# Average case analysis of Quick-Sort (distinct elements)

Kashif Murtaza (kashif.murtaza@pucit.edu.pk)
(Please inform me about errors if any)

***Courtesy:***

1. ***Mark Allen Weiss:*** *Data Structures and Algorithm Analysis in C++*
2. ***CLRS:*** *Introduction to Algorithms: Appendix A*

$T(n) = \Theta(1)$ For $n \le 1$ (base case)

Partition step takes $\Theta(n)$ time and the array can be partitioned into one of the following possibilities depending upon the pivot:

$$T(n) = \begin{cases} T(0) + T(n-1) + \Theta(n) \\ T(1) + T(n-2) + \Theta(n) \\ T(2) + T(n-3) + \Theta(n) \\ \qquad\qquad \vdots \\ T(n-1) + T(0) + \Theta(n) \end{cases}$$

Technically, T(0) will never occur in the code, but assuming a base case that handles empty array, we allow T(0) as well. All the above possibilities have equal probability to occur, so the average running time is sum of all possibilities divided by the total number of possibilities:

$$T(n) = \frac{1}{n}\sum_{i=0}^{n-1}(T(i) + T(n-i-1) + \Theta(n)) \quad \text{.......................... (1)}$$

$\forall i \ \ T(i)$ Occurs exactly 2 times in summation of above equation so:

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

Replacing $\Theta(n)$ with $cn$ and distributing summation

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

$$\Rightarrow T(n) = cn + \frac{2}{n}\sum_{i=0}^{n-1}T(i)$$

$$\Rightarrow nT(n) = cn^2 + 2\sum_{i=0}^{n-1}T(i) \quad \text{............................................. (2)}$$

Evaluating equation (2) at n-1 we get

$$(n-1)T(n-1) = c(n-1)^2 + 2\sum_{i=0}^{n-2} T(i) \text{ ..........................................} \textbf{(3)}$$

Subtracting equation (3) from equation (2) we get:

$$nT(n) - (n-1)T(n-1) = cn^2 - c(n-1)^2 + 2T(n-1)$$

$$\Rightarrow nT(n) = (n+1)T(n-1) + 2cn - c$$

$$\Rightarrow \frac{T(n)}{n+1} = \frac{T(n-1)}{n} + \frac{2c}{n+1} - \frac{c}{n(n+1)} \quad \text{........................................}(4)$$

Evaluating equation (4) at n-1 we get

$$\frac{T(n-1)}{n} = \frac{T(n-2)}{n-1} + \frac{2c}{n} - \frac{c}{n(n-1)}$$

Putting the value of $\frac{T(n-1)}{n}$ in equation (4) we get:

$$\frac{T(n)}{n+1} = \frac{T(n-2)}{n-1} + \frac{2c}{n} - \frac{c}{n(n-1)} + \frac{2c}{n+1} - \frac{c}{n(n+1)} \quad \text{....... ...................}(5)$$

Now again using iteration method, we find the value of $\frac{T(n-2)}{n-1}$ using equation (4) and put that value in equation (5). After k iterations we get:

$$\frac{T(n)}{n+1} = \frac{T(n-k)}{n-k+1} + \sum_{i=n-k+2}^{n+1} \frac{2c}{i} - \sum_{j=n-k+1}^{n} \frac{c}{j(j+1)}$$

The value of k should be n-1 to reach the base case and we get:

$$\frac{T(n)}{n+1} = \frac{T(1)}{2} + 2c\sum_{i=3}^{n+1} \frac{1}{i} - c\sum_{j=2}^{n} \frac{1}{j(j+1)}$$ : The first series is harmonic and is bounded by lgn

$$\Rightarrow \frac{T(n)}{n+1} = c_1 + O(\lg n) - c\sum_{j=2}^{n} \left(\frac{1}{j} - \frac{1}{j+1}\right)$$ : Using partial fractions $\frac{1}{j(j+1)} = \frac{1}{j} - \frac{1}{j+1}$

$$\Rightarrow \frac{T(n)}{n+1} = c_1 + O(\lg n) - c\left(\frac{1}{2} - \frac{1}{n+1}\right)$$ : Telescopic series

$$\Rightarrow \frac{T(n)}{n+1} = O(\lg n) \qquad \Rightarrow T(n) = O(n \lg n)$$