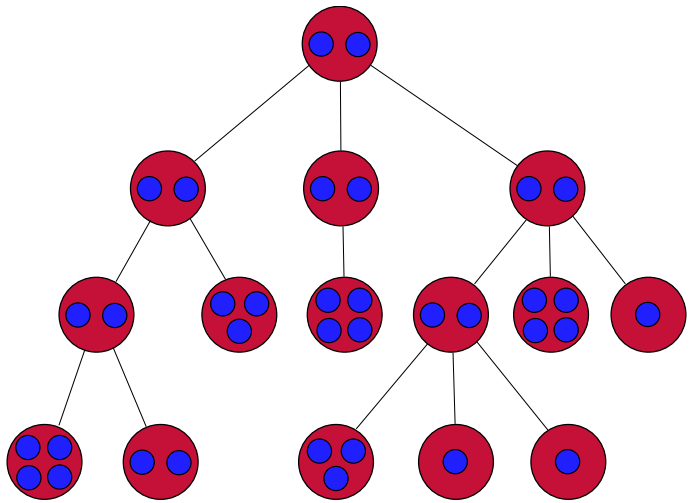
# Random Records and Cuttings in Split Trees

Cecilia Holmgren

(INRIA Rocquencourt, Paris)

Nordita, Stockholm, 01 November 2010



## Aim of Study

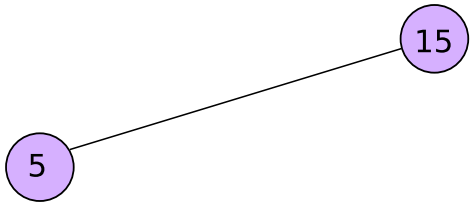
- To find the **asymptotic distribution of the number of records in random split trees**. (This number is equal in distribution to the number of cuts needed to eliminate this type of tree.)

# The Binary Search Tree is an Example of a Split Tree

# The Binary Search Tree is an Example of a Split Tree

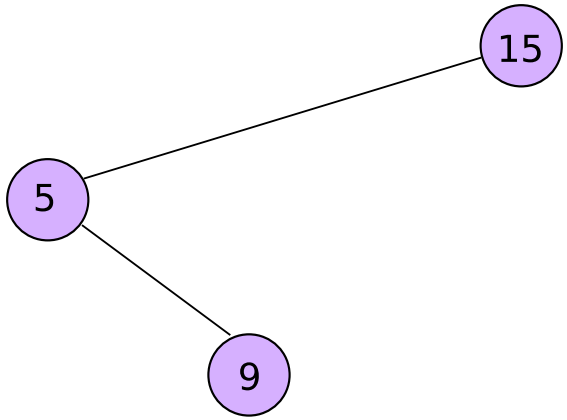

Randomly draw a number, which we call a key, from the set $\{1, 2 \ldots, 30\}$, and associate it to the root.

# The Binary Search Tree is an Example of a Split Tree



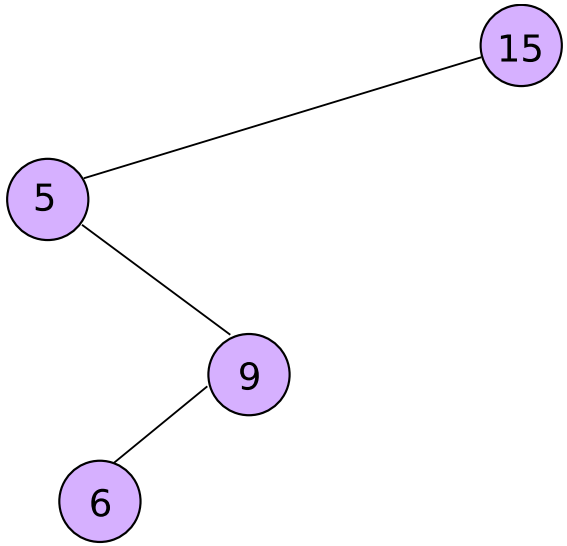

Randomly draw a new number from the remaining numbers in $\{1, 2 \ldots, 30\}$, and associate it to the left child if it is smaller than the root's key and to the right child if it is larger.

# The Binary Search Tree is an Example of a Split Tree

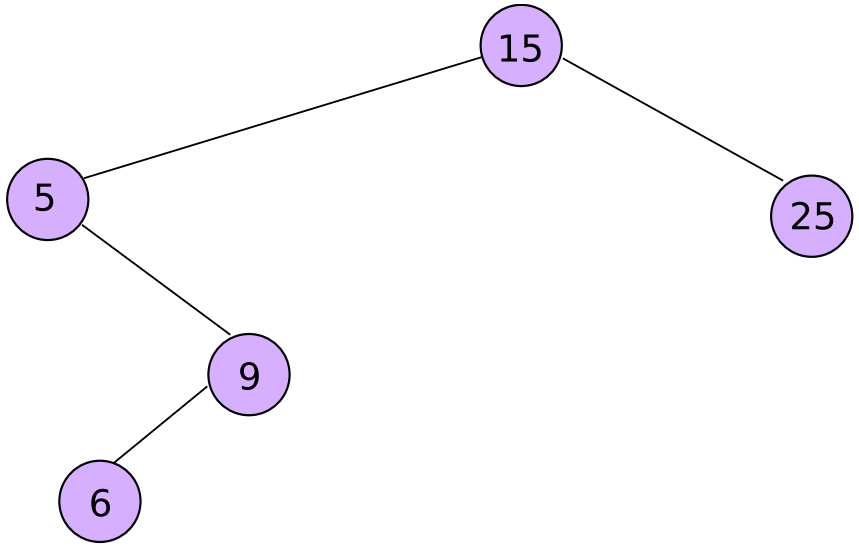



Proceed recursively in each subtree, by comparing the new drawn key by the current root's key.

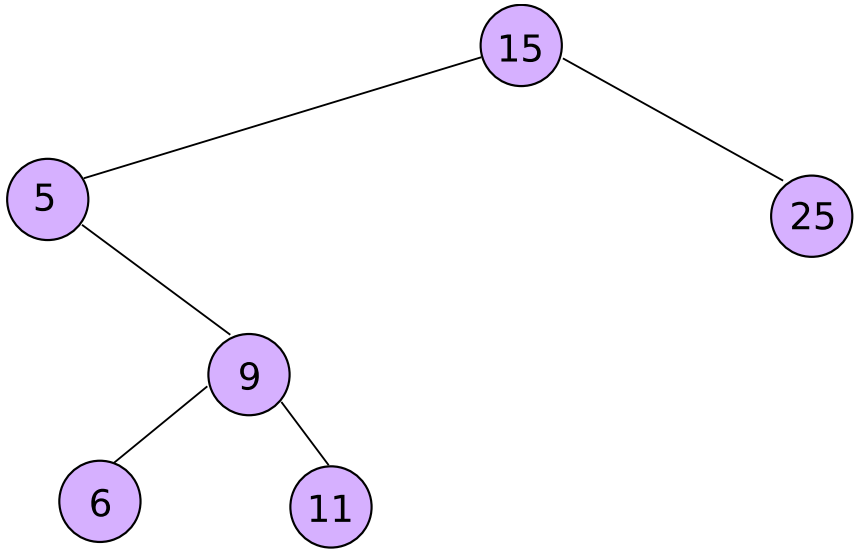
# The Binary Search Tree is an Example of a Split Tree



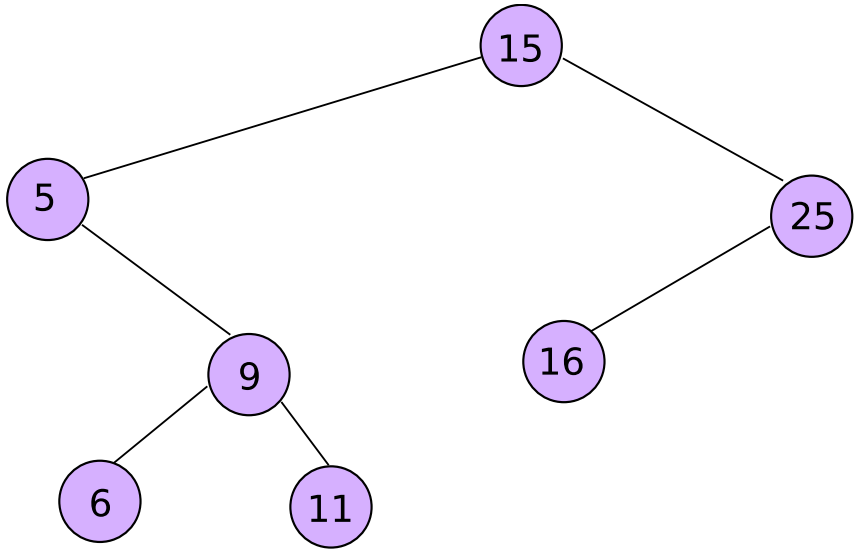
# The Binary Search Tree is an Example of a Split Tree



