Design and evaluation of Cost-Aware Information Centric Routers

A. Araldo
araldo@lri.fr

D. Rossi
dario.rossi@enst.fr

F. Martignon
fabio.martignon@lri.fr

Paris, 26 September 2014
perso.telecom-paristech.fr/~araldo

Outline

- Motivations
  - Need for investigating economic implications of ICN
- Design of Cost-Aware (CoA) caching
  - Goal: reduction of ISP cost
  - Implementable in ICN routers, interoperable
- Evaluation through simulation
  - Noticeable cost savings (~15%)
  - Savings are robust w.r.t. different scenarios

Motivations: ICN benefits

“Named data is a better abstraction for today’s communication problems” [CoNEXT09]

“CCN offers much better security, delivery efficiency, and disruption tolerance” [CoNEXT09]

Motivations: ICN benefits

“Named data is a better abstraction for today’s communication problems” [CoNEXT09]

“CCN offers much better security, delivery efficiency, and disruption tolerance” [CoNEXT09]

Is this enough to foster ICN adoption?

Motivations: ICN benefits

"Named data is a better abstraction for today's communication problems" [CoNEXT09]

"CCN offers much better security, delivery efficiency, and disruption tolerance" [CoNEXT09]

Is this enough to foster ICN adoption?

"Money plays the largest part in determining the course of history"
Karl Marx

Motivations: ICN benefits

"Named data is a better abstraction for today's communication problems" [CoNEXT09]

"CCN offers much better security, delivery efficiency, and disruption tolerance" [CoNEXT09]

Is this enough to foster ICN adoption?

"Money plays the largest part in determining the course of history"
Karl Marx

Scientists are idealists.
They aim to design efficient systems

Internet players are pragmatic.
They will keep on operating as today if they see no clear **economic incentives** to switch to ICN, event if it is better

Related work on ICN

Several studies with an engineering viewpoint
[IFIP13], [ComCom13], [Elsevier12], [ICNP13], [Globecom13], [CNSM13]

- Hit ratio, link load, latency

Few studies on the economic impact
[ComMag12], [INFOCOM13], [SocialCom13], [IFIP13]

- Interactions between different ISPs, considered as atomic entities

We jointly consider the two viewpoints:

we engineer the ISP network with the goal of
minimizing the cost
related to the inter-ISP link utilization

Design of a Cost-Aware network

- Design goals
  - Flexibility
  - Simplicity
  - Robustness
  - Interoperability
- Where to inject cost-awareness
  - Naming
  - Routing
  - Forwarding
  - Caching

Design:caching is a rich tool

<sth>-aware caching

Design:caching is a rich tool

<sth>-aware caching

- congestion
- latency

Design:caching is a rich tool

<sth>-aware caching

- congestion
- latency
- cost

Design:caching is a rich tool

<sth>-aware caching

- congestion
- latency
- cost
- ...

Design: price heterogeneity

If you want to get cost savings, increase hit ratio

Design: price heterogeneity

~~If you want to get cost savings, increase hit ratio~~

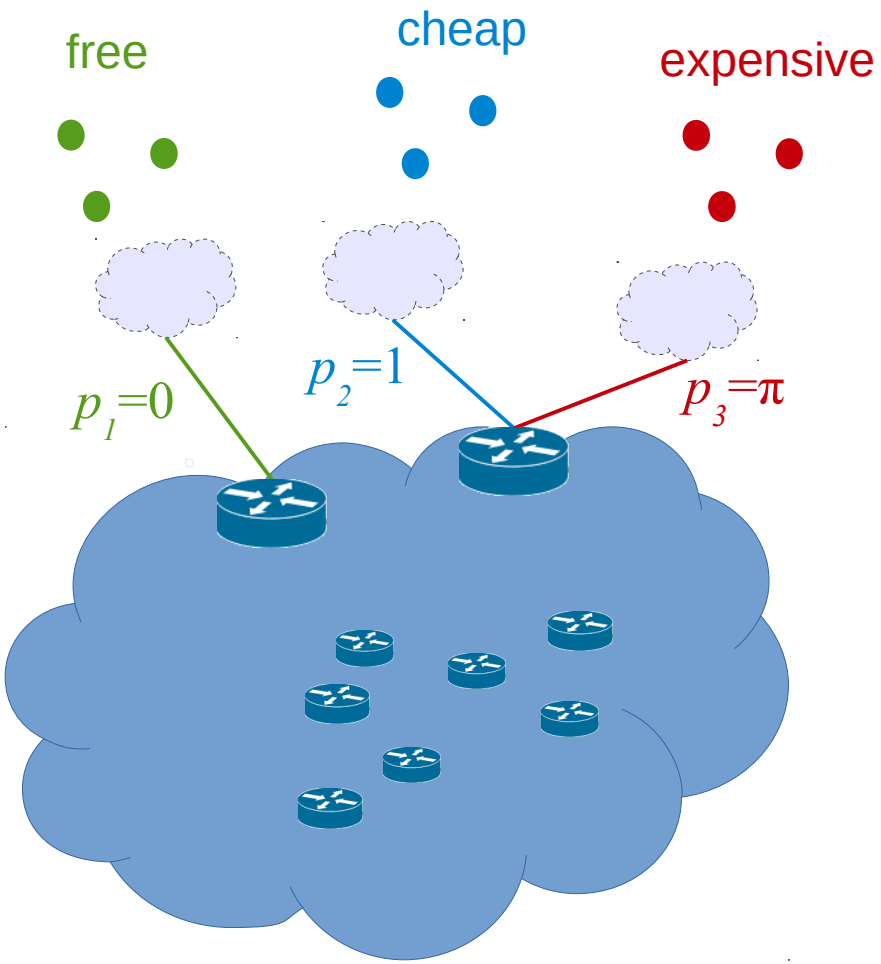
NOT NECESSARILY TRUE

We analytically proved [Globecom14]
a **tradeoff** between hit ratio and cost

Design: price heterogeneity

~~If you want to get cost savings, increase hit ratio~~

NOT NECESSARILY TRUE



We analytically proved [Globecom14] a **tradeoff** between hit ratio and cost

Basic idea:

- Not all objects have the same value
  - Maximizing the hit ratio blindly across all links is not effective
- Exploit **price heterogeneity**
- *Cache more expensive objects*

Design: where to inject cost-awareness

[Usenix97]: replacement algorithms to evict objects with lowest cost

- The notion of cost is generic
- Algorithms are quite complex
- The cost of all the cached objects must be saved and considered in the computation

Design: where to inject cost-awareness

[Usenix97]: replacement algorithms to evict objects with lowest cost

- The notion of cost is generic
- Algorithms are quite complex
- The cost of all the cached objects must be saved and considered in the computation

NOT IMPLEMENTABLE AT LINE SPEED

Design: where to inject cost-awareness

[Usenix97]: replacement algorithms to evict objects with lowest cost

- The notion of cost is generic
- Algorithms are quite complex
- The cost of all the cached objects must be saved and considered in the computation

NOT IMPLEMENTABLE AT LINE SPEED

Replacement policy is not the good place to inject cost awareness

We need a solution that is stateless and light

Design: Cost-Aware (CoA) decision policy

Given an object *o* :

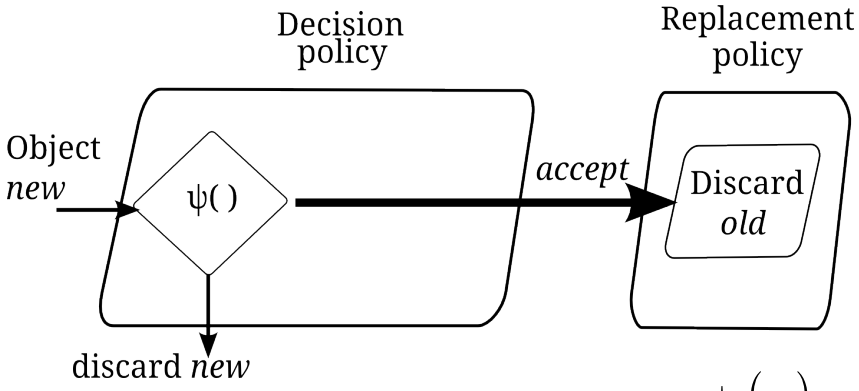
- request rate: λ_o

We proved [Globecom14] that:

- To maximize **hit ratio**: store the objects with highest λ_o

Design: Cost-Aware (CoA) decision policy

- Classic schemes infer λ_o



$$\psi(o) = \begin{Bmatrix} 1 & LCE \\ X & Unif \\ \dots & \dots \end{Bmatrix}$$

Given an object *o* :

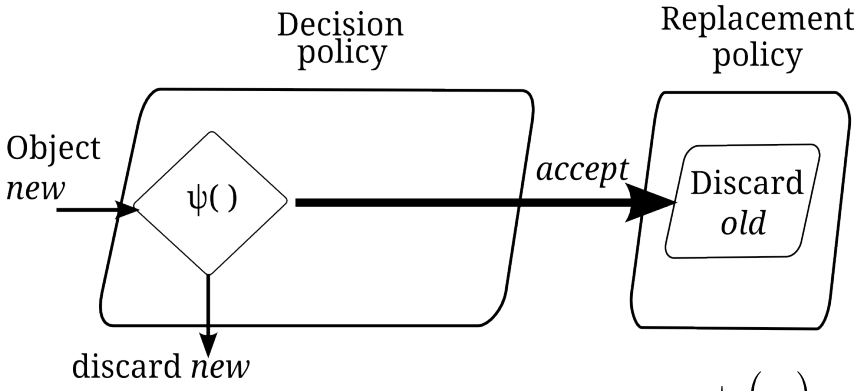
- request rate: λ_o

We proved [Globecom14] that:

- To maximize **hit ratio**: store the objects with highest λ_o

Design: Cost-Aware (CoA) decision policy

- Classic schemes infer λ_o



$$\psi(o) = \begin{Bmatrix} 1 & LCE \\ X & Unif \\ \dots & \dots \end{Bmatrix}$$

Given an object o :

