QuickSort

Analysis I: A Decomp-osition Principle

Design and Analysis of Algorithms I

Average Running Time of QuickSort

<u>QuickSort Theorem</u> : for every input array of length n, the average running time of QuickSort (with random pivots) is O(nlog(n)).

<u>Note</u> : 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)

Preliminaries

Fix input array A of length n

Sample Space Ω = all possible outcomes of random choices in QuickSort (i.e., pivot sequences)

Key Random Variable : for $\sigma \in \Omega$

$C(\sigma)$ = # of comparisons between two input elements made by QuickSort (given random choices σ)

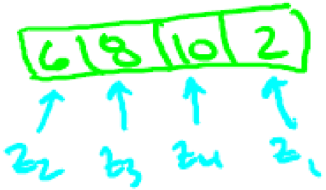
Lemma: running time of QuickSort dominated by comparisons.

There exist constant c s.t. for all $\sigma \in \Omega$, $RT(\sigma) \leq c \cdot C(\sigma)$

Remaining goal : E[C] = O(nlog(n))

Building Blocks

<u>Note</u> can't apply Master Method [random, unbalanced subproblems]



[A = final input array]

<u>Notation</u> : z_i = i^{th} smallest element of A

For $\sigma \in \Omega$, indices i< j

$X_{ij}(\sigma)$ = # of times z_i,z_j get compared in QuickSort with pivot sequence σ

Fix two elements of the input array. How many times can these two elements get compared with each other during the execution of QuickSort?

- 1
- 0 or 1
- 0, 1, or 2
- Any integer between 0 and $n-1$

Reason : two elements compared only when one is the pivot, which is excluded from future recursive calls.

Thus : each X_{ij} is an "indicator" (i.e., 0-1) random variable

A Decomposition Approach

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

$X_{ij}(\sigma)$ = # of comparisons between z_i and z_j

Thus : $\forall \sigma, C(\sigma) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}(\sigma)$

complicated

simple

By Linearity of Expectation : $E[C] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} E[X_{ij}]$

Since $E[X_{ij}] = 0 \cdot Pr[X_{ij} = 0] + 1 \cdot Pr[X_{ij} = 1] = Pr[X_{ij} = 1]$

Thus : $E[C] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} Pr[z_i, z_j \text{ get compared}] \quad (*)$

Next video

Gopal Shangari

A General Decomposition Principle

1. Identify random variable Y that you really care about

2. Express Y as sum of indicator random variables :

$$Y = \sum_{l=1}^{m} X_e$$

3. Apply Linearity of expectation :

$$E[Y] = \sum_{l=1}^{m} Pr[X_e = 1]$$

"just" need to understand these!

Gopal Shangari

QuickSort

Analysis II:
The Key Insight

Design and Analysis
of Algorithms I

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(nlog(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)

The Story So Far

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

$X_{ij}(\sigma)$ = # of comparisons between z_i and z_j

i^{th}, j^{th} smallest entries in array

Recall : $E[C] = \sum_{i=1}^{n-1} \sum_{k=i+1}^{n} Pr[X_{ij} = 1]$ $= Pr[z_i\ z_j\ get\ compared]$

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

Gopal Shangari

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

Note : 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

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

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

Gopal Tim Shangari

QuickSort

Analysis III: Final Calculations

Design and Analysis of Algorithms I

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(nlog(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., pivot choices)

NEXTCORE AI

The Story So Far

$$E[C] = 2\sum_{i=1}^{n-1}\sum_{j=1}^{n}\frac{1}{j-i+1}$$

How big can this be ?

(*)

<= n choices for i

$\theta(n^2)\ \ terms$

Note : for each fixed i, the inner sum is

$$\sum_{j=i+1}^{n}\frac{1}{j-i+1} = 1/2 + 1/3 + \ldots$$

$$So \ \ E[C] \le 2\cdot n\cdot\sum_{k=2}^{n}\frac{1}{k}$$

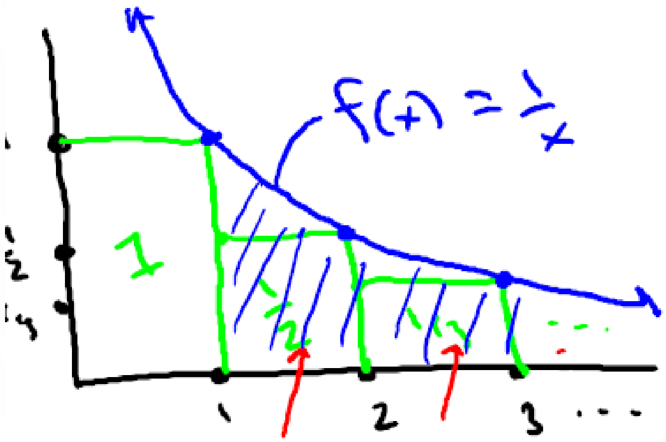
Claim : this is <= ln(n)

Completing the Proof

$$E[C] \leq 2 \cdot n \cdot \sum_{k=2}^{n} \frac{1}{k}$$

$$Claim \quad \sum_{k=2}^{n} \frac{1}{k} \leq \ln n$$

Proof of Claim



$$So \quad \sum_{k=2}^{n} \frac{1}{n} \leq \int_{1}^{n} \frac{1}{x} dx$$

$$= \ln x \, \Big|_{1}^{n}$$

$$= \ln n - \ln 1$$

$$= \ln n$$

Q.E.D. (CLAIM)

So :
E[C] <=
2n ln n

Q.E.D.