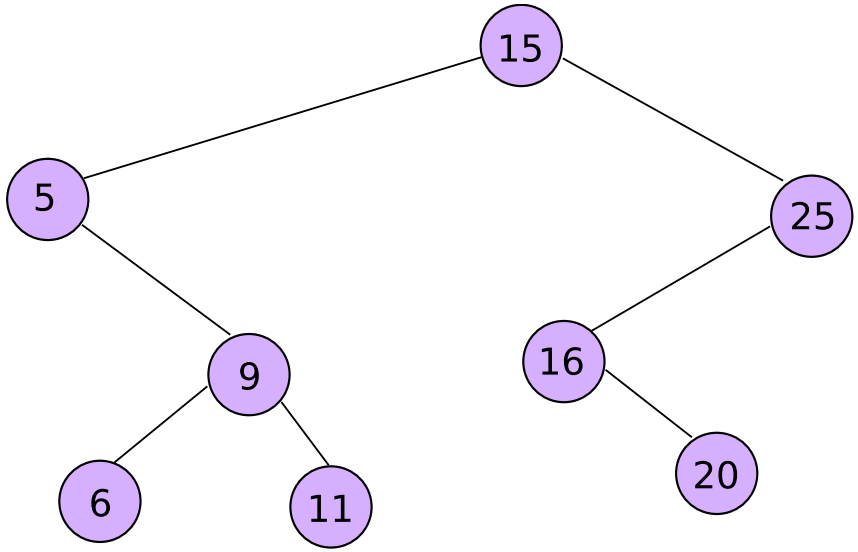
# The Binary Search Tree is an Example of a Split Tree



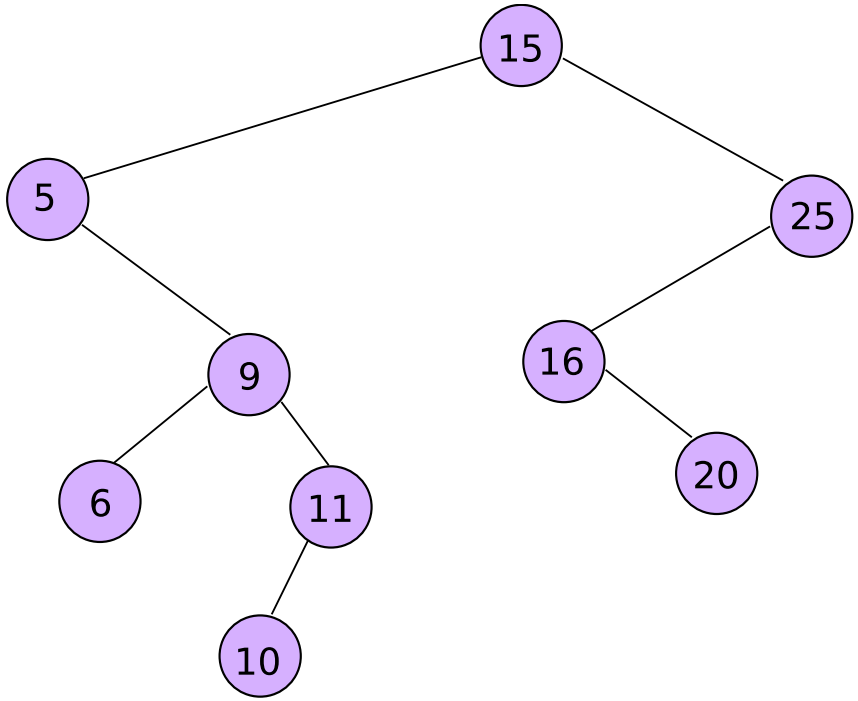
# The Binary Search Tree is an Example of a Split Tree



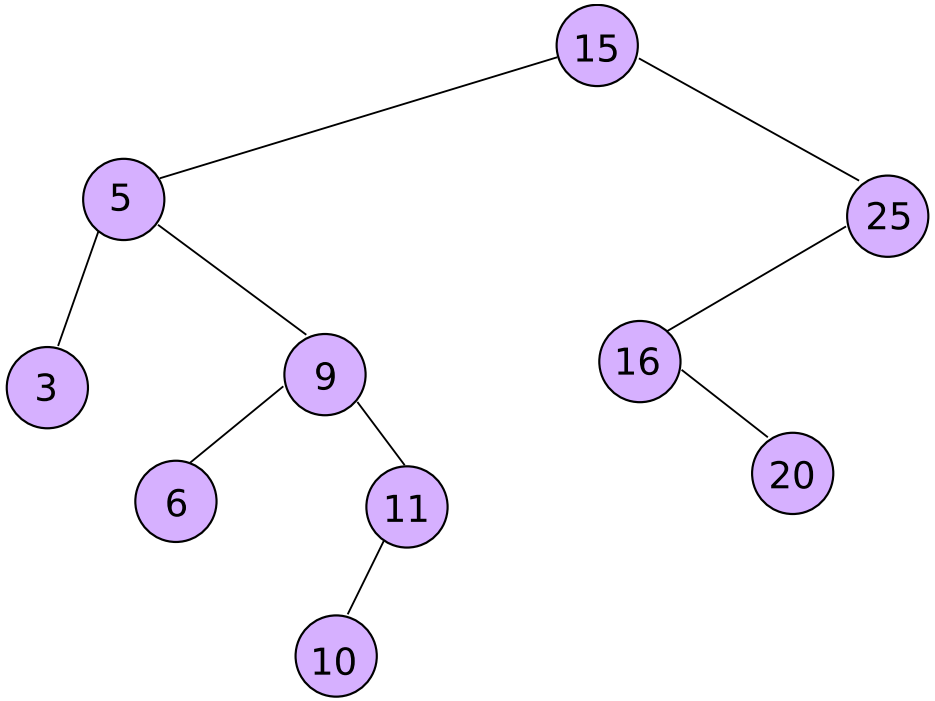
# The Binary Search Tree is an Example of a Split Tree



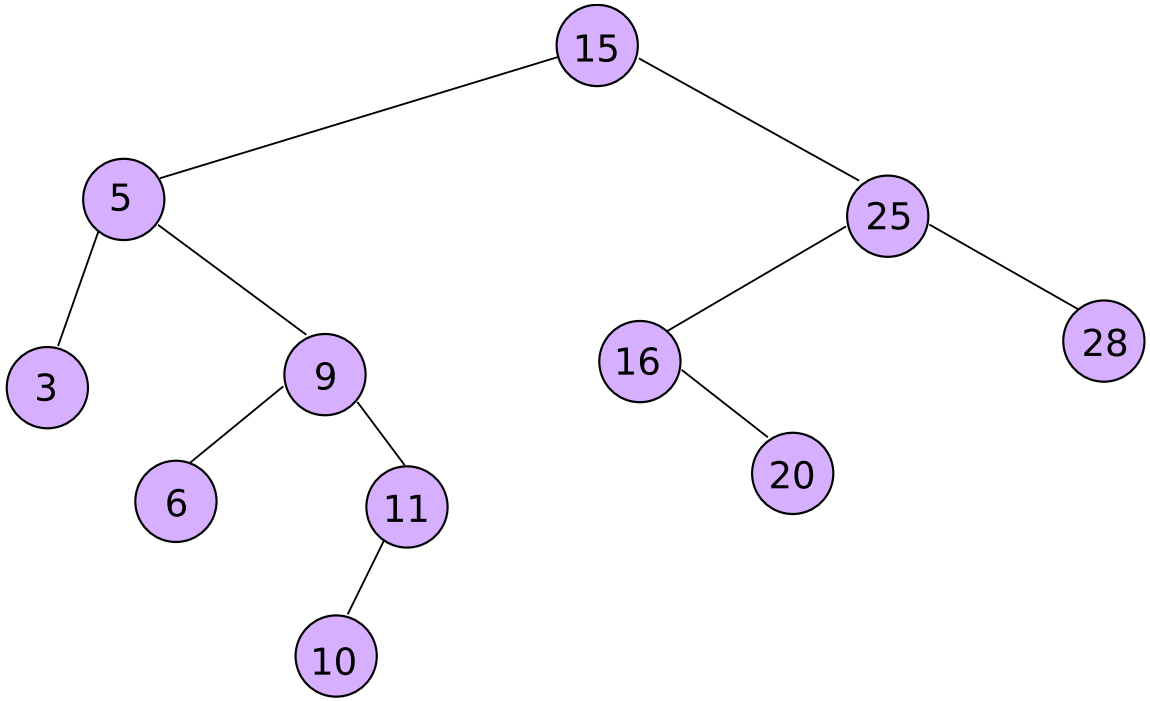
# The Binary Search Tree is an Example of a Split Tree