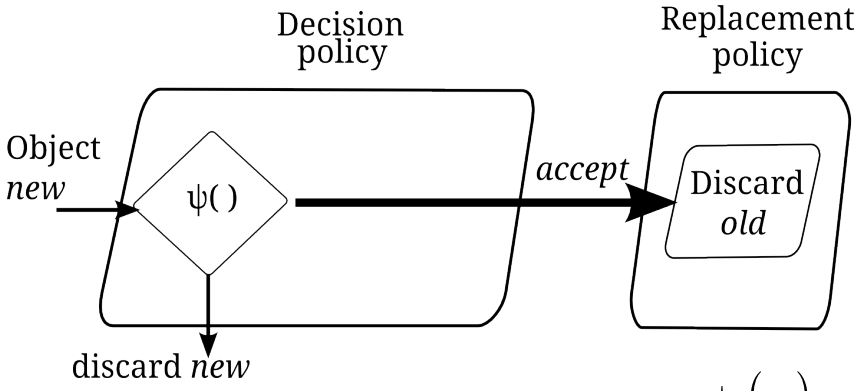
- request rate: λ_o
- price: π_o
- cost: $c_o = \lambda_o \cdot \pi_o$

We proved [Globecom14] that:

- To maximize **hit ratio**: store the objects with highest λ_o
- To minimize **cost**: store the objects with highest c_o

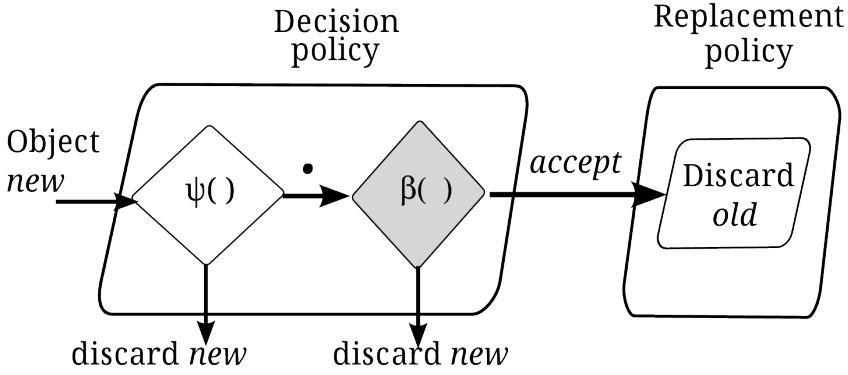
Design: Cost-Aware (CoA) decision policy

- Classic schemes infer λ_o



$$\psi(o) = \begin{Bmatrix} 1 & LCE \\ X & Unif \\ \dots & \dots \end{Bmatrix}$$

- CoA infers c_o



$$\beta(o) = K \cdot \pi_o$$

Given an object o :

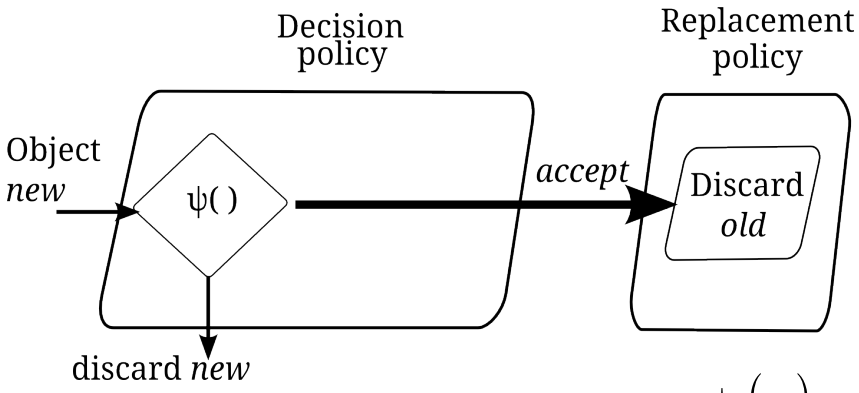
- request rate: λ_o
- price: π_o
- cost: $c_o = \lambda_o \cdot \pi_o$

We proved [Globecom14] that:

- To maximize **hit ratio**: store the objects with highest λ_o
- To minimize **cost**: store the objects with highest c_o

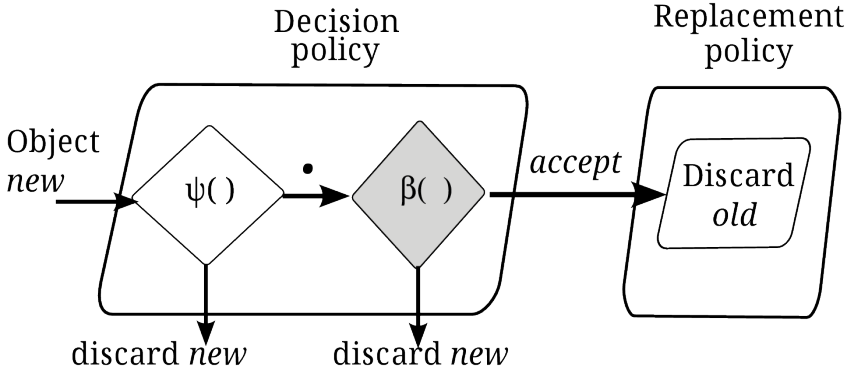
Design: Cost-Aware (CoA) decision policy

- Classic schemes infer λ_o



$$\psi(o) = \begin{cases} 1 & LCE \\ X & Unif \\ \dots & \dots \end{cases}$$

- CoA infers c_o



$$\beta(o) = K \cdot \pi_o \qquad E\left[\psi(o) \cdot \beta(o)\right] = X$$

s.t.

Given an object o :

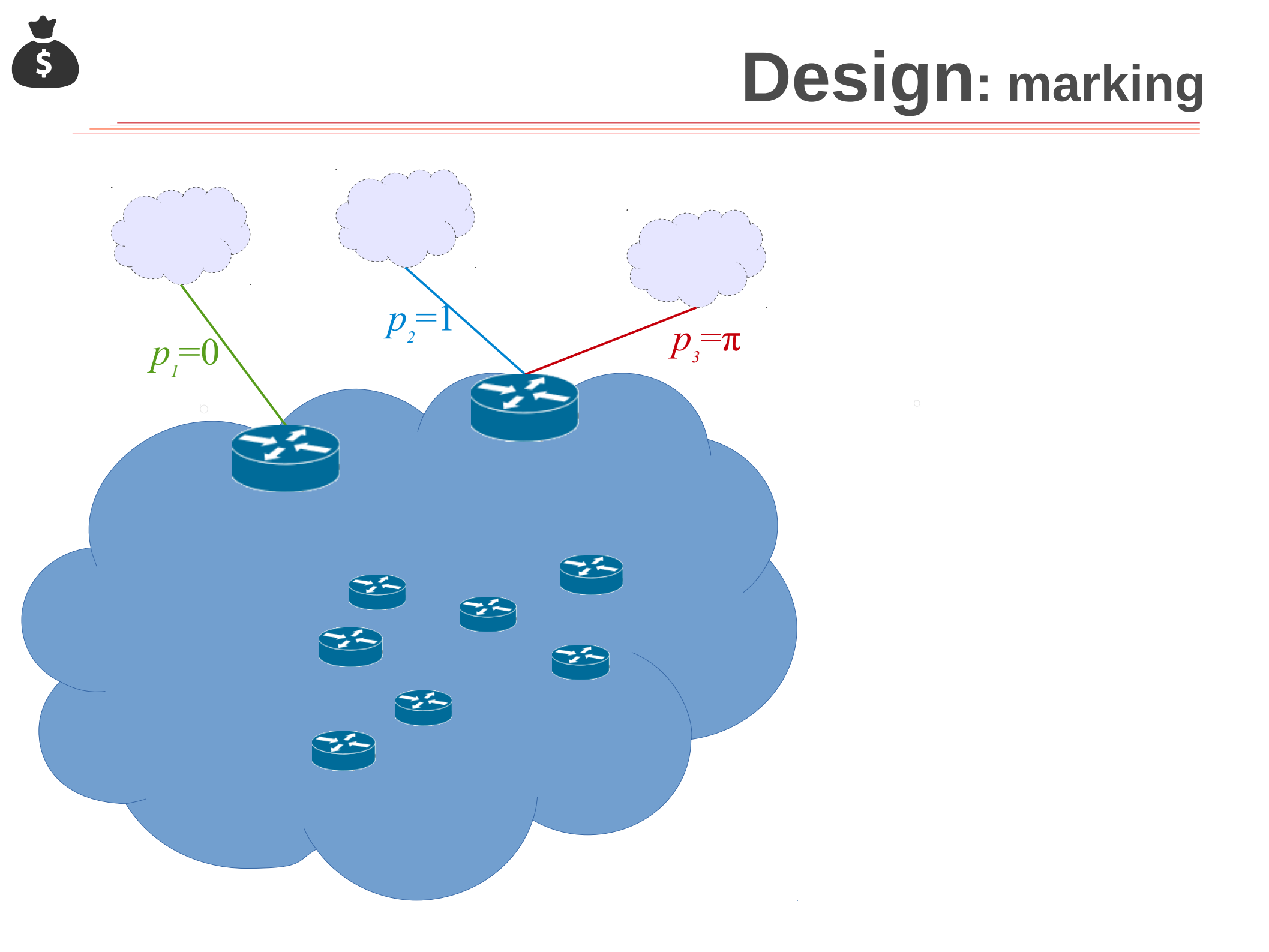
- request rate: λ_o
- price: π_o
- cost: $c_o = \lambda_o \cdot \pi_o$

We proved [Globecom14] that:

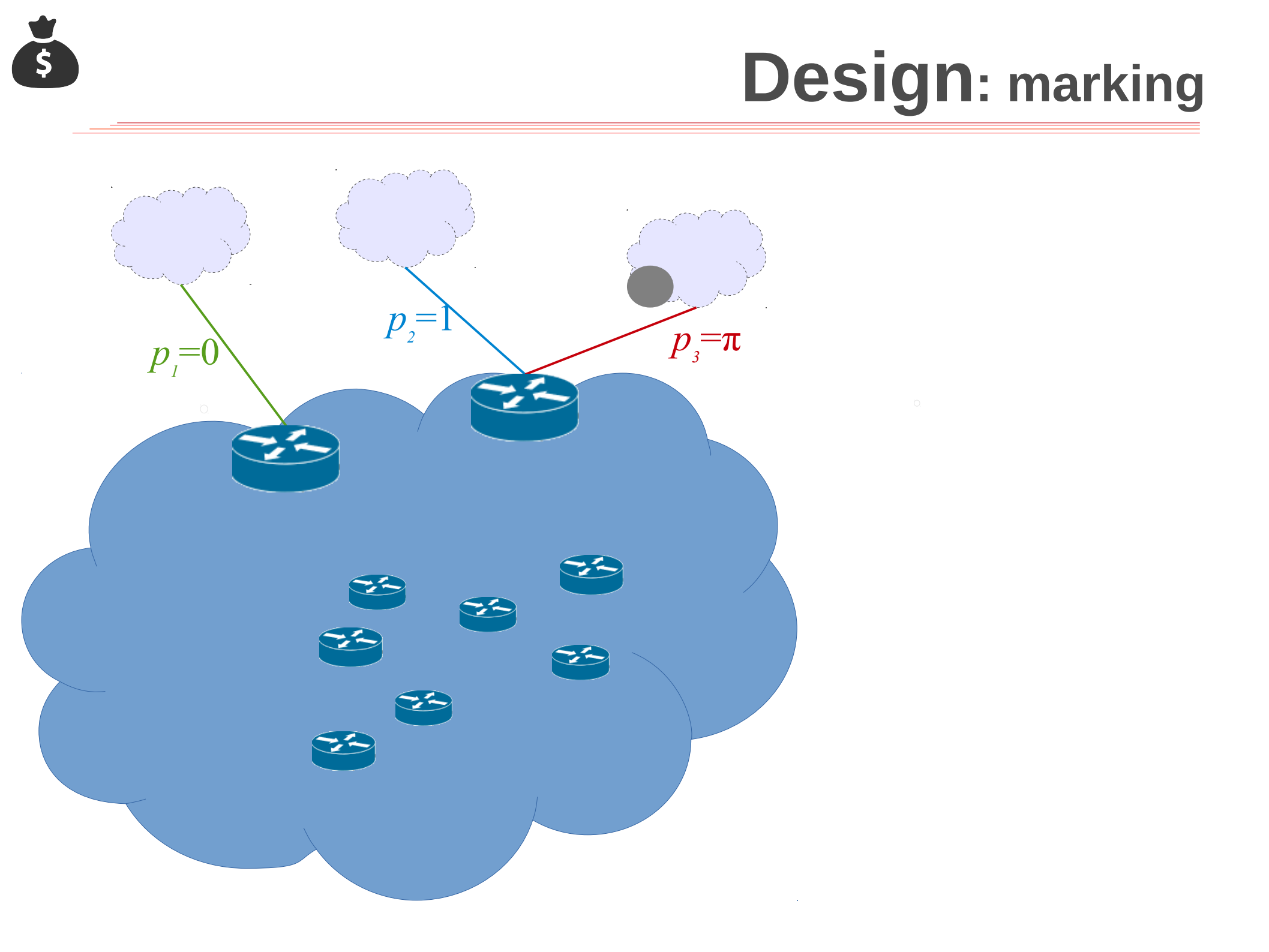
- To maximize **hit ratio**: store the objects with highest λ_o
- To minimize **cost**: store the objects with highest c_o

