Incentive-Compatible Caching and Inter-Domain Traffic Engineering in CCN

Xun Shao, Hitoshi Asaeda

2016-05-19

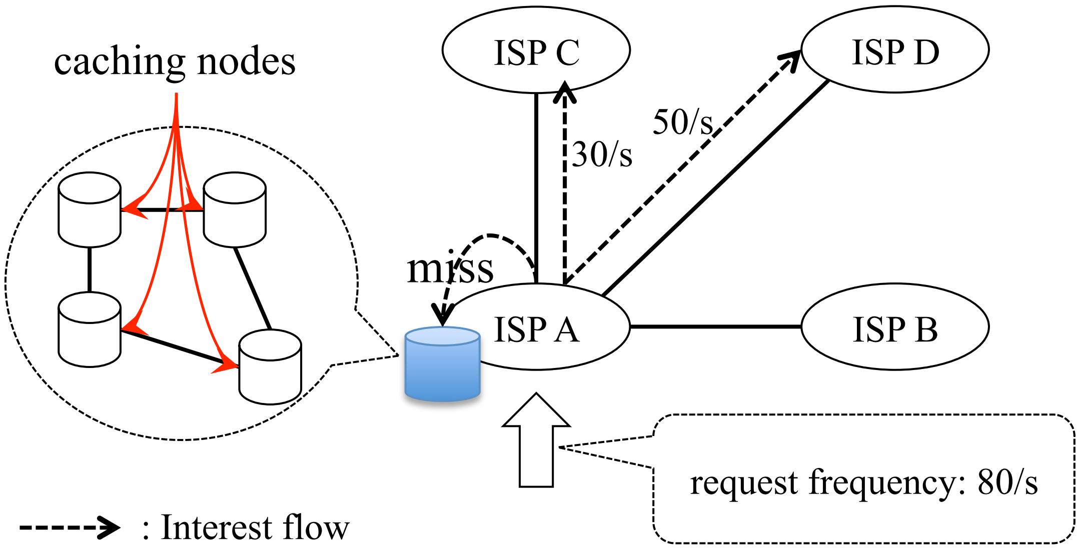
National Institute of Information and Communications Technology (NICT)

Outline

- Caching and inter-domain traffic engineering in CCN
- Joint optimization of caching and inter-domain traffic engineering for a single ISP (Feng et al. 2015)
- Interaction of multiple peering ISPs in caching and inter-domain traffic engineering (Pacifici et al. 2016)
- Problems and opportunities for future research

Caching and inter-domain traffic engineering in CCN

- Premises and assumptions on caching and inter-domain traffic engineering in CCN
  - The intra-domain cache nodes are abstracted as a single node
  - ISP determines which content to cache according to the properties of content
  - ISP has full control of both outbound traffic and inbound traffic



- Opportunities and challenges to ISPs:
  - Make better decision in caching (what to cache) and inter-domain routing (to whom to send specific interests)
  - Explore the opportunities in coordination with neighboring ISPs considering the business relationships (free-settle peering, transit, etc.)