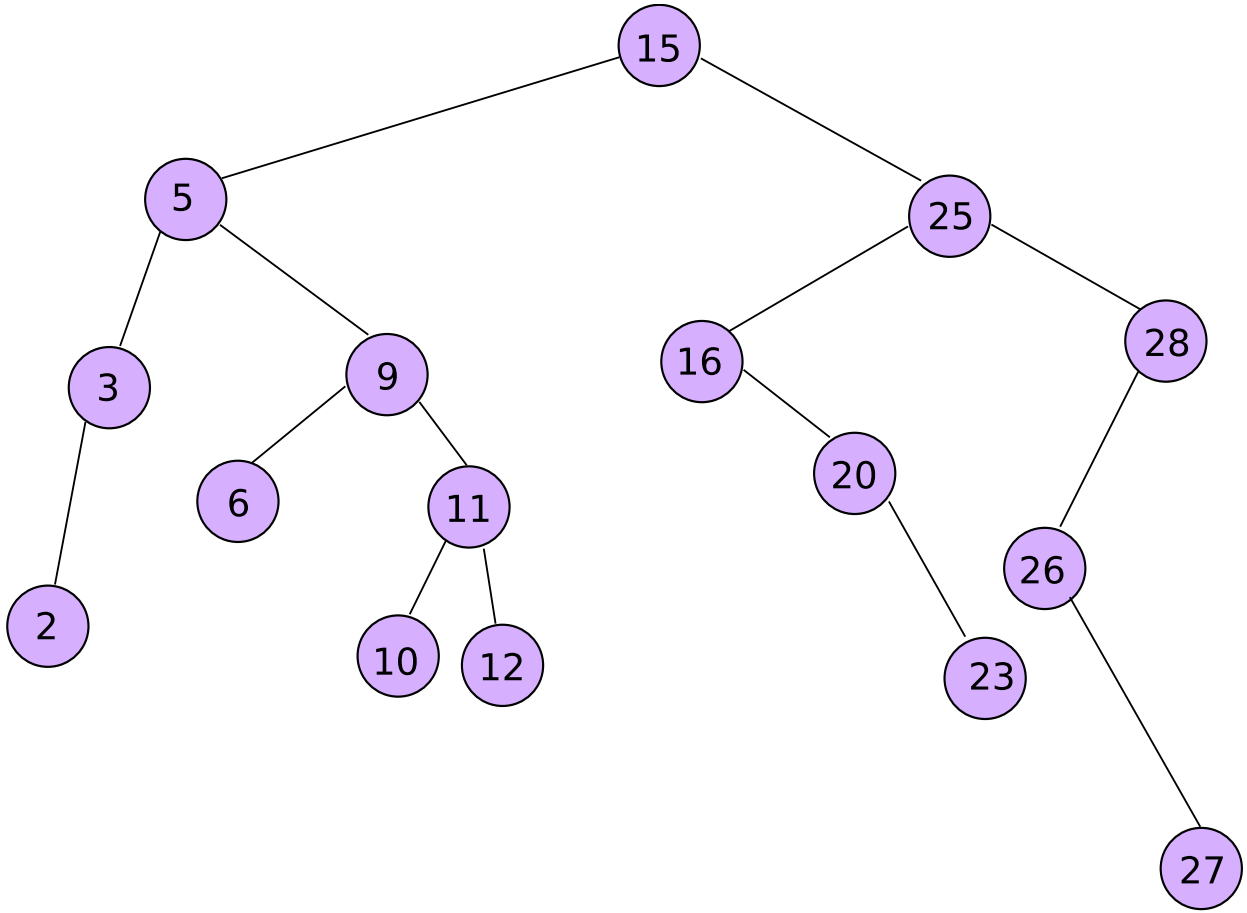
# The Binary Search Tree is an Example of a Split Tree



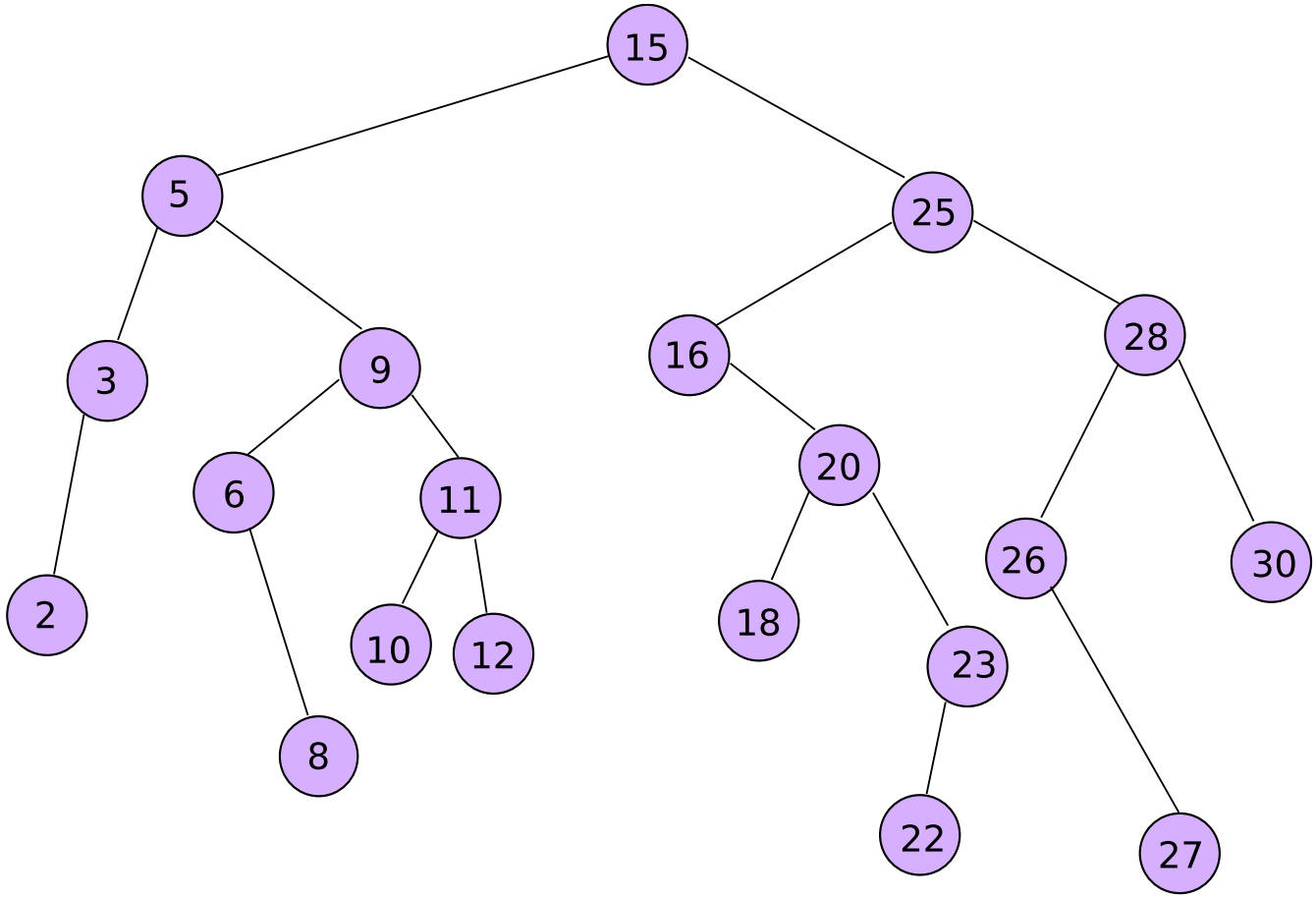
# The Binary Search Tree is an Example of a Split Tree



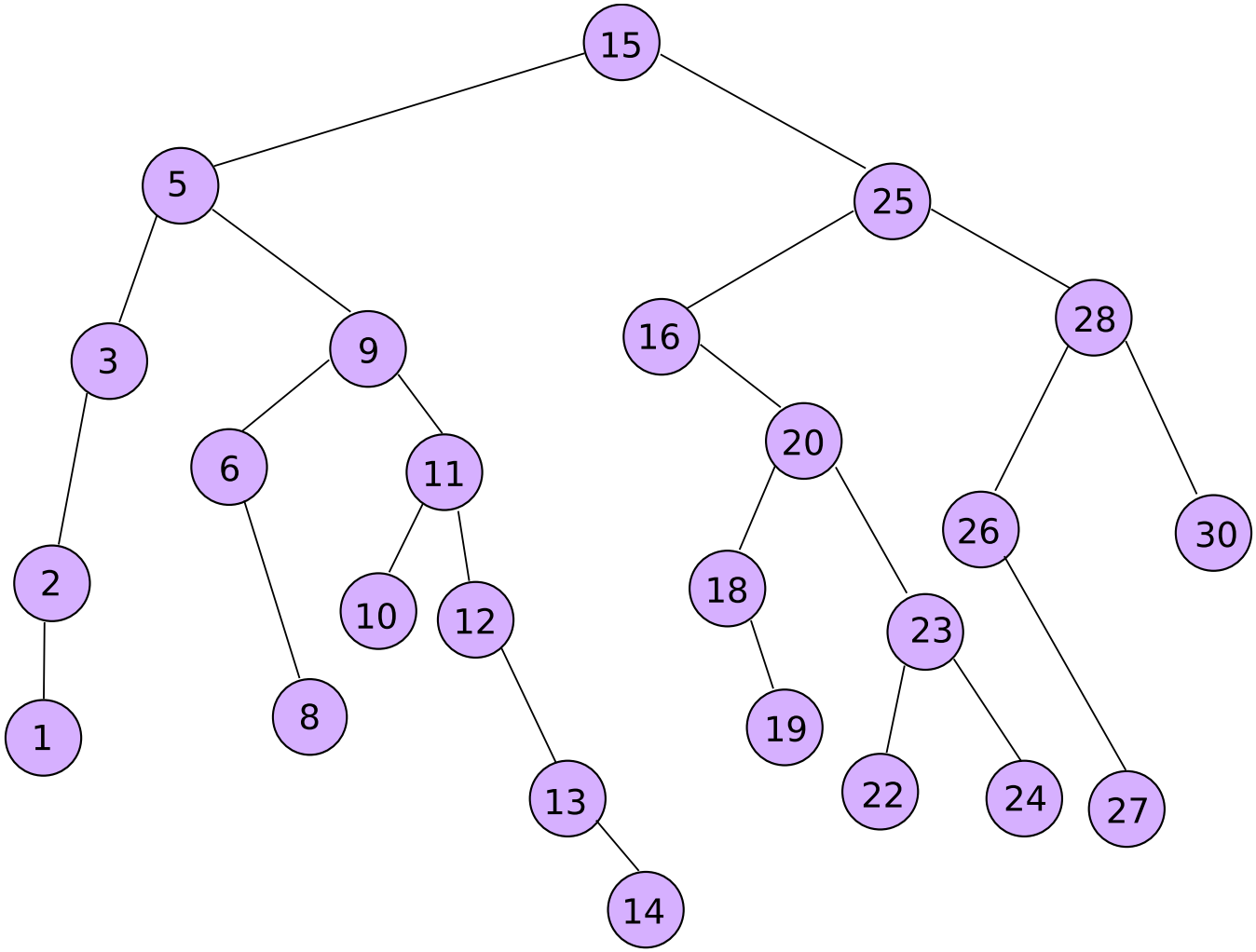
# The Binary Search Tree is an Example of a Split Tree



