Content search and availability estimation in mobile opportunistic networks

Suzan Bayhan (University of Helsinki), Esa Hyytiä (Aalto Univ), Jussi Kangasharju (UH), Jörg Ott (Aalto Univ, TU Munich)

University of Helsinki (UH), Dept. of Computer Science

Collaborative Networking Research Group (CoNe)

http://www.hiit.fi/u/bayhan

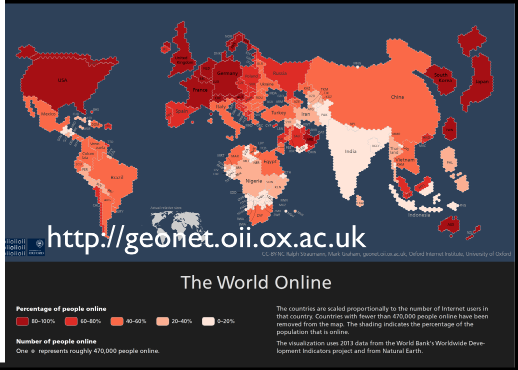
November 24, 2015, Cambridge University NetOS Group Seminars

Aalto University

How would one retrieve some content stored in a remote mobile device if there is no Google-like service and nodes are moving around in the network?

Mobile opportunistic networks

- Network of mobile devices with wireless communication interface
- Intermittent connections, but mobile nodes, *store-carry-forward*



Useful information often found locally, *homophily, spatial locality*

No or unreliable infrastructure, +50% forecasted global population will remain offline in 2017

Tight control on content and users (e.g., censorship, tracking)

Per-bit billing vs. almost-free network capacity

Challenges and solutions

Challenges

- Sporadic contacts (delay-tolerant applications)
- Time-varying network topology
- Lack of precise knowledge
- Energy-limited devices

Solutions:

- Introduce redundancy, i.e., multi-copy multi-hop routing protocols
- Exploit predictability of human contacts (scheduled lives!)

How to find content without Google?

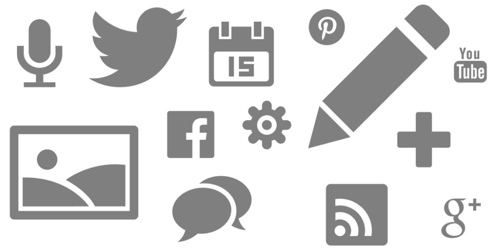
- Ask every one
  - Epidemic (flooding)
- Ask the nodes in the same/similar community
  - DelQueue: geo-community [Fan 2011], Seeker-assisted search [Bayhan2013]
- Ask some nodes based on some criteria
  - Announced experience [Liu 2014], Random walk
- Wait till meeting one of the content providers
  - Direct delivery [Sermpezis2014]
- Do not ask, wait for somebody to deliver!
  - Push based approach (pub/sub)
- Design your own Google!
  - Hash-based mapping of content [Talipov 2013]

Three components of opportunistic search

User

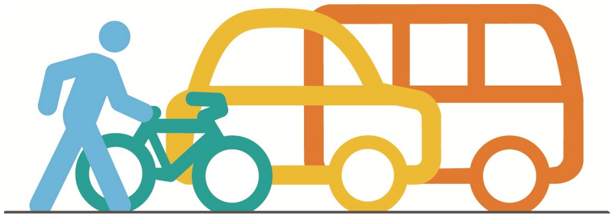
Limited tolerance to waiting

Limited energy



Content

Scarce or abundant item



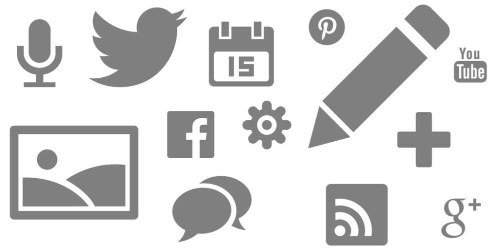
Network mobility

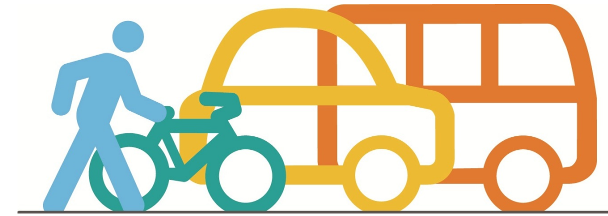
Many contacts?

Many diverse contacts?

Images from: http://www.onewebcms.com/en/home/default.aspx and http://www.dotherightmix.eu/

Three components of opportunistic search





User

Limited tolerance to waiting

Limited energy

↓

Message lifetime (TTL)

Message hop-limit to bound the cost

Content

Scarce or abundant item

↓

Content availability α

Network mobility

Many contacts?

Many diverse contacts?

↓

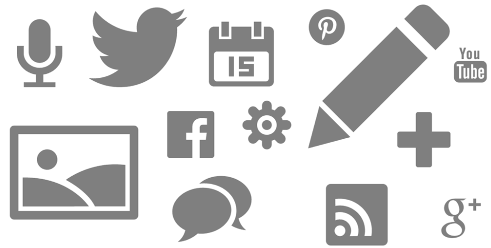
Real traces

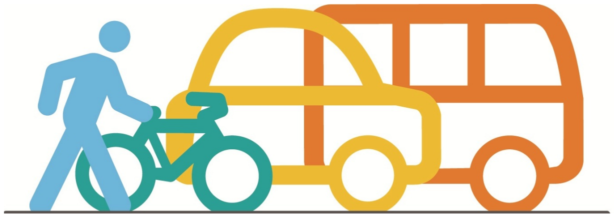
Neighborhood growth

Temporal distance to content

Images from: http://www.onewebcms.com/en/home/default.aspx and http://www.dotherightmix.eu/

Three components of opportunistic search





User
Limited tolerance to waiting
Limited energy

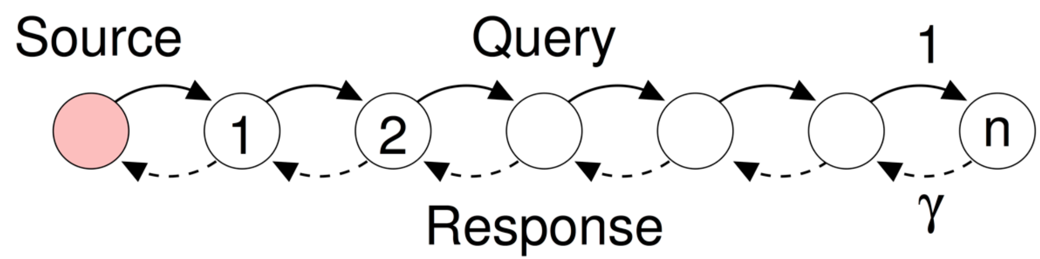
Content
Scarce or abundant item

Network mobility
Many contacts?
Many diverse contacts?

> How these components affect the (optimal) search strategy and performance (success, delay, cost)?

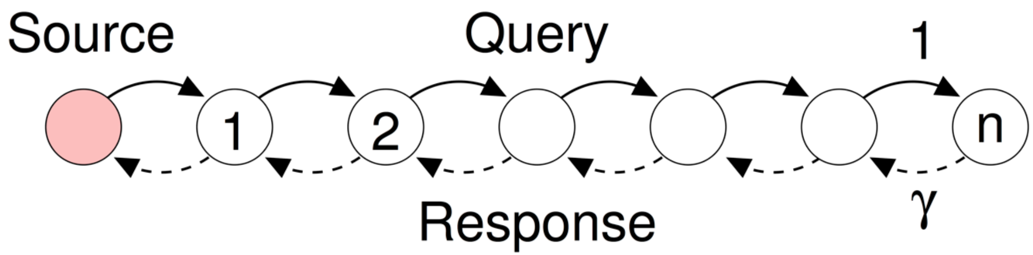
Images from: http://www.onewebcms.com/en/home/default.aspx and http://www.dotherightmix.eu/

Optimal search depth in a linear network



- Query: Travels to right and a possible response to left (content discovery, forward path)
- Response: every link remains available with probability γ (content delivery, return path)
- Transmission cost for each link: *e*
- Each discovered content has some value *v*
  - Bernoulli case: a node either has the content, or not with single availability parameter *p*

Optimal search depth in a linear network



- Static: Searching node determines the search depth (number of hops). Nodes route the query and response, if any
- Dynamic: Each relaying node decides to stop the search or route to the next node based on the content availability and the cost

Optimal depth: utility maximization problem

- \# of nodes queried
- content availability for dynamic schemes

Utility = Expected value of content – (expected cost of forward path
+expected cost of return path)

