Congestion control and in-network caching

D. Saucez[△], L. A. Grieco[◇], C. Barakat[△]
[△]Inria Sophia Antipolis ({first.last@inria.fr})
[◇]Politecnico di Bari (a.grieco@poliba.it)

Motivation

- Information-Centric Networking can massively rely on in-network on-path caching
  - most popular contents tend to be cached close to the consumers
  - least popular contents tend to be cached farther

Motivation

- Information-Centric Networking can massively rely on in-network on-path caching
  - most popular contents tend to be cached close to the consumers
  - least popular contents tend to be cached farther
- **How does in-network on-path caching impact the retrieval time?**

Motivation

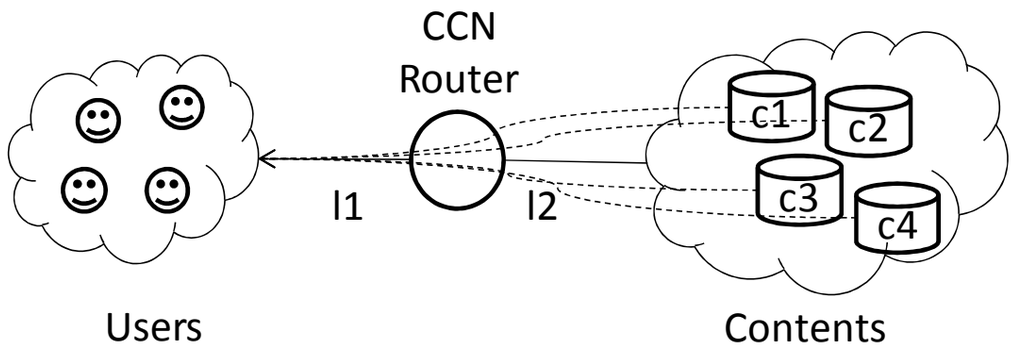
- Information-Centric Networking can massively rely on in-network on-path caching
  - most popular contents tend to be cached close to the consumers
    - least popular contents tend to be cached farther
- **How does in-network on-path caching impact the retrieval time?**
- **How does it influence the fairness?**

Motivation

- Information-Centric Networking can massively rely on in-network on-path caching
  - most popular contents tend to be cached close to the consumers
  - least popular contents tend to be cached farther
- **How does in-network on-path caching impact the retrieval time?**
- **How does it influence the fairness?**
- **How does in-network on-path caching impact server load?**

Hypotheses 1/2

- One *consumer* site initiates ALL the Interests
- One (other) site initiates ALL the Data packets (content *producer*)
- Consumer and producer sites connected by a chain of LRU caches of length *H*
- Every link with delay *d*, total delay *H*d*



Example for *H*=1

Hypothesis 2/2

- Congestion is controlled by and only by the requester with "*Additive Increase Multiplicative Decrease*" (AIMD)

- Throughput for content *c* in AIMD given by

$$T(c) = \frac{K}{RTT(c)\sqrt{p(c)}}$$

Hypothesis 2/2

- Congestion is controlled by and only by the requester with "*Additive Increase Multiplicative Decrease*" (AIMD)

- Throughput for content *c* in AIMD given by

$$T(c) = \frac{K}{RTT(c)\sqrt{p(c)}}$$

How does in-network on-path caching impact the retrieval time?

- RTT for a content is the RTT to the first node that caches the content
- The average position is given by the hit rate $\omega_j(c)$ of nodes in the chain
- The average delay for *c* is then

$$RTT(c) = d \sum_{i=1}^{H} i\omega_i(c) \prod_{j<i} [1 - \omega_j(c)]$$

How does in-network on-path caching impact the retrieval time?

