Caching performance of CCN under multipath routing (and more...)

Giuseppe Rossini (speaker)
Dario Rossi

About this presentation

- Goals
  - Hands on assessment of several CCN aspects via simulation
  - Evaluation scenario realistic (and large scale) as possible
- Large scale, realistic scenarios in terms of
  - Chunk, File, Cache sizes
  - Network topologies
  - Content Popularity

Introduction

- What is CCN?
- New networking Paradigm
  - Files subdivided in chunks
  - Data transfer receiver oriented
  - From a network of routers to a network of caches
- How does it work?
  - A client sends an interest for a data chunk
  - The interest if forwarded along one or more paths toward the content
  - The data is given back either from one of the repositories or from a cache along the path

Caching scenarios

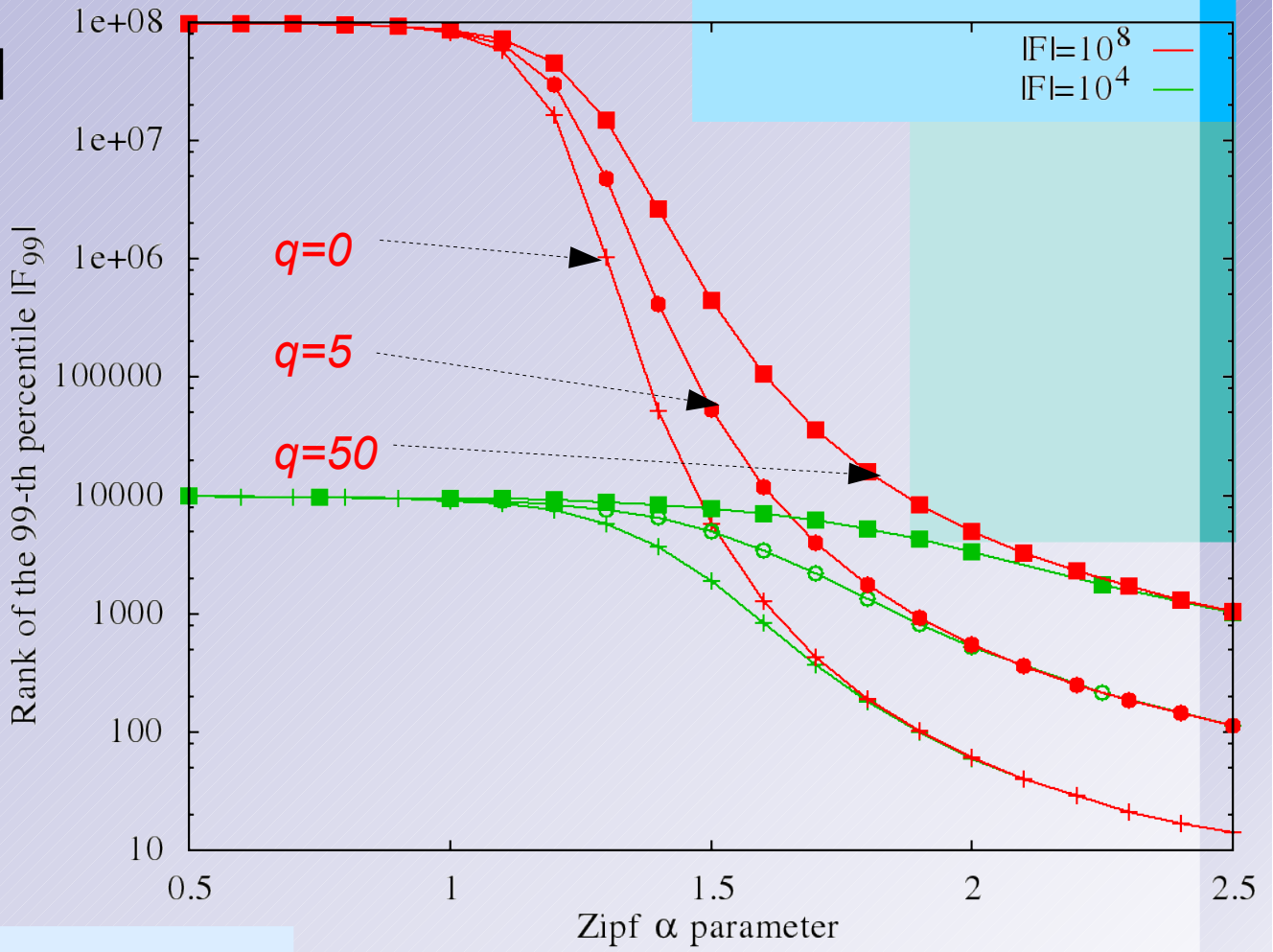
- Large spectrum of values among the whole literature, but not all very sound
- Sometimes very easy job for caches
  - ”Easy” popularity model parameters
  - Large caches with respect to catalog size
- First goal: build a reasonable, fairly large scale scenario
- Second goal: play with several CCN design decisions

