Chapter 7
Randomization

Algorithm Theory
WS 2017/18

Fabian Kuhn

Randomized Quicksort

Quicksort:

S	v

$S_\ell < v$	v	$S_r > v$

function Quick (S: sequence): sequence;

{returns the sorted sequence S}

begin

if $\#S \leq 1$ then **return** S

else { choose pivot element v in S;

partition S into S_ℓ with elements $< v$,

and S_r with elements $> v$

return | Quick(S_ℓ) | v | Quick(S_r) |

end;

Randomized Quicksort Analysis

Randomized Quicksort: pick uniform random element as pivot

Running Time of sorting n elements:

- Let's just count the number of comparisons
- In the partitioning step, all $n - 1$ non-pivot elements have to be compared to the pivot
- **Number of comparisons:**

$$\boldsymbol{n-1\ +\ \#\text{comparisons in recursive calls}}$$

- **If rank of pivot is r:**
  recursive calls with $\boldsymbol{r-1}$ and $\boldsymbol{n-r}$ elements

Law of Total Expectation

- Given a random variable X and
- a set of events $A_1, \ldots, A_k$ that partition Ω
  - E.g., for a second random variable Y, we could have
$$A_i \coloneqq \{\omega \in \Omega : Y(\omega) = i\}$$

Law of Total Expectation

$$\mathbb{E}[X] = \sum_{i=1}^{k} \mathbb{P}(A_i) \cdot \mathbb{E}[X \mid A_i] = \sum_{y} \mathbb{P}(Y = y) \cdot \mathbb{E}[X \mid Y = y]$$

Example:

- X: outcome of rolling a die
- $A_0 = \{X \text{ is even}\}, \;\; A_1 = \{X \text{ is odd}\}$

Randomized Quicksort Analysis

Random variables:

- C: total number of comparisons (for a given array of length n)
- R: rank of first pivot
- C_ℓ, C_r: number of comparisons for the 2 recursive calls

$$\mathbb{E}[C] = n - 1 + \mathbb{E}[C_\ell] + \mathbb{E}[C_r]$$

Law of Total Expectation:

$$\begin{aligned}\mathbb{E}[C] &= \sum_{r=1}^{n} \mathbb{P}(R = r) \cdot \mathbb{E}[C|R = r] \\ &= \sum_{r=1}^{n} \mathbb{P}(R = r) \cdot (n - 1 + \mathbb{E}[C_\ell|R = r] + \mathbb{E}[C_r|R = r])\end{aligned}$$

Randomized Quicksort Analysis

We have seen that:

$$\mathbb{E}[C] = \sum_{r=1}^{n} \mathbb{P}(R = r) \cdot (n - 1 + \mathbb{E}[C_\ell | R = r] + \mathbb{E}[C_r | R = r])$$

Define:

- $\boldsymbol{T(n)}$: expected number of comparisons when sorting n elements

$$\mathbb{E}[C] = T(n)$$
$$\mathbb{E}[C_\ell | R = r] = T(r-1)$$
$$\mathbb{E}[C_r | R = r] = T(n-r)$$

Recursion:

$$\boldsymbol{T(n) = \sum_{r=1}^{n} \frac{1}{n} \cdot \big(n - 1 + T(r-1) + T(n-r)\big)}$$

$$\boldsymbol{T(0) = T(1) = 0}$$

Randomized Quicksort Analysis

Theorem: The expected number of comparisons when sorting n elements using randomized quicksort is $T(n) \leq 2n \ln n$.

Proof:

$$T(n) = \sum_{r=1}^{n} \frac{1}{n} \cdot \big(n - 1 + T(r-1) + T(n-r)\big), \qquad T(0) = 0$$

Randomized Quicksort Analysis

Theorem: The expected number of comparisons when sorting n elements using randomized quicksort is $T(n) \leq 2n \ln n$.

Proof:

$$T(n) \leq n - 1 + \frac{4}{n} \cdot \int_1^n x \ln x \, dx$$

$$\int x \ln x \, dx = \frac{x^2 \ln x}{2} - \frac{x^2}{4}$$

Alternative Analysis

Array to sort: [7 , 3 , 1 , 10 , 14 , 8 , 12 , 9 , 4 , 6 , 5 , 15 , 2 , 13 , 11]

Viewing quicksort run as a tree:

Comparisons

- Comparisons are only between pivot and non-pivot elements
- Every element can only be the pivot once:
  → every 2 elements can only be compared once!