Average acceptance ratio does not change

The acceptance is just biased toward expensive objects

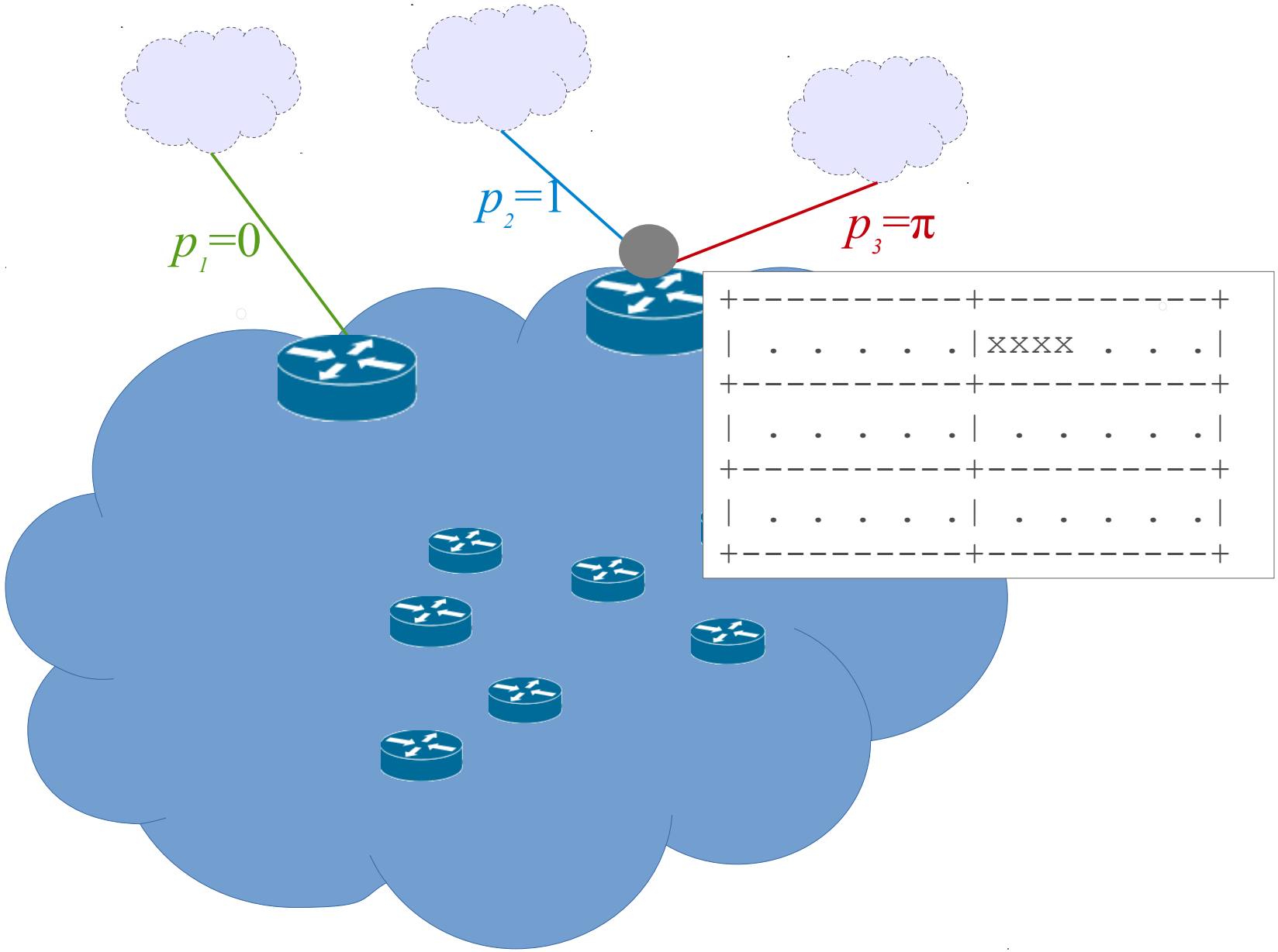


Design: marking

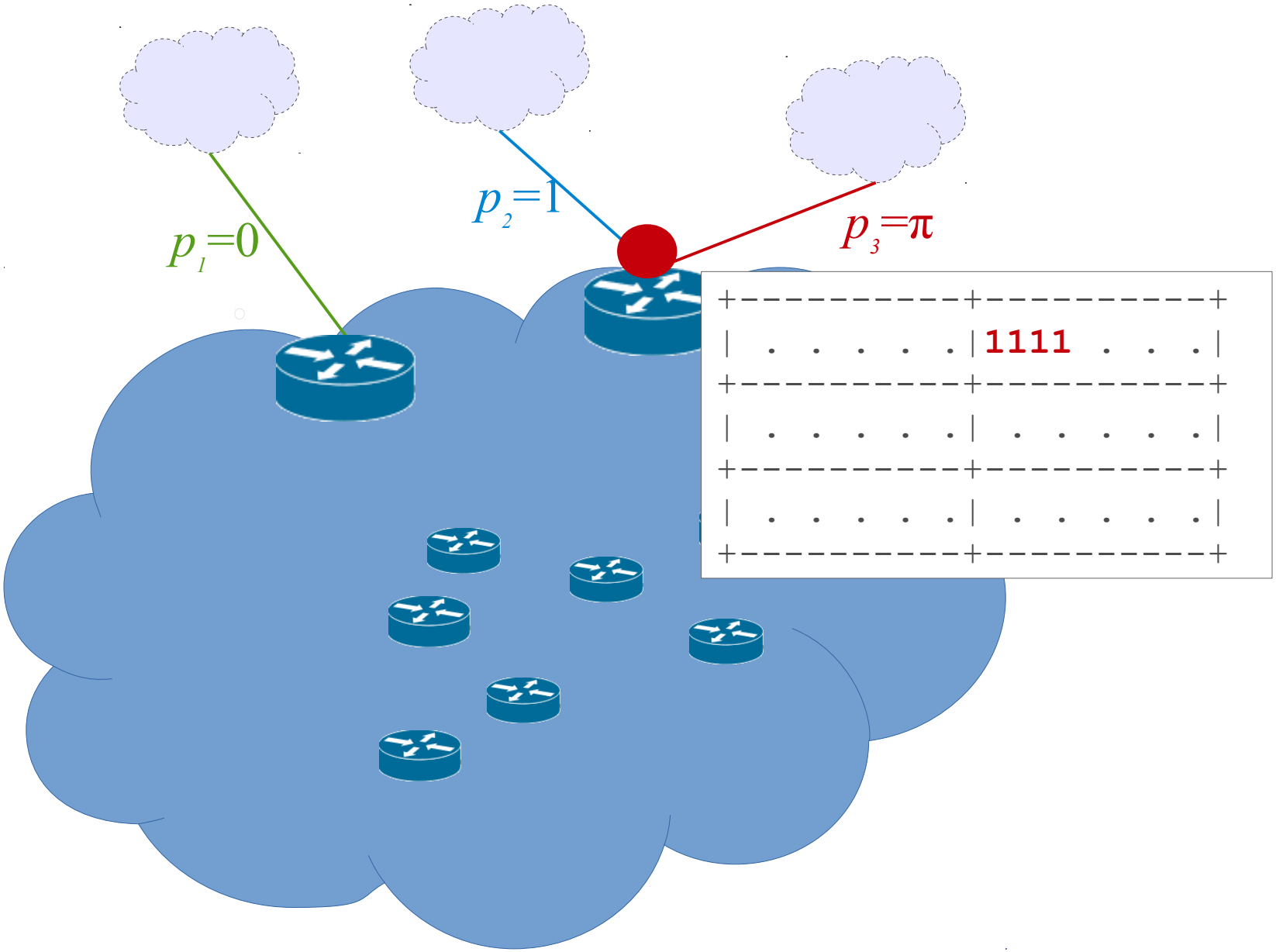


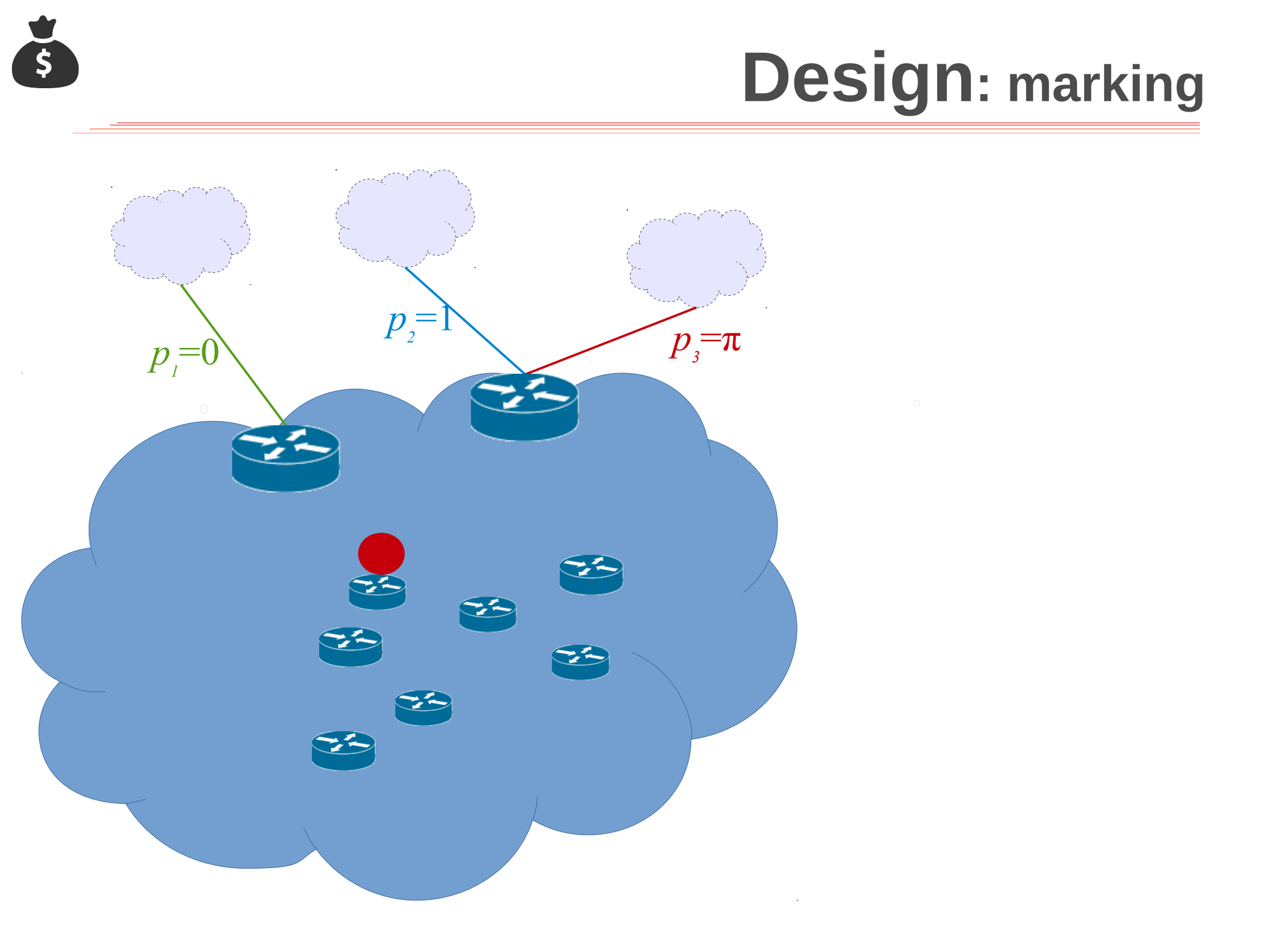
Design: marking

Design: marking



Design: marking





Design: marking

Design: properties of CoA

- Easy to implement
  - Only slight modifications of routers
    - Border routers have to mark objects with price
    - Modify the probability of the decision policy
- Interoperable
  - It is independent from the specific ICN architecture

Evaluation: terms of comparison

- Cost-Blind
  - `LCE`: always accept *o*
  - `Unif`: accept *o* with a fixed probability
- Cost-Aware policies
  - `CoA`: our proposal
  - `Optimum`: fill a priori the cache with the objects with the highest cost $\lambda_o \pi_o$

Evaluation: terms of comparison

- Cost-Blind
  - `LCE`: always accept *o*
  - `Unif`: accept *o* with a fixed probability
- Cost-Aware policies
  - `CoA`: our proposal
  - `Optimum`: fill a priori the cache with the objects with the highest cost $\lambda_o \pi_o$

Not implementable

Evaluation: terms of comparison

- Cost-Blind
  - `LCE`: always accept *o*
  - `Unif`: accept *o* with a fixed probability
- Cost-Aware policies
  - `CoA`: our proposal
  - `Optimum`: fill a priori the cache with the objects with the highest cost $\lambda_o \pi_o$
- Others
  - `Ideal-Blind`, `Ideal-CoA`

Not implementable

Design: scenario

Zipf skew	α	0.8, **1**, 1.2
System scale	$\|C\|/\|O\|$	$10^2/10^4$, **$10^3/10^5$**, $10^4/10^6$, $10^5/10^7$, $10^6/10^8$
Cache to catalog ratio	$\|C\|/\|O\|$	**$10^3/10^5$**, $10^3/10^6$, $10^3/10^7$, $10^3/10^8$
Price ratio	π	1, 2, 5, **10**, 100
Catalog split	$\bar{s} = (s_1, s_2, s_3)$	**$\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)$**, $\left(0,\frac{1}{4},\frac{3}{4}\right)$, $\left(0,\frac{1}{2},\frac{1}{2}\right)$, $\left(0,\frac{3}{4},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{4},\frac{1}{4},\frac{1}{2}\right)$, $\left(\frac{1}{4},\frac{1}{2},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{2},\frac{1}{4},\frac{1}{4}\right)$, $\left(\frac{1}{2},0,\frac{1}{2}\right)$, $\left(\frac{3}{4},0,\frac{1}{4}\right)$, $\left(\frac{3}{4},\frac{1}{4},0\right)$
Replacement policy		LRU

Default values

Design: scenario

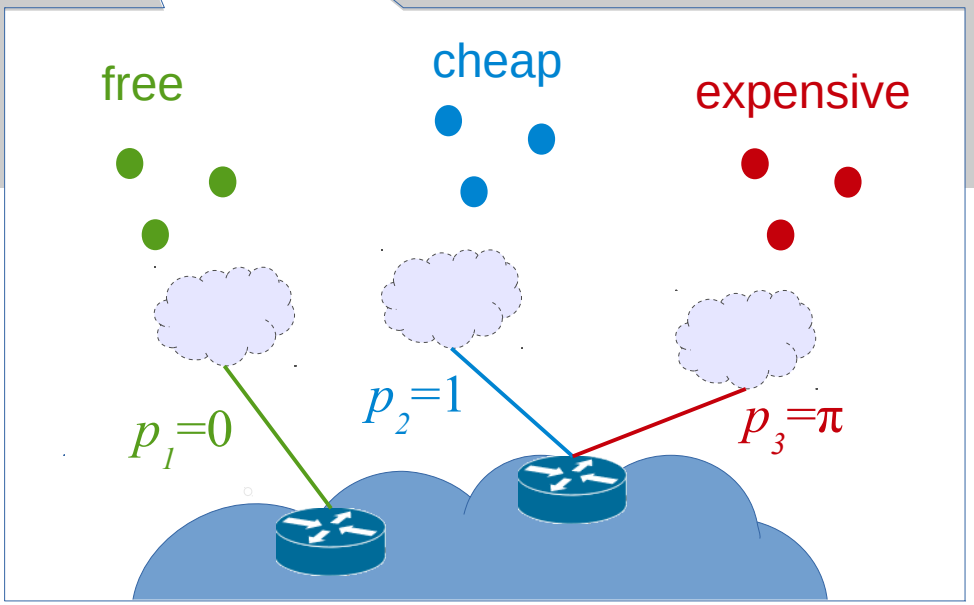
Zipf skew	α	0.8, **1**, 1.2
System scale	$\|C\|/\|O\|$	$10^2/10^4$, **$10^3/10^5$**, $10^4/10^6$, $10^5/10^7$, $10^6/10^8$
Cache to catalog ratio	$\|C\|/\|O\|$	**$10^3/10^5$**, $10^3/10^6$, $10^3/10^7$, $10^3/10^8$
Price ratio	π	1, 2, 5, **10**, 100
Catalog split	$\bar{s} = (s_1, s_2, s_3)$	**$\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)$**, $\left(0,\frac{1}{4},\frac{3}{4}\right)$, $\left(0,\frac{1}{2},\frac{1}{2}\right)$, $\left(0,\frac{3}{4},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{4},\frac{1}{4},\frac{1}{2}\right)$, $\left(\frac{1}{4},\frac{1}{2},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{2},\frac{1}{4},\frac{1}{4}\right)$, $\left(\frac{1}{2},0,\frac{1}{2}\right)$, $\left(\frac{3}{4},0,\frac{1}{4}\right)$, $\left(\frac{3}{4},\frac{1}{4},0\right)$
Replacement poli		LRU

