Practical Bounds on Optimal Caching with Variable Object Sizes

Daniel S. Berger    Nathan Beckmann    Mor Harchol-Balter
Carnegie Mellon University

Caches are Everywhere

Goal: minimize cache miss ratio

miss ratio = \( \frac{\text{# requests not served from cache}}{\text{total # requests}} \)
Key Question: how much further can miss ratios be improved?

Def: OPT = lowest miss ratio possible on a given trace

Results on 2016 CDN trace. Cache size: 4GB.
Defining OPT

Def: OPT = lowest miss ratio possible on a given trace

= offline optimal miss ratio on a given trace

Constraints:

1. Limited cache size
2. Gets to see full request trace
3. No prefetching

⇒ admit an object only when it is requested
Finding OPT

What is OPT?

No! Belady assumes equal sizes!

9 orders of magnitude variability!

So, can we find OPT?

Unfortunately, NP-hard

In fact, strongly NP complete

Can we approximate OPT?
## OPT Approximation Algorithms

<table>
<thead>
<tr>
<th>Technique</th>
<th>Time Complexity</th>
<th>Approximation Guarantee</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFMA [STOC’97]</td>
<td>$O(N^2)$</td>
<td>$O(\log \text{cache size})$</td>
</tr>
<tr>
<td>LP rounding [SODA’99]</td>
<td>$O(N^5)$</td>
<td>$O(\log \left(\frac{\text{max size}}{\text{min size}}\right))$</td>
</tr>
<tr>
<td>LocalRatio [JACM’01]</td>
<td>$O(N^3)$</td>
<td>4</td>
</tr>
</tbody>
</table>

State-of-the-art 4-approximation not practical
## OPT Approximation Algorithms

<table>
<thead>
<tr>
<th>Technique</th>
<th>Time Complexity</th>
<th>Approximation Guarantee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best prior approximation</td>
<td>$O(N^3)$</td>
<td>4</td>
</tr>
</tbody>
</table>

*Traces are not adversarial in practice*

$\Rightarrow$ Probabilistic assumptions

- Independent Reference Model (IRM)
- Large systems: #objects, cache size
## Our Main Result

<table>
<thead>
<tr>
<th>Technique</th>
<th>Time Complexity</th>
<th>Approximation Guarantee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best prior approximation</td>
<td>$O(N^3)$</td>
<td>4</td>
</tr>
<tr>
<td>Flow-offline optimum (FOO)</td>
<td>$O(N^2 \log^2 N)$</td>
<td>1</td>
</tr>
</tbody>
</table>

Traces are not adversarial in practice

⇒ Probabilistic assumptions

Independent Reference Model (IRM)

Large systems: #objects, cache size

On trace with strong correlations: error < 0.14%
How does FOO attain OPT for large systems?

How to get OPT fast:

1. Detailed OPT ILP (NP-hard)
2. Interval ILP (NP-hard)
3. Interval LP relaxation (Ω(N^{3.5}))
4. FOO Min Cost Flow graph (O(N^2 \log^2 N))

Trace:

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>b</th>
<th>a</th>
</tr>
</thead>
<tbody>
<tr>
<td>X_{(a,1)}</td>
<td>X_{(a,2)}</td>
<td>X_{(a,3)}</td>
<td>X_{(a,4)}</td>
<td>X_{(a,5)}</td>
</tr>
</tbody>
</table>

DVs for object a:
How does FOO attain OPT for large systems?

**How to get OPT fast:**

1. Detailed OPT ILP (NP-hard)
2. Interval ILP (NP-hard)
3. Interval LP relaxation ($\Omega(N^{3.5})$)
4. FOO Min Cost Flow graph ($O(N^2\log^2 N)$)

**How to prove FOO’s correctness:**

1. Non-integer decision vars (DVs) always exist
2. Precedence relation, which forces integer DVs under IRM
3. Coupon collector problem in large systems
4. Integer DVs almost surely
## Our Main Result

<table>
<thead>
<tr>
<th>Technique</th>
<th>Time Complexity</th>
<th>Approximation Guarantee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best prior approximation</td>
<td>$O(N^3)$</td>
<td>4</td>
</tr>
<tr>
<td>Flow-offline optimum (FOO)</td>
<td>$O(N^2 \log^2 N)$</td>
<td>1</td>
</tr>
</tbody>
</table>

Computable with up to $10^7$ requests
Empirical Results

**Key Question:** how much further can miss ratios be improved?

![Graph showing miss ratios from 2001 to 2018](chart.png)

- Miss Ratio:
  - 2001: 0.3
  - 2015: 0.2
  - 2017: 0.3
  - 2018: 0.3

- 30% gap

- Are we there yet?

- Could be optimal... or not
Results for Other Configurations

Key Question: how much further can miss ratios be improved?

- **CDN**
  - Small Cache: 30% gap
  - Large Cache: 45% gap

- **WebApp**
  - Small Cache: 61% gap
  - Large Cache: 51% gap

- **Storage**
  - Small Cache: 15% gap
  - Large Cache: 41% gap
## Conclusion

<table>
<thead>
<tr>
<th>Technique</th>
<th>Time Complexity</th>
<th>Approximation Guarantee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best prior approximation</td>
<td>$O(N^3)$</td>
<td>4</td>
</tr>
<tr>
<td>Flow-offline optimum (FOO)</td>
<td>$O(N^2 \log^2 N)$</td>
<td>1</td>
</tr>
</tbody>
</table>

Actually can do: $O(N \log N)$

**Implication:** large potential for new caching policies

⇒ e.g., 60% improvement possible for WebApps
Practical Bounds on Optimal Caching with Variable Object Sizes

Daniel S. Berger  Nathan Beckmann  Mor Harchol-Balter
Carnegie Mellon University

Source code and data: available at GitHub/dasebe/optimalwebcaching