Depends on
- content availability distribution
- \# of nodes queried

- \# of nodes queried
- Reliability of the return links

Optimal depth: utility maximization problem

- # of nodes queried
- content availability for dynamic schemes

Utility = Expected value of content – (expected cost of forward path
+expected cost of return path)

Depends on
- content availability distribution
- # of nodes queried

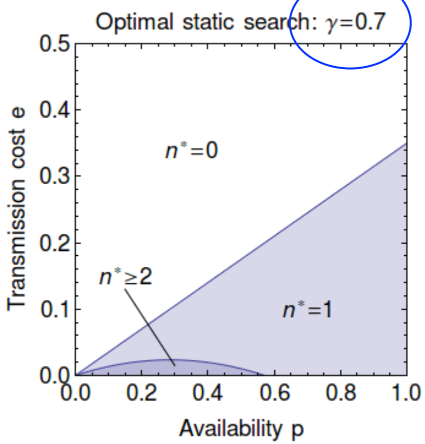
- # of nodes queried
- Reliability of the return links

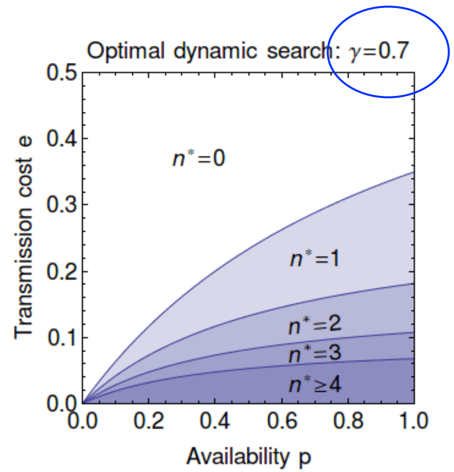
Stop search when the next node does not bring any improvement in utility

$$U_{n+1} - U_n \leq 0$$

Static scheme

Dynamic



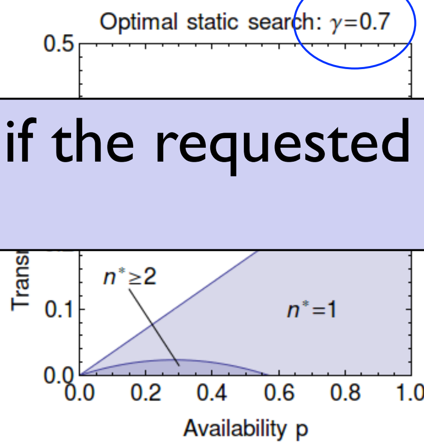


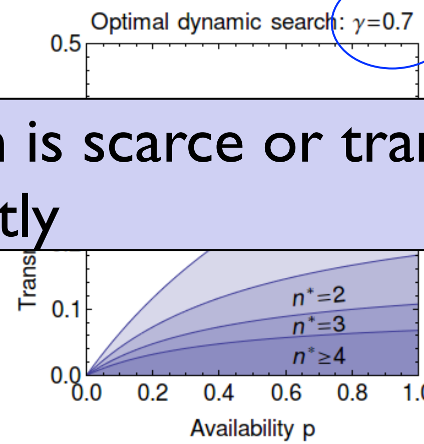
conservative most of the time

higher hop counts thanks to the capability of stopping the search

Static scheme

Dynamic





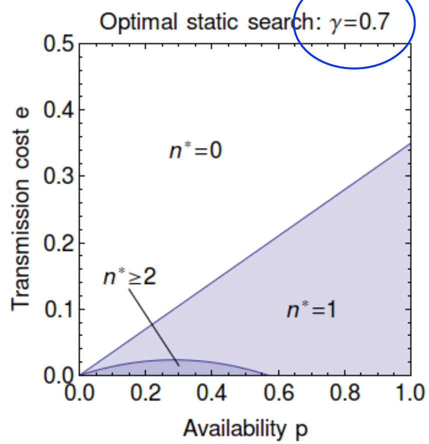
higher hop counts

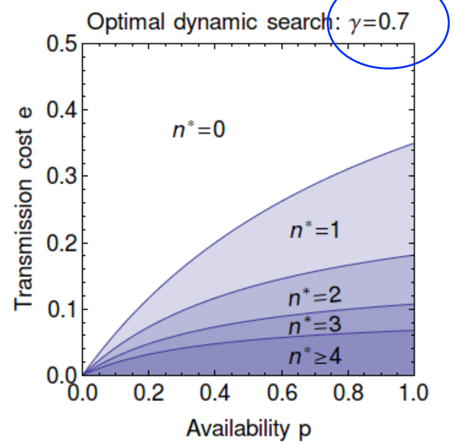
Don’t search if the requested item is scarce or transmission is very costly

Static scheme

Dynamic

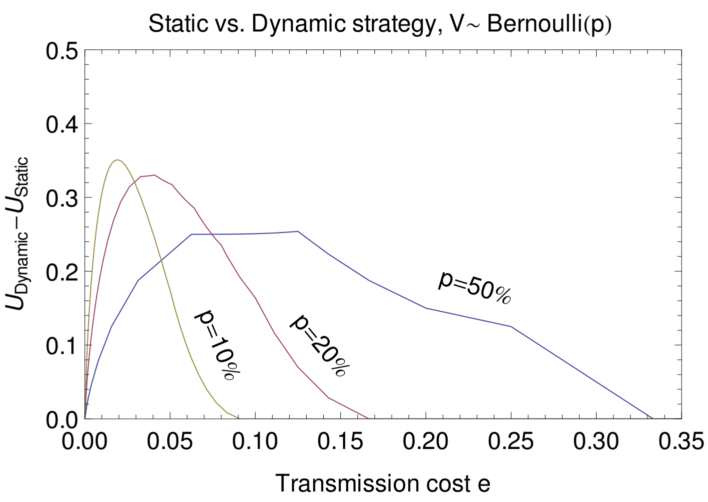
conservative most of the time



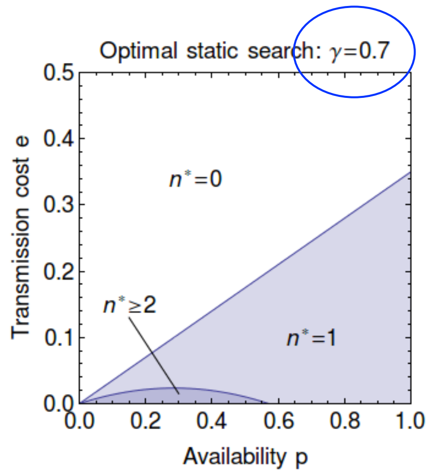


higher hop counts thanks to the capability of stopping the search

Difference in Utility: Dynamic-Static

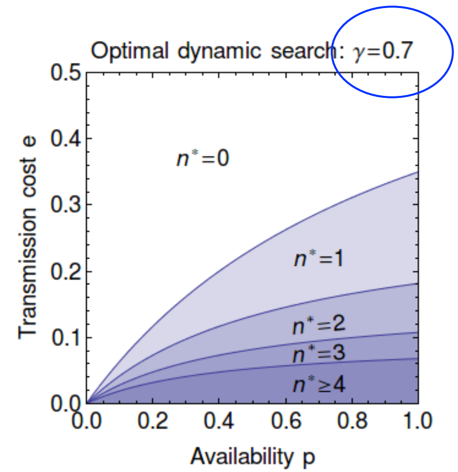


Static scheme



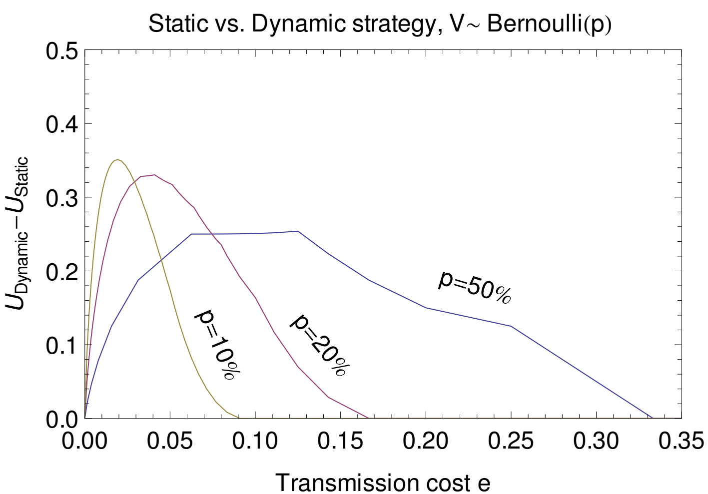
conservative most of the time

Dynamic



higher hop counts thanks to the capability of stopping the search

Difference in Utility: Dynamic-Static



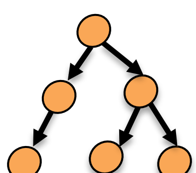
Improvement by dynamic strategy is higher for low availability

Nodes should be able to stop or forward the search messages!

