Design and Analysis of Algorithms I

QuickSort

Analysis II: The Key Insight

Average Running Time of QuickSort

QuickSort Theorem : for every input array of length n, the average running time of QuickSort (with random pivots) is $O(n\log(n))$.

Note : holds for every input. [no assumptions on the data]

- recall our guiding principles !
- "average" is over random choices made by the algorithm (i.e., the pivot choices)

Tim Roughgarden

The Story So Far

$C(\sigma)$ = # of comparisons between input elements

$X_{ij}(\sigma)$ = # of comparisons between $\underbrace{z_i \text{ and } z_j}_{i^{th},\ j^{th} \text{ smallest entries in array}}$

<u>Recall</u> : $E[C] = \sum_{i=1}^{n-1} \sum_{k=i+1}^{n} Pr[X_{ij} = 1]$

$= Pr[z_i\ z_j\ get\ compared]$

<u>Key Claim</u> : for all i < j, Pr[z_i, z_j get compared] = 2/(j-i+1)

Tim Roughgarden

Proof of Key Claim

$\Pr[z_i, z_j$ get compared $] = 2/(j-i+1)$

Fix z_i, z_j with $i < j$

Consider the set $z_i, z_{i+1}, \ldots, z_{j-1}, z_j$

Inductively : as long as none of these are chosen as a pivot, all are passed to the same recursive call.

Consider the first among $z_i, z_{i+1}, .., z_{j-1}, z_j$ that gets chosen as a pivot.

1. If z_i or z_j gets chosen first, then z_i and z_j get compared
2. If one of $z_{i+1}, \ldots, z_{j-1}$ gets chosen first then z_i and z_j are never compared [split into different recursive calls]

KEY INSIGHT

Proof of Key Claim (con'd)

1. z_i or z_j gets chosen first => they get compared
2. one of $z_{i+1},\ldots,z_{j-1}$ gets chosen first => z_i , z_j never compared

<u>Note</u> : Since pivots always chosen uniformly at random, each of $z_i,z_{i+1},\ldots,z_{j-1},z_j$ is equally likely to be the first

$\Rightarrow$Pr[z_i,z_j get compared] = 2/(j-i+1)

Choices that lead to case (1)

Total # of choices

<u>So</u> : $$E[C] = \sum_{i=1}^{n-1} \sum_{j=1}^{n} \frac{2}{j-i+1}$$

[Still need to show this is O(nlog(n))

Tim Roughgarden