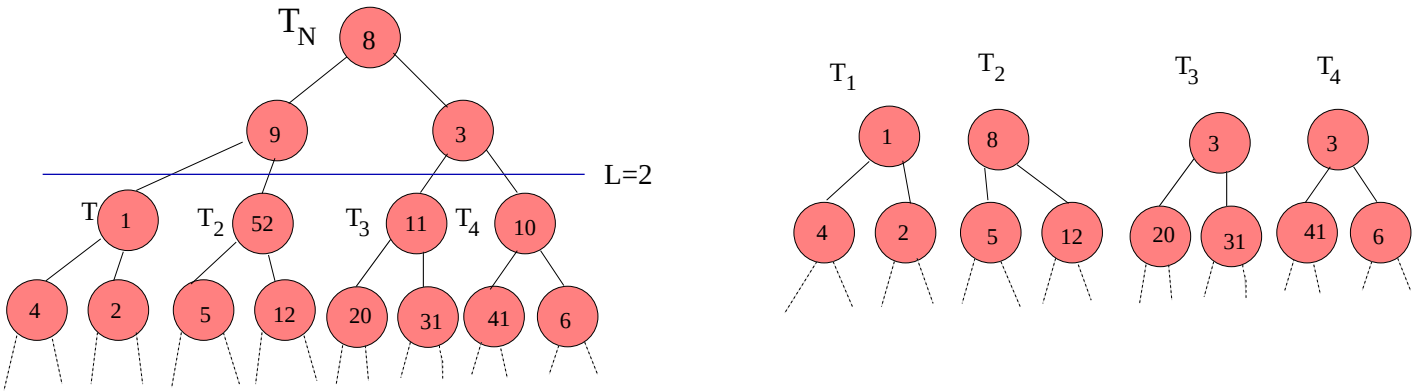


Figure: *The subtrees T_1, T_2, T_3, T_4 at depth $L = 2$ are considered. This example has $\Lambda_1 = 1$, $\Lambda_2 = 8$, $\Lambda_3 = 3$ and $\Lambda_4 = 3$.*

Advantage to Considering Records in Subtrees continued

Recall that Λ_i is the smallest value of the λ_v's from vertex i to the root of T_N. By using the Chebyshev inequality, the total number of records given Λ_i in the subtrees T_i rooted at depth L i.e.

$$\sum_{i=1}^{b^L} X(T_i)_{\Lambda_i},$$

can be approximated by the conditional expected value

$$\sum_{i=1}^{b^L} \mathbf{E}(X(T_i)_{\Lambda_i} \mid T_i, \Lambda_i).$$

The conditional expectation of $X(T_i)_{\Lambda_i}$

Let for each vertex $v \in T_i$, I_v be the indicator that λ_v is the minimum value given T_i and Λ_i i.e. the smallest value of the values λ_v's from the vertex i to the root of T_N. Thus,

$$\mathbf{E}(X(T_i)_{\Lambda_i} \mid T_i, \Lambda_i) = \sum_{v \neq v_i} \mathbf{E}(I_v).$$

Let $h_i(v)$ be the depth of v in T_i i.e. the distance between v and the root i. By using that the values λ_v's are independent Exp(1) random variables one easily get

$$\mathbf{E}(I_v) = \frac{1 - e^{-h_i(v)\Lambda_i}}{h_i(v)}.$$

Thus,

$$\mathbf{E}(X(T_i)_{\Lambda_i} \mid T_i, \Lambda_i) = \sum_{v \neq v_i} \frac{1 - e^{-h_i(v)\Lambda_i}}{h_i(v)}.$$

The conditional expectation of $\mathbf{X(T_i)_{\Lambda_i}}$ continued

Let N_i be the number of balls in T_i. Since most vertices in T_i have depths in the strip

$$\mu^{-1} \ln N_i - \ln^{0.6} N_i \leq h(v) \leq \mu^{-1} \ln N_i + \ln^{0.6} N_i$$

(i.e. these are the good vertices in T_i) the sum

$$\mathbf{E}(X(T_i)_{\Lambda_i} \mid T_i, \Lambda_i) = \sum_{v \neq v_i} \frac{1 - e^{-h_i(v)\Lambda_i}}{h_i(v)}$$

is close to

$$(1 - e^{-\mu^{-1} \ln N_i \Lambda_i}) \sum_{v \neq v_i} \frac{1}{h_i(v)}. \tag{4}$$

Again because most vertices are good we can use Taylor expansion about $\mu^{-1} \ln N_i$ for approximating $\frac{1}{h_i(v)}$ in (4).

Approximation of $\mathbf{X(T_N)}$ depending on N_v's and λ_v's

Recall that the purpose of these calculations is to find an approximation of $X(T_N)$ since we showed that this number could be approximated by

$$\sum_{i=1}^{b^L} \mathbf{E}(X(T_i)_{\Lambda_i} \mid T_i, \Lambda_i).$$

The result we get is an approximation of $X(T_N)$ depending on the subtree sizes N_i for $h(i) = L$ and Λ_i i.e. the smallest value of the values λ_v's from the vertex i to the root of T_N. We want to use a triangular array theorem and therefore we need independent random variables. Neither the N_i's or the Λ_i's are independent! However, from the approximation we got it is quite easy to find an approximation of $X(T_N)$ depending on $N_v,\ h(v) \leq L$ and $\lambda_v,\ h(v) \leq L$. Thus, at least the λ_v's are independent.