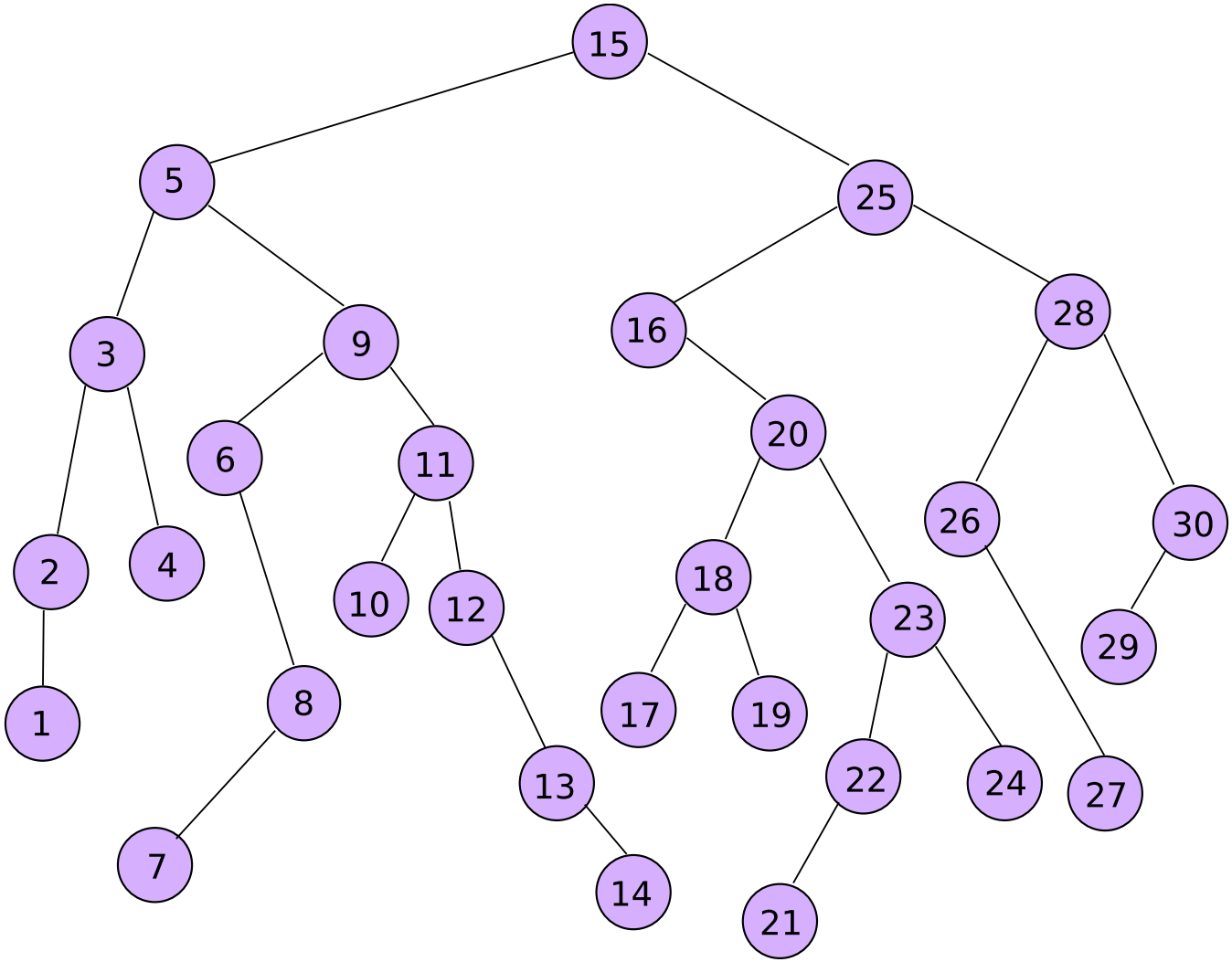
# The Binary Search Tree is an Example of a Split Tree



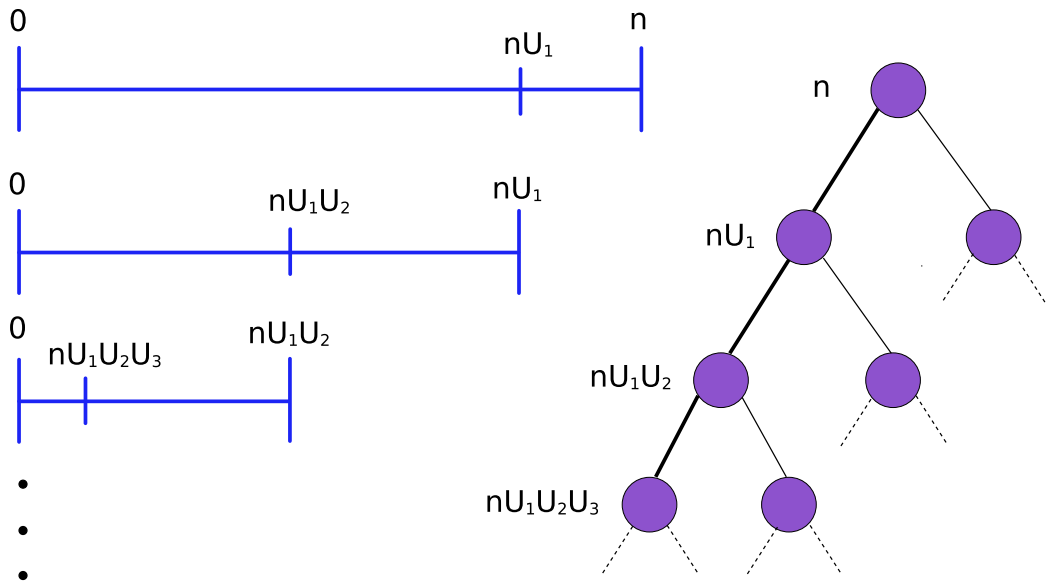
# The Binary Search Tree is an Example of a Split Tree



# The Binary Search Tree is an Example of a Split Tree



# 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 is distributed as $\lfloor nU \rfloor$, where $U$ is a uniform $U(0, 1)$ random variable. Similarly the right subtree is distributed as $n - \lfloor nU \rfloor$.
- All subtree sizes can be explained in this manner. If a subtree rooted at $v$ has size $V$, the size of its left subtree is $\stackrel{d}{=} \lfloor VU_v \rfloor$.