Content availability and cost determines the optimal search depth

Content availability and cost determines the optimal search depth

How about the user and more realistic settings (general topologies)?



Search on more realistic topologies

- Assume uniform mobility characteristics, uniform content distribution
- User's tolerance to waiting: T for each step of the search
- # of nodes message reaches under T and hop limitation h is $M = N_h(T)$

Search on more realistic topologies

- Assume uniform mobility characteristics, uniform content distribution
- User's tolerance to waiting: T for each step of the search
- # of nodes message reaches under T and hop limitation h is M = Nh(T)

Search success for content with availability α and M replicas maximum:

$$P_s = \sum_{m=1}^{M} Pr\{\text{m content providers are discovered}\}$$
$$\times Pr\{\text{at least one of m responses reaches } n_s\}$$

Search on more realistic topologies

- Assume uniform mobility characteristics, uniform content distribution
- User's tolerance to waiting: T for each step of the search
- # of nodes message reaches under T and hop limitation h is M = Nh(T)

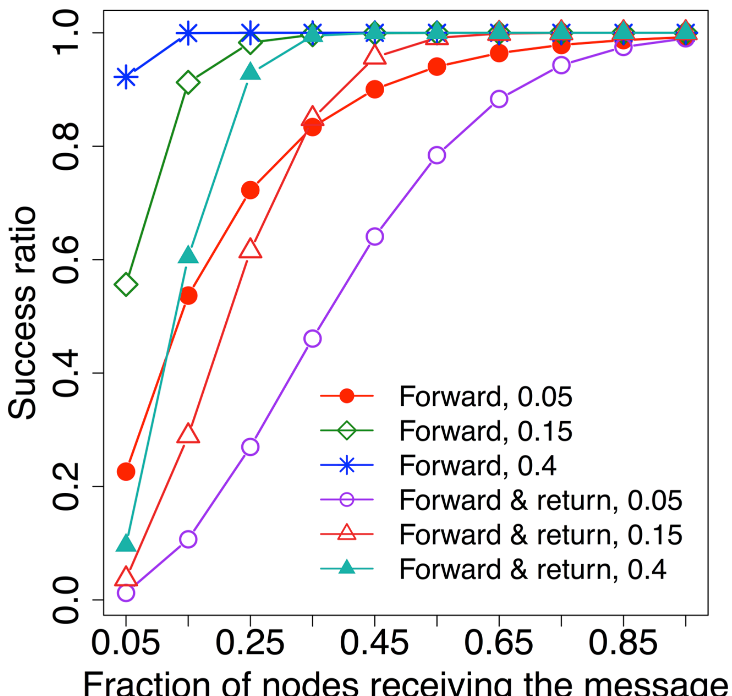
Search success for content with availability α and M replicas maximum:

$$P_s = \sum_{m=1}^{M} Pr\{\text{m content providers are discovered}\} \times Pr\{\text{at least one of m responses reaches } n_s\}$$

$$P_s = 1 - (1 - \alpha\gamma)^M \quad \text{where} \quad \gamma = \frac{M}{N-1}$$

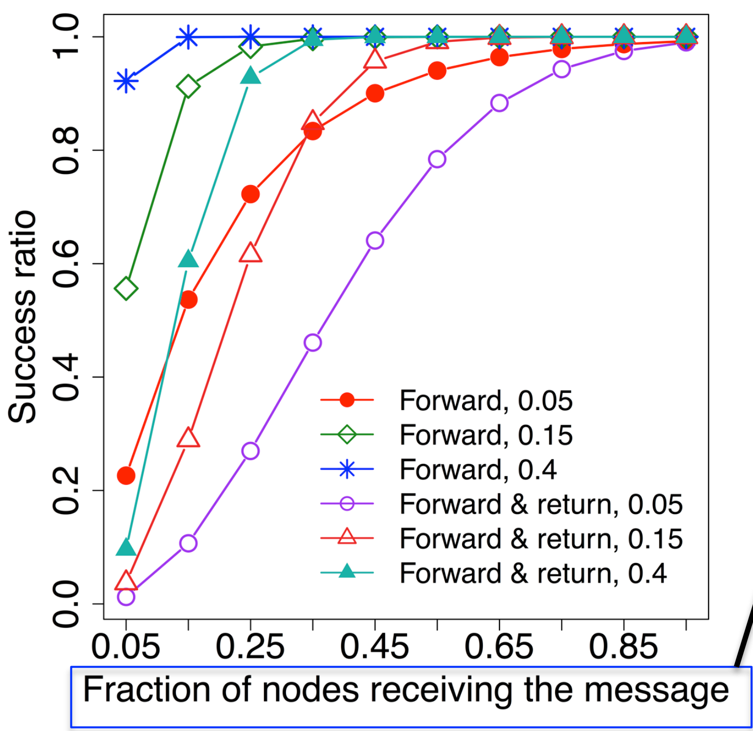
Neighborhood growth

$$P_s = 1 - (1 - \alpha\gamma)^M$$



Neighborhood growth

$$P_s = 1 - (1 - \alpha\gamma)^M$$



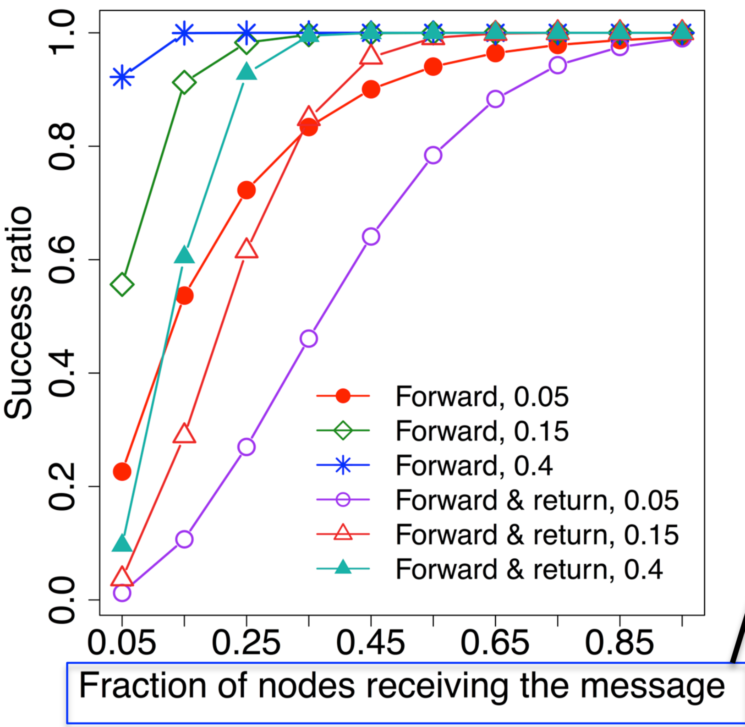
Time limited h-hop neighborhood

f(Mobility, hop limit, time limit)

For static networks, less challenging to model, e.g., [Wang ICN 2015]

Neighborhood growth

$$P_s = 1 - (1 - \alpha\gamma)^M$$

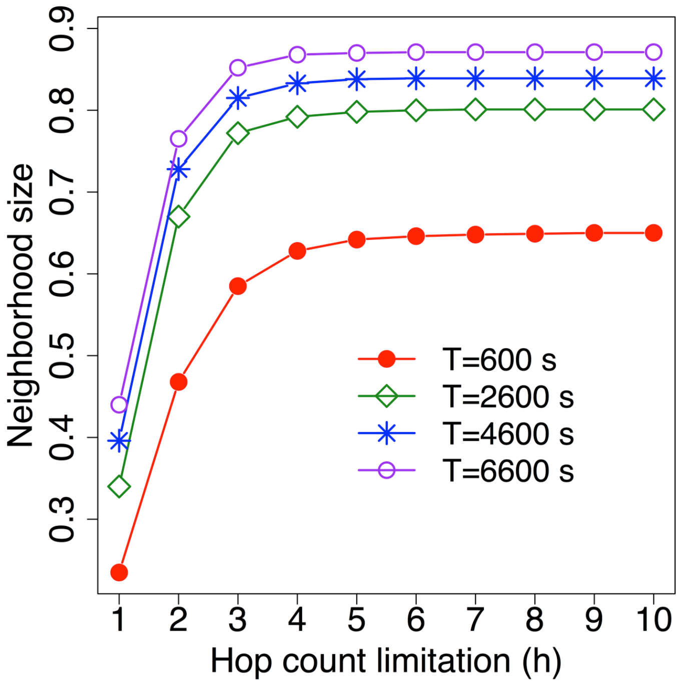


Time limited h-hop neighborhood
f(Mobility, hop limit, time limit)