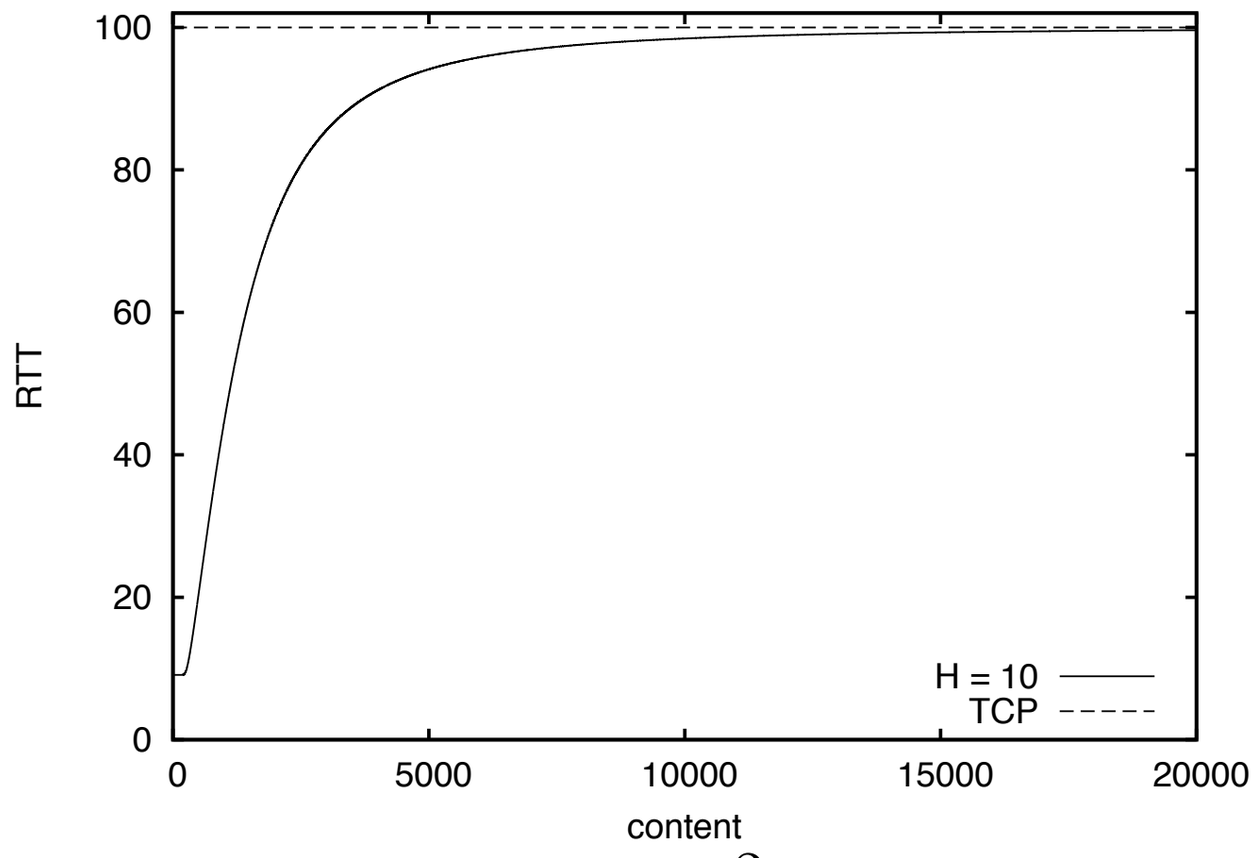
- RTT for a content is the RTT to the first node that caches the content
- The average position is given by the hit rate $\omega_j(c)$ of nodes in the chain
- The average delay for *c* is then