# The m-ary Search Trees are Examples of Split Trees

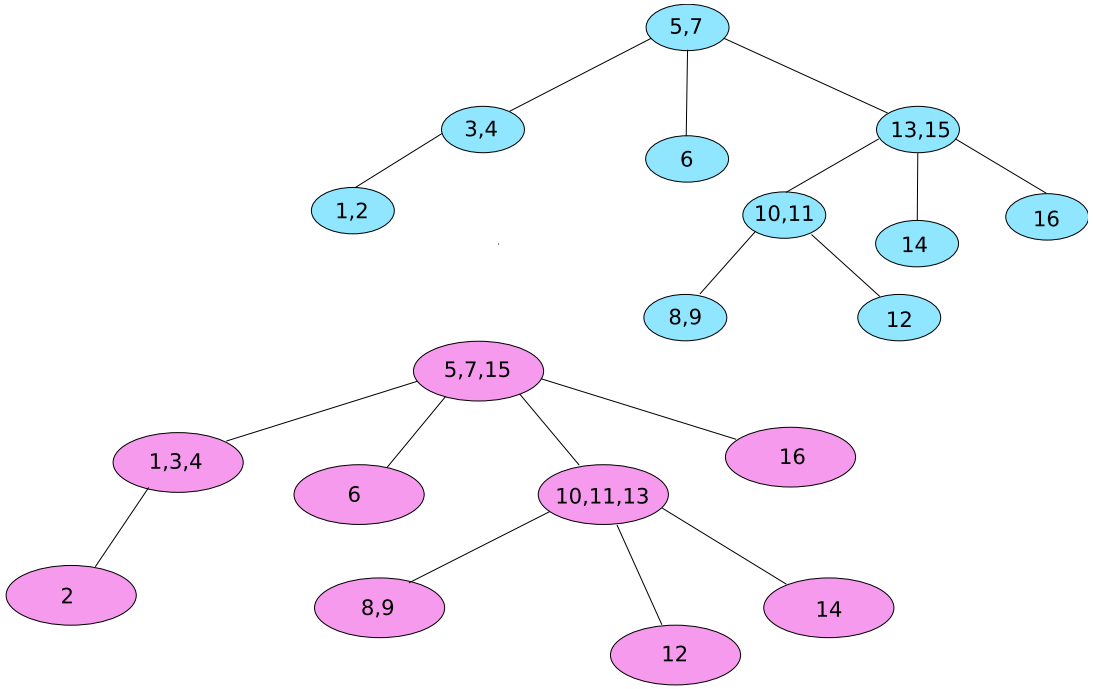
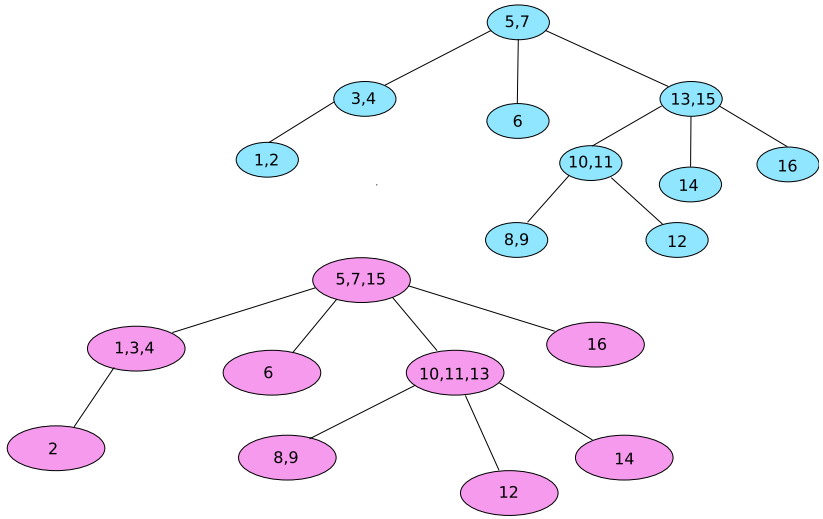



Figure: *The m-ary search trees are generalisations of the binary search tree where $m = 2$. The figure shows a 3-ary and a 4-ary search tree constructed from the sequence 7,5,15,3,4,6,1,13,11,10,2,16,8,9,14,12.*

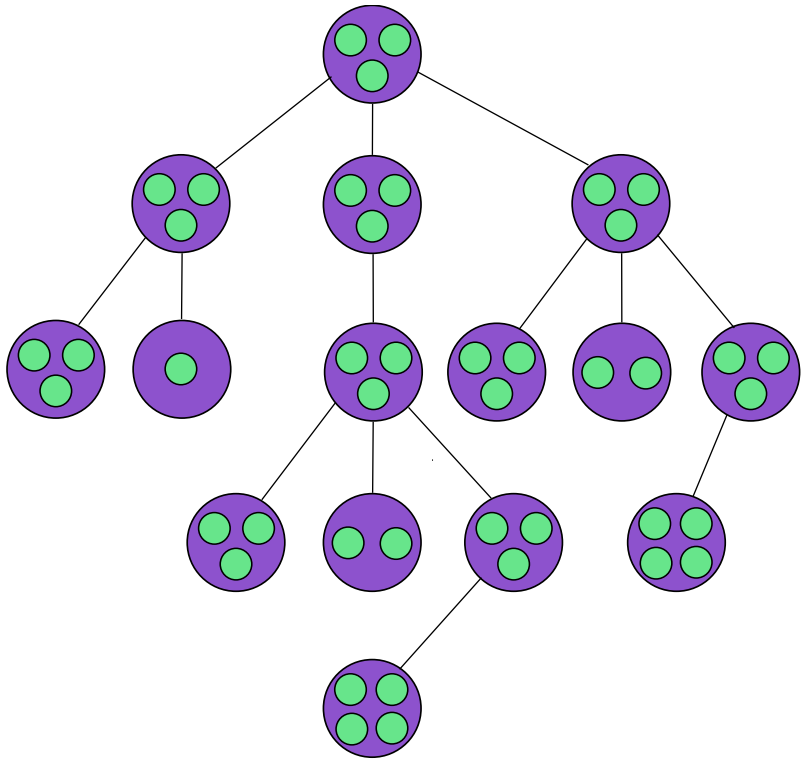
# The m-ary Search Trees cont.




- The proportions of the number of keys in the $m$ subtrees of the root are given by the lengths of the sub-intervals created if we do $m-1$ random cuts of a [0,1] interval.
- Let $(n_1, n_2, \ldots, n_m)$ be the vector of the subtree sizes for the children of the root. Then $(n_1, n_2, \ldots, n_m)$ is distributed as a multinomial vector $(n, V_1, \ldots, V_m)$, where the $V_i$'s are distributed as the minimum of $m-1$ uniform $U(0,1)$ r.v..

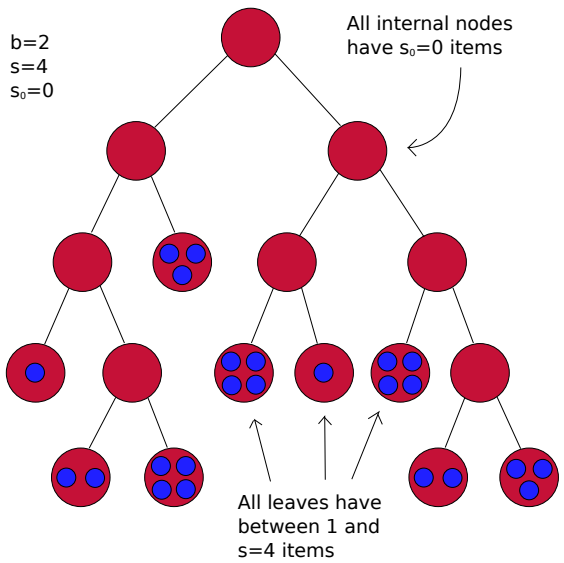
# What is a Split Tree?

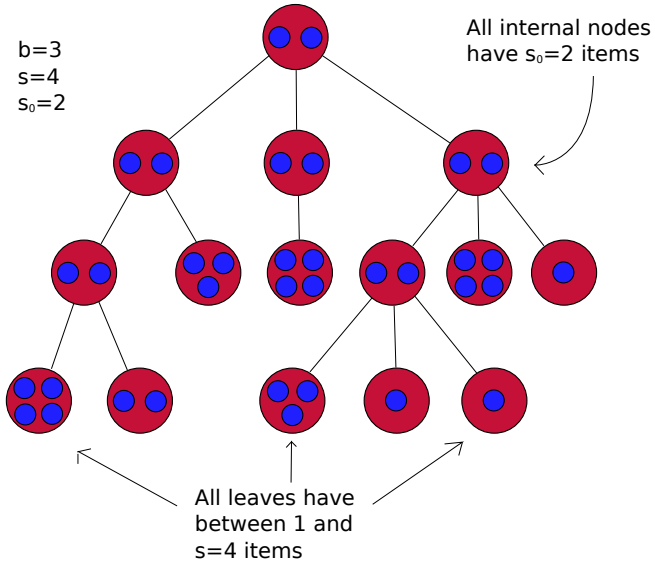
(Devroye 1998)



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

# The Recursive Construction of a Split Tree




- Let $n_v$ denote the cardinality of a node $v$.
- The splitting procedure starts in the root and is only carried on as long $n_v > s$.
- Given the cardinality $n_v > s$ and the split vector $\mathcal{V}_v = (V_1, V_2 \ldots, V_b)$, the cardinalities $(n_{v_1}, n_{v_2}, \ldots, n_{v_b})$ of the $b$ subtrees rooted at $v_1, v_2, \ldots, v_b$ are distributed as

$$\text{Multinomial}\Big(n_v - s_0, V_1, V_2, \ldots, V_b\Big).$$

# Examples of Split Trees

- The class of split trees includes many important random trees of logarithmic height, such as **binary search trees**, **m-ary search trees**, **quadtrees**, **median of $(2k+1)$-trees**, **simplex trees**, **tries** and **digital search trees**.

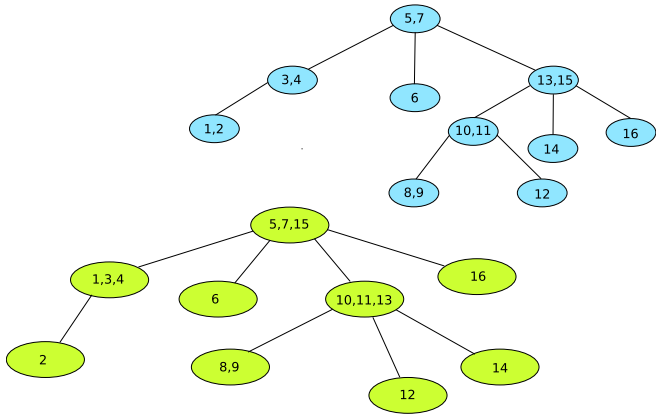



Figure: *A 3-ary and a 4-ary search tree constructed from the sequence* $7, 5, 15, 3, 4, 6, 1, 13, 11, 10, 2, 16, 8, 9, 14, 12$.