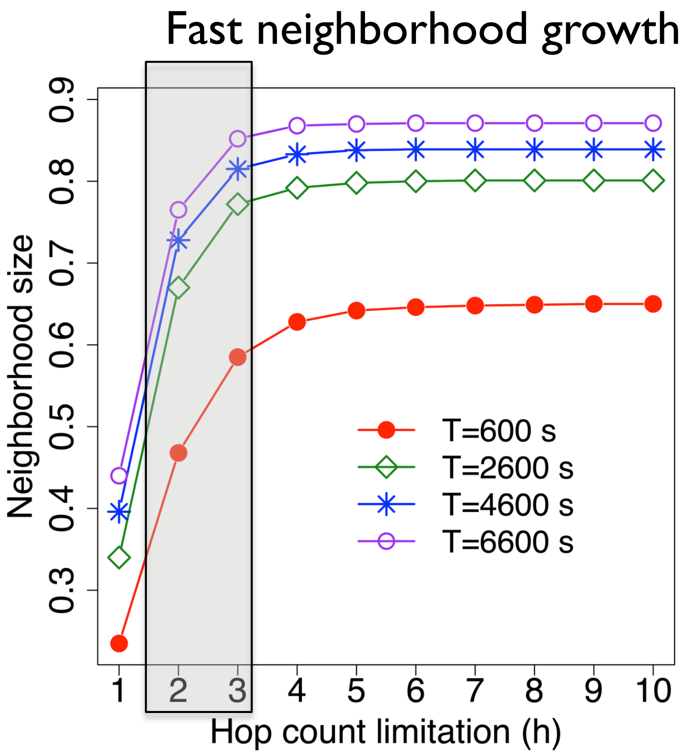
For static networks, less challenging to model, e.g., [Wang ICN 2015]

Our approach: derive from real mobility traces to understand how neighborhood grows

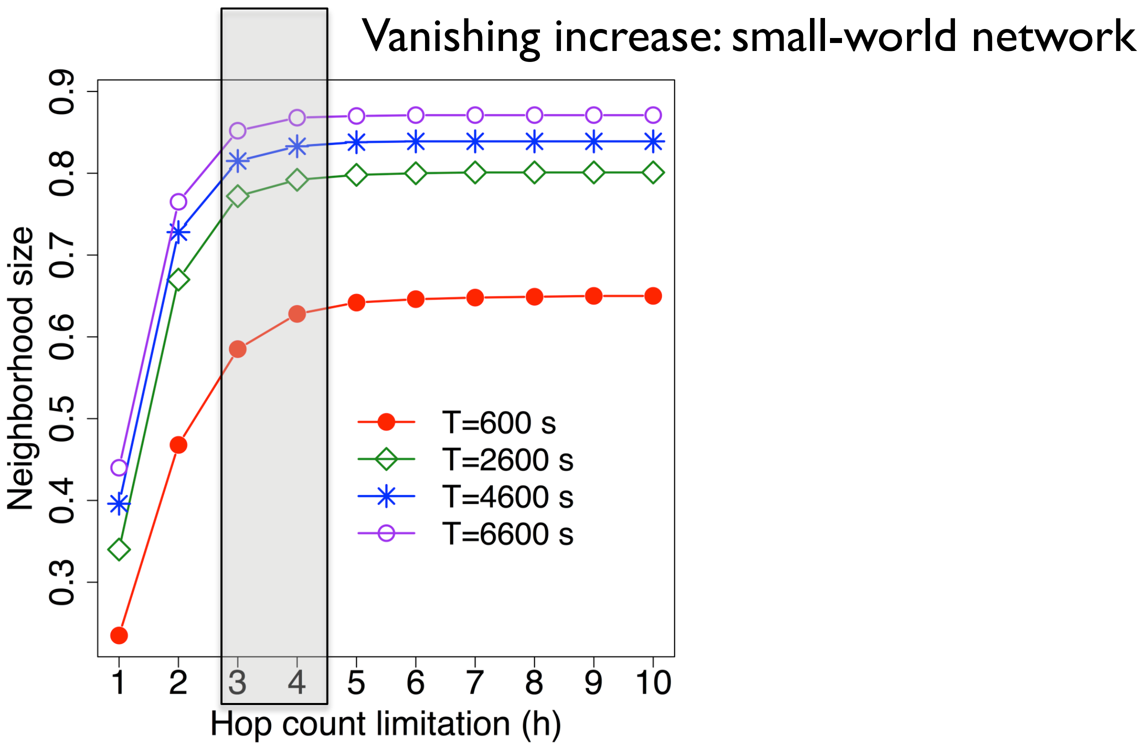
Analysis of Infocom06 trace, 98 nodes



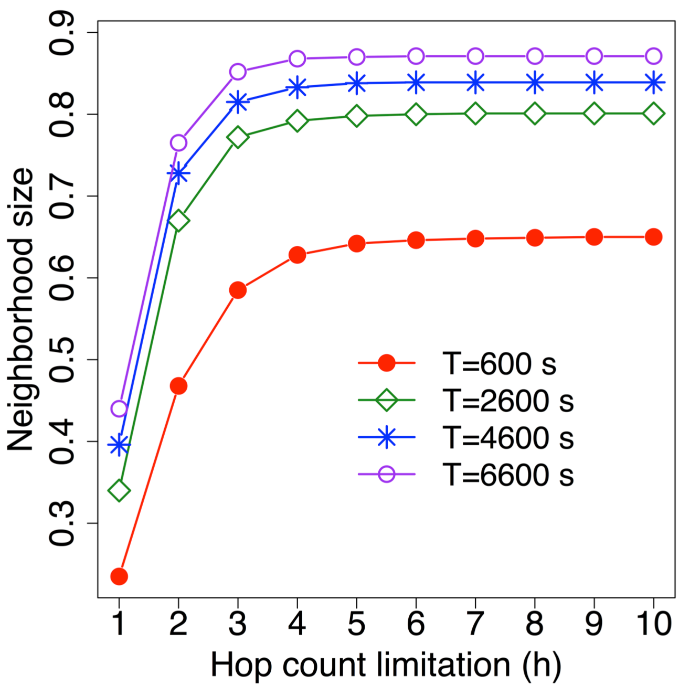
Analysis of Infocom06 trace, 98 nodes



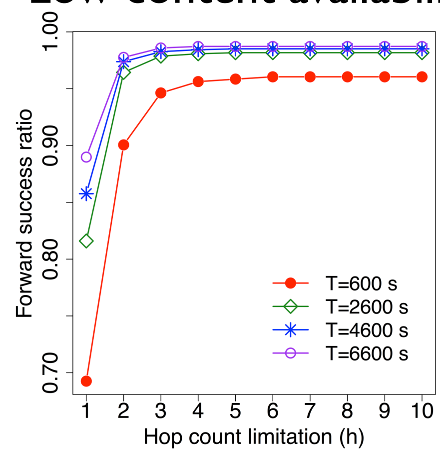
Analysis of Infocom06 trace, 98 nodes



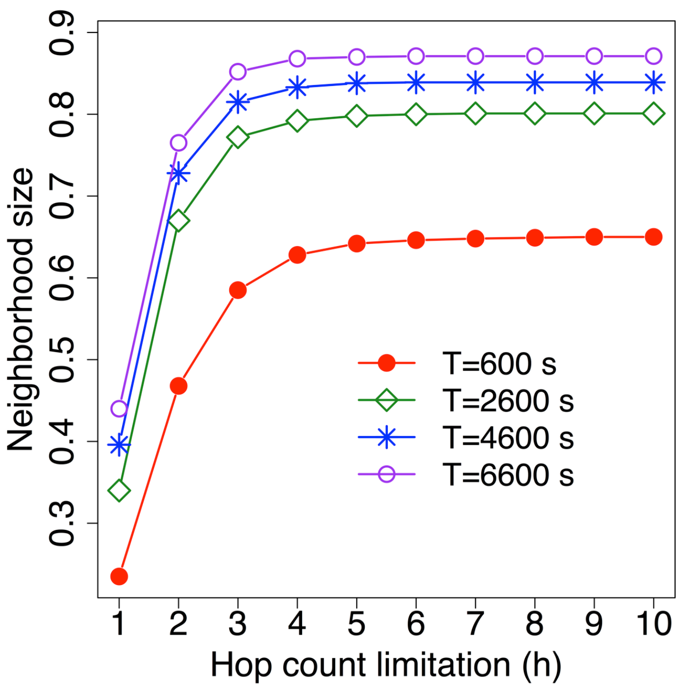
Analysis of Infocom06 trace, 98 nodes



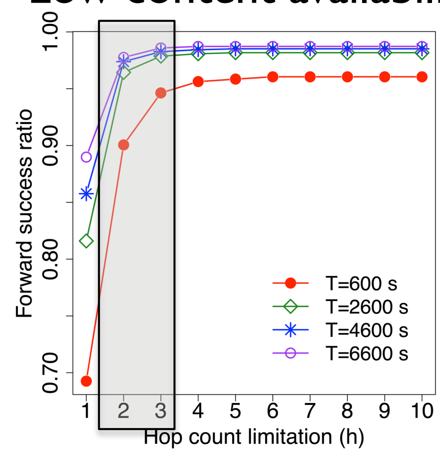
Low content availability, %5 availability