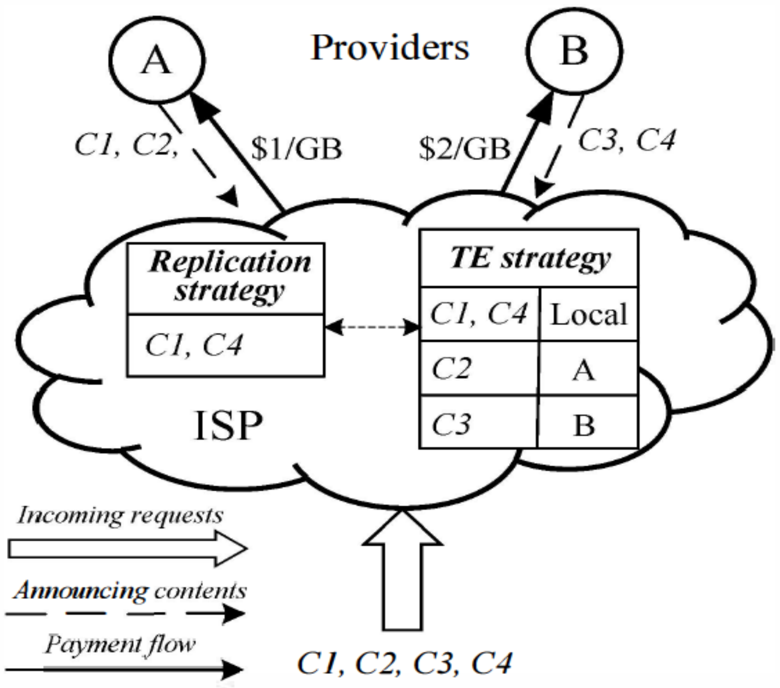
Joint Optimization of Content Replication and Traffic Engineering in ICN

Authors: Z. Feng et al.
Proc. IEEE LCN 2015

Introduction

- The opportunity of jointly optimizing caching and inter-domain traffic engineering for CCN-enabled ISP is explored
- A jointly optimization frame work for caching and inter-domain traffic engineering is introduced
- Simulations show that the proposed method can increase the ISP’s profit significantly

A toy example

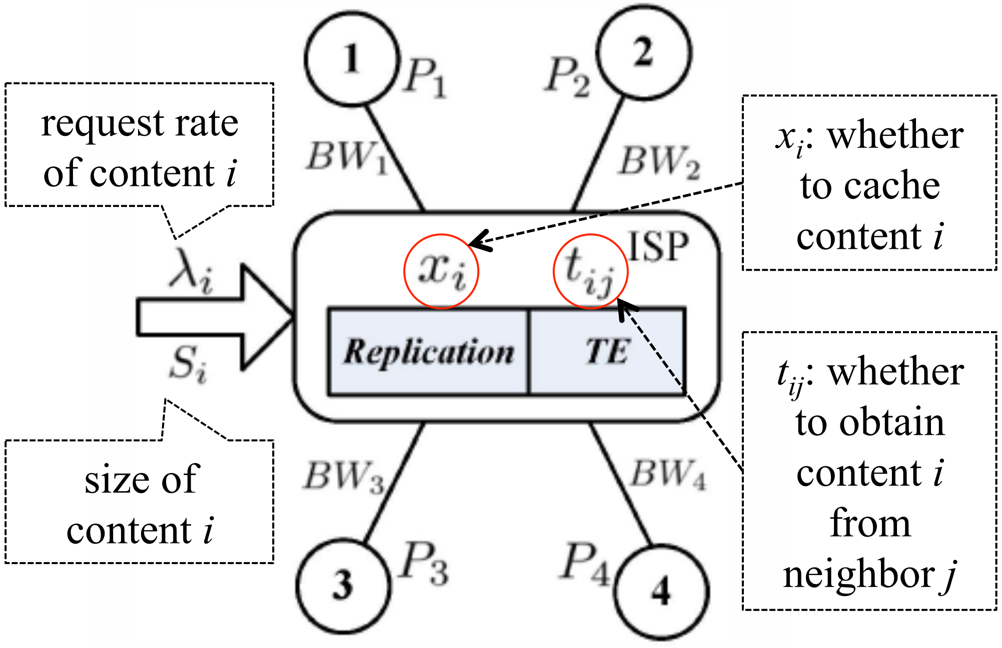


Contents	Requested frequency (times/s)	Size (GB)	Monetary cost ($/s) if not cached
C1	5	0.1	$0.5
C2	2.5	0.1	$0.25
C3	2	0.2	$0.8
C4	2	0.1	$0.4

Cache capacity: 0.2 GB

Strategies	Cache	Monetary cost ($/s)
Popularity prioritized	C1, C2	$1.2
Price prioritized	C3	$1.15
Joint optimization	C1, C4	$1.05

