The Expected Height of Randomly Built Binary Search Trees Lecture Notes for ADS-MIRI

Amalia Duch

February 19, 2020

Let us define a *randomly built binary search tree* of n nodes (containing n different keys) as a binary search tree that is built by inserting successively the elements of a *random permutation* of the n different keys in the tree.

We say that a permutation π_n of n different keys is *random* if the $n!$ possible arrangements of the keys are equiprobable.

In particular, let us observe that in a random permutation π_n any of the n elements has the same probability $(1/n)$ of being the first one. Indeed, let element i-th $(1 \leq i \leq n)$ be the element that appears at the i-th position of a sequence in which all the n different keys are ordered increasingly. Then, the i-th element has a probability $p_i = 1/n$ of being the first one of a random permutation π_n of n different keys. Consequently, a randomly built binary search tree of size n (containing n nodes) would have the i-th element as its root with probability $p_i = 1/n$ and therefore a left subtree of size $i-1$ and a right subtree of size $n-i$ with the same probability.

Let us define now as X_n the expected height of a randomly built binary search tree of size $n > 0$. Then,

$$\begin{aligned}
X_n &= 1 + \sum_{i=1}^{n} p_i \max(X_{i-1}, X_{n-i}), && (1)\\
&= 1 + \frac{1}{n}\sum_{i=1}^{n} \max(X_{i-1}, X_{n-i}), && (2)\\
&\leq 1 + \frac{1}{n}\sum_{i=1}^{n} X_{i-1} + X_{n-i}, && (3)\\
&= 1 + \frac{2}{n}\sum_{i=1}^{n} X_{i-1}. && (4)
\end{aligned}$$

Let us observe therefore that, for $n > 0$, Recurrence 4 is similar to the recurrence for the average number of comparisons Q_n of the Quick sort algorithm over a vector of n elements, which is:

$$Q_n = n + 1 + \frac{2}{n} \sum_{i=1}^{n} Q_{i-1}, \tag{5}$$

with $Q_0 = 1$. To solve Recurrence 5 let us observe first that, for $n > 1$

$$Q_{n-1} = n + \frac{2}{n-1} \sum_{i=1}^{n-1} Q_{i-1}.$$

And therefore,

$$nQ_n - (n-1)Q_{n-1} = 2n + 2Q_{n-1},$$

rearranging and dividing both sides of previous equation by $n(n+1)$ we obtain that

$$\frac{Q_n}{n+1} = \frac{2}{n+1} + \frac{Q_{n-1}}{n}. \tag{6}$$

Telescoping and solving for Q_n we obtain that $Q_n = 2(n+1)H_{n+1}$, where H_n is the n-th harmonic number. Since $H_n = \Theta(\ln n)$ we have that $Q_n = \Theta(n \ln n)$.

Recurrence 4 is very similar also to the expected *Internal Path Length (IPL)* of a randomly built binary search tree. The IPL of a tree is defined as the sum of the lengths of all the paths in the tree form the root to the internal nodes. In particular, for randomly built binary search trees of n nodes we have that, the expected IPL(n) satisfies[1]

$$IPL(n) = n - 1 + \frac{2}{n} \sum_{i=0}^{n-1} IPL(i),$$

and therefore $IPL(n) = \Theta(n \ln n)$.

Let us now define as D_n the expected depth of a node in a randomly built binary search tree of n nodes. Observing that the sum of all these expected depths is $IPL(n)$[2] and, as we said, there are exactly n nodes, we have that $D_n = \Theta(\ln n)$. Therefore, one should reasonably expect, as it is the case, that $X_n = \Theta(\ln n)$.

However, it is not possible to obtain this result from Recurrence 4. Indeed, solving Recurrence 4 directly, using generating functions or via Roura's Continuous Master Theorem we obtain that $X_n = O(n)$, which is true but is not a tight upper bound. In fact, this upper bound was expected from Recurrence 4 since the recurrence is the same as the one that corresponds to visit all the nodes of the binary tree.

It turns out that, defining the random variable $Y_n = 2^{X_n}$ it is possible to obtain the logarithmic tight bound. Let us see how.

[1] Explain why.

[2] Explain why.

From Recurrence 2 we can derive the following inequality for Y_n,

$$Y_n \leq \frac{4}{n}\sum_{i=0}^{n-1} Y_i \tag{7}$$

Multiplying by n, and summing up to n we obtain that

$$nY_n \leq 4\sum_{i=0}^{n} Y_i - 4Y_n$$

And multiplying by z^n and summing up for all n we obtain:

$$\sum_{n\geq 0} nY_n z^n \leq \sum_{n\geq 0}\sum_{i=0}^{n} Y_i z^n - 4\sum_{n\geq 0} Y_n z^n \tag{8}$$

Defining the generating function $Y(z) = \sum_{n\geq 0} Y_n z^n$ the equation:

$$\sum_{n\geq 0} nY_n z^n = \sum_{n\geq 0}\sum_{i=0}^{n} Y_i z^n - 4\sum_{n\geq 0} Y_n z^n$$

yields to the differential equation:

$$Y'(z) = \frac{4Y(z)}{1-z}$$

with solution

$$\begin{aligned} Y(z) &= \frac{c}{(1-z)^4}{}^3 && (9)\\ &= \sum_{n\geq 0}\binom{n+3}{3} z^n && (10) \end{aligned}$$

Therefore

$$\begin{aligned} Y_n &\leq \binom{n+3}{3} && (11)\\ &= \frac{(n+3)(n+2)(n+1)}{6} && (12) \end{aligned}$$

And taking logarithms we have that $X_n = O(\ln n)$. It must be also that $X_n = \Omega(\ln n)$ which implies that $X_n = \Theta(\ln n)$.