Caching scenarios(2)

c	*Chunk size*	10 KB
F	*File size*	Up to 10^4 chunks (10MB) (geom distributed)
|F|	*Number of files*	Up to 10^8
|F|F	*Catalog size (in bytes)*	Up to 10^{15} bytes (1PB)
C	*Cache size*	Up to 10^6 chunks (10 GB)
C/|F|F	*Cache/catalog ratio*	$[10^{-5},10^{-1}]$
α	*Zipf shaping factor*	[0.5,2.5]
q	*Mzipf plateau*	{0,5,50}
λ	*Arrival rate*	[1,10]Hz
W	*Control window width*	1 chunk
R	*Number of paths*	{1,2}
C_R	*Cache replacement policy*	FIFO, LRU,UNIF, BIAS
C_D	*Cache decision policy*	LCE,LCD, FIX(P)
Net	*Network topology*	Geant, Abilene, Dtelekom,Tiger,Qwest,Level3

Popularity model

- Really hard to find a dial on the right model
- Mandelbrot Zipf seems the most accreditate
- $P(i) = C/(i+q)^{\alpha}$
  - q = plateau
  - α = shaping factor



Caching scenario

- Often in the following we will refer to the Youtube scenario. This means
  - A catalog of 10^8 files
  - File size average 10MB (geometrically distributed)
- Common used values for the system of caches will be
  - 10KB chunk size
  - 10GB caches
  - Thus a cache/catalog ratio of 10^{-5}

Topology and routing

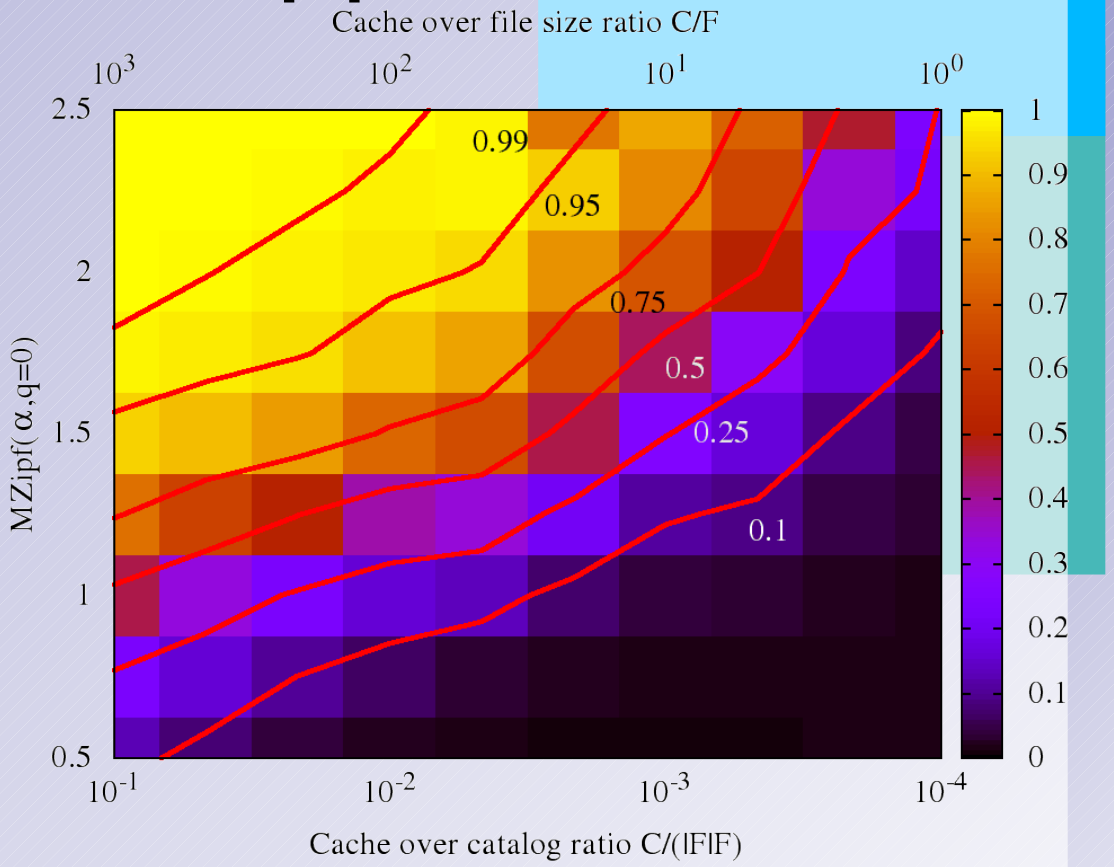
- Single (Dijkstra) vs multipath
  - Shortest path toward the closest repository
  - Multiple paths toward the closest repository
  - Shortest paths toward multiple repositories
- Different strategies to dial with multiple paths
  - Parallel (using multiple paths at the same time)
  - Alternate
  - Retention of the strategy for the first chunk(s)
- Topologies
  - Traditional 15-nodes tree
  - Realistic topologies