System model and solutions

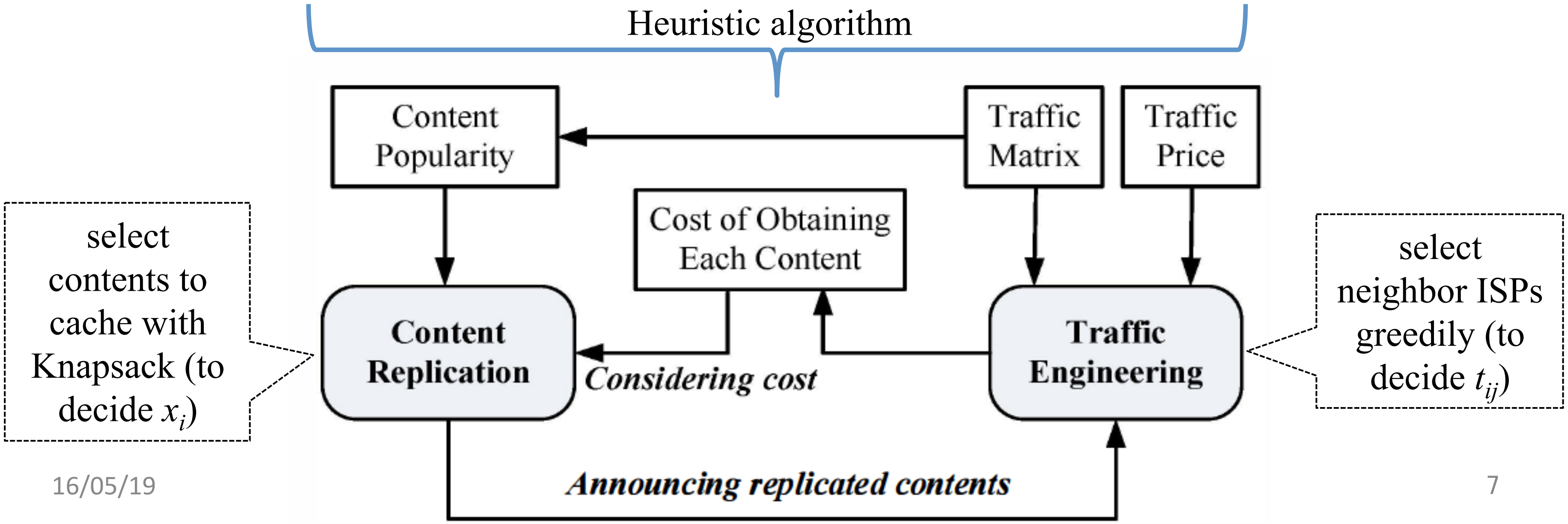


$$\text{Maximize} \quad \sum_{i=1}^{N}\sum_{j=1}^{M} \lambda_i S_i P_j t_{ij}$$

$$\text{Subject to}: \quad \sum_{i=1}^{N} S_i x_i \leq \text{Storage}$$

$$\sum_{i=1}^{N} \lambda_i S_i t_{ij} \leq \beta \times BW_j$$

NP−hard problem even without x_i !