Analysis of Infocom06 trace, 98 nodes

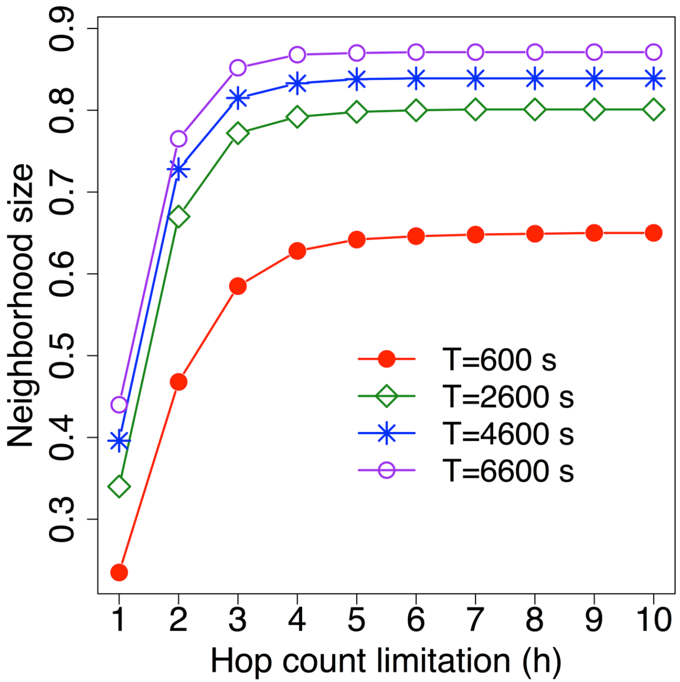


Low content availability, %5 availability

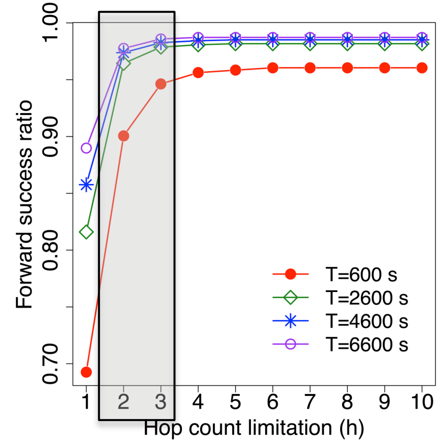


Second hop brings the highest improvement

Analysis of Infocom06 trace, 98 nodes

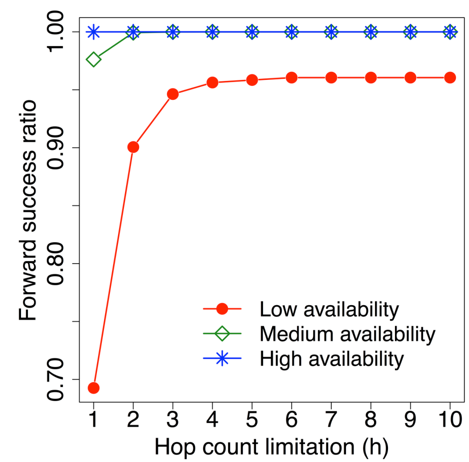


Low content availability, %5 availability

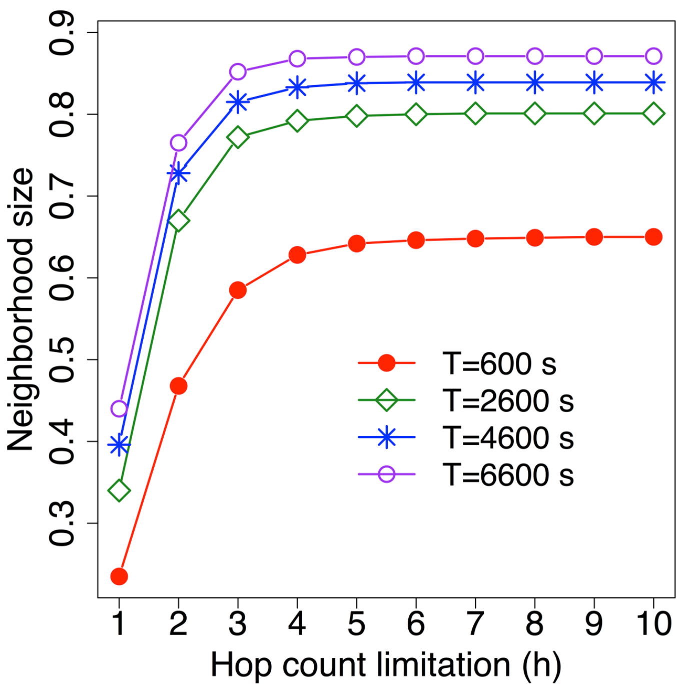


Second hop brings the highest improvement

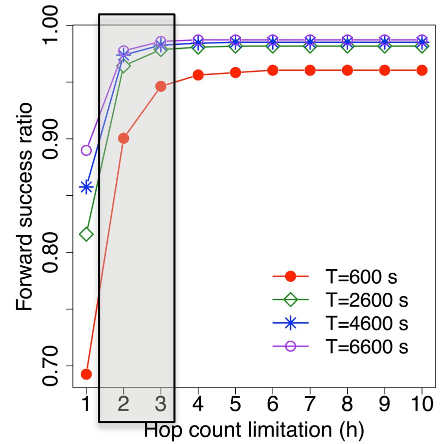
Short T, 10 mins



Analysis of Infocom06 trace, 98 nodes

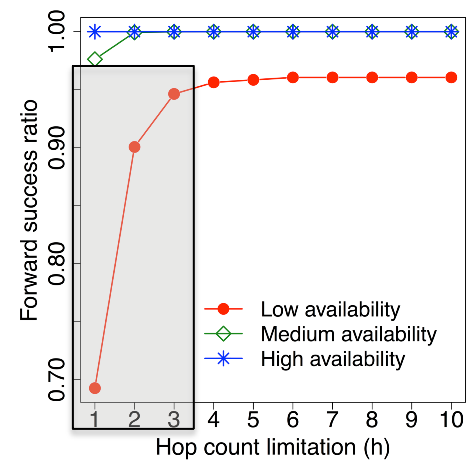


Low content availability, %5 availability



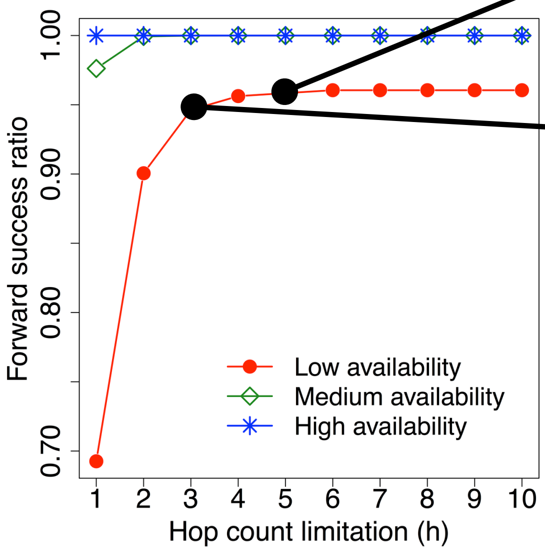
Second hop brings the highest improvement