Decision vs Replacement policies

- Replacement policies: ”decide which element should be replaced by the new one”
  - Random
  - FIFO
  - LRU (Least Recently Used)
  - BIAS (mixed RANDOM & LFU)
- Decision policies:”decide if caching or not an incoming element”
  - LCE (Leave Copy Everywhere)
  - LCD (Leave Copy Down)
  - Fixed probability P

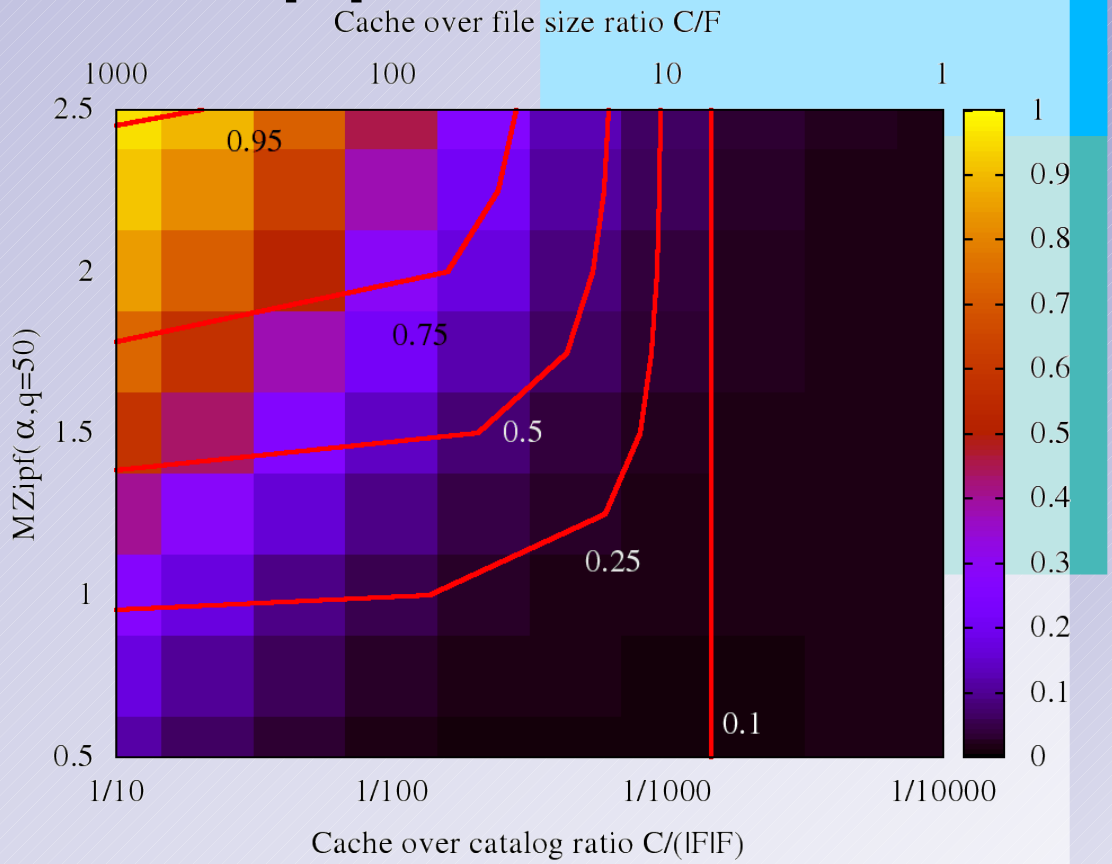
Performance: appetizer

- Non trivial dependencies (C/F, C/F|F|) cannot scale down the problem
- The problem is indeed trivial or impossible depending on what are the reasonable values for the scenario