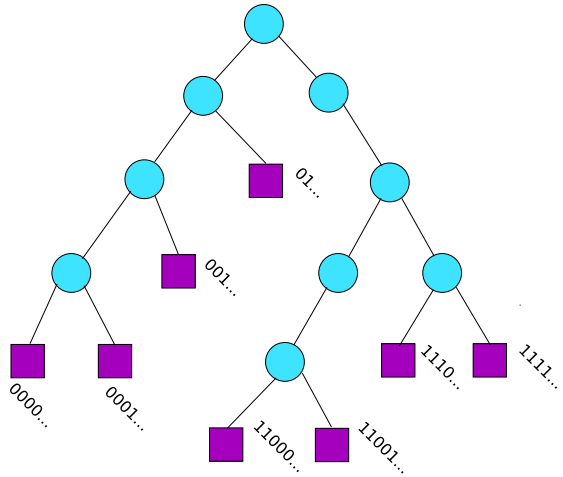
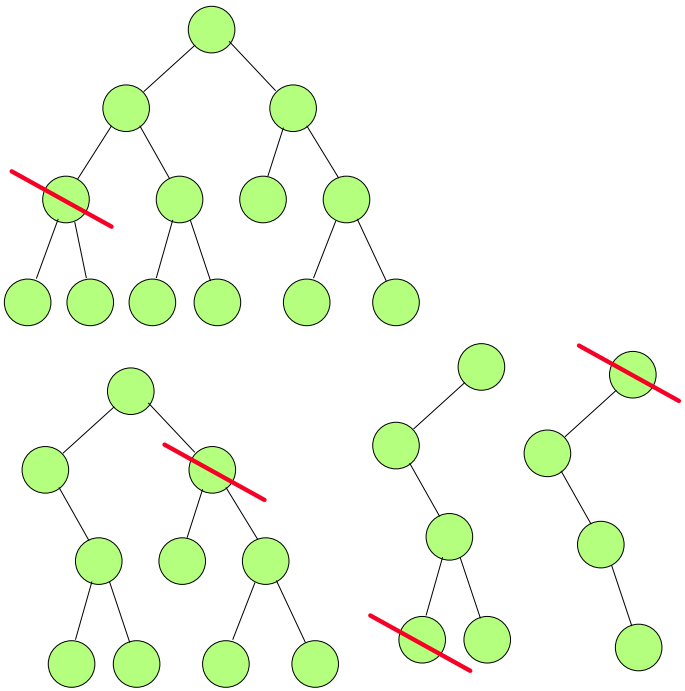



Figure: *A trie built from the strings* $0000\ldots$, $0001\ldots$, $001\ldots$, $01\ldots$, $11000, \ldots$, $11001, \ldots$, $1110\ldots$ *and* $1111, \ldots$.

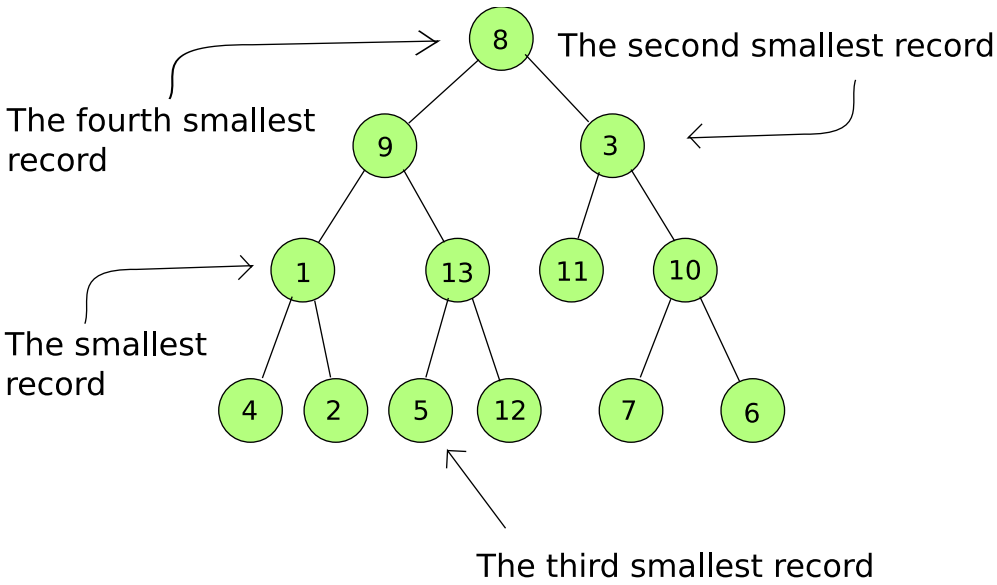
# What is a Cutting in a Rooted Tree?

- Choose one node at random.
- Cut in this node so that the tree separates into two parts, and keep only the part containing the root.
- Continue recursively until the root is cut. Let $X(T)$ denote the (random) number of cuts.



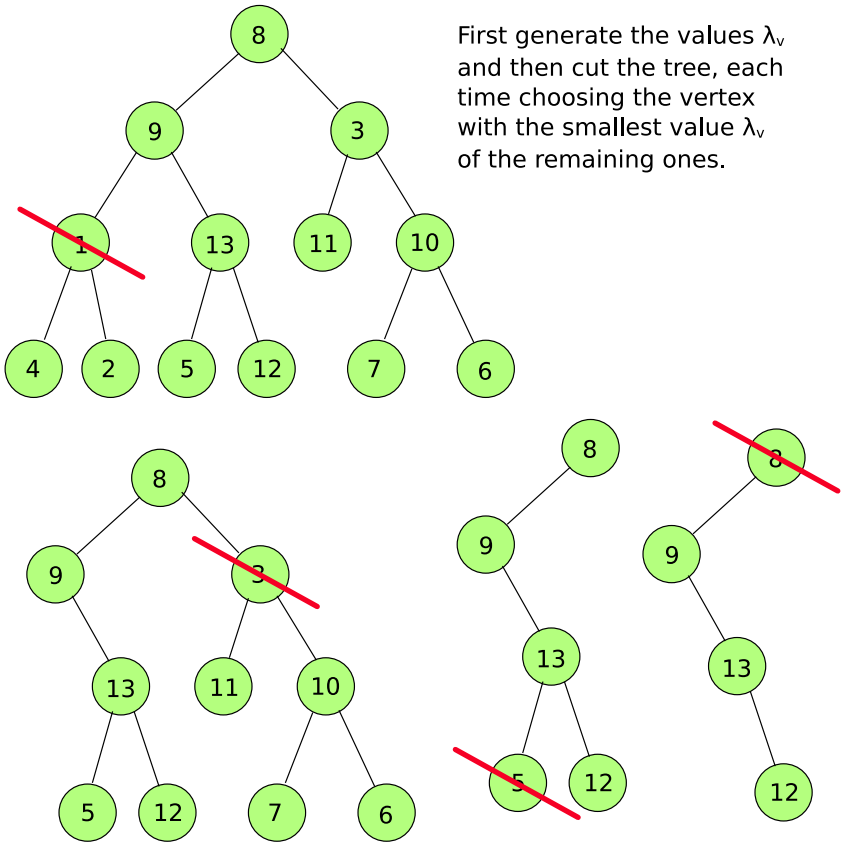
# What is a Record in a Rooted Tree?

- Let each node $v$ have a random value $\lambda_v$ attached to it. Assume that these values are i.i.d. with a continuous distribution.
- A value $\lambda_v$ is a **record** if it is the smallest value in the path from the root to $v$.



# Records and Cuttings in Rooted Trees

- The number of cuts $X(T)$ is equal in distribution to the number of records. (Janson 2004)
  Think! A node $v$ is cut at some time if and only if $\lambda_v$ is a record.



## Aim of Study

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

# Background

- Cutting down trees first introduced by Meir and Moon (1970). Essentially two random tree models have been considered:
- **In the first model the trees have height of order $\sqrt{n}$.** *Panholzer, Fill and Kapur have studied e.g., the well-known Cayley tree. Janson (2004) generalised their results and showed that the numbers of records (or cuts) of conditioned Galton–Watson trees are asymptotically Rayleigh distributed. A recent approach by Addario-Berry, Broutin and Holmgren is to show this result by defining a cutting down procedure for the Brownian continuum random trees of Aldous.*
- **In the second model the trees have height of order $\log n$.** *A large class of trees in this model are the random split trees. Janson (2004) showed that for the complete binary tree the number of cuts is asymptotically weakly 1-stable. Drmota, Iksanov, Moehle and Roesler, recently used analytic methods to show that the number of cuts in the random recursive tree is also weakly 1-stable.*

## Cuttings in Relation to Physics

- The number of cuttings in rooted trees is related to coalescent theory in Physics.
- In coalescent theory one studies the physical phenomenon when several blocks merge into one block. There is a markov process with transition probabilities $\lambda_{b,k}$ which gives the rate at which any $k$-tuple of blocks merges when there are $b$ blocks in total.
- Martin and Goldschmidt (2005) showed that the number of cuttings in a random recursive tree corresponds to the number of collision events that take place until there is just a single block in the Bolthausen-Sznitman coalescent.

# The Main Theorem

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

## Main Theorem

Suppose that $n \to \infty$. Then

$$\left( X(T_n) - \frac{\alpha n}{c \ln n} - \frac{\alpha n \ln\ln n}{c \ln^2 n} \right) \Big/ \frac{\alpha n}{c^2 \ln^2 n} \xrightarrow{d} -W, \qquad (1)$$

where $c$ and $\alpha$ are constants and $W$ has an infinitely divisible distribution more precisely a **weakly 1-stable distribution** with characteristic function

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

where $C$ is a constant.

