infOnAlg-01a: Online Algorithms	**Lecture 04**	**Date: 14.11.2018**
Instructor: Sebastian Berndt		**Scribe: Torben Neumann**

1 Ramdomized algorithm for caching

Marking algorithms are a general class of paging algorithms. Marking algorithms work on phases, in which each phase is the longest sequence of requests in which no more than k distinct pages are requested. The competitive ratio of deterministic algorithms is k. We will use a randomized algorithm, which evict random unmarked pages to have a better competitive ratio than the deterministic algorithms.

2 MARK

- When a page p is requested, mark it.
- When a page needs to be evicted, evict an uniformly chosen unmarked page. If no such unmarked page exists, unmark all pages.

Observation 1 *All pages are unmarked at the beginng of a phase.*

Theorem 2 *Mark is $2H_k$-competitive with $H_k = \sum_{i=1}^{k} \frac{1}{i}$*

Proof We fix a request sequence σ with layers $l_0, l_1, \ldots$. During phase i, there are two kinds of request to a page p:

1. p was in the cache during phase $i-1$ ("old request")
2. p was not in the cache during phase $i-1$ ("new request").

To maximize the number of faults of MARK, we assume that all new requests are made before the old request. Let n_i be the number of the unique new requests thus $k - n_i$ is the number of unique old requests. Note that $n_1 = k$.

□

Observation 3 *MARK faults on every new request.*

Proof Consider the $j-th$ old request p. At this time, the cache has $n_i + j - 1$ pages from this phase. The probability that p is still in the cache is thus:

$$\frac{\#\text{ unmarked old pages in cache}}{\#\text{ unmarked old pages}} = \frac{k-(n_i+j-1)}{k-(j-1)}$$

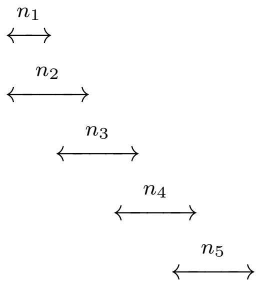
The probability for a page fault is thus:

$$1 - \frac{k-(n_i+j-1)}{k-(j-1)} = \frac{k-(j-1)-k+(n_i+j-1)}{k-(j-1)} = \frac{k-j+1-j+n_i+j-1}{k-j+1} = \frac{n_i}{k-j+1}$$

We thus have:

$$\begin{aligned}
\mathbb{E}\{\#\text{ faults of MARK during phase i}\} &\leq n_i + \sum_{j=1}^{k-n_i} \frac{n_i}{k-j+1} \\
&= n_i + n_i \cdot (\frac{1}{k} + \frac{1}{k-1} + ... + \frac{1}{n_i+1}) \\
&\leq n_i + n_i \cdot (\frac{1}{2} + \frac{1}{3} + ... + \frac{1}{j}) \\
&= n_i \cdot H_k
\end{aligned}$$

Hence $\mathbb{E}\{\#\text{ faults of MARK}\} \leq H_k \cdot \sum_i n_i$. $(\star)$
Between phases $i-1$ and i, there are exactly $k+n_i$ unique page requests. Every algorithm has at most k of these pages in its cache at the beginning of phase $i-1$. Hence, every algorithm has n_i page faults during phase $i-1$ and i.
For example:
phase: 1 | 2 | 3 | 4 | 5



with $n_1, ..., n_5$ being the page faults during the respective phases.
Let f_i be the number of page faults during a phase i.
Then: $f_1 + \sum_{i\geq 2}^{f}[f_{i-1} + f_i] \geq n_1 + \sum_{i\geq 2} n_i$. Hence $2(\sum_i f_i) \geq \sum_{i\geq 1} n_i$ and thus $\sum_{i\geq 1} f_i \geq \frac{1}{2} \cdot (\sum_{i\geq 1} n_i)$.
Hence, we have $OPT(\sigma) = \sum_{i\geq 1} f_i \geq \frac{1}{2} \cdot (\sum_{i\geq 1} n_i)$. $(\star\star)$
Combining $(\star)$ and $(\star\star)$ gives:
$\mathbb{E}[MARK(\sigma] \overset{(\star)}{\leq} H_k \cdot \sum_{i\geq 1} n_i) \overset{(\star\star)}{\leq} 2H_k OPT(\sigma)$ □

Corollary 4 *There are online problem such that randomized algorithms can be much better than deterministic ones.*

Oblivious adversary	vs.	adaptive adversary
fixed instance in before		construct instance based on current run of algorithm

Can we give a lower bound on the competitive ratio of randomized algorithm? Let $\mathcal{X}$ be a set of instances for some minimum problem and (A) be a set of algorithms for this problem.