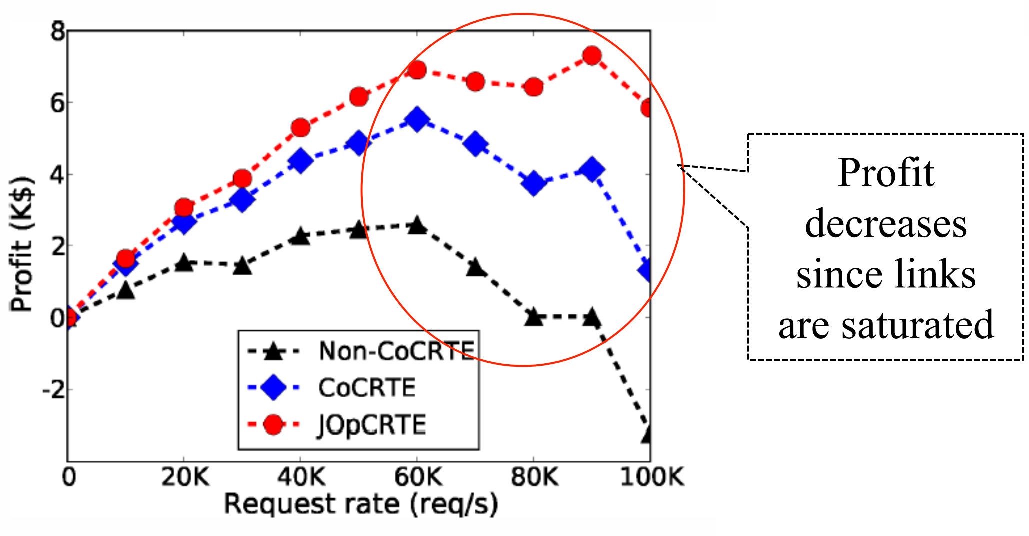


Evaluation

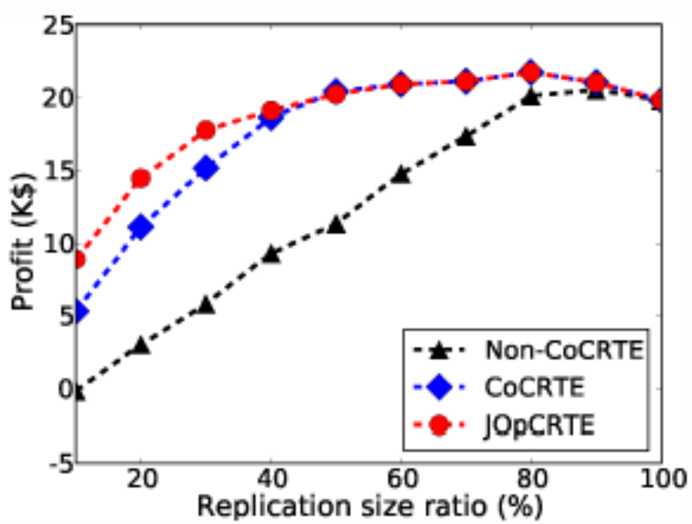
ASN	ISP Name	# Providers	# Customers	# Peers	# IP prefixes of neighbors[7]
AS 8002	Stealth Communications	2	3	21	3215
AS 25973	Global Telecom & Technology	3	26	13	5153
AS 5400	British Telecom	5	103	49	14655
AS 14744	Internap	6	36	1	401
AS 3209	Vodafone	7	120	65	7235
AS 7713	Telkom Indonesia International	8	137	19	8326

Request rate	Sizes of contents	Average size of content	Bandwidth of links	Transit fee
Zipf's law	Pareto distribution	1.7 MB	40Gbps	U(0.1, 0.2)

JOpCRTE: the proposed method; **Non-CoCRTE**: Greedy algorithm with popularity; **CoCRTE**: Greedy algorithm with price