$$RTT(c) = d\sum_{i=1}^{H} i\omega_i(c)\prod_{j<i}[1-\omega_j(c)]$$

Probability of reaching node *i*

Short RTT for popular contents

- 1,000,000 contents
- 10,000 caching entries total



$\alpha = 2$

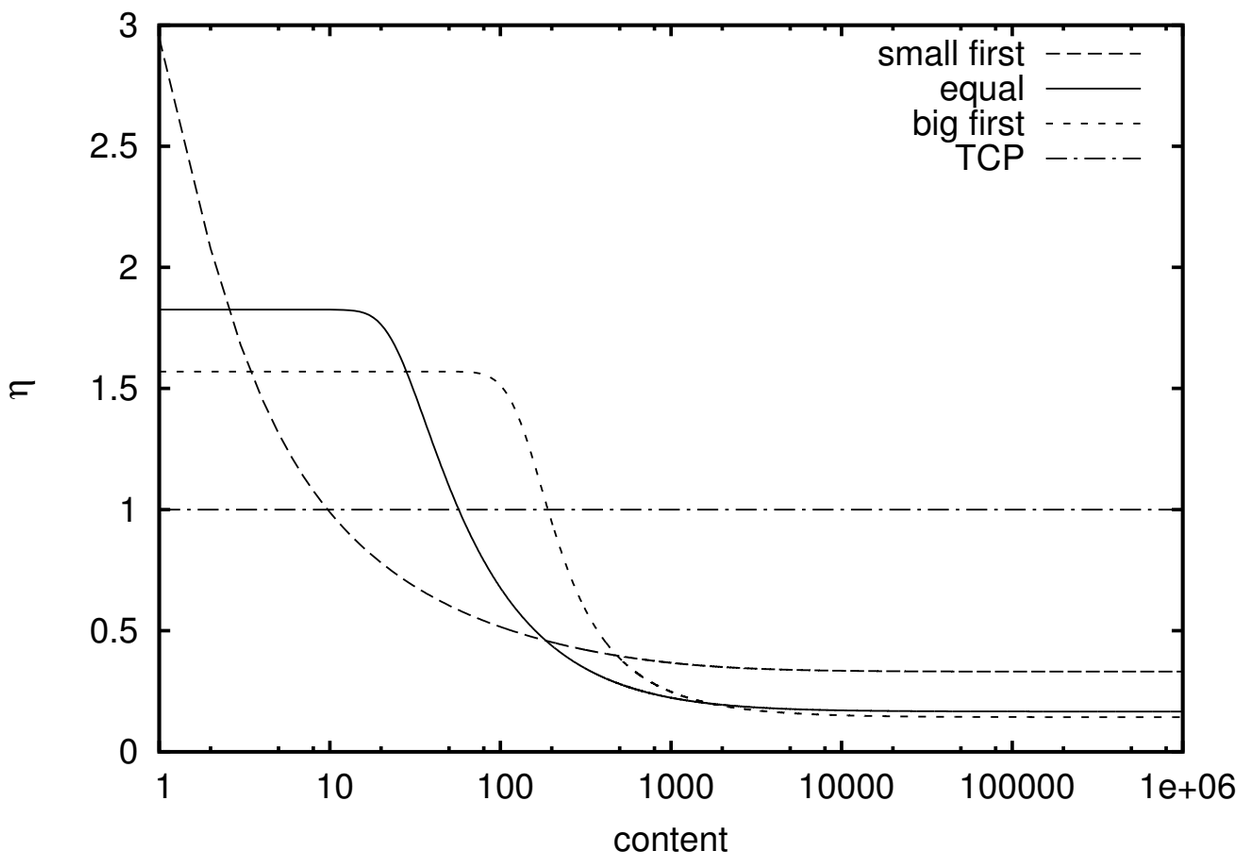
How does it influence the fairness?

- Baseline: TCP (i.e., no cache) throughput
- Is the throughput gain identical for every of the *N* downloads?
- Metric: ratio of throughput with and without cache, for any download *i*

