CSE 6311
Advanced Computational Models and Algorithms

Instructor: Prof. Gautam Das
Lecture node April 7, 2009
CSE UTA
Na Li
Outline

• Randomized Algorithm for Quick Sort
• Las Vegas Alg. & Monte Carlo Alg.
Outline

• Randomized Algorithm for Quick Sort
• Las Vegas Alg. & Monte Carlo Alg.
Randomized Algorithm for Quick Sort

- Randomized Algorithm of Quick Sort
- Analysis of Randomized Algorithm
Randomized Algorithm for Quick Sort

• Randomized Algorithm of Quick Sort
• Analysis of Randomized Algorithm
Randomized Algorithm for Quick Sort

Quick Sort:
• Input: $S=\{x_1, x_2, \ldots, x_n\}$
• Output: A sorted list of numbers

Quick Sort ($S$):
1) **Randomly** pick a pivot element $P$ from $S$
2) Partition $S$ into two parts, elements on the left part are not larger than $P$ and those on the right part are not smaller than $P$. $L \leq P \leq R$
3) Quick Sort (Left)
4) Quick Sort (Right)
Randomized Algorithm for Quick Sort

• Randomized Algorithm of Quick Sort
• Analysis of Randomized Algorithm
Analysis of Randomized Algorithm

Steps:
1) The relationship of $X_{ij}$ and $T(n)$
   $X_{ij}$ is a boolean variable, indicating whether $i$ was compared with $j$
   ($X_{ij}=1$, if $i$ is compared with $j$, otherwise, $X_{ij}=0$.)

2) The relationship of $E[T(n)]$ and $E[X_{ij}]$

3) The relationship of $P_{ij}$ and $E[X_{ij}]$
   $P_{ij}$ is the probability that $i$ is compared with $j$.

4) Calculate $E[T(n)]$
Analysis of Randomized Algorithm

Step 1: \[ T(n) = \sum_{i=1}^{n-1} \sum_{j>i}^{n} X_{ij} \]

Step 2: \[ E[T(n)] = E[\sum_{i=1}^{n-1} \sum_{j>i}^{n} X_{ij}] = \sum_{i=1}^{n-1} \sum_{j>i}^{n} E[X_{ij}] \]

<table>
<thead>
<tr>
<th>X_{1,2}</th>
<th>X_{1,3}</th>
<th>X_{1,4}</th>
<th>X_{1,5}</th>
<th>...</th>
<th>X_{n-1,n}</th>
<th>( \sum_{i} \sum_{j} X_{ij} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st round</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>84</td>
</tr>
<tr>
<td>2nd round</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>79</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>infinite rounds</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>Average ( X_{ij} ) = ( \frac{\sum_{(\text{round})} X_{ij}}{# \text{of rounds}} )</td>
<td>0.76</td>
<td>0.68</td>
<td>0.9</td>
<td>0.7</td>
<td>0.54 ( E[T(n)] )</td>
<td></td>
</tr>
</tbody>
</table>
Analysis of Randomized Algorithm

Step 3:

$$P_{ij} = E[X_{ij}]$$

Through the example below, we know that $i$ and $j$ are compared only if one ends up as a descendent of the other.

eg. \{1, 2, 3, 4, 5\}

First, pick 2

Second, pick 4
Analysis of Randomized Algorithm

Step3:
So for i and j, there are three possible cases:

For Case 1 and Case 2, i and j will be compared.
For Case 3, they will not be compared.
Analysis of Randomized Algorithm

Step 3:

If we select elements between \(i+1\) and \(j-1\), \(i\) and \(j\) will never be compared.

If we select elements \(i\) or \(j\), \(i\) and \(j\) will be compared.

Conclusion:

\[
P_{ij} = \frac{2}{j - i + 1}
\]
Analysis of Randomized Algorithm

Step 4:

\[ E[T(n)] = \sum_{i=1}^{n-1} \sum_{j>i}^{n} \frac{2}{j-i+1} = \sum_{i=1}^{n-1} \sum_{k=2}^{n-i+1} \frac{2}{k} \]

\[ \leq \sum_{i=1}^{n} \sum_{k=1}^{n} \frac{2}{k} = 2n \sum_{k=1}^{n} \frac{1}{k} = O(n \lg n) \]
Outline

• Randomized Algorithm for Quick Sort
• Las Vegas Alg. & Monte Carlo Alg.
Las Vegas Alg. & Monte Carlo Alg.

Las Vegas Alg.
1) It always produces the right answer.
2) Its running time is a random variable.

Monte Carlo Alg.
1) The answer is approximate.
2) Its running time is bounded by a fixed approximated/deterministic value.
Las Vegas Alg. & Monte Carlo Alg.

**Min Cut Graph:**

Find the **smallest** set of edges so that removing them will make the graph disconnected.

For **deterministic algorithm**, for each pair of nodes, run Maximum Flow Algorithm to find the min cut and select the smallest set.

For **randomized algorithm**, assume the size of the opt min cut is k, if the graph has n vertices, the number of edges is not smaller than nk/2, since the degree of each node is at least k.
Las Vegas Alg. & Monte Carlo Alg.

The min cut in the graph below is 2.