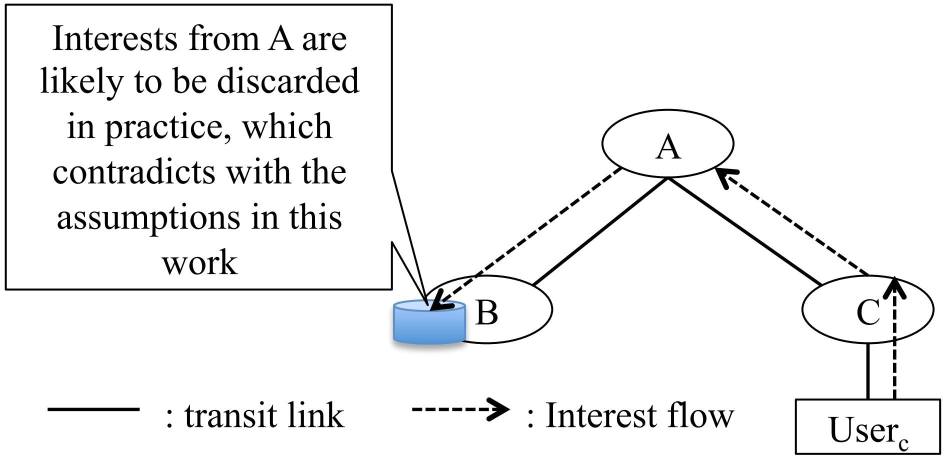
Profit of increasing request rates (Zipf α=0.7, replication size ratio=0.2)



Profit of increasing replication size ratio (request rate=90k/s, Zipf α=0.7)

Summary

- A jointly optimization frame work for caching and inter-domain traffic engineering which is difficult for IP networking was introduced
- Simulations show that the proposed method can increase ISP's profit up to 66%
- Cache level coordination among ISPs is not considered
- A disputable assumption about routing policy in this work:



Coordinated Selfish Distributed Caching for Peering Content-Centric Networks

Authors: V. Pacifici et al.

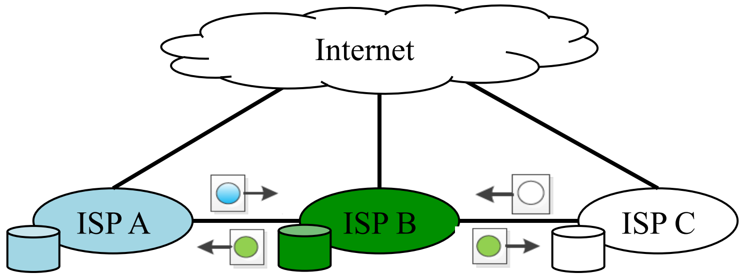
IEEE/ACM Trans. Networking, March 2016

Introduction

- A model of the interaction between the caches managed by peering ISPs is introduced
- Peering ISPs can converge to a stable configuration efficiently by avoiding simultaneous updates
- The analytical results are validated using simulations on the measured peering topology of more than 600 ISP

Problem definition and model

- Cache coordination of peering ISPs:
  - Method: ISP advertises the content names in its local cache to peering neighbors periodically
  - Objective: obtain contents from peering ISPs to save transit fee



- The decision variant C_i of ISP i : the content set in its local cache

request frequency to content o

- The cost for ISP i to obtain content o: $C_i^o(\mathcal{C}_i, \mathcal{C}_{-i}) = w_i^o \begin{cases} \alpha_i & \text{if } o \in \mathcal{L}_i \cup \mathcal{R}_i \\ \gamma_i & \text{otherwise,} \end{cases}$

alpha$_i$: unit cost for ISP i to obtain contents from local cache or peering ISP's cache
gama$_i$: unit traffic fee for ISP i to obtain contents from transit ISP

- Total cost of ISP i: $C_i(\mathcal{C}_i, \mathcal{C}_{-i}) = \alpha_i \sum_{\mathcal{L}_i \cup \mathcal{R}_i} w_i^o + \gamma_i \sum_{\mathcal{O} \setminus \{\mathcal{L}_i \cup \mathcal{R}_i\}} w_i^o$

where: $\mathcal{L}_i = \mathcal{C}_i \cup \mathcal{H}_i$ $\quad \mathcal{R}_i = \bigcup_{j \in \mathcal{N}(i)} \mathcal{L}_j$

C_i: the content set in ISP i's local cache
H_i: the original contents hosted in ISP i's network

A toy example showing the oscillation of cached contents

- Scenario :
  - ISP 1 and ISP 2 are in content-peering relationship
  - The capacity of both the caches equals “2”
  - The popularity of the contents: A > B > C > D