|O| catalog size
|C| cache space

Default values

Design: scenario

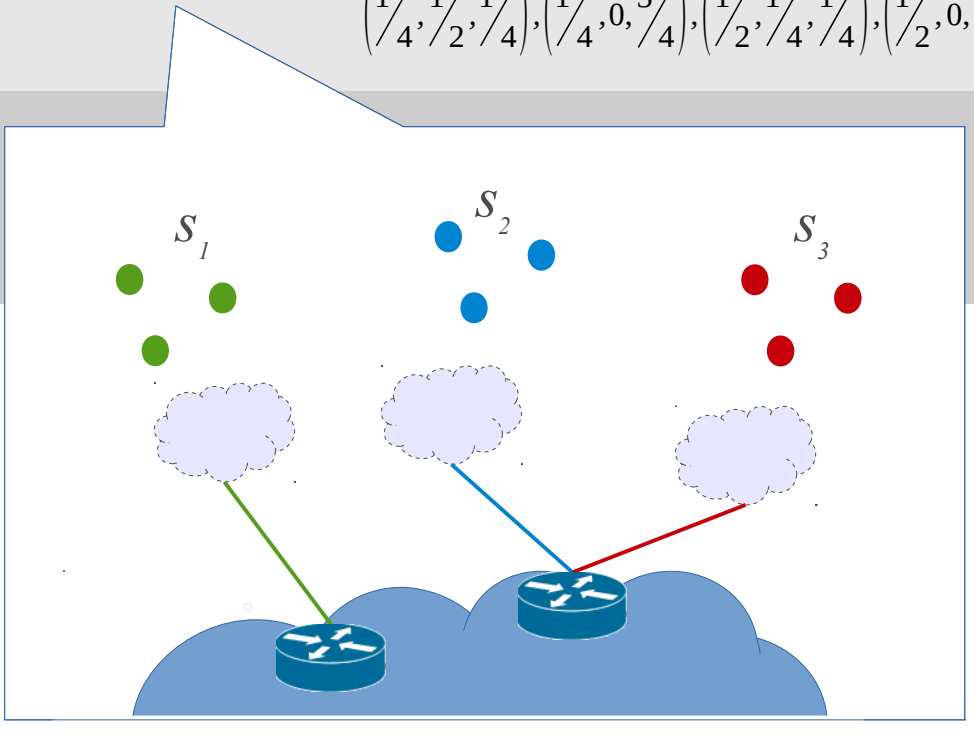
Zipf skew	α	0.8, **1**, 1.2
System scale	$\|C\|/\|O\|$	$10^2/10^4$, **$10^3/10^5$**, $10^4/10^6$, $10^5/10^7$, $10^6/10^8$
Cache to catalog ratio	$\|C\|/\|O\|$	**$10^3/10^5$**, $10^3/10^6$, $10^3/10^7$, $10^3/10^8$
Price ratio	π	1, 2, 5, **10**, 100
Catalog split	$\bar{s} = (\ldots, s_3)$	**$\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)$**, $\left(0,\frac{1}{4},\frac{3}{4}\right)$, $\left(0,\frac{1}{2},\frac{1}{2}\right)$, $\left(0,\frac{3}{4},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{4},\frac{1}{4},\frac{1}{2}\right)$, $\left(\frac{1}{4},\frac{1}{2},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{2},\frac{1}{4},\frac{1}{4}\right)$, $\left(\frac{1}{2},0,\frac{1}{2}\right)$, $\left(\frac{3}{4},0,\frac{1}{4}\right)$, $\left(\frac{3}{4},\frac{1}{4},0\right)$
Replacement policy		



Default values

Design: scenario

Zipf skew	α	0.8, **1**, 1.2
System scale	$\|C\|/\|O\|$	$10^2/10^4$, **$10^3/10^5$**, $10^4/10^6$, $10^5/10^7$, $10^6/10^8$
Cache to catalog ratio	$\|C\|/\|O\|$	**$10^3/10^5$**, $10^3/10^6$, $10^3/10^7$, $10^3/10^8$
Price ratio	π	1, 2, 5, **10**, 100
Catalog split	$\bar{s} = (s_1, s_2, s_3)$	**$\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)$**, $\left(0,\frac{1}{4},\frac{3}{4}\right)$, $\left(0,\frac{1}{2},\frac{1}{2}\right)$, $\left(0,\frac{3}{4},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{4},\frac{1}{4},\frac{1}{2}\right)$, $\left(\frac{1}{4},\frac{1}{2},\frac{1}{4}\right)$, $\left(\frac{1}{4},0,\frac{3}{4}\right)$, $\left(\frac{1}{2},\frac{1}{4},\frac{1}{4}\right)$, $\left(\frac{1}{2},0,\frac{1}{2}\right)$, $\left(\frac{3}{4},0,\frac{1}{4}\right)$, $\left(\frac{3}{4},\frac{1}{4},0\right)$
Replacement policy		



Default values

Design: scenario

Zipf skew	α	0.8, **1**, 1.2
System scale	$\|C\|/\|O\|$	$10^2/10^4$, **$10^3/10^5$**, $10^4/10^6$, $10^5/10^7$, $10^6/10^8$
Cache to catalog ratio	$\|C\|/\|O\|$	**$10^3/10^5$**, $10^3/10^6$, $10^3/10^7$, $10^3/10^8$
Price ratio	π	1, 2, 5, **10**, 100
Catalog split	$\bar{s} = (s_1, s_2, s_3)$	**$(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$**, $(0, \frac{1}{4}, \frac{3}{4})$, $(0, \frac{1}{2}, \frac{1}{2})$, $(0, \frac{3}{4}, \frac{1}{4})$, $(\frac{1}{4}, 0, \frac{3}{4})$, $(\frac{1}{4}, \frac{1}{4}, \frac{1}{2})$, $(\frac{1}{4}, \frac{1}{2}, \frac{1}{4})$, $(\frac{1}{4}, 0, \frac{3}{4})$, $(\frac{1}{2}, \frac{1}{4}, \frac{1}{4})$, $(\frac{1}{2}, 0, \frac{1}{2})$, $(\frac{3}{4}, 0, \frac{1}{4})$, $(\frac{3}{4}, \frac{1}{4}, 0)$
Replacement policy		LRU