Short T, 10 mins



Search for scarce item benefits significantly from multi-hop search

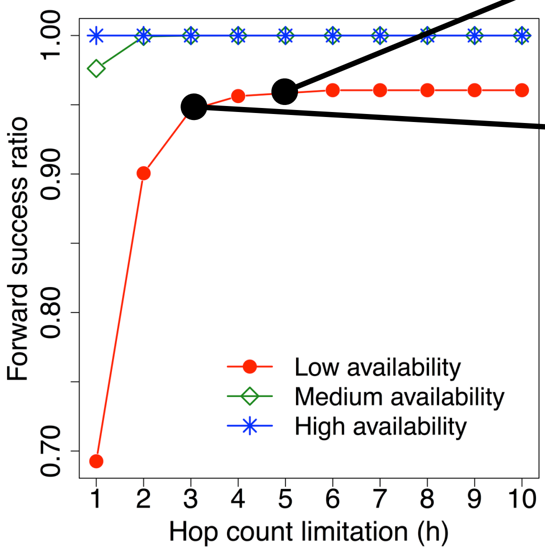
Optimal hop vs. fair-benefit hop



Optimal hop: hop count providing highest possible performance

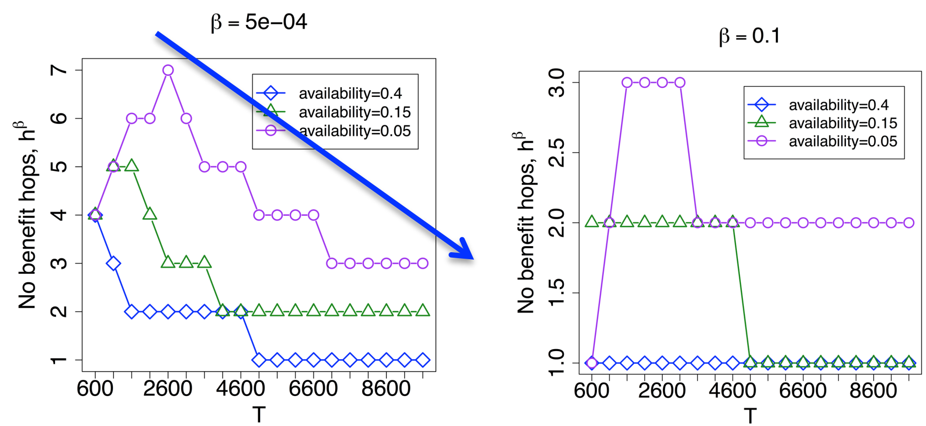
Fair-benefit hop: hop count beyond which performance improvement is "insignificant" (e.g., user experience does not change significantly)

Optimal hop vs. fair-benefit hop



Optimal hop: hop count providing highest possible performance

Fair-benefit hop: hop count beyond which performance improvement is "insignificant" (e.g., user experience does not change significantly)



Lower hop count for increasing T, increasing availability → shrinking network diameter

Effective temporal distance to content

- Networks have different mobility characteristics resulting in different effective temporal/ hop distance to content

Effective temporal distance to content

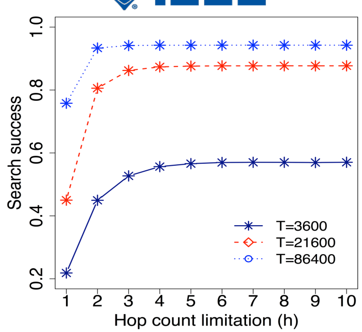
- Networks have different mobility characteristics resulting in different effective temporal/ hop distance to content
- Effective temporal/hop distance to content and from the content: Maximum distance, be it hops or time, which ensures that 90% of the paths between a searching node and a content provider is lower than this distance

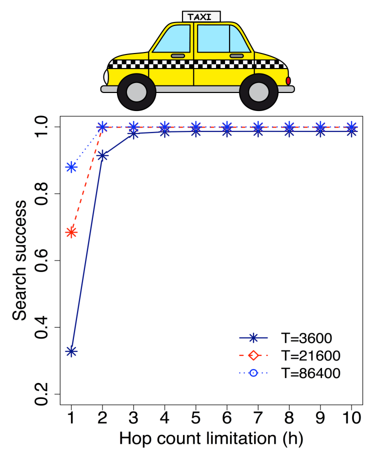
Effective temporal distance to content

- Networks have different mobility characteristics resulting in different effective temporal/hop distance to content
- Effective temporal/hop distance to content and from the content: Maximum distance, be it hops or time, which ensures that 90% of the paths between a searching node and a content provider is lower than this distance
- ONE simulations of hop-limited search
- Infocom06 (98 nodes) and Cabspotting (460 cars)

Effective temporal distance to content

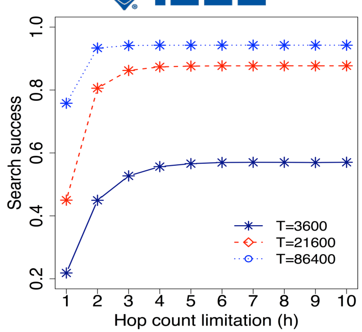
- Networks have different mobility characteristics resulting in different effective temporal/ hop distance to content
- Effective temporal/hop distance to content and from the content: Maximum distance, be it hops or time, which ensures that 90% of the paths between a searching node and a content provider is lower than this distance
- ONE simulations of hop-limited search
- Infocom06 (98 nodes) and Cabspotting (460 cars)

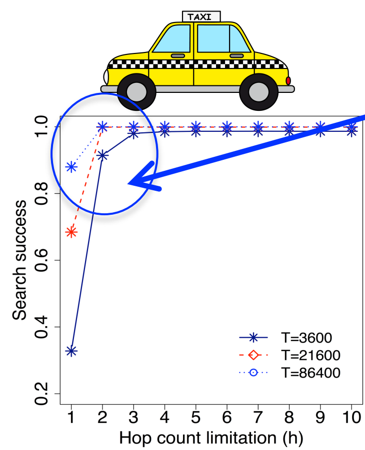




Effective temporal distance to content

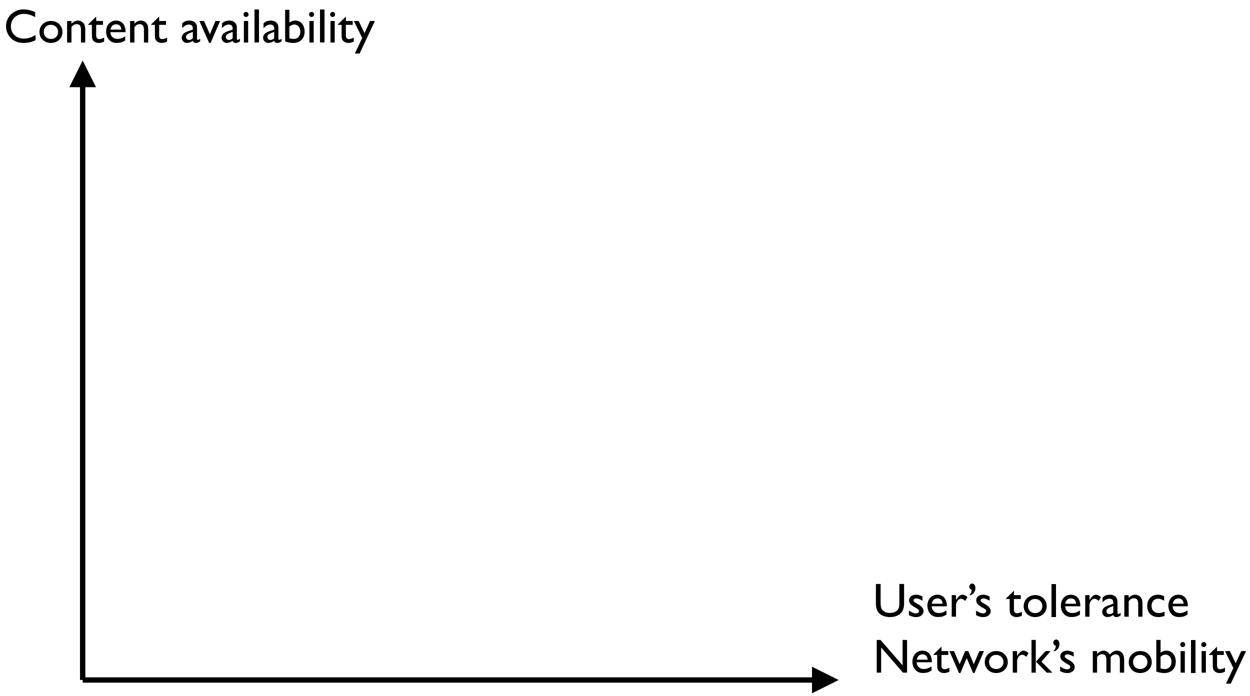
- Networks have different mobility characteristics resulting in different effective temporal/ hop distance to content
- Effective temporal/hop distance to content and from the content: Maximum distance, be it hops or time, which ensures that 90% of the paths between a searching node and a content provider is lower than this distance
- ONE simulations of hop-limited search
- Infocom06 (98 nodes) and Cabspotting (460 cars)