Time sequence	Contents in ISP1’s cache	Contents in ISP2’s cache
t0	A, B	A, B
t1	C, D	C, D
t2	A, B	A, B
t3	C, D	C, D
……	……	……

The proposed algorithms to avoid cache oscillation

Algorithm 1 : Cache-or-Wait (CoW):

Independent set : A set I is an independent set of peering graph G if it does not contain peering ISPs

algorithm summary : In time slot t, the ISPs in the independent set I_t update cache to minimize their total cost. The ISPs not belonging I_t are not allowed to update cache, and have to wait for their time slot

- Pick $\mathcal{I}_t$.
- Allow ISPs $i \in \mathcal{I}_t$ to change their cached items from $\mathcal{C}_i(t-1)$ to $\mathcal{C}_i(t)$,
- For all $j \notin \mathcal{I}_t$, $\mathcal{C}_j(t) = \mathcal{C}_j(t-1)$.
- At the end of the time slot inform the ISPs $j \in \mathcal{N}(i)$ about the new cache contents $\mathcal{C}_i(t)$

The time for all the ISPs to finish cache eviction is too long

Algorithm 2 : Cache-No-Wait(CnW):

algorithm summary : all the ISPs are allowed to update cache independently; after the cache eviction, ISPs are required to acknowledge their neighbors about the updated contents

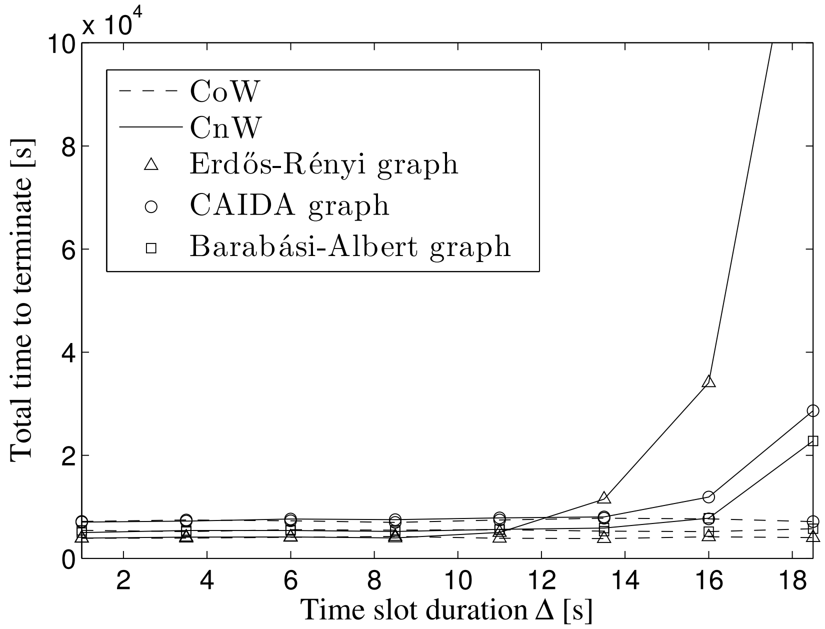
- Every ISP $i \in N$ is allowed to change its cached items from $\mathcal{C}_i(t-1)$ to $\mathcal{C}_i(t)$.
- At the end of the time slot ISP i informs the ISPs $j \in \mathcal{N}(i)$ about the new cache contents $\mathcal{C}_i(t)$