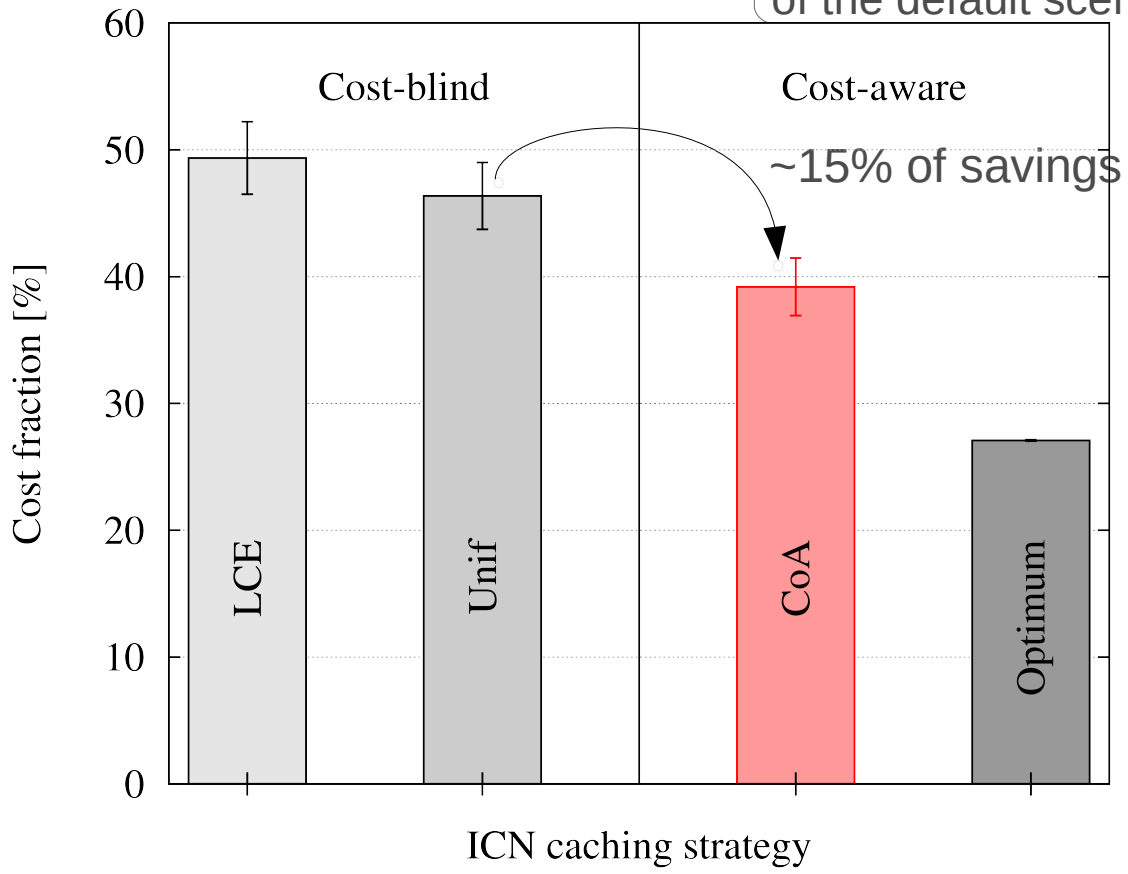
Default values

Evaluation : reduction at a glance

- Caching (even if naïve) permits savings

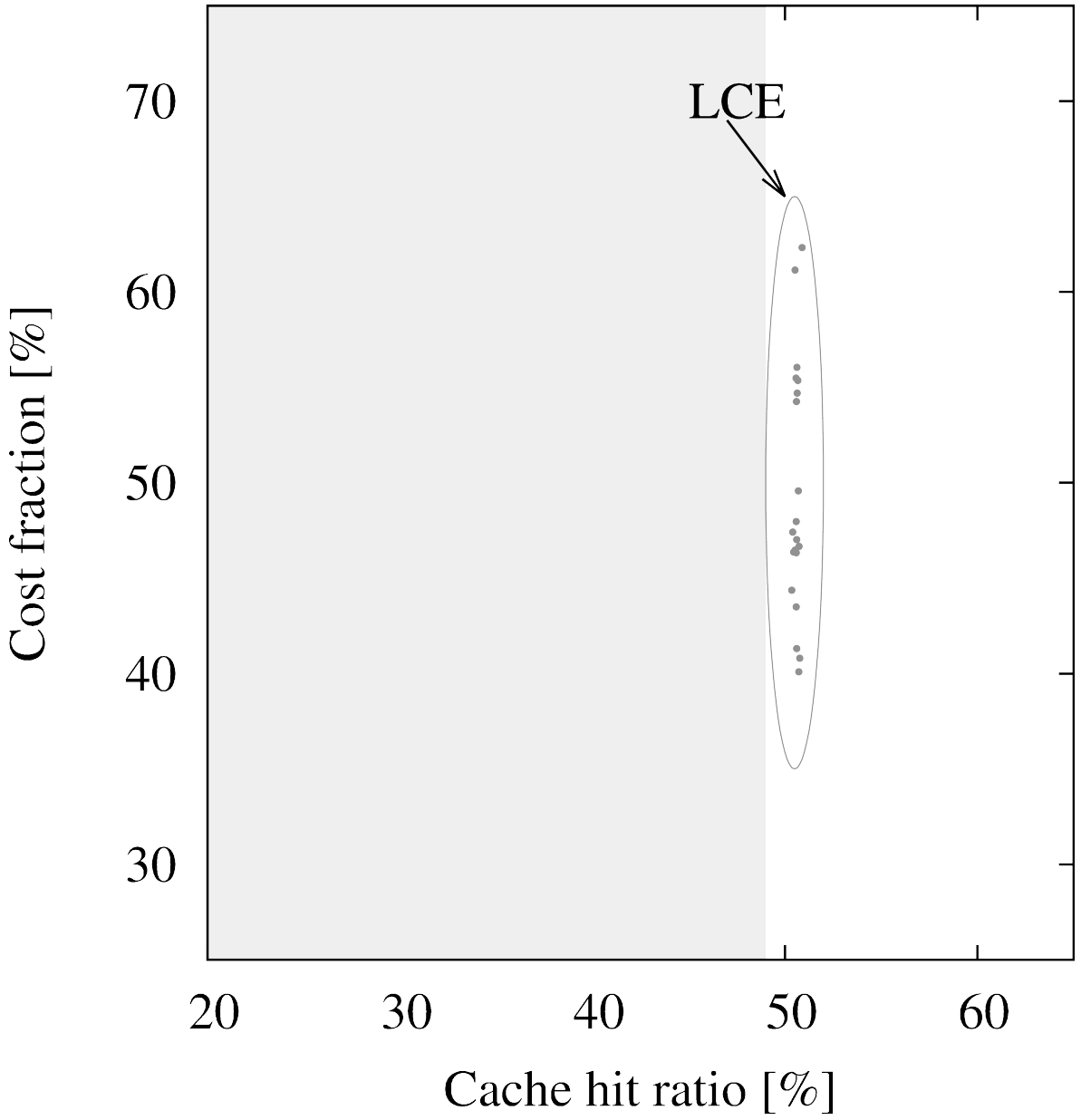
- If an ISP is using `Unif` to max hit, it could save ~15% of its current cost by switching to `CoA`.
- It's an operational expenditure saving
  ➡ big number

Results over 20 runs of the default scenario

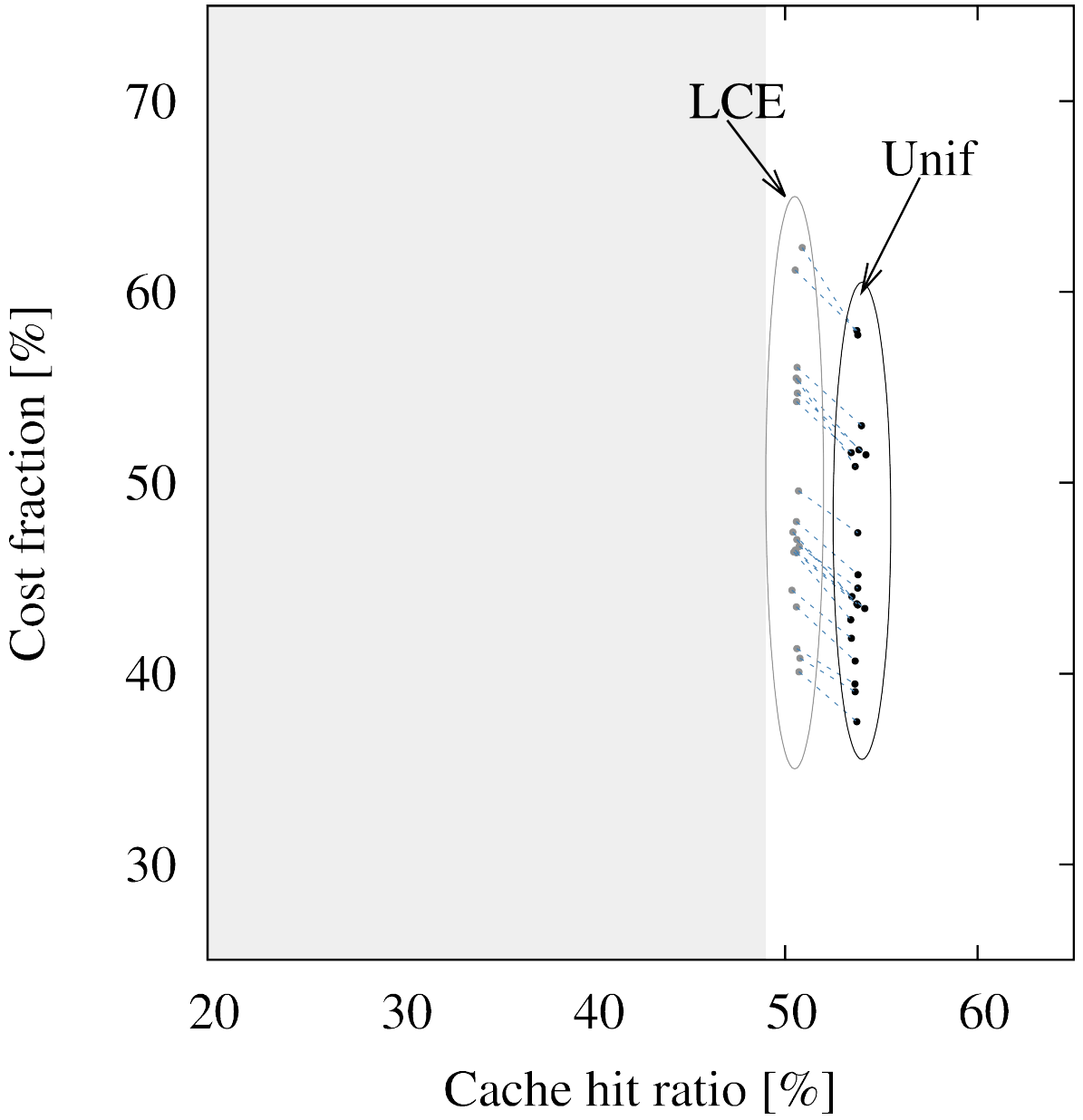


$$\text{cost fraction(S)} = \frac{\text{cost with strategy S}}{\text{cost without caching}}$$