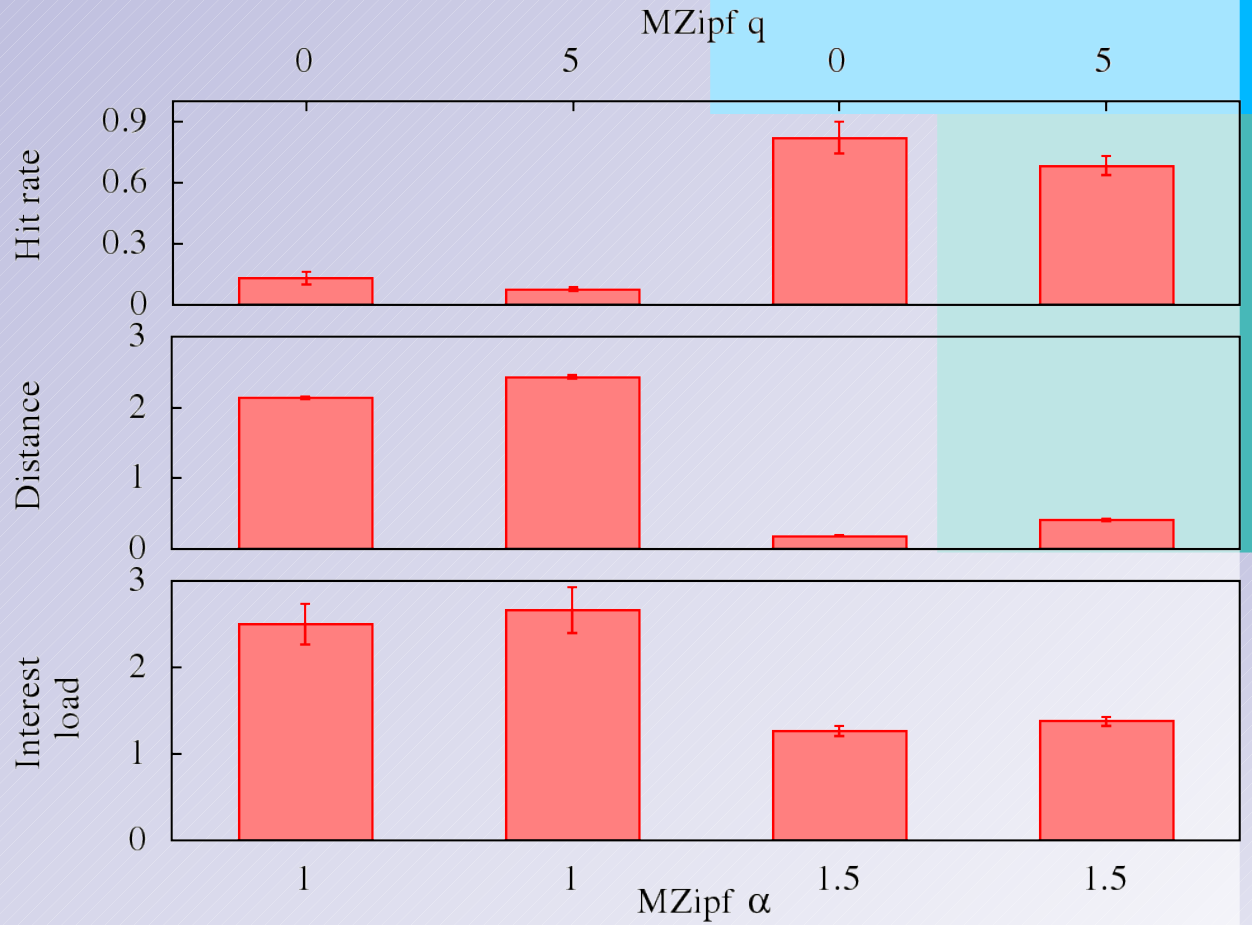
Performance: appetizer

- Non trivial dependencies (C/F, C/F|F|) cannot scale down the problem
- The problem is indeed trivial or impossible depending on what are the reasonable values for the scenario