Both CoW and CnW are proved to converge to stable state theoretically

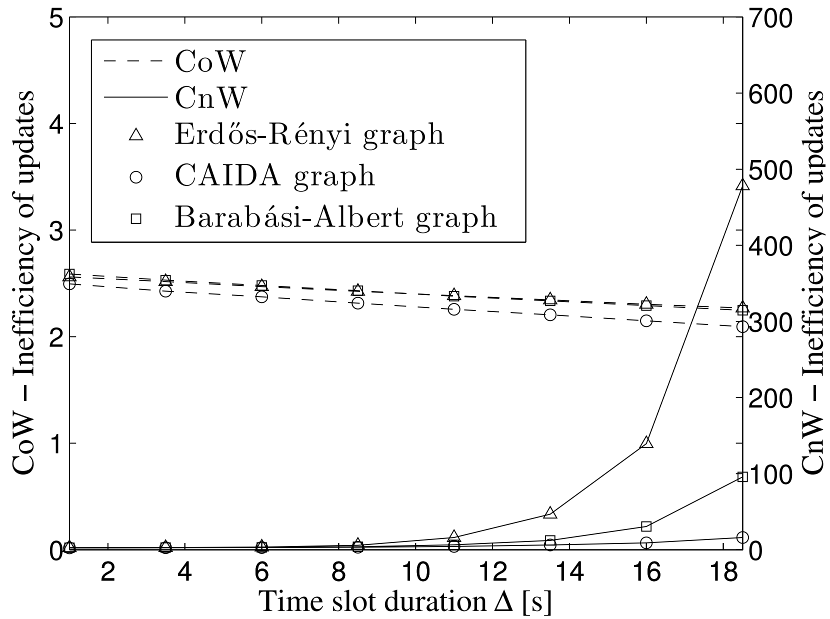
Validation with simulations

Simulation settings:

- topologies: CAIDA graph, ER graph, BA graph
- 616 ISPs with average degree 9.66
- alpha = 1, gama = 10, cache capacity is "10"



Average time needed to terminate as a function of the time slot duration Δ for three different peering graphs and algorithms COW and CNW.



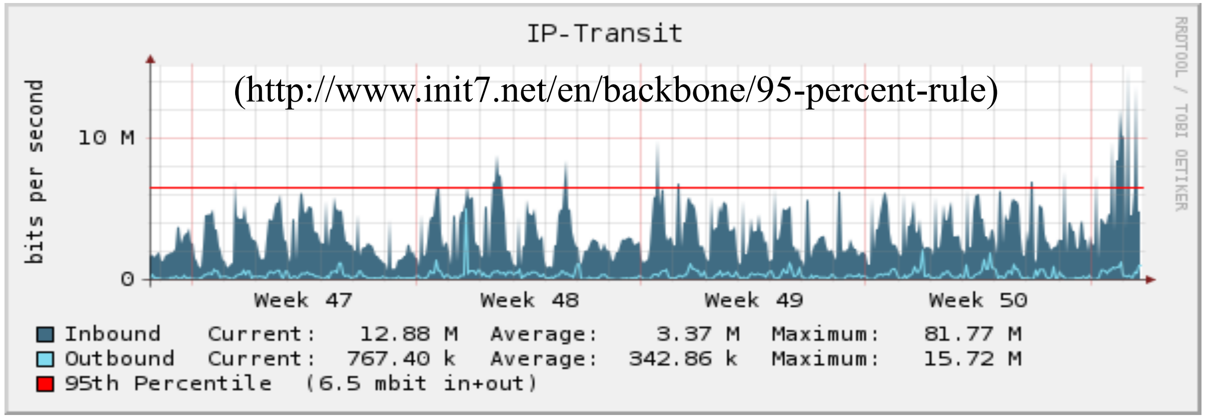
Average inefficiency as a function of the time slot duration Δ for three different peering graphs and algorithms COW and CNW.

Summary

- A model of the interactions between the caches managed by peering ISPs in CCN was proposed
- Synchronizing algorithms to avoid simultaneous cache evictions were introduced for fast convergence to a stable cache configuration
- This work focused on the convergence of the algorithms rather than the ISPs' benefit from content–peering

Problems and opportunities for future research

- Lack of in-depth study with practical situations
  - e.g. considering 95 percentile measurement rule, there are opportunities to further improve the ISP coordination benefits



- Lack of incentive mechanisms for ISPs to extend the cooperation targets
  - e.g. to enable the following coordination