- W.l.o.g., assume that the elements to sort are $1, 2, \ldots, n$
- Elements i and j are compared if and only if either i or j is a pivot before any element $h: i < h < j$ is chosen as pivot
  - i.e., iff i is an ancestor of j or j is an ancestor of i

$$\mathbb{P}(\text{comparison betw.}\, i \text{ and } j) = \frac{2}{j - i + 1}$$

Counting Comparisons

Random variable for every pair of elements (i, j):

- $$\boldsymbol{X_{ij}} = \begin{cases} 1, & \text{if there is a comparison between } i \text{ and } j \\ 0, & \text{otherwise} \end{cases}$$

Number of comparisons: $\boldsymbol{X}$

$$X = \sum_{i<j} X_{ij}$$

- What is $\mathbb{E}[X]$?

Randomized Quicksort Analysis

Theorem: The expected number of comparisons when sorting n elements using randomized quicksort is $T(n) \leq 2n \ln n$.

Proof:

- **Linearity of expectation:**
  For all random variables $X_1, \dots, X_n$ and all $a_1, \dots, a_n \in \mathbb{R}$,

$$\mathbb{E}\left[\sum_{i}^{n} a_i X_i\right] = \sum_{i}^{n} a_i \mathbb{E}[X_i] .$$

Randomized Quicksort Analysis

Theorem: The expected number of comparisons when sorting n elements using randomized quicksort is $T(n) \leq 2n \ln n$.

Proof:

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

Quicksort: High Probability Bound

- We have seen that the number of comparisons of randomized quicksort is $O(n \log n)$ in expectation.

- Can we also show that the number of comparisons is $O(n \log n)$ with high probability?

- **Recall:**

  On each recursion level, each pivot is compared once with each other element that is still in the same “part”

Counting Number of Comparisons

- We looked at 2 ways to count the number of comparisons
  - recursive characterization of the expected number
  - number of different pairs of values that are compared

Let's consider yet another way:

- Each comparison is between a pivot and a non-pivot
- How many times is a specific array element x compared as a non-pivot?

Value x is compared as a non-pivot to a pivot once in every recursion level until one of the following two conditions apply:

1. x is chosen as a pivot
2. x is alone

Successful Recursion Level

- Consider a specific recursion level ℓ
- Assume that at the beginning of recursion level ℓ, element x is in a sub-array of length K_ℓ that still needs to be sorted.
- If x has been chosen as a pivot before level ℓ, we set $K_\ell \coloneqq 1$

Definition: We say that recursion level ℓ is successful for element x iff the following is true:

$$K_{\ell+1} = 1 \quad \text{or} \quad K_{\ell+1} \leq \frac{2}{3} \cdot K_\ell$$

Successful Recursion Level

Lemma: For every recursion level ℓ and every array element x, it holds that level ℓ is successful for x with probability at least $^1/_3$, independently of what happens in other recursion levels.

Proof:

Number of Successful Recursion Levels

Lemma: If among the first ℓ recursion levels, at least $\log_{3/2}(n)$ are successful for element x, we have $K_\ell = 1$.

Proof:

Chernoff Bounds

- Let $X_1, \dots, X_n$ be independent 0-1 random variables and define $p_i := \mathbb{P}(X_i = 1)$.
- Consider the random variable $X = \sum_{i=1}^{n} X_i$
- We have $\mu := \mathbb{E}[X] = \sum_{i=1}^{n} \mathbb{E}[X_i] = \sum_{i=1}^{n} p_i$

Chernoff Bound (Lower Tail):

$$\forall \boldsymbol{\delta} > \mathbf{0}: \ \mathbb{P}(\boldsymbol{X} < (\mathbf{1} - \boldsymbol{\delta})\boldsymbol{\mu}) < \boldsymbol{e}^{-\boldsymbol{\delta}^2 \boldsymbol{\mu}/2}$$

