Distributed Caching Algorithms for Content Distribution Networks

(Sem Borst, Varun Gupta, Anwar Walid)

Y. Wu

June 2, 2010

Outline

- Introduction
- Problem Formulation
- Symmetric Scenario
  - Intra-level
  - Inter-level
- Numerical Experiments
- Conclusion

Introduction

- Springs up of V generated vid
- Broa

Large volumes of traffic!

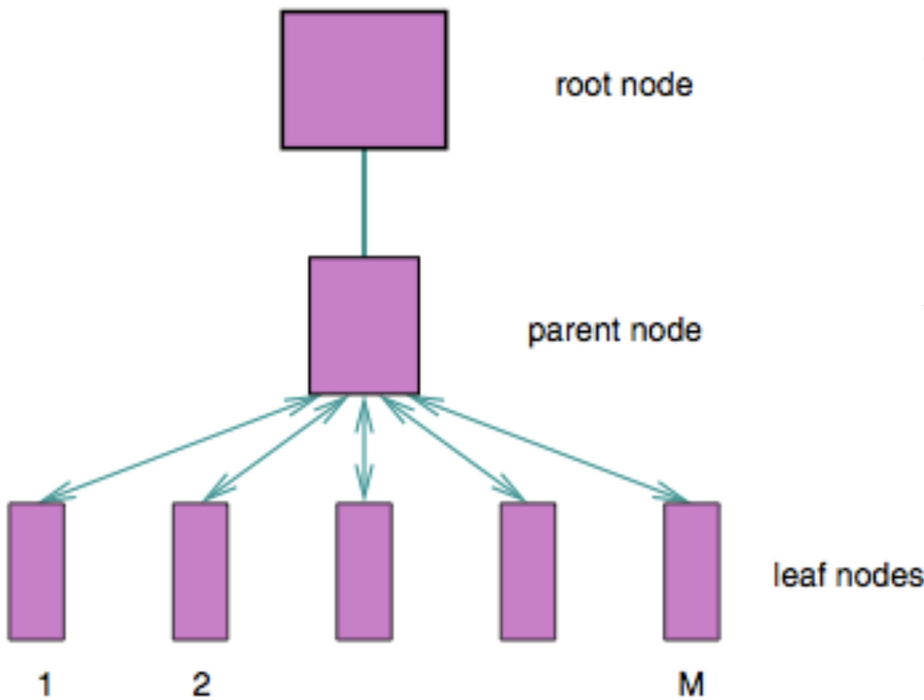
Introduction

- Trade storage for bandwidth
  - Store popular contents closer to network edge to reduce traffic load, expense and performance bottlenecks.
- Caching Strategy
  - Prediction of Demand
  - Content Placement
  - Optimal Dimensioning of Cache

Low-complexity, distributed
yet with implicit coordination

Problem Formulation

- A cache cluster consisting of M 'leaf' nodes indexed 1,…,M.



Size of Caches:

$$B_0, B_1, \ldots, B_M$$

Size of items:

$$s_1, s_2, \ldots, s_N$$

Problem Formulation

- Some other important parameters
  - d_{in} the demand for the n-th content item in node i
  - c_0 the unit cost incurred from root to parent
  - c_i the unit cost incurred from parent to node i
  - c_{ij} the unit cost from node j to node i
  - x_{in} indicates whether the n-th content is stored in node i
  - x_{ijn} indicates whether requests for the n-th content at node I are served from j.

x_{in}, x_{ijn} are 0-1 decision variable

Problem Formulation

- Optimization-I (Expenses)

$$\min \quad \sum_{i=1}^{M} \sum_{n=1}^{N} s_n d_{in} \big((c_0 + c_i) x_{i,-1n} + c_i x_{i0n} + \sum_{j \neq i} c_{ij} x_{ijn} \big)$$

$$\text{sub} \quad \sum_{n=1}^{N} s_n x_{in} \leq B_i, \quad i = 0, \ldots, M$$

$$x_{ijn} \leq x_{jn}, \quad i = 1, \ldots, M, i \neq j = 0, 1, \ldots, M, \forall n$$

$$x_{in} + x_{i,-1n} + x_{i0n} + \sum_{j \neq i} x_{ijn} \geq 1, \quad i = 1, \ldots, M, \forall n,$$