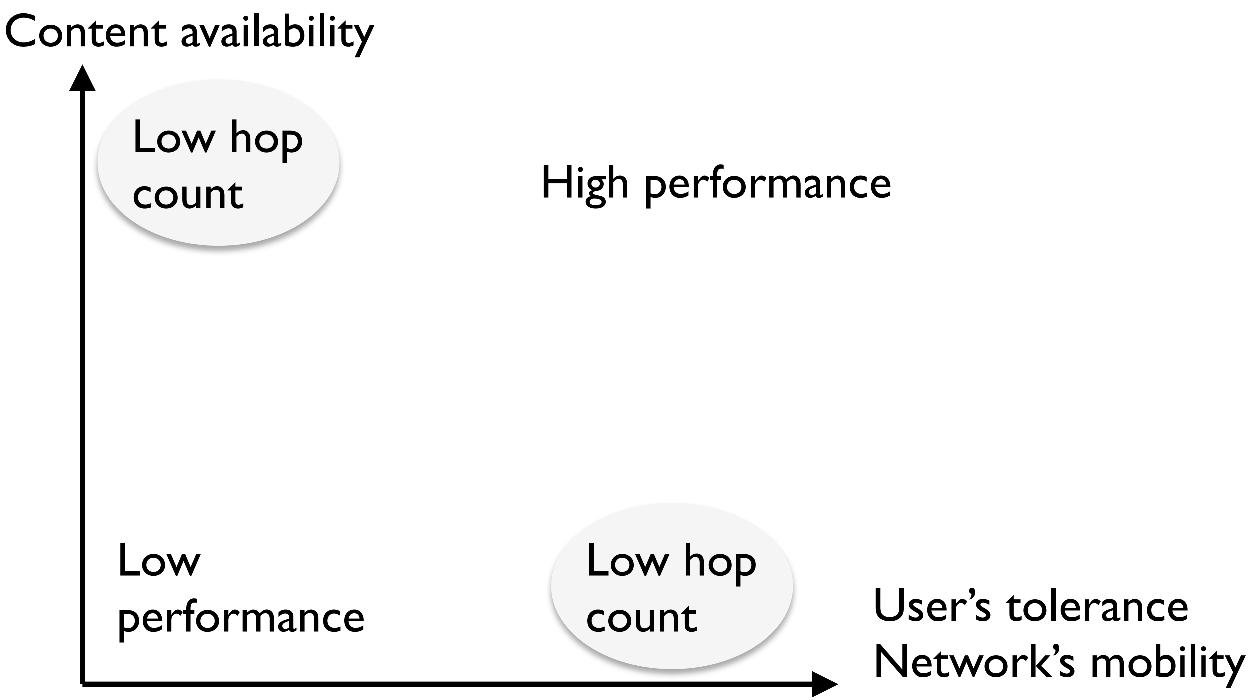


- Higher performance due to Capspotting trace having lower temporal distance to content.
- Low availability: 15 mins vs 4 hours under flooding-search

