

File Sharing in Distributed Networks with Minimum Delay

Mohsen Sardari

Georgia Institute of Technology

August 13, 2009

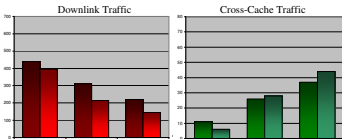
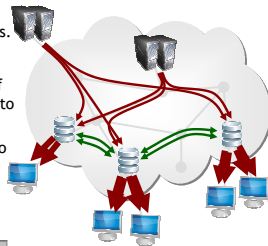
joint work with [Emina Soljanin](#) (Alcatel-Lucent)

Open CDN Provides a Platform for Advanced Content Distribution Algorithms

Provides a control platform that enables a new breed of content distribution algorithms.

Example: cache-based load balancing

- Instruct caches to store a disjoint set of items and allow serving users attached to other caches.
- A larger set of items can be delivered to users in the network.
- Cooperative caching shifts load from peering links to the core network.



Leverage available network capacity to lower peering link and server load!

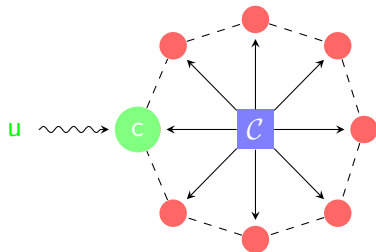
All rights reserved © Alcatel-Lucent 2009

Alcatel-Lucent

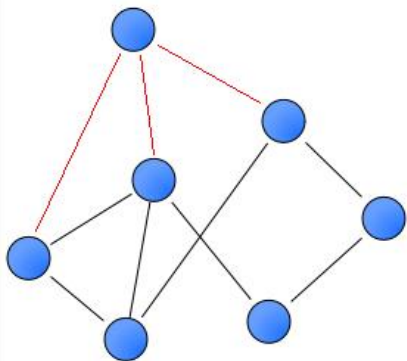
Network Model

Focus: simple networks with two levels of hierarchy.

- ▶ \mathcal{C} is the controller node that facilitates the cooperation between caches.
- ▶ for each user there is a primary cache (green) which can use the information stored in all the other caches to serve the user.



Problem: What is the minimum storage required such that each node can recover a file of size k by accessing the memory of its own and of its neighbors?

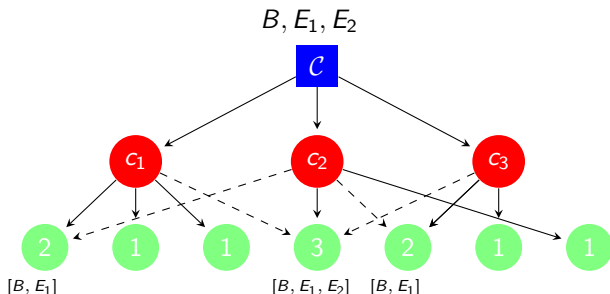


$$\begin{array}{ll} \text{minimize} & \sum_v s_v \\ \text{subject to} & A_G \cdot S \geq k \end{array}$$

A_G : adjacency matrix
 s_v : storage at node v
 S : vector of s_v

Coding in Heterogeneous Networks

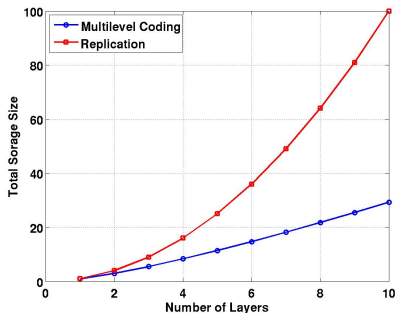
Networks with heterogeneous users can benefit from coding.
Consider a scalable video m with three layers: **Base layer B** , **1st enhancement layer E_1** and **2nd enhancement layer E_2** . There are three levels of users. Every user needs B in order to start playback. For higher quality 2nd and 3rd level users need to have $[B, E_1]$ and $[B, E_1, E_2]$, respectively.