Problem Formulation

- Optimization II (Saving)

$$
\begin{aligned}
\max \; & \sum_{i=1}^{M} \sum_{n=1}^{N} s_n d_{in} \Big((c_0 + c_i) x_{in} + c_0 x_{i0n} + \sum_{j \neq i} (c_0 + c_i - c_{ij}) x_{ijn} \Big) \\
\text{sub} \; & \sum_{n=1}^{N} s_n x_{in} \leq B_i, \quad i = 0, \ldots, M \\
& x_{ijn} \leq x_{jn}, \quad i = 1, \ldots, M, i \neq j = 0, 1, \ldots, M, \forall n \\
& x_{in} + x_{i0n} + \sum_{j \neq i} x_{ijn} \leq 1, \quad i = 1, \ldots, M, \forall n,
\end{aligned}
$$

Transport cost savings achieved by transferring data to leaf i from leaf j instead of root

Symmetric Scenario

- Leaf nodes are symmetric in terms of bandwidth costs, demand characteristics and cache sizes
  - $c_i = c, c_{ij} = c', d_{in} = d_n, B_i = B(i = 1,\ldots,M)$
- Relaxation towards LP
  - x_{in}, x_{ijn} can have fractional values but not more than 1.
  - Chunks?

$$c'' := c + c_0 - c'$$

$$u_n := \min\{1, x_{0n} + \sum_{i=1}^{M} x_{in}\}$$

$$\Longrightarrow$$

$$\max \sum_{n=1}^{N} s_n d_n (M c'' u_n + c' \sum_{i=1}^{M} x_{in} + (c' - c) \sum_{i=1}^{M} x_{i0n}) \quad (1)$$

$$\text{sub} \sum_{n=1}^{N} s_n x_{0n} \le B_0 \quad (2)$$

$$\sum_{n=1}^{N} s_n x_{in} \le B, \quad i = 1, \ldots, M \quad (3)$$

$$u_n \le 1, \quad n = 1, \ldots, N \quad (4)$$

$$u_n \le x_{0n} + \sum_{i=1}^{M} x_{in}, \quad n = 1, \ldots, N \quad (5)$$

$$x_{in} \le 1, \quad i = 0, \ldots, M, n = 1, \ldots, N \quad (6)$$

$$x_{i0n} \le x_{0n}, \quad i = 1, \ldots, M, n = 1, \ldots, N \quad (7)$$

$$x_{i0n} + x_{in} \le 1, \quad i = 1, \ldots, M, n = 1, \ldots, N, \quad (8)$$

Symmetric Scenario

- Some observations
  - Assume $x_{in} = x_n$ ....
  - $x_{0n} + x_n \leq 1$
  - Optimality requires $u_n = 1$ if $u_n < x_{0n} + \sum_{i=1}^{M} x_{in}$

$$c''' := M(c_0 + c) - (M-1)c' = Mc'' + c'$$

$$p_n := u_n - x_{0n}$$

$$q_n := (x_{0n} + \sum_{i=1}^{M} x_{in} - u_n) / (M-1)$$

$$\Rightarrow$$

$$\max \quad \sum_{n=1}^{N} s_n d_n (c''' p_n + c'(M-1)q_n + M c_0 x_{0n}) \quad (9)$$

$$\text{sub} \quad \sum_{n=1}^{N} s_n x_{0n} \le B_0 \quad (10)$$

Knapsack-like

$$\sum_{n=1}^{N} s_n (p_n + (M-1)q_n) \le MB \quad (11)$$

$$p_n + x_{0n} \le 1, \qquad n = 1, \ldots, N \quad (12)$$

$$q_n + x_{0n} \le 1 \qquad n = 1, \ldots, N \quad (13)$$

1. q_n will be zero most of the time unless $p_n = 1$
2. $p_n = q_n = 0$ when $x_{0n} = 1$