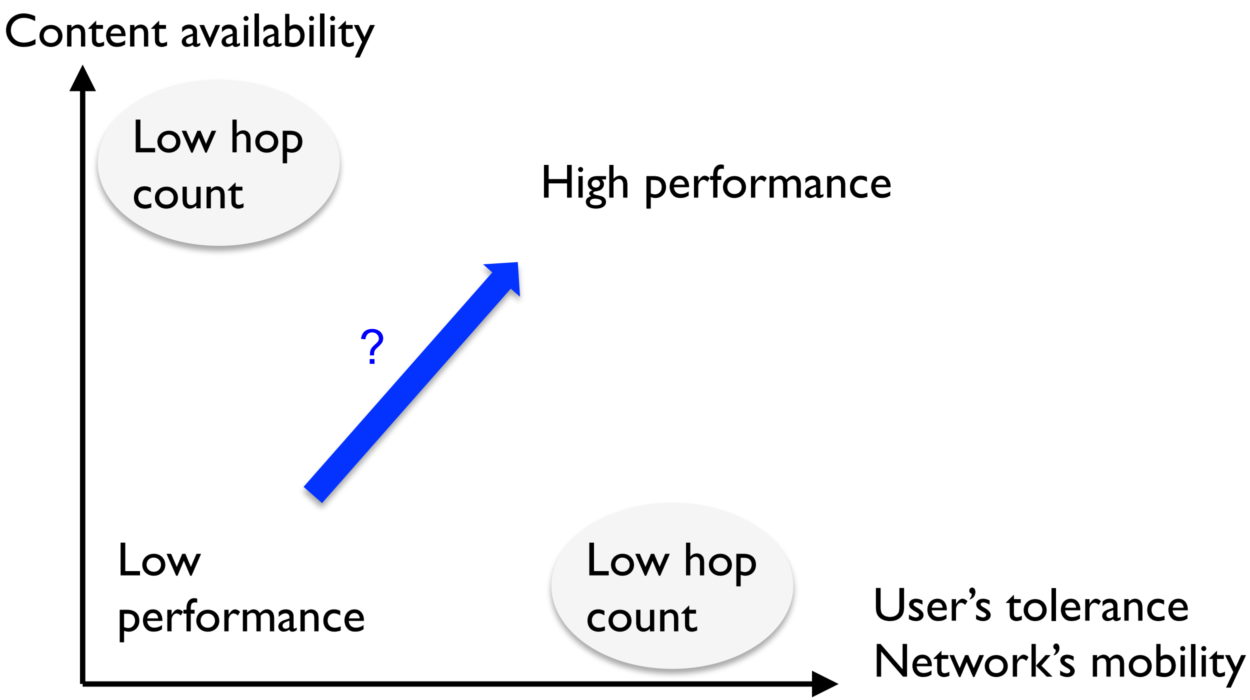
Operation region



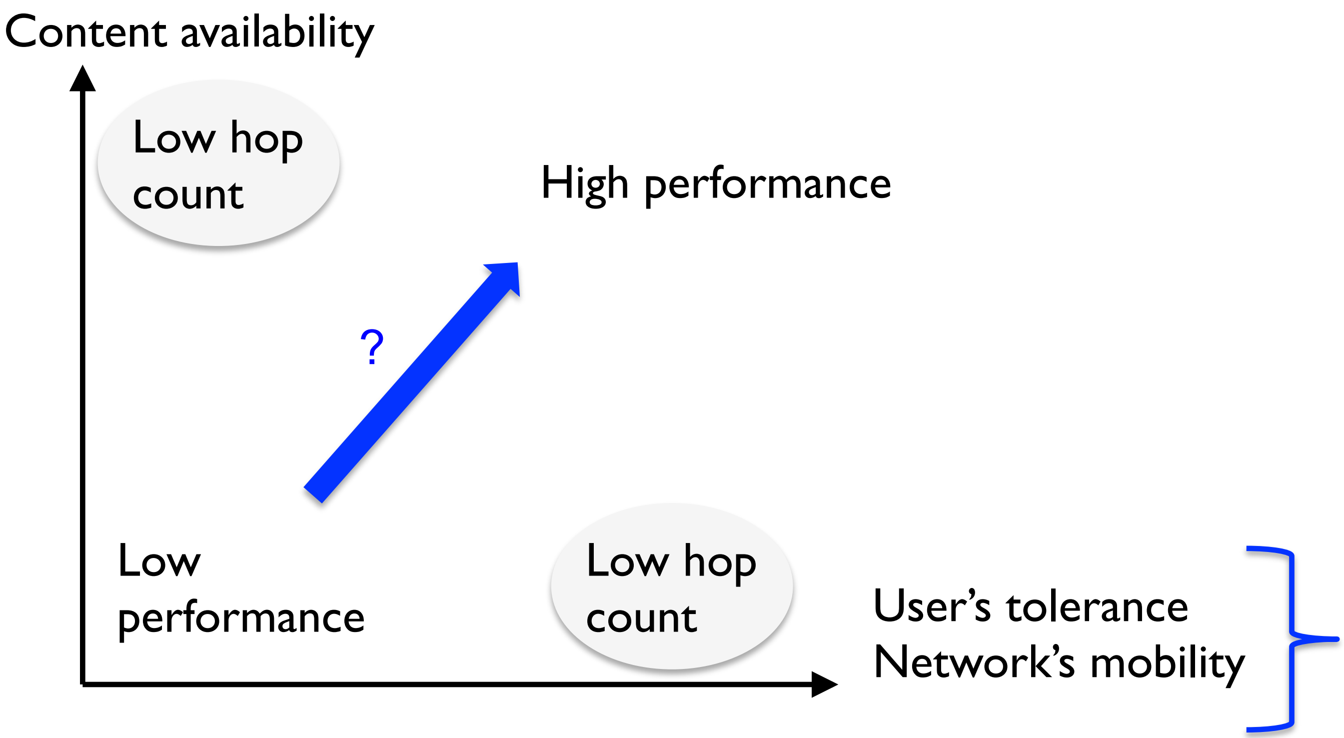
Operation region



Operation region



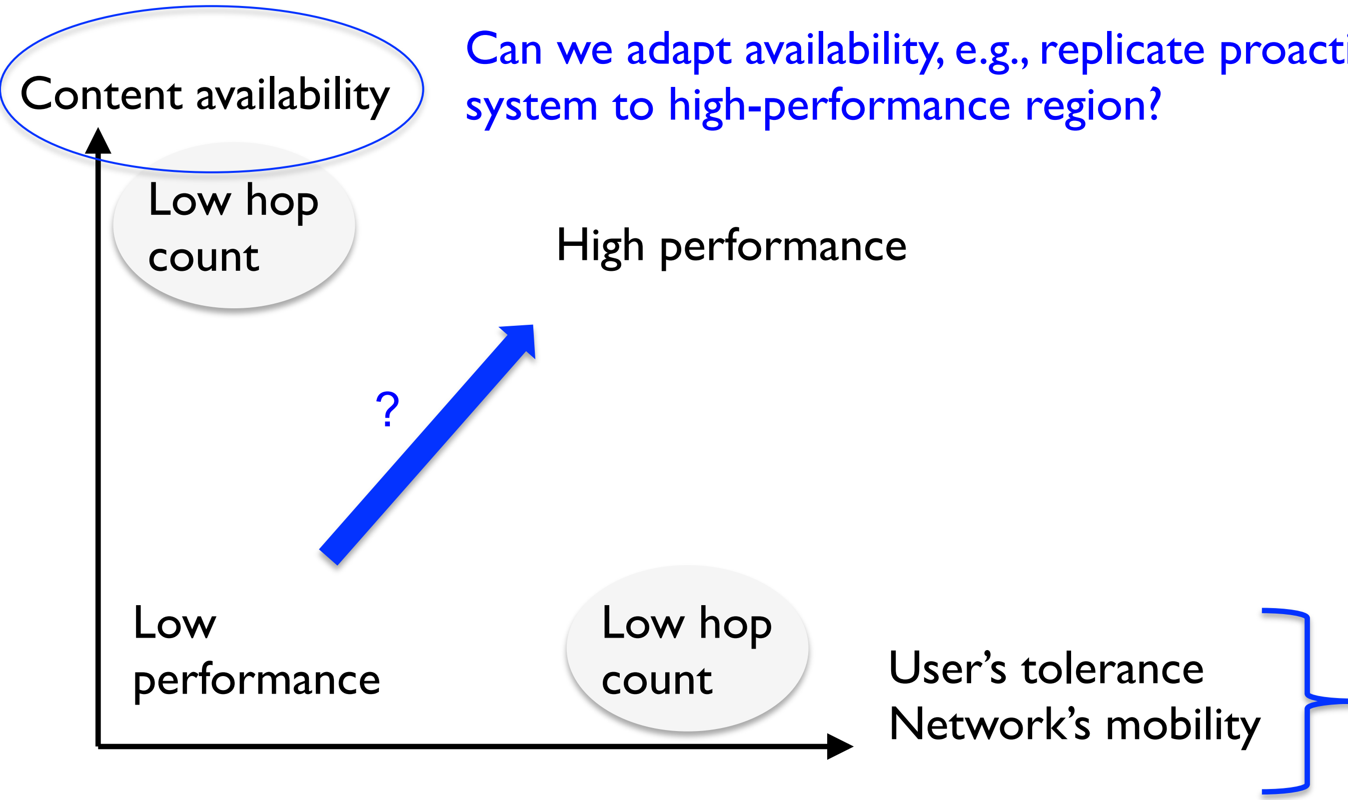
Operation region



May not be easy to push to higher-performance region

Operation region

Can we adapt availability, e.g., replicate proactively, to push the system to high-performance region?



May not be easy to push to higher-performance region

Availability estimation

Content availability

Can we adapt availability, e.g., replicate proactively, to push the system to high-performance region?

Yes, we can!

Availability estimation

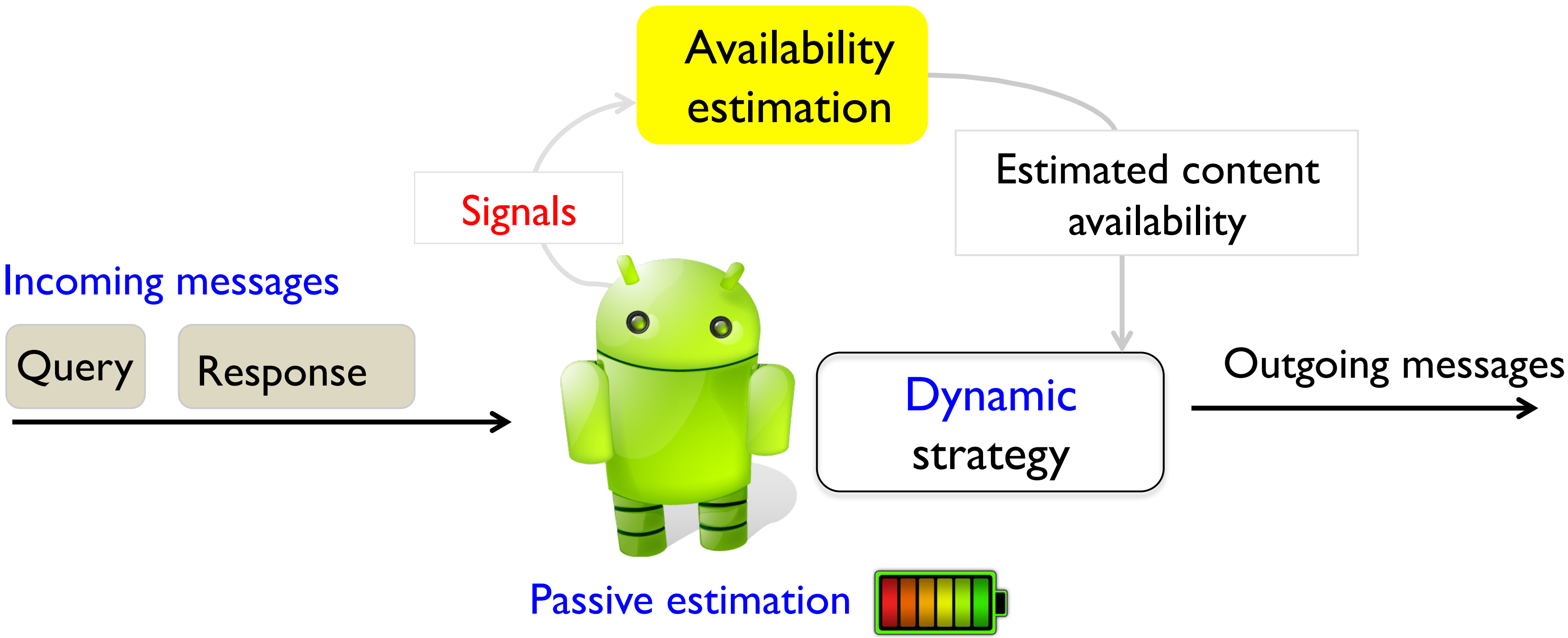
Content availability

Can we adapt availability, e.g., replicate proactively, to push the system to high-performance region?

Yes, we can!

Infer the operation region (estimate the availability) first to take an appropriate action


Incoming messages
Query
Response
Forwarding strategy
Outgoing messages



Signals: number of hops, number of queries, number of content providers

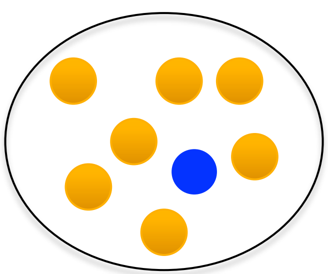
Assumptions

1. Uniform distribution of content: every node is equally likely to be a provider
2. Uniform interest distribution: every node searches for content c with equal probability

Assumptions

1. Uniform distribution of content: every node is equally likely to be a