Applying a Theorem for Triangular Arrays

- The normalized $X(T_N)$ in the Main Theorem can be expressed as

$$-\Big(\sum_{h(v)\le L} \xi_v + \sum_{i=1}^{N} \xi_i' \Big) + o_p(1),$$

where $\xi_v := \frac{N_v \mu^{-1} \ln N}{N} \cdot e^{-\lambda_v \mu^{-1} \ln N}$ and the ξ_i''s are r.v.'s only depending on the N_v's with $h(v) = L$.

- Conditioned on the N_v's, the ξ_v's are independent r.v.'s since the λ_v's are independent, and the ξ_i''s are deterministic. Thus, given the N_v's, $\{\xi_v\} \bigcup \{\xi_i'\}$ is a triangular array.
- **The purpose is to use a classical central limit theorem for convergence of a sum of triangular null arrays to infinitely divisible distributions.**

The Triangular Array Theorem Requires Theorem 2.1

- The limit theorem for convergence of a sum of triangular null arrays to infinitely divisible distributions requires that three conditions for the null array are fulfilled.
- Theorem 2.1 shows that these conditions are fulfilled for $\{\xi_v\} \bigcup \{\xi_i'\}$ conditioned on the N_v's (with ξ_i' deterministic).
- Theorem 2.1 shows that the limit theorem for null arrays can be applied to $\sum_{h(v)\leq L} \xi_v + \sum_{i=1}^{N} \xi_i'$ given the N_v's. The limit theorem implies that this sum converges in distribution to a r.v. W, with an infinitely divisible distribution that does not depend on the given N_v's.
- **Thus, the Main Theorem is proved i.e. $X(T_N)$ normalized converges to $-W$ with an infinitely divisible distribution.**

Theorem 2.1

Let Ω_L be the σ-field generated by $\{N_v,\ h(v) \le L\}$.
Theorem 2.1
Suppose that $N \to \infty$ and choose any constant $c > 0$.
Conditioning on the σ-field Ω_L the following hold

(i) $\sup_v \mathbf{P}\big(\xi_v > x \big| \Omega_L\big) \to 0$ for every $x > 0$, *i.e.* $\{\xi_v\}$ *is a null array*

(ii) $\sum_{h(v)\le L} \mathbf{P}\big(\xi_v > x \big| \Omega_L\big) \xrightarrow{p} \nu(x,\infty) = \dfrac{\mu^{-1}}{x}$ for every $x > 0$,

(iii) $\sum_{h(v)\le L} \mathbf{E}\big(\xi_v \mathbf{1}[\xi_v \le c] \big| \Omega_L\big) + \sum_{i=1}^{N} \xi_i' \xrightarrow{p} K$, K *is a constant*

(iv) $\sum_{h(v)\le L} \mathbf{Var}\big(\xi_v \mathbf{1}[\xi_v \le c] \big| \Omega_L\big) \xrightarrow{p} \mu^{-1} c.$

Proof of Theorem 2.1

- Theorem 2.1, which implies the Main Theorem has a technical proof. The idea is to use the Chebyshev inequality for proving that the sums in (ii), (iii) and (iv) are sharply concentrated about their mean values.
- Important Observation: The sums in (ii), (iii) and (iv) only depend on the subtree sizes $\{N_v,\ h(v) \leq L\}$.
- Recall that N_v for v at depth k, is close to $NW_1W_2 \dots W_k$, where $W_r,\ r \in \{1, \dots k\}$ are independent r.v.'s distributed as the components V_i in the splitting vector.
- Let $Y_k := -\sum_{r=1}^{k} \ln W_r$. Note that $NW_1W_2 \dots W_k = Ne^{-Y_k}$. In a binary search tree, Y_k is distributed as a $\Gamma(k,1)$ r.v. since $W_r \stackrel{d}{=} V_i \stackrel{d}{=} U$, where U is a uniform $U(0,1)$ r.v..

Proof of Theorem 2.1 (continued)

- For general split trees there is usually no simple distribution function for Y_k; instead renewal theory is used.
- Define the renewal function

$$U(t) = \sum_{k=1}^{\infty} b^k \mathbf{P}(Y_k \leq t) = \sum_{k=1}^{\infty} F_k(t), \tag{5}$$

  and let $F(t) := F_1(t) = b\mathbf{P}(W_i \leq t)$.
- For $U(t)$ we obtain the following renewal equation

$$U(t) = F(t) + \sum_{k=1}^{\infty} (F_k * F)(t) = F(t) + (U * F)(t).$$

- For $t \to \infty$ the solution of this equation is

$$U(t) = (\mu^{-1} + o(1))e^t.$$

Conclusions

- It was tested whether the Janson method for determining the asymptotic distribution of the number of records (or cuts) in a deterministic complete binary tree could be extended to random split trees.
- It was shown that with modifications, the Janson method could be used for determining the asymptotic distribution of the number of records (or cuts) **in the binary search tree**, which is one well-characterized type of split tree.
- Further, by also introducing renewal theory, the method of proof used for the binary search tree could be generalized **to cover all split trees**.
- The results show that for the entire large class of random split trees the normalized **number of records (or cuts) has asymptotically a weakly 1-stable distribution**.

Acknowledgements

- Professor Svante Janson, Uppsala University.