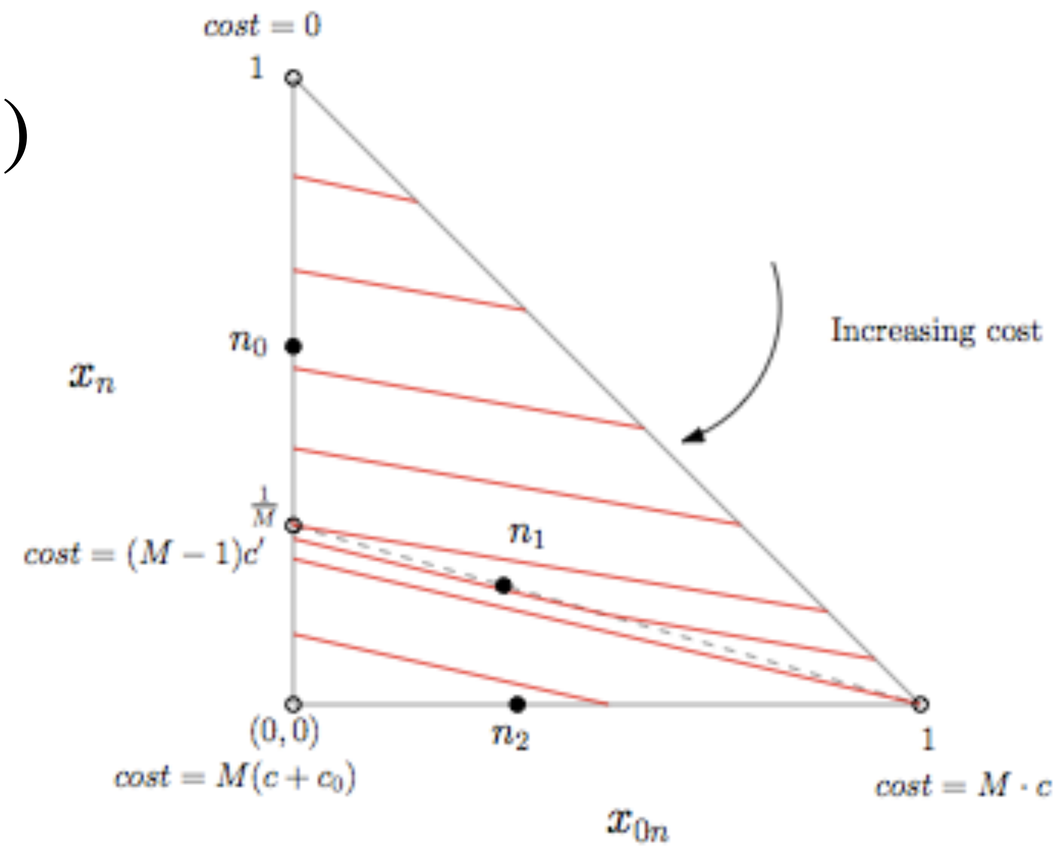
Symmetric Scenario

- Case A: $Mc \geq (M-1)c'$
- Case B: $Mc < (M-1)c'$

It is more advantageous to store content in the leaf nodes than in the parent node

Symmetric Scenario

(x_{0n}, x_n)



Symmetric Scenario

- Boundary points
  - n_0, n_1, n_2
  - Items $1,\ldots,n_0-1$ are cached in all the leaf nodes.
  - Items $n_0+1,\ldots n_1-1$ have a single copy in the parent node only
  - Items $n_2+1,\ldots,N$ are not cached anywhere.

One of the following must be true:
(1) Item n0 has a single copy at a leaf node
(2) Item n1 has a single copy at the parent node
(3) Item n2 is not cached

There can be at most two items which are in a non-vertex configuration!