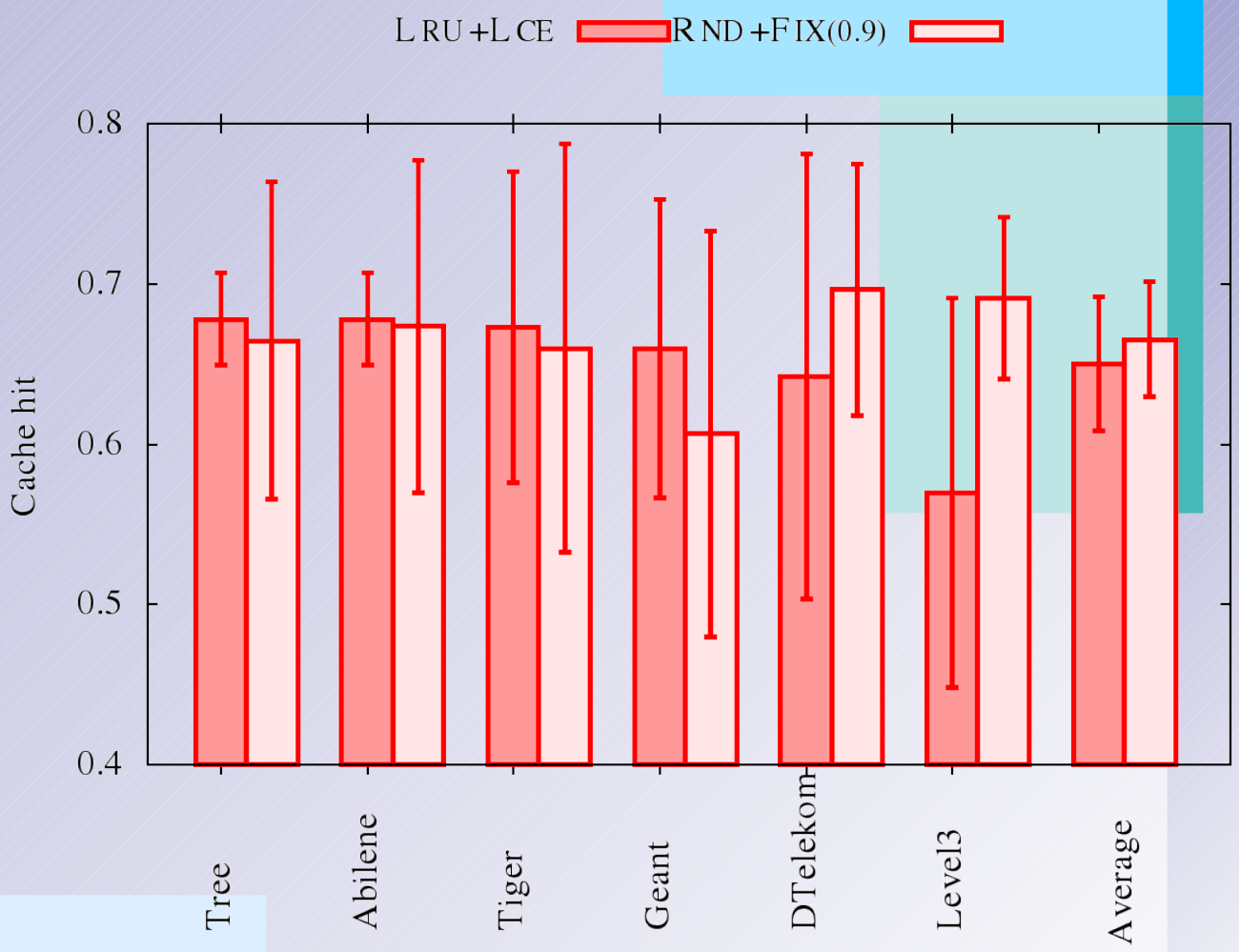
Performance at a glance

- Geant topology, Youtube scenario
- On the average, shaping factor influences most
- Varying the plateau has a borderline effect
- What about varying policies and topologies?