Storage

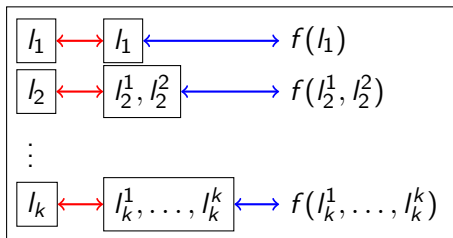
Claim. When the number of layers is k (each of unit size), if the vector S is feasible, then

$$|S| \geq k \sum_{i=1}^k \frac{1}{i} \approx k \log(k)$$



Achieving the Minimum Storage

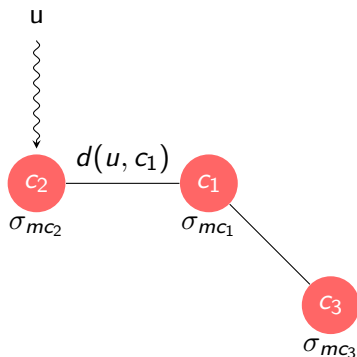
The minimum can be achieved by partitioning the layers (l) and saving independent linear combinations.



Note that $|f(l_i^1, \dots, l_i^i)| = 1/i$.

Decentralized

Modeling a system with Cost



Consider a set of users of size n and a set of caches of size k .

We assume that each user demands one movie.

σ_{mc} : fraction of movie m stored in cache c

$d(u, c_i)$: distance between user u and cache c_i

$f_{u,c}$: fraction by which user u receives from cache c

$m(u)$: requested of user u

S : cache capacity

Constraint on Number of Users

Transferring data from adjacent caches and pre-fetching enforces a limit on the number of simultaneous users, denoted by D .

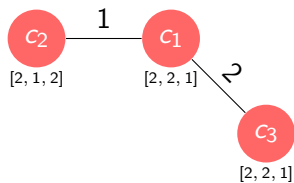
We need to solve this linear program:

$$\begin{array}{llllll} \text{minimize} & \sum_{u,c} f_{uc} d(u, c) & & & & \\ \text{subject to} & f_{uc} & \leq & \sigma_{m(u)c} & \forall u, c & \\ & \sum_c f_{uc} & = & 1 & \forall u & \text{User's demand} \\ & \sum_u f_{uc} & \leq & D & \forall c & \text{Limit on number of users} \\ & \sum_m \sigma_{mc} & \leq & S & \forall c & \text{Capacity of cache} \\ & f_{uc}, \sigma_{mc} & \leq & 1 & \forall c, u, m & \end{array}$$

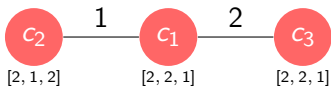
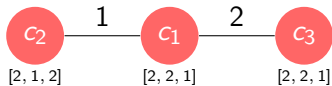
Structure of the Solution

Example. Suppose that we have three movies m_1, m_2, m_3 and 15 users. Number of requests for m_1, m_2, m_3 is 6, 5, 4, respectively. Each cache can serve upto 5 users at a time.

At each cache, the triple $[u_1, u_2, u_3]$ shows the number of local users requesting m_1, m_2, m_3 .



$D = 5$ and number of requests for m_1, m_2, m_3 is 6, 5, 4, respectively. Also, $d(u, c) = 0$ if c is primary for u .



m_1	m_1	m_1
m_2	m_2	m_3

m_1	m_1	m_1
$m_2/2$	$m_2/2$	$m_3/2$
$m_3/3$	$m_3/3$	$m_3/3$

$$\begin{cases} \text{delay} & 3 + 4 + 4 = 11 \\ \text{storage} & 6 \end{cases}$$

$$\begin{cases} \text{delay} & 11/6 + 3/2 + 8/3 = 6 \\ \text{storage} & 14/3 \end{cases}$$

Conclusion and Future Work

- ▶ Coding is a potential solution for decentralized cooperative CDN networks
- ▶ By using coding we can effectively reduce the gap between optimal fractional solutions and integral solutions
- ▶ Finding the optimal storage scheme for Multilevel Coding in general graph is of interest
- ▶ Finding the gap between the value achieved by decentralized Multilevel Coding and optimal LP solution is of interest
- ▶ Penalty variables need to be studied thoroughly for practical scenarios