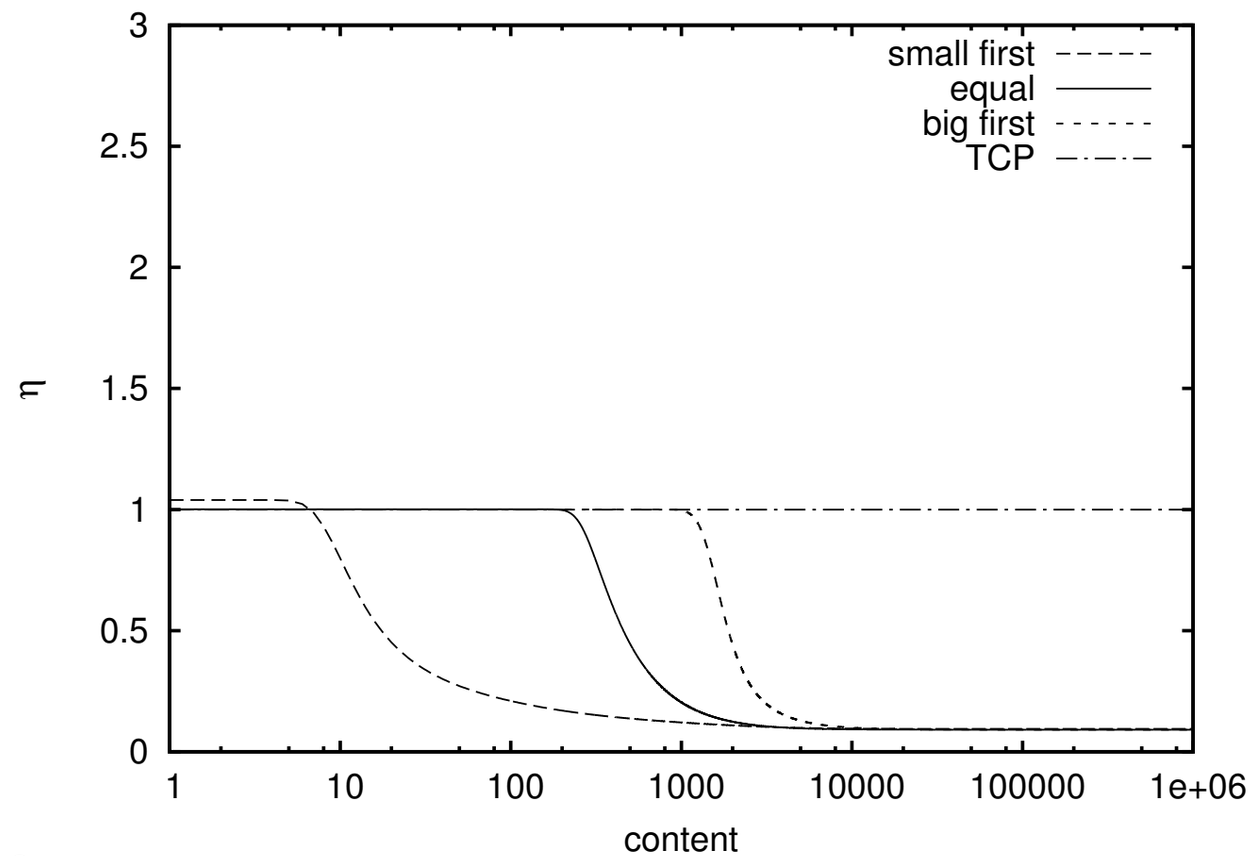
$$\eta(c_i) = \frac{T(c_i)}{\hat{T}(c_i)} \approx \frac{1/RTT(c_i)}{\frac{\sum_{j=1}^{N} 1/RTT(c_j)}{N}}$$

Negative impact for least popular contents

- Individually, very popular contents might not gain that much...
- ... but least popular lose a lot (up to $\frac{1}{H+1}$)



