COT 6936: Topics in Algorithms

Giri Narasimhan

ECS 254A / EC 2443; Phone: x3748

giri@cs.fiu.edu

http://www.cs.fiu.edu/~giri/teach/COT6936_S10.html

https://online.cis.fiu.edu/portal/course/view.php?id=427

Randomization

- Randomized Algorithms: Uses values generated by random number generator to decide next step
- Often easier to implement and/or more efficient
- Applications
  - Used in protocol in "Ethernet Cards" to decide when it next tries to access the shared medium
  - Primality testing & cryptography
  - Monte Carlo simulations

QuickSort vs Randomized QuickSort

QuickSort

- Pick a fixed pivot
- Partition input based on pivot into two sets
- Recursively sort the two partitions

Randomized QuickSort

- Pick a random pivot
- Partition input based on pivot into two sets
- Recursively sort the two partitions

QuickSort: Probabilistic Analysis

- Expected rank of pivot = n/2 (Why?)
- Thus expected size of sublists after partition = n/2
- Hence the recurrence $T(n) = 2T(n/2) + O(n)$
- Average time complexity = $T(n) = O(n \log n)$

New Quicksort: Randomized Analysis

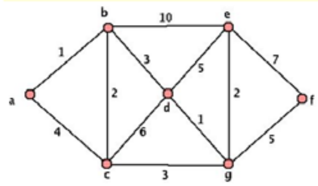
- Let X_{ij} be a random variable representing the number of times items i and j are compared by the algorithm.
- Expected time complexity = expected value of sum of all random variables X_{ij}.
- $Pr(X_{ij} = 1) = 2/(j - i + 1)$ (Why?)
- T(n) = ?

Analysis

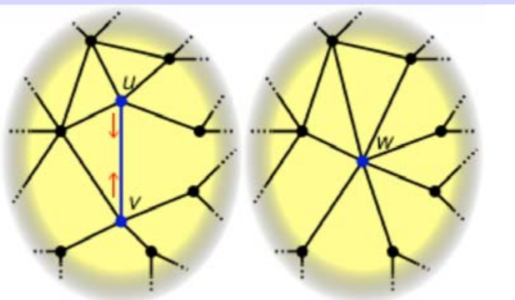
- $\Sigma_i \Sigma_j 2/(j-i+1) = \Sigma_i \Sigma_{k=2..n-i+1} 2/k$

Cut-Sets & Min-Cuts

- Example 1: ({a,b,c,d}, {e,f,g})
  - Weight = 19
- Example 2: ({a,b,g}, {c,d,e,f})
  - Weight = 30
- Example 3: ({a}, {b,c,d,e,f,g})
  - Weight = 5



Edge Contraction



http://en.wikipedia.org/wiki/Edge_contraction

Edge Contractions and Min-Cuts

- Lemma: If you are not contracting an edge from the cut-set, edge contractions do not affect the size of min-cuts.
- Observation: Most edges are not part of the min-cut.
- Idea: Use randomization

Randomized Algorithms: Min-Cut

- Assume that the Min-cut is of size k
- Pick a random edge
- Prob {edge is not in Min-cut} ≥ 1 – 2/n (why?)
- Prob {Min-cut is output} ≥ 2/n(n – 1) (why?)

Monte Carlo vs Las Vegas

- Monte Carlo algorithms: sometimes incorrect, but with bounded probability
  - One-sided versus two-sided errors
- Las Vegas algorithms: always correct, but with variable run times

Chain Hashing

- Balls and Bins Model
  - Throw m balls into n bins
  - Location of each ball chosen independently and uniformly at random
- Interesting questions to ask
  - How many balls in a bin on the average?
  - How many bins are empty?
  - How many balls in the fullest bin?
  - If m=n, how many bins are expected to have > 1 ball in it?

Power of Two Choices

- Hashing with **two** hash functions
  - Dramatically reduces the expected size of the largest bin while doubling the average search cost.
- Dynamic Resource Allocation
  - Multiple identical resources to choose from
    - Find load of each one and pick least loaded
    - Pick random resource
    - Sample 2 random resources and pick less loaded one

Bloom Filters

- Used to test set membership by using bit arrays to indicate which positions have been hashed to.

0	0	1	1	0	1	0	0	1

- Use k hash functions instead of 1.
- How large should k be for given error bound?

Breaking symmetry

- Many users want to share a resource
  - Want to pick a permutation quickly
  - Hash to 2^b bits and sort them
  - If $b = 3\log_2 n$ then two users will have distinct hash values with probability 1-1/n