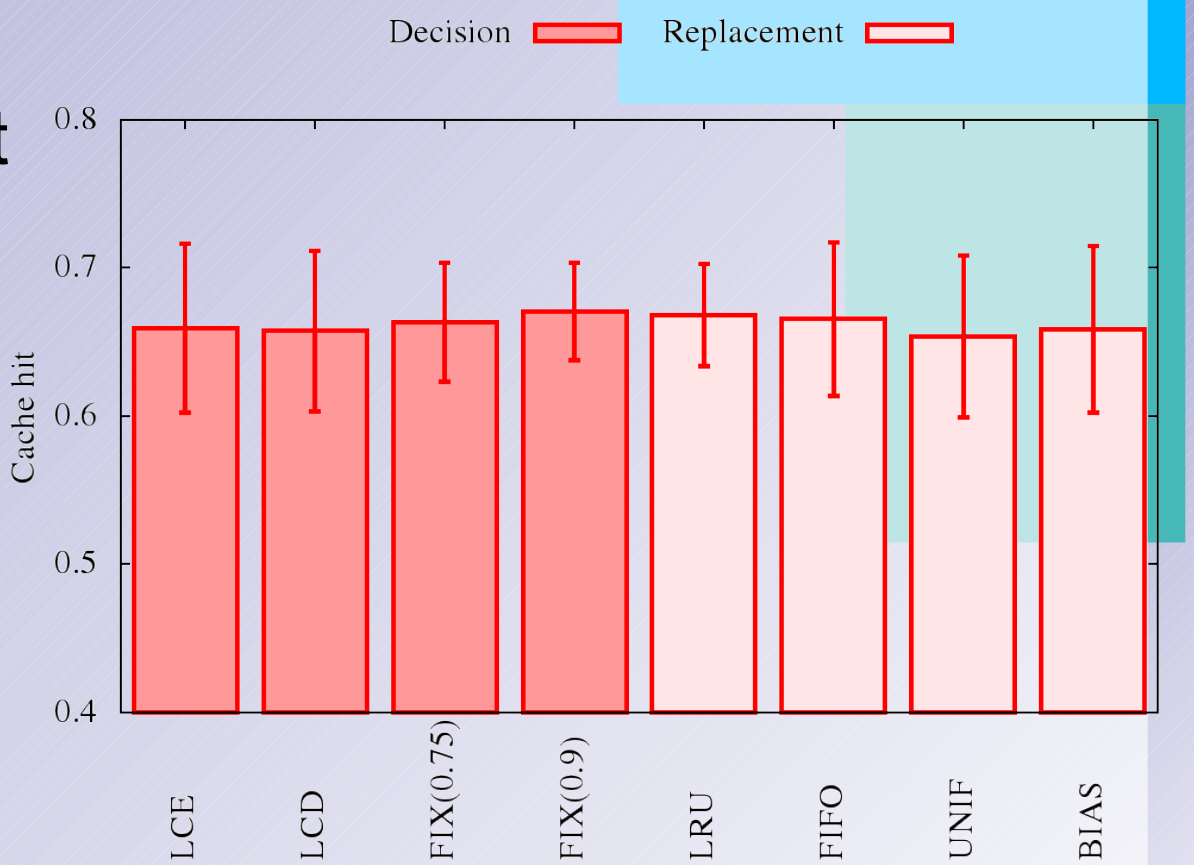
Performance: topologies

- Zipf shaping factor α = 1.5
- On the average not very influenced by the topology
- Even the choice of different policies seems not affecting too much