## Infinitely Divisible Distributions

- A triangular array is a sequence of random variables $Z_{n,j},\ 1 \leq j \leq n$, so that the variables in each row, $n$, are independent and identically distributed. Typically the variables in different rows are not independent.
- A random variable $Z$ has an infinitely divisible distribution, if and only if, for all $n$, there is a triangular array $Z_{n,j},\ 1 \leq j \leq n$, such that

$$Z \stackrel{d}{=} \sum_{j=1}^{n} Z_{n,j}.$$

## $\alpha$-Stable Distributions

A distribution of a random variable $Z$ is $\alpha$-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 $X(T_n)$ by a sum of i.i.d. r.v. derived from $\lambda_v$ 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.*
- 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 $\log_2 n$

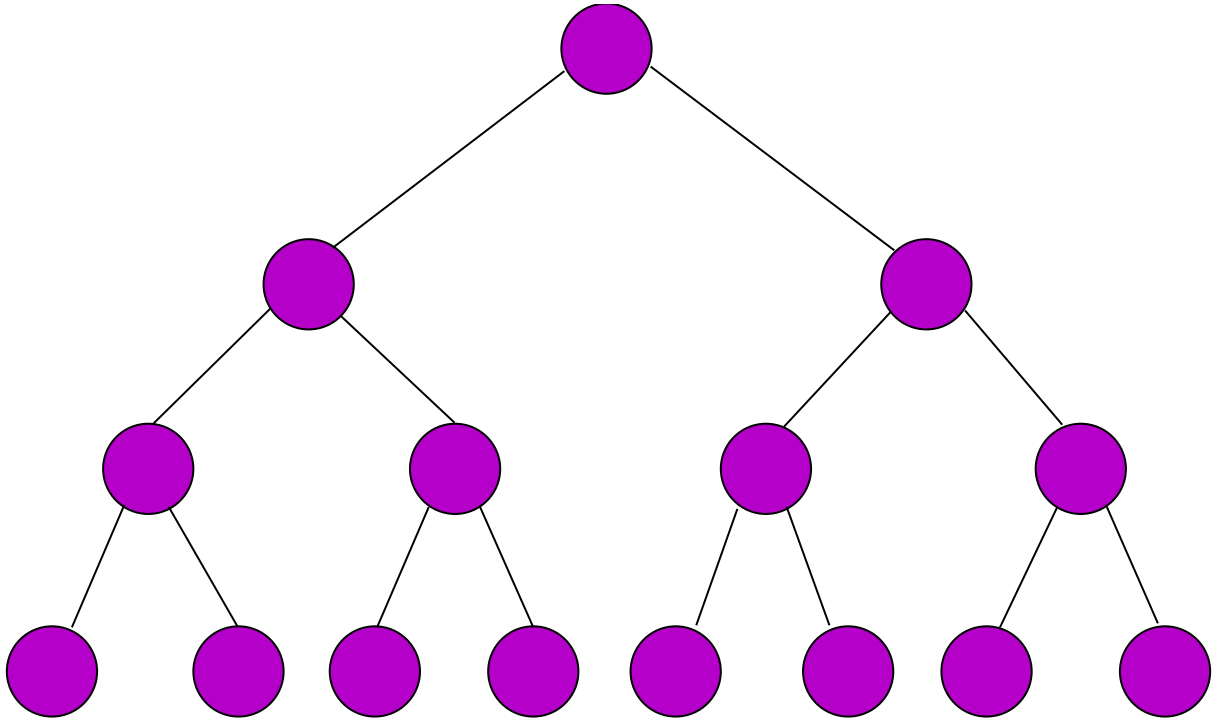


Figure: *A complete binary tree. All nodes except the leaves have two children.*

# Split Trees: Most Nodes Close to Depth $c \ln n$.

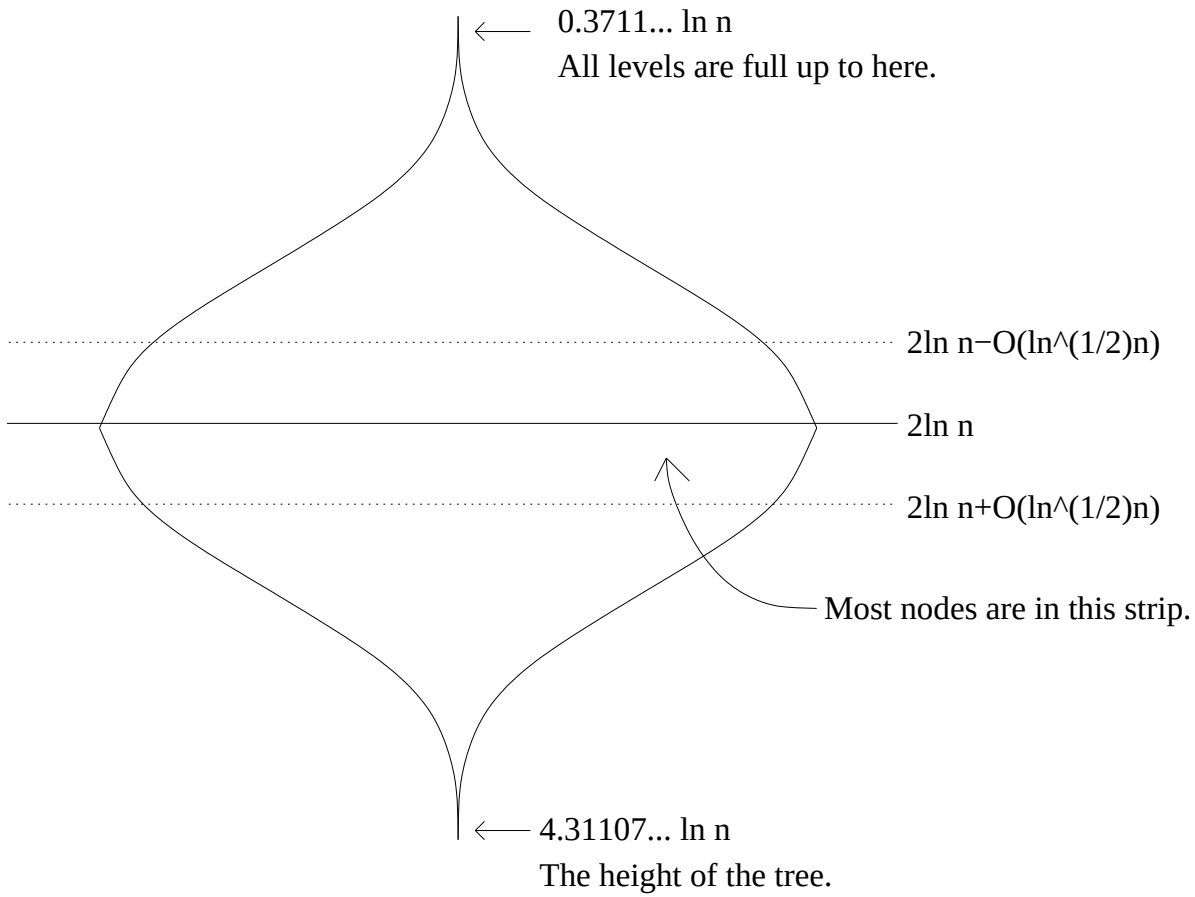



Figure: *This figure illustrates the shape of the binary search tree. The root is at the top. The horizontal width represents the number of nodes at each level. Most nodes are in a strip of width* $\mathcal{O}(\sqrt{\ln n})$ *around* $2 \ln n$.