$\alpha = 1.1$



$\alpha = 2$

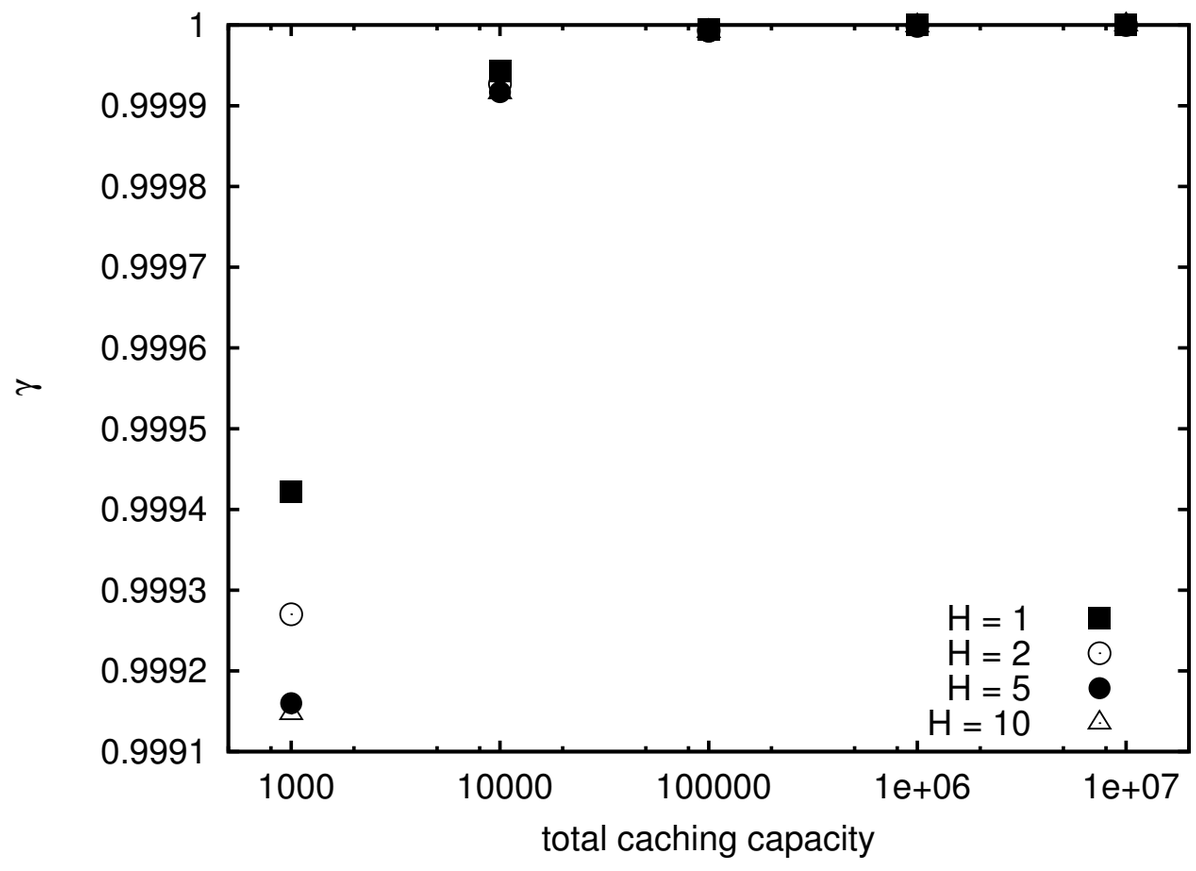
How does in-network on-path caching impact server load?

- Hypothesis: processing cost and data size is the same for every content
- Metric: ratio of server link usage with and without cache, for any content *c*

$$\gamma = 1 - \frac{\Lambda(l_{H+1})}{\Lambda(\hat{l}_{H+1})} = 1 - \frac{\sum_{i=1}^{N} \frac{\prod_{j=1}^{H}[1-\omega_j(c_i)]}{RTT(c_i)}}{\sum_{j=1}^{N} \frac{1}{RTT(c_j)}}$$

Server load is reduced

- Limited impact of the chain length on server load



$\alpha = 2$

Conclusion

- What's happening if AIMD is used by ICN clients?
  - RTT is a function of the popularity when caches are used
    - Throughput is a function of the invert of RTT
    - Popular contents obtain more resources than the others
- We MUST think about this problem when we will design a congestion control

Question to the RG

- Is the RG ready to work on this?
  - Who?

Congestion control and in-network caching

?? || /**/

more details: http://hal.inria.fr/hal-00719793/