Evaluation : cost and hit trade off

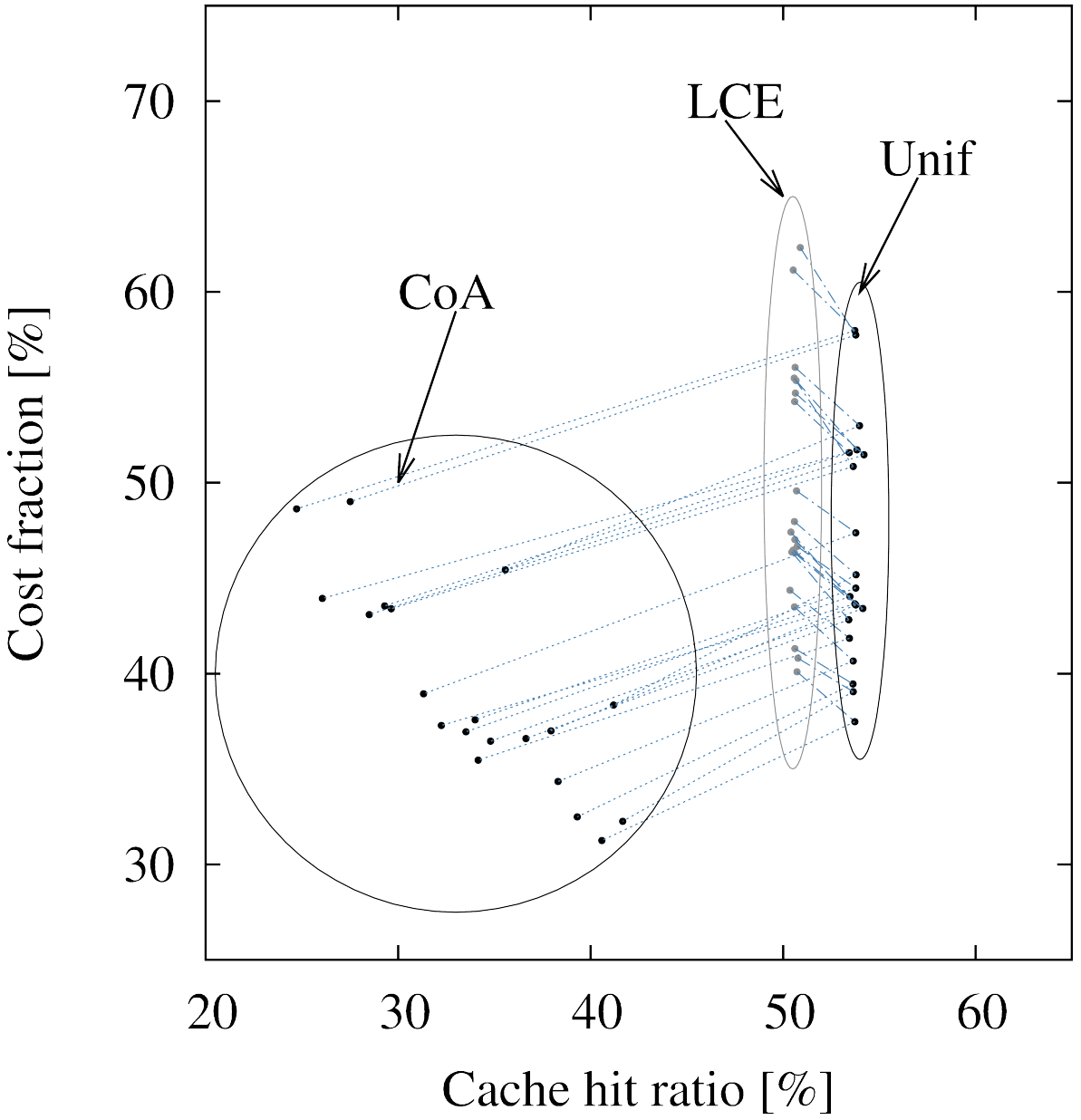


Evaluation : cost and hit trade off

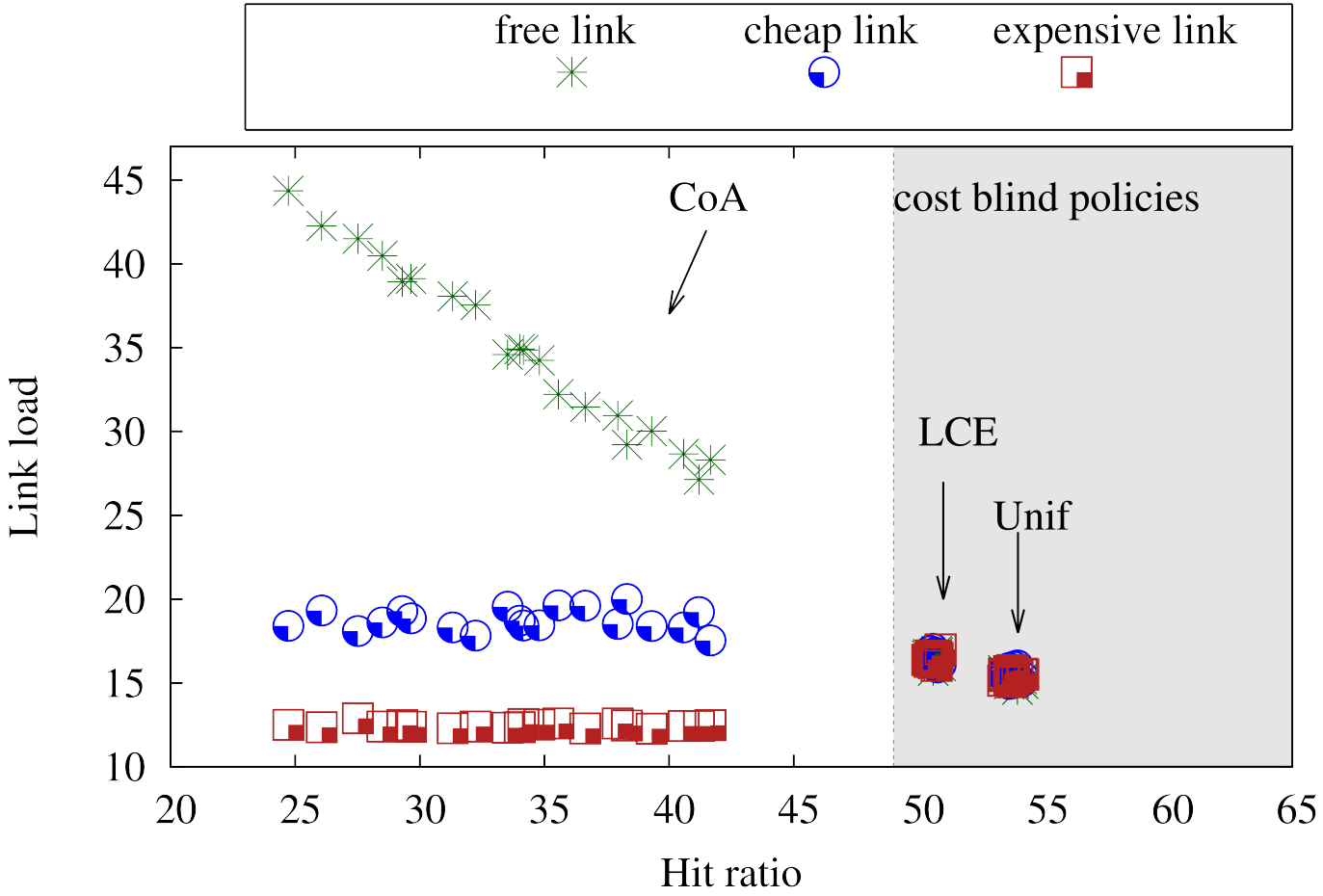


Evaluation : cost and hit trade off

- Cost reduction and hit maximization may be **conflicting goals**
- If you try to maximize hit ratio, you may loose potential cost savings



Evaluation : cause of reduction



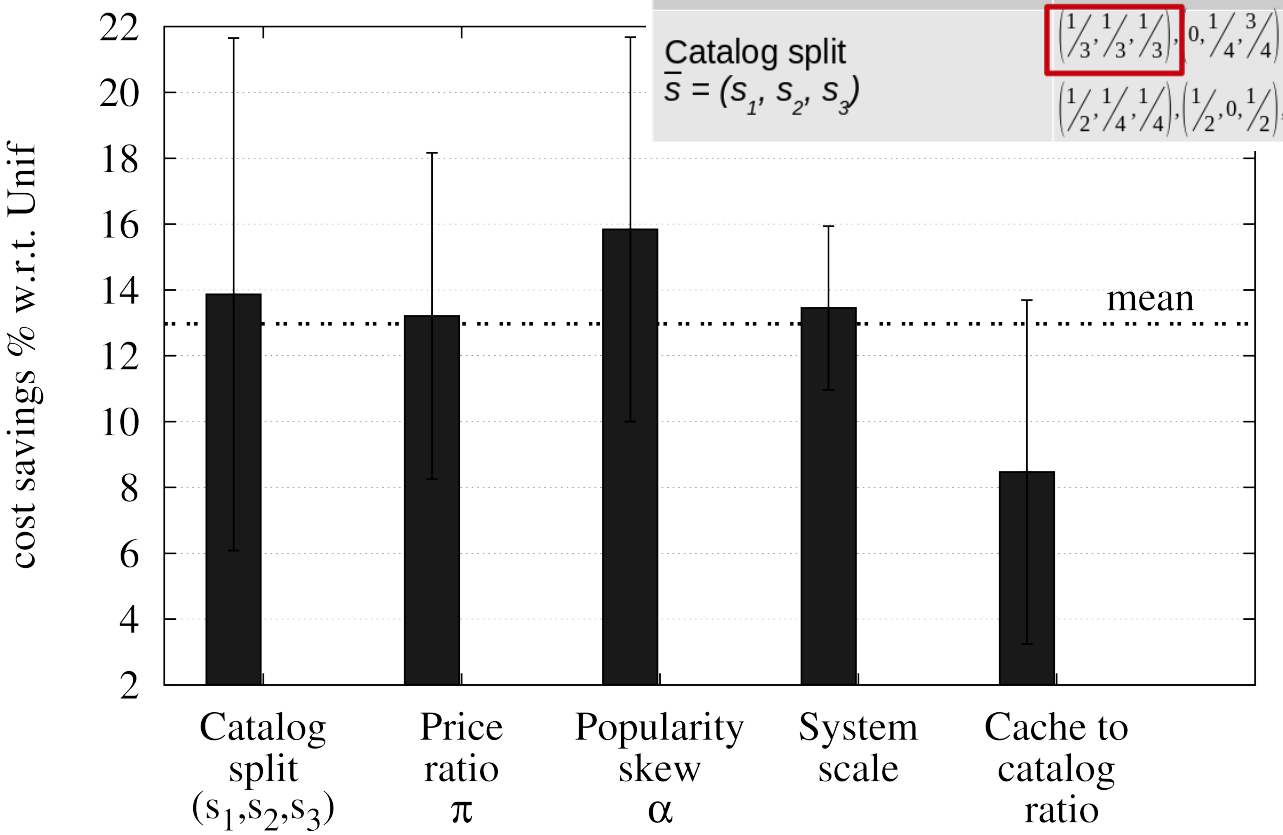
$$load(l) = \frac{\text{downloads from } l}{\text{requests}}$$