# Subtree Sizes in Split Trees

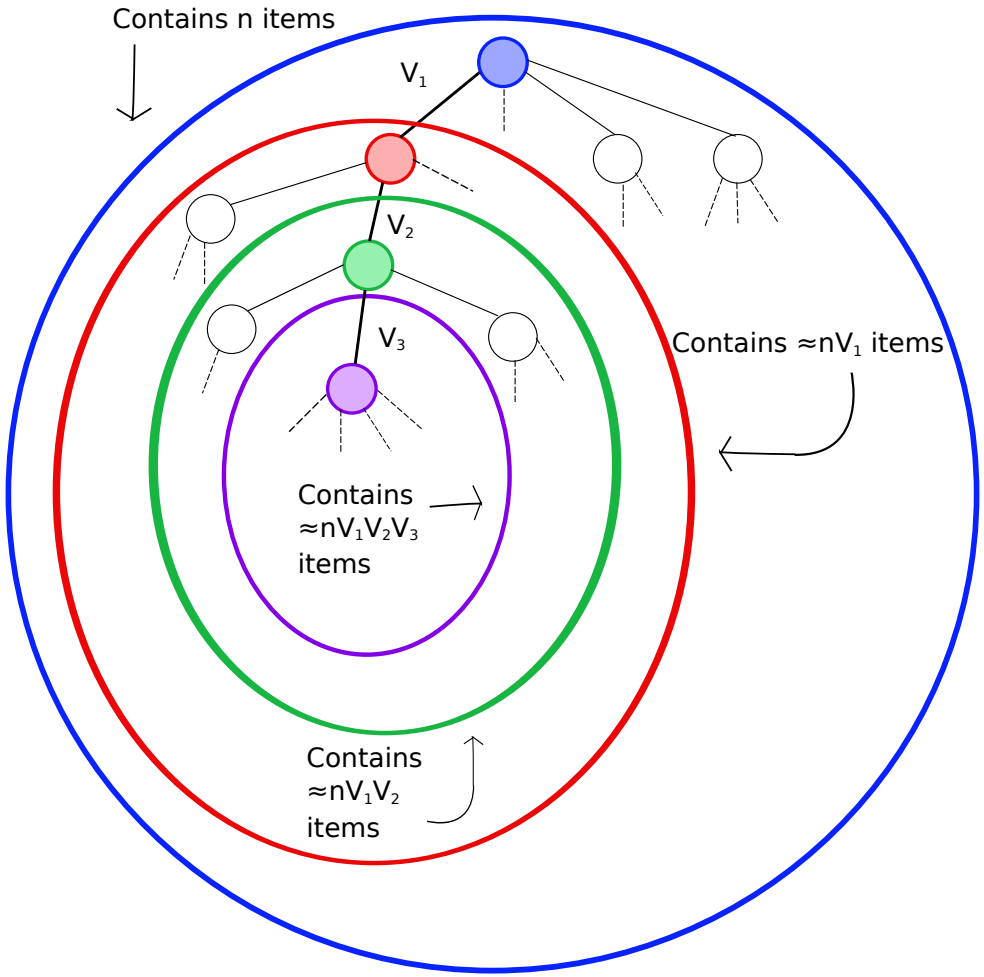



Figure: *Given all split vectors in the tree, $n_v$ for $v$ at depth $k$ is close to $nV_1V_2 \ldots V_k$, where the $V_r$'s are i.i.d. random variables distributed as the components in the split vector.*

## Subtree Sizes in Split Trees

- In a split tree with $n$ items, given the root's split vector $\mathcal{V}_\sigma = (V_1, \ldots, V_b)$, the numbers of items in the subtrees rooted at the root's children are close to $nV_1, \ldots, nV_b$.
- Let $n_v$ be the number of items in the subtree rooted at node $v$. Given all split vectors in the tree, $n_v$ for $v$ at depth $k$ is close to

$$nV_1V_2 \ldots V_k,$$

where $V_r$, $r \in \{1, \ldots, k\}$ are independent and identically distributed (i.i.d.) random variables (r.v).
The $V_r$'s are given by the split vectors associated with the nodes in the unique path from $v$ to the root.

## "Good" and "Bad" Vertices in Split Trees

- There is a central limit theorem for the depth of nodes so that "most" nodes lie at $c \ln n + \mathcal{O}\left(\sqrt{\ln n}\right)$. Devroye (1998)
- Let $d(v)$ denote the depth of a node $v$ in the split tree $T_n$. A node $v$ is called **good** if

$$c \ln n - \ln^{0.6} n \leq d(v) \leq c \ln n + \ln^{0.6} n,$$

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

# Advantage of Considering Records in Subtrees

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

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

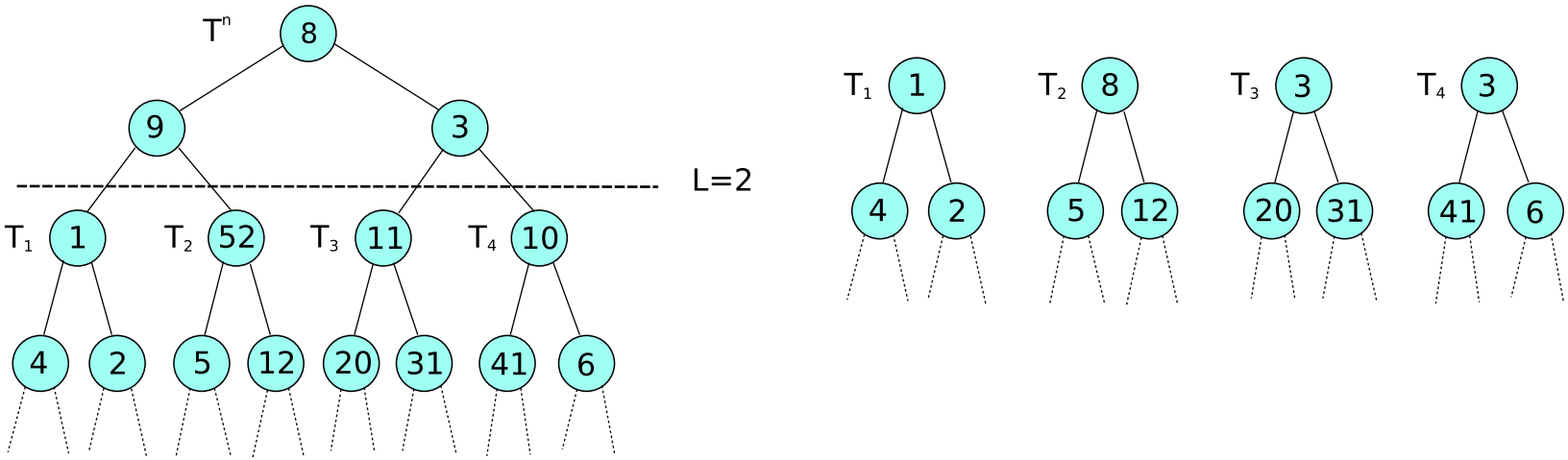



Figure: *The subtrees $T_1$, $T_2$, $T_3$, $T_4$ at depth $L = 2$ are considered. This example has $\Lambda_1 = 1$, $\Lambda_2 = 8$, $\Lambda_3 = 3$ and $\Lambda_4 = 3$.*

# Applying a Theorem for Triangular Arrays

- Using that $X(T_n) \approx \sum_{i=1}^{b^L} X(T_i)_{\Lambda_i}$, the normalized $X(T_n)$ in the Main Theorem can be expressed as

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

  where $\xi_v := \frac{n_v c \ln n}{n} \cdot e^{-\lambda_v c \ln n}$ and the $\xi_i'$'s are r.v.'s only depending on the $n_v$'s with $d(v) = L$.
- Conditioned on the $n_v$'s, the $\xi_v$'s are independent r.v.'s since the $\lambda_v$'s are independent, and the $\xi_i'$'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

The limit theorem for triangular null arrays requires that three conditions for the null array are fulfilled.

## Theorem 2

Suppose that $n \to \infty$ and choose any constant $C > 0$, then

$$(i) \quad \sup_v \mathbf{P}\left(\xi_v > x \middle| n_v\right) \to 0 \quad \text{for every } x > 0, \ \textit{i.e. } \{\xi_v\} \textit{ is a null array}$$

$$(ii) \quad \sum_{d(v)\le L} \mathbf{P}\left(\xi_v > x \middle| n_v\right) \xrightarrow{p} \nu(x,\infty) = \frac{c}{x} \quad \text{for every } x > 0,$$

$$(iii) \quad \sum_{d(v)\le L} \mathbf{E}\left(\xi_v \mathbf{1}[\xi_v \le C] \middle| n_v\right) + \sum_{i=1}^{n} \xi_i' \xrightarrow{p} K, \quad K \textit{ is a constant}$$

$$(iv) \quad \sum_{d(v)\le L} \mathbf{Var}\left(\xi_v \mathbf{1}[\xi_v \le C] \middle| n_v\right) \xrightarrow{p} cC.$$

## Theorem 2 implies the Main Theorem

- Recall that the normalized $X(T_n)$ in the Main Theorem can be expressed as $-\left(\sum_{d(v)\leq L}\xi_v + \sum_{i=1}^{n}\xi_i^{'}\right) + o_p(1)$.
- Theorem 2 shows that the necessary conditions for $\{\xi_v\}\bigcup\{\xi_i^{'}\}$ are fulfilled so that the limit theorem for convergence of sums of null arrays to infinitely divisible distributions can be applied to $\sum_{d(v)\leq L}\xi_v + \sum_{i=1}^{n}\xi_i^{'}$.
- **Thus, the Main Theorem is proved i.e. the normalized $X(T_n)$ converges to an infinitely divisible distribution. In particular the measure $\nu(x,\infty) = \frac{c}{x}$ in Theorem 2 implies that this distribution is weakly 1-stable.**

# Proof of Theorem 2

- Theorem 2, 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 $\{n_v,\ d(v) \leq L\}$.
- Recall that $n_v$ for $v$ at depth $k$, is close to $nV_1V_2\dots V_k$, where $V_r,\ r \in \{1,\dots k\}$ are independent r.v.'s distributed as the components $V_i$ in the split vector.
- Let $Y_k := -\sum_{r=1}^{k} \ln V_r$. Note that $nV_1V_2\dots V_k = ne^{-Y_k}$. In a binary search tree, $Y_k$ is distributed as a $\Gamma(k,1)$ r.v. since $V_r \stackrel{d}{=} U$, where $U$ is a uniform $U(0,1)$ r.v..

# Proof of Theorem 2 (continued)

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

$$U(t) = \sum_{k=1}^{\infty} b^k \mathbf{P}(Y_k \leq t) = \sum_{k=1}^{\infty} F_k(t), \tag{3}$$

  and let $F(t) := F_1(t) = b\mathbf{P}(V_i \leq t)$.
- 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

$$U(t) = (c + o(1))e^t.$$

# 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 both for introducing me to this problem area and for stimulating discussions and guidance throughout the work.