Chernoff Bound (Upper Tail):

$$\forall \boldsymbol{\delta} > \mathbf{0}: \ \mathbb{P}(\boldsymbol{X} > (\mathbf{1} + \boldsymbol{\delta})\boldsymbol{\mu}) < \left(\frac{\boldsymbol{e}^{\boldsymbol{\delta}}}{(\mathbf{1} + \boldsymbol{\delta})^{\mathbf{1}+\boldsymbol{\delta}}}\right)^{\boldsymbol{\mu}} < \boldsymbol{e}^{-\boldsymbol{\delta}^2 \boldsymbol{\mu}/3}$$

holds for $\delta \leq 1$

Chernoff Bounds, Example

Assume that a fair coin is flipped n times. What is the probability to have

1. less than $n/3$ heads?

2. more than $0.51n$ tails?

3. less than ${}^{n}/_{2} - \sqrt{c \cdot \; n \ln n}$ tails?

Proof of Chernoff Bound

- Independent Bernoulli random variables $X_1, X_2, \dots, X_n$
- $\mathbb{P}(X_i = 1) \geq p_i, X \coloneqq \sum_{i=1}^n X_i, \mu \coloneqq \sum_{i=1}^n p_i \geq \mathbb{E}[X]$

Chernoff Lower Tail: $\mathbb{P}(X < (1-\delta)\mu) < e^{-\delta^2\mu/2}$

Recall

- Markov Inequality: Given non-negative rand. var. $Z \geq 0$

$$\forall t > 0\colon\ \mathbb{P}(Z > t) < \frac{\mathbb{E}[Z]}{t}$$

- Independent random variables Y, Z:

$$\mathbb{E}[Y \cdot Z] = \mathbb{E}[Y] \cdot \mathbb{E}[Z]$$

Proof of Chernoff Bound

- $\mathbb{P}(X_i = 1) \geq p_i, X \coloneqq \sum_{i=1}^{n} X_i, \mu \coloneqq \sum_{i=1}^{n} p_i \geq \mathbb{E}[X]$

Chernoff Lower Tail: $\mathbb{P}(\boldsymbol{X} < (\mathbf{1} - \boldsymbol{\delta})\boldsymbol{\mu}) < \boldsymbol{e}^{-\boldsymbol{\delta}^2\boldsymbol{\mu}/2}$

Proof of Chernoff Bound

- $\mathbb{P}(X_i = 1) \geq p_i, X \coloneqq \sum_{i=1}^{n} X_i, \mu \coloneqq \sum_{i=1}^{n} p_i \geq \mathbb{E}[X]$

Chernoff Lower Tail: $\mathbb{P}(\boldsymbol{X} < (\mathbf{1} - \boldsymbol{\delta})\boldsymbol{\mu}) < \boldsymbol{e}^{-\boldsymbol{\delta}^2\boldsymbol{\mu}/\mathbf{2}}$

Proof of Chernoff Bound

- $\mathbb{P}(X_i = 1) \geq p_i,\, X \coloneqq \sum_{i=1}^{n} X_i,\, \mu \coloneqq \sum_{i=1}^{n} p_i \geq \mathbb{E}[X]$

Chernoff Lower Tail: $\mathbb{P}(\boldsymbol{X} < (\boldsymbol{1} - \boldsymbol{\delta})\boldsymbol{\mu}) < \boldsymbol{e}^{-\boldsymbol{\delta}^2\boldsymbol{\mu}/\boldsymbol{2}}$

Number of Comparisons for x

Lemma: For every array element x, with high probability, as a non-pivot, x is compared to a pivot at most $O(\log n)$ times.

Proof:

Number of Comparisons for x

Lemma: For every array element x, with high probability, as a non-pivot, x is compared to a pivot at most $O(\log n)$ times.

Proof:

Number of Comparisons

Theorem: With high probability, the total number of comparisons is at most $O(n \log n)$.

Proof:

Types of Randomized Algorithms

Las Vegas Algorithm:

- always a correct solution
- running time is a random variable

- **Example:** randomized quicksort, contention resolution

Monte Carlo Algorithm:

- probabilistic correctness guarantee (**m**ostly **c**orrect)
- fixed (deterministic) running time

- **Example:** primality test