- Cost unaware policies distribute load blindly across all links
- On the contrary, CoA effectively **minimizes the load on paid link**
- The paid link load is constant, even when hit ratio decreases

CoA effectively **distributes the additional miss-stream to the free link**

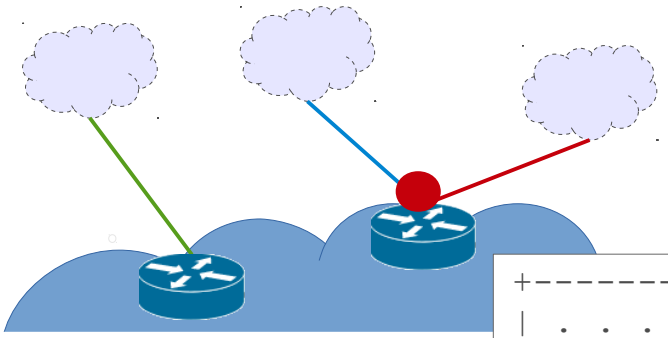
Evaluation : robustness

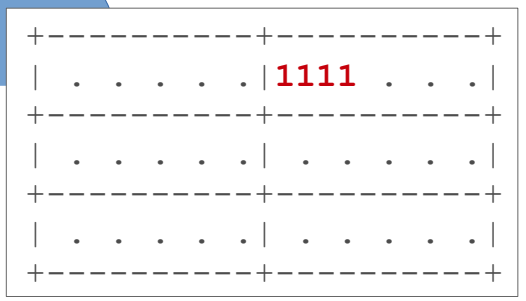
Zipf skew α	0.8, **1**, 1.2
System scale $\|C\|/\|O\|$	$10^2/10^4$, **$10^3/10^5$** $10^4/10^6$, $10^5/10^7$, $10^6/10^8$
Cache to catalog ratio $\|C\|/\|O\|$	**$10^3/10^5$** $10^3/10^6$, $10^3/10^7$, $10^3/10^8$
Price ratio π	1, 2, 5, **10**, 100
Catalog split $\bar{s} = (s_1, s_2, s_3)$	**$(\frac{1}{3},\frac{1}{3},\frac{1}{3})$**, $(0,\frac{1}{4},\frac{3}{4})$, $(0,\frac{1}{2},\frac{1}{2})$, $(0,\frac{3}{4},\frac{1}{4})$, $(\frac{1}{4},0,\frac{3}{4})$, $(\frac{1}{4},\frac{1}{4},\frac{1}{2})$, $(\frac{1}{4},\frac{1}{2},\frac{1}{4})$, $(\frac{1}{4},0,\frac{3}{4})$, $(\frac{1}{2},\frac{1}{4},\frac{1}{4})$, $(\frac{1}{2},0,\frac{1}{2})$, $(\frac{3}{4},0,\frac{1}{4})$, $(\frac{3}{4},\frac{1}{4},0)$

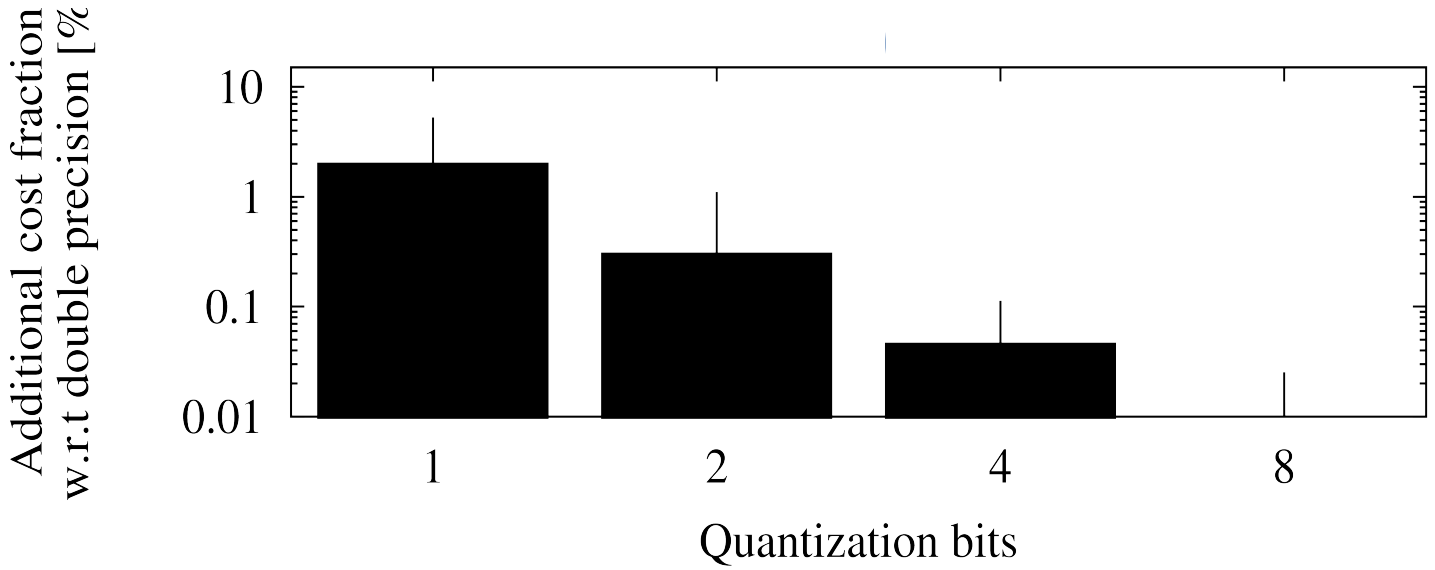


500 simulation runs, accounting for over $8 \cdot 10^9$ requests

Evaluation : implementation constraints







- 2 - 4 bits are sufficient to indicate cost

Conclusion

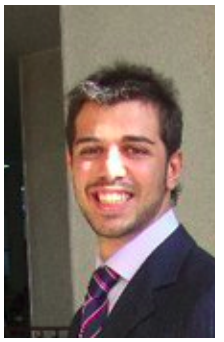
- Hit vs cost trade-off
- New caching schema (CoA) to reduce ISP cost
  - Cache more the expensive objects
  - Easily implementable in an ICN router
- Consistent savings under different scenarios

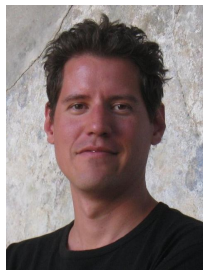
References

- [CoNEXT09] Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H., & Braynard, R. L. “Networking named content”, in *ACM CoNEXT*, 2009
- [Globecom14] A. Araldo, M. Mangili, F. Martignon, D. Rossi, “Cost-aware caching: optimizing cache provisioning and object placement in ICN”, in *IEEE Globecom*, 2014
- [ICCIT13] Barakat, C., Kalla, A., Saucez, D., & Turletti, T. “Minimizing Bandwidth on Peering Links with Deflection in Named Data Networking”, in *ICCIT*, 2013.
- [Usenix97] Cao, P., & Irani, S. “Cost-Aware WWW Proxy Caching Algorithms”, in *Usenix symposium on internet technologies and systems*, 1997.
- [ComMag12] P. Agyapong and M. Sirbu. “Economic incentives in information-centric networking: Implications for protocol design and public policy”. In IEEE Communications Magazine, vol. 50(12), pages 18–26, Dec. 2012.
- [INFOCOM13]V. Pacifici and G. Dan. “Content-peering dynamics of autonomous caches in a content-centric network”. In IEEE INFOCOM, 2013.
- [SocialCom13] F. Kocak, G. Kesidis, T.-M. Pham, and S. Fdida. “The effect of caching on a model of content and access provider revenues in information-centric networks”. In IEEE SocialCom, 2013.
- [IFIP13] T.-M. Pham, S. Fdida, and P. Antoniadis. “Pricing in Information-Centric Network Interconnection. In IFIP Networking”, 2013.
- [ComCom13] G. Rossini and D. Rossi, “Evaluating CCN multi-path interest forwarding strategies,” Computer Communications, Apr. 2013.
- [Elsevier12] W. K. Chai, D. He, I. Psaras, and G. Pavlou, “Cache Less for More in Information-centric Networks,” Elsevier Computer Communications, 2012.
- [ICNP13] Y. Wang, Z. Li, G. Tyson, S. Uhlig, and G. Xie, “Optimal Cache Allocation for Content-Centric Networking,” ICNP, 2013.
- [Globecom13] M. Mangili, F. Martignon, and A. Capone, “A Comparative Study of Content-Centric and Content-Distribution Networks : Performance and Bounds,” in IEEE Globecom, 2013.
- [CNSM13] D. Tuncer, M. Charalambides, R. Landa, and G. Pavlou, “More control over network resources: An ISP caching perspective,” in CNSM, 2013.

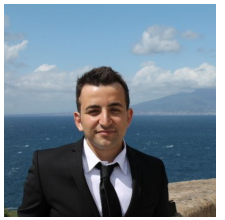
ccnSim

- Open source simulation tool
- written in C++ under the Omnet++ framework
- modular
- Performance assessment in large scale scenarios
- V0.3 release:
  http://www.enst.fr/~drossi/ccnSim
- Help us improve it: send emails with suggestions, problems, …
- Scenarios and howto to reproduce these results


G Rossini


D.Rossi


A Araldo


R Chiocchetti


M.Tortelli

Thanks

perso.telecom-paristech.fr/~araldo