
Presented by: Shayak Lahiri
Agenda

• Problem Statement
• Overview
• Problem formulation
• Proposed algorithm
• Analysis
• Conclusion
• Q&A
Overview

**Caching:** Storing pages at intermediate levels to reduce time(cost) of future accesses to the same page.

- **Simple model:** Equal page sizes and equal fetching costs
- **Generalized caching:** Unequal page sizes (web) and unequal fetching costs (location).
- **Bit Model:** The fetching cost of a page is proportional to the size of the page.
  - Minimize the total network traffic
- **Fault Model:** Fetching cost of each page is uniform. Sizes vary.
  - Minimize the user wait time

The general version of caching is NP-Hard for both online and offline cases. It’s a formulation of Knapsack Problem.
Problem Statement

• An online algorithm with a \(O(\log^2 k)\) competitive ratio for General case.
  • First algorithm for this problem that is sub linear in \(k\)

• An improved \(O(\log k)\) competitive algorithm for the special cases of Bit Model and the Fault Model.
  • Previously \(O(\log^2 k)\) competitive algorithm was known.
Problem Formulation

Step 1: $O(\log k)$ competitive algorithm for the fractional problem.
- Based on the primal-dual framework for online packing and covering problems.
- Add exponentially many Knapsack-cover inequalities to get around the integrality gap of $\Omega(k)$.

Step 2: Get a randomized integral solution
- Generate a distribution of cache states, while bounding the costs.
- Prove validity of states and maintenance throughout the execution.
Problem Formulation

1. Size of cache is $k$
2. Pages: $n$ pages with sizes $w_1 \leq w_2 \leq \ldots \leq w_n$
3. For a set $S$ of pages, $W(S) = \sum_{i \in S} w_i$
4. Cost of fetching page $p = c_p$
5. Fault Model $c_p = 1$; Bit Model $c_p = w_p$; General $c_p$ is arbitrary
6. $x(p,j)$: indicator variable for page $p$ is evicted between $j^{th}$ and $j+1^{th}$ request.
7. $t(p,j)$: time when page $p$ is requested for the $j^{th}$ time.
8. $r(p,t)$: number of times page $p$ is requested until time $t$.
9. $B(t) = \{p \mid r(p,t) \geq 1\}$: set of pages requested until time $t$
10. $p_t$ : Page requested at time $t$.
11. $x(p_t, r(p,t)) = 0$ (page requested is always in the cache)

Minimize $\sum_{p=1}^{n} \sum_{j=1}^{r(p,t)} c_p \cdot x(p, j)$

For any time $t$, $\sum_{p \in B(t) \setminus \{pt\}} w_p \cdot x(p, r(p, t)) \geq W(B(t)) - k$

For any $p,j$: $x(p, j) \in \{0,1\}$
Problem Formulation

Introducing exponentially many Knapsack Cover inequalities:

• For \( S \subseteq B(t) \), such that \( p_t \in S, W(S) > k \),
  \[ \sum_{p \in S \setminus \{p_t\}} w_p \cdot x(p, r(p, t)) \geq W(S) - k \]
  for each such set \( S \).

• We also truncate size of the page to \( W(S) - k \).
  \[ \sum_{p \in S \setminus \{p_t\}} \min(w_p, W(S) - k) \cdot x(p, r(p, t)) \geq W(S) - k \]

Observation: Given a fractional solution \( x \), if the knapsack cover inequality is violated for a set \( S \), then it is also violated for a set \( S' \) obtained by removing all the evicted pages from \( S \).

Result: \( x(i, j) \leq 1 \) always, thus \( x(i, j) \geq 0 \) is sufficient as a constraint.
Problem Formulation

Primal:

Minimize $\sum_{p=1}^{n} \sum_{j=1}^{r(p,t)} c_p \cdot x(p, j)$

For time $t$, and set of requested pages $S \subseteq B(t)$ such that $p_t \in S$

$\sum_{p \in S \setminus \{p_t\}} \min(w_p, W(S) - k) \cdot x(p, r(p, t)) \geq W(S) - k$

For any $p, j$: $x(p, j) \geq 0$

Dual:

Maximize $\sum_{t} \sum_{S \subseteq B(t), p_t \in S} (W(S) - k) y(t, S)$

For each page $p$ and the $j$th time it's requested:

$\sum_{t = t(p, j) + 1}^{t(p, j + 1) - 1} \sum_{S \mid p \in S} \min\{w_p, W(S) - k\} y(t, S) \leq c_p$
Algorithm

Fractional Caching Algorithm: At time t, when page $p_t$ is requested:

1. Set $x(p_t, r(p_t, t))$ to 0
2. Until all primal constraints corresponding to time t are satisfied, do:
   3. Assume the primal constraint of a minimal set, $S$ is not satisfied
      1. Increase variable $y(t,S)$ continuously; for each variable $x(p, j)$ such that $p \in S \setminus \{p_t\}$
      2. If $x(p,j) = 1$, then remove $p$ from $S$.
      3. If $x(p,j) = 0$ and $\sum_{t=t(p,j)+1}^{t(p,j+1)-1} \sum_{p \in S} \min\{w_p, W(S)-k\} y(t,S) = c_p$, then $x(p,j) = 1/k$.
      4. If $1/k \leq x(p,j) < 1$, increase $x(p,j)$ by the following function:
$$\frac{1}{k} \exp(\frac{1}{c_p}[\sum_{t=t(p,j)+1}^{t(p,j+1)-1} \sum_{p \in S} \min\{w_p, W(S)-k\} y(t,S)] - c_p)$$
Analysis

Contribution to the primal cost:

• Increasing \( x(p,j) \) from 0 to \( 1/k \) \((C_1)\)
• increasing \( x(p,j) \) using the exponential function. \((C_2)\)

It can be shown that:

1. \( C_1 \leq (\ln k + 1) \). Profit of feasible dual solution
2. \( C_2 \leq (\ln k + 1) \). Profit of feasible dual solution

Thus \( C_1 + C_2 \leq (\ln k + 1) \). Profit of feasible dual solution

• The profit of any dual solution is a lower bound on the optimal solution.
• Thus, by weak duality, this algorithm is \( O(\log k) \) competitive.
Rounding the fractional solution online

Let
1. \( x_p = x(p, r(p,t)) \)
2. \( \gamma \geq 1 \)
3. \( y_p = \min(\gamma x_p, 1) \).

Let \( \mu \) be a distribution on a subset of pages. \( \mu \) is said to be consistent with \( y \) if \( \mu \) induces the distribution \( y \) on the page set defined by:

\[ \forall p: \sum_D A^D_p \cdot \mu(D) = y_p \]

Where \( A^D_p = 1 \), if \( p \in D \), else \( p = 0 \).

Required conditions:
1. **Size property**: For any \( D \) with \( \mu(D) > 0 \), \( W(D) \geq W(B(t)) - k \).
2. **Bounded Cost property**: If \( y \) changes to \( y' \), while incurring a fractional cost of \( d \), \( \mu \) can be changed to \( \mu' \) with a cost of \( \beta d \).
Rounding: General Cost Model and Bit Model

**Lemma:** For any subset $D$ that is balanced with respect to $y$, the sum of sizes of all pages in $D$ is at least $W(B(t)) - k$

**Proves:** Size property holds.

**Lemma:** Given any distribution $\mu$ over balanced sets that is consistent with $y$, if $y$ changes to $y'$ incurring fractional cost $d$, then $\mu$ can be modified to another distribution $\mu'$ over balanced sets consistent with $y'$ such that the total cost incurred is at most $10d$.

**Proves:** Bounded Cost Property holds.
Rounding: Fault Model

**Lemma:** Let $G$ be a good grouping with respect to $y$. Then any balanced set $D$ with respect to $G$ has size at least $W(S) - k$.

**Proves:** Size property holds.

**Lemma:** As the solution $y$ changes over time, we can maintain a good grouping $G$ and a consistent distribution on balanced sets with an amortized cost at most a constant times a fractional cost.

**Proves:** Bounded Cost Property holds.
Conclusion

✓ Theorem: There exist randomized algorithms for the generalized caching problem that are:
  • $O(\log k)$ competitive for the Bit-Model.
  • $O(\log k)$ competitive for the Fault-Model.
  • $O(\log^2 k)$ competitive for the general model.

✓ An improvement over previously known algorithms for these problems.

✓ A unified framework for caching algorithms which are substantially simpler than previously used algorithms.

✗ Assumes that requests must be served in the order of arrival.

✗ Performance degrades in a pipelined structure.
Thank you