Efficient Proactive Caching for Supporting Seamless Mobility

Vasilios Siris, Xenofon Vasilakos, George Polyzos
PhD student, AUEB, MMLAB
mm.aueb.gr/~xvas
xvas@aueb.gr
Problem

- Reduce propagation delay
  - $f(\#\text{network hops})$
Approach (1/2)

- **Proactively** fetch data-objects to attachment points
- Is this a *typical* proactive caching approach?
Approach (2/2)

- Handoff mobility probabilities $q_1, q_2$
- Exploit **Individual** mobility & requests
  - *Not* data-popularities
Efficient Proactive Caching (EPC)

• Individual requests imply higher demand for cache space

• Congestion pricing for cache storage
  – Efficient cache utilization

  ➢ EPC trades cache space (price) for reduced delay (delay cost)
Outline

1. EPC in a **flat** cache structure
2. EPC in a **two-level** cache hierarchy
3. Evaluation
EPC IN A FLAT CACHE STRUCTURE
Flat cache structure

- Decision Rule:
  \[
  \begin{cases} 
  1 & \text{if } q \left( D_R - D_L \right) \geq p_l \\
  0 & \text{if } q \left( D_R - D_L \right) < p_l 
  \end{cases}
  \]

- **Autonomous** prefetching/caching
Flat cache structure

• Step-wise decision procedure
  – **Optimal** selection of cached objects?
Flat cache structure

• **Optimal** selection of cached objects?

  1. **Objects with different sizes**
     • Optimization is identical to 0/1 Knapsack Problem
     • NP-hard problem

  2. **Optimal for equal-size objects**
     • For each cache and each request, order by
       \[ q \cdot (D_R - D_L) \]
EPC IN A TWO-LEVEL CACHE HIERARCHY
Hierarchical cache structure

source

mid-level caches

leaf caches

$q_{l1}$ $q_{m1}$ $q_{mN}$

17/06/2014
Hierarchical cache structure

- Leaf nodes solve 2 flat cache problems:
  1. Delay $D_R$
  2. Delay $D_M$

- Requires cooperation

\[ D_{mid}^R - D_{mid}^M \geq P_{mid} \]
Finding an optimal solution?

- Data Placement Problem
  - Different object sizes => **NP-complete**
  - Equal size objects => high polynomial degree time
EVALUATION
Evaluation

Comparison with a naive, an optimal, and an oracle scheme
Evaluation

Comparison with a naive, an optimal, and an oracle scheme
Evaluation

Comparison with a naive, an optimal, and an oracle scheme
Evaluation

(a) $D_M/D_L = 5$

(b) $D_M/D_L = 2$
Evaluation

Comparison with a naive, an optimal, and an oracle scheme

Gain Over No Caching

MC/TC

NAIVE
EPC SKD50%
EPC SKD70%
EPC SKD90%
OPT SKD50%
OPT SKD70%
OPT SKD90%
ORACLE

17/06/2014
A distributed mobility support solution tailored to individual user mobility/requests that exploits user mobility and uses congestion pricing
Bibliography

[1] V. A. Siris, X. Vasilakos and G. C. Polyzos, "Efficient proactive caching for supporting seamless mobility," *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium on a*, Sydney, NSW, 2014, pp. 1-6.

[2] X. Vasilakos, V. A. Siris, G. C. Polyzos, and M. Pomonis. "Proactive selective neighbor caching for enhancing mobility support in information-centric networks." In *Proceedings of the second edition of the ICN workshop on Information-centric networking*, pp. 61-66. ACM, 2012.

[3] V.A. Siris, X. Vasilakos, and G. C. Polyzos. "A Selective Neighbor Caching Approach for Supporting Mobility in Publish/Subscribe Networks." In *FIFTH ERCIM WORKSHOP ON EMOBILITY*, p. 63. 2011.