Intra-level

- $B_0 = 0, s_n = 1 (n = 1, \ldots, N)$
- Local Greedy Algorithm
- Local Greedy Gen Algorithm

Intra-level

- Local Greedy Algorithm
  - In case of a request of item n at node i, if n is not stored at I and it has higher utility than some item m that is currently stored at I, then replace m by n.
- Utility Functions
  -

$$utility = \begin{cases} c' d_n & \text{If has been stored in the cluster} \\ c''' d_n & \text{Not Stored} \end{cases}$$

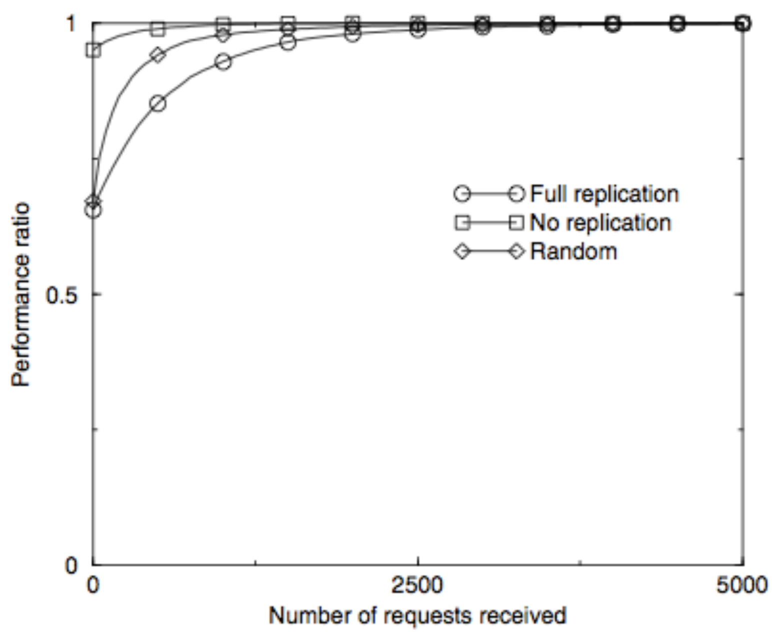
Intra-level

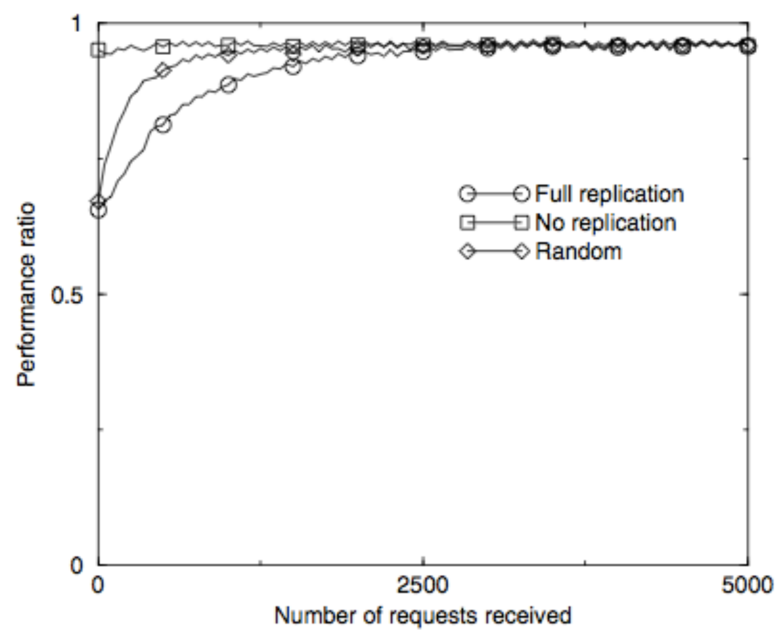
- Performance Guarantee
  - Symmetric popularities: 4/3 from optimal
  - Arbitrary popularities: 2 from optimal
- The problem?
  - Local Greedy Gen Algorithm…

Inter-Level

- $c_{ij} = 0$
- Simple greedy algorithm
  - Each node aims at maximizing the its own hit rate
  - $\dfrac{(M-1)c_{\min} + Mc_0}{(M-1)c_{\min} + (2M-1)c_0} \geq \dfrac{M}{2M-1}$ from optimal.

Numerical Experiments





Numerical Experiments

- Observations:
  - Local greedy performs well
  - Scenario with no replication appears to be most favorable one, due to the fact that in optimal placement only a small number of items are fully replicated.

Conclusion

- Very brave to do the simplification when you can't figure it out…("Without loss of generality")
- Gaps between the algorithm and the problem model
- Proof of the algorithm's performance bounds is the key part, though not presented…