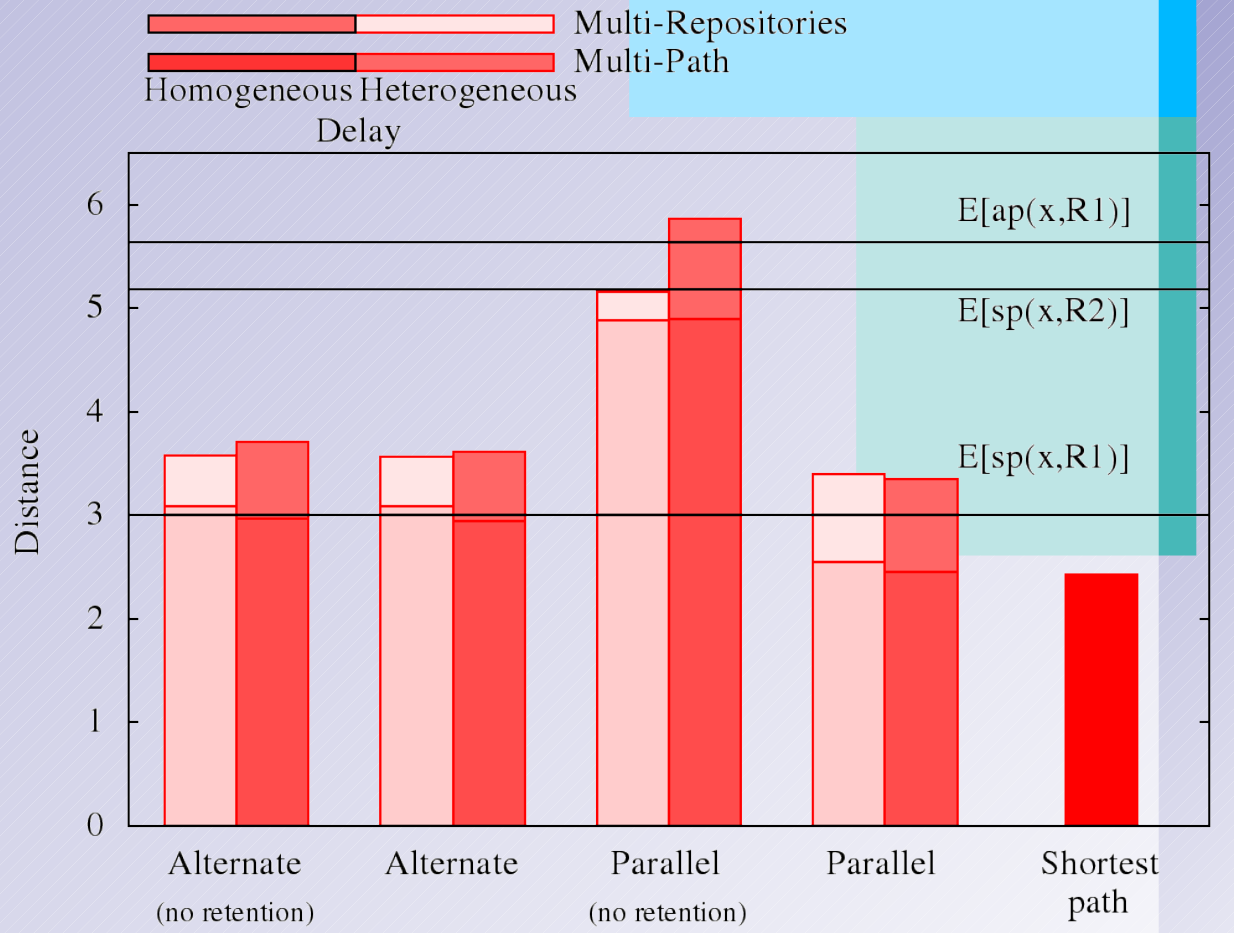
Performance: decision/replacement

- Different decision/replacement policies have little impact on the performance
- This is a good news, as simple LCE/RND policy can be employed
- This is a bad news, as for coarse scenarios few can be improved



Performance: multipath

- Zipf shaping factor $\alpha = 1$
- Having multiple paths may lower performance
- In fact, having longer paths will surely increase the average distance
- Difference between multi-{rep,path} is quite slight



Summary

- Simple replacement policies achieve comparable performance to usual reference
- Multipath can be potentially harmful for caching
- The crucial point is the understanding of the popularity models

Thanks for your attention