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_S12.html

https://moodle.cis.fiu.edu/v2.1/course/view.php?id=174

COT 6936: Topics in Algorithms

Online Algorithms

How to Analyze Online Algorithms?

- Competitive analysis
  - Compare with optimal offline algorithm (OPT)
- Algorithm A is α-competitive if there exists constants b such that for every sequence of inputs σ:
  - $cost_A(\sigma) \leq \alpha cost_{OPT}(\sigma) + b$

How to Analyze Rand Online Algorithms?

- Algorithm A is α-competitive if there exists constants b such that for every sequence of inputs σ:
  - $cost_A(\sigma) \leq \alpha cost_{OPT}(\sigma) + b$
- Randomized Algorithm R is α-competitive if there exists constants b such that for every sequence of inputs σ:
  - $E[cost_R(\sigma)] \leq \alpha cost_{OPT}(\sigma) + b$

Adversary provides request sequence at start

Randomized Online algorithms

- Lower bound does not apply to randomized algorithms
  - Lower bound on randomized algorithms = H_k
- Proof uses 2 main principles
  - Cover time of a random walk on K_{k+1} is kH_k
  - Lower bound on competitiveness of randomized algorithms equals competitiveness of best deterministic algorithm A on "worst-case" distribution on request sequence
- H_k is k-th Harmonic number and < ln(k) + 1

Randomized Algorithm: RANDOM

- On a miss:
  - Evict an item chosen uniformly at random from all k items
- RANDOM is k-competitive

Marker Algorithm

- Each of k pages has a marker bit
- Algorithm proceeds in rounds with invariant:
  - At start of round all pages are unmarked
- In each round
  - If request is a hit: mark page
  - If request is a miss:
    - If all pages are marked: start next round and unmark all pages
    - Replace (arbitrary) unmarked page and mark it
- Intuition: k pages accessed each round

Marker Algorithm

- Marker algorithm is k-competitive
  - In each round, algorithm has k misses
  - OPT has at least one miss because k+1 distinct pages are accessed including the last access from previous round

Randomized Marker Algorithm

- Each of k pages has a marker bit
- Algorithm proceeds in rounds with invariant:
  - At start of round all pages are unmarked
- In each round
  - If request is a hit: mark page
  - If request is a miss:
    - If all cache pages are marked: start next round by unmarking all locations
    - Else evict (randomly) unmarked page and mark it
- This algorithm is $2H_k$-competitive

Only change