Theorem 5 (Yao's principle)
Let X be a random variable on $\mathcal{X}$ and A be a random variable on $\mathcal{A}$. Then

$$\max_{x\in\mathcal{X}}\{\mathbb{E}[A(x)]\} \geq \min_{a\in\mathcal{A}}\{\mathbb{E}[a(X)]\}.$$

Proof

- $\mathbb{E}[A(x)] = \sum_{a\in\mathcal{A}} Pr[A=a]\cdot a(x)$
- $\mathbb{E}[a(X)] = \sum_{x\in\mathcal{X}} Pr[X=x]\cdot a(x)$

As $\sum_{x\in\mathcal{X}} Pr[X=x]=1$, we have

$$\begin{aligned}
\max_{x\in\mathcal{X}}\{\mathbb{E}[A(x)]\} &= \max_{x\in\mathcal{X}}\{\sum_{a\in\mathcal{A}} Pr[A=a]\cdot a(x)\}\\
&\underset{\text{max us. avg.}}{\geq} \sum_{x\in\mathcal{X}} Pr[X=x]\cdot\sum_{a\in\mathcal{A}} Pr[A=a]\cdot a(x)\\
&\underset{\text{reorder}}{=} \sum_{a\in\mathcal{A}} Pr[A=a]\cdot\sum_{x\in\mathcal{X}} Pr[X=x]\cdot a(x)\\
&\underset{\text{min us. avg.}}{\geq} \min_{a\in\mathcal{A}}\{\sum_{x\in\mathcal{X}} Pr[X=x]\cdot a(x)\}\\
&= \min_{a\in\mathcal{A}}\{\mathbb{E}[a(X)]\}.
\end{aligned}$$

□

To use this, we interpret a randomized algorithm A as a set of deterministic algorithms (one for each fixed random string). We only need a good random variable X for our input.

Theorem 6 *There is no randomized algorithm for the caching problem with ratio $r \leq H_k$.*

Proof Consider $k+1$ pages $p_1, ..., p_{k+1}$ and construct a random sequence $x = x_1x_2...x_n$
$P_r[x_i = p_j] = \frac{1}{k+1}$ for all i, j.
Any deterministic online algorithm a has a cache of size k. The propability for a page fault at time i is thus at least $\frac{1}{k+1}$. Hence $\mathbb{E}[a(X)] \geq \frac{n}{k+1}$ and thus for all randomized algorithms
$\max_{x\in\mathcal{X}}\{\mathbb{E}[A(x)]\} \geq \frac{n}{k+1}$.
Consider FIF(Furthest in future) and remember that it has one page fault per phase. How long is a phase on average?

Start with empty set $s = \emptyset$
Consider the following experiment:

1. $s = \emptyset$; $T = 0$
2. while $|s| < k$:
    1. Choose p uniformly from $p_1, ..., p_{k+1}$
    2. $s = s \cup \{p\}$
    3. $T = T + 1$
    4. return T

Compute $\mathbb{E}[T]$:
For $i = 1, ..., k+1$, let t_i be the time to collect the i-th page after $(i-1)$ pages were collected. Then $\mathbb{E}[T] = \mathbb{E}[\sum_{i=1}^{k+1} t_i = \sum_{i=1}^{k+1} \cdot \mathbb{E}[t_i]$. As the probability to collect a new page is

$$\frac{\#\text{new pages}}{\#\text{pages}} = \frac{k+1-(i-1)}{k+1},$$

we have

$$\mathbb{E}[t_i] = \frac{k+1}{k+1-(i-1)} = \frac{k+1}{k-i+2}.$$

Hence

$$\sum_{i=1}^{k+1} \cdot \mathbb{E}[t_i] = \sum_{i=1}^{k+1} \frac{k+1}{k-i+2} = \frac{k+1}{k+1} + \frac{k+1}{k} + ... + \frac{k+1}{1} = (k+1) \cdot H_{k+1}.$$

Hence, each phase has expected lentgh $(k+1) \cdot H_{k+1}$ and thus
$\mathbb{E}[FIF(X)] = \frac{n}{(k+1) \cdot H_{k+1}}$. As $\mathbb{E}[A(x)] \geq \frac{n}{k+1}$, we have

$$\frac{\mathbb{E}[FIF(X)]}{\mathbb{E}[A(x)]} \geq \frac{\frac{n}{k+1}}{\frac{n}{(k+1) \cdot H_{k+1}}} = H_{k+1}.$$

□