Online Algorithms

For input sequence $P_1, P_2, \ldots, P_n$ an online algorithm must produce an output given $P_1, P_2, \ldots, P_i$ (without seeing the ones after) for each i.

Example: Page replacement in cache

Loading parts from disk into memory when you need it.

k is cache size (#pages)

At ith page request Pi the cache contains some k pages. If pi is not in cache (page fault) some page must be evicted from cache to make room for pi then pi is added to the cache.

The cost of a page replacement alg A on a sequence $P_1, \ldots P_n$ is $f_A(p_1, \ldots P_n) = \#$ faults on $P_1, \ldots, P_n$.

Online algorithm must decide what page to evice without knowing the future requests.

Algorithms

Least Recently Used (LRU)

Evict page whose most recent request occured furthest in the past.

Least Frequently Used (LFU)

Evict page that has been requested least often.

Marking algorithm

Approximates LRU. Set a bit on each entry in page table when it's accessed. Then you know what's recently been accessed. (with randomization).

FIFO

Evict page that has been in cache the longest.

How do we decide best online algorithm?

1. Worst-case performance

- LRU (new page very time = n)
- Least frequently used, (new page every time = n)
- FIFO (new page every time = n)

2. Average case performance

Assume all incoming pages are equally probable. m = total number of pages possibly requested.

Expected # page faults on sequence of randomly, uniformly, independently chosen pages.

- LRU, LFU, FIFO, expected = $(1 - \frac{k}{m})n$

Average case isn't typically useful, unless analyzing program traces.

3. Competitive Analysis

Have the algorithms compete against each other. How doe the online algorithm performance compare to the optimal/best offline algorithm?

n online algorithm A is c-competitive if for all $P_1 P_2 P_n$, $f_A(p_1, \ldots, p_n) \leq c f_{OPT}(p_1, \ldots, p_n) + b$

Theorem: LRU and FIFO are k-competitive.

Theorem: If A is a deterministic online algorithm, for paging, then $C \geq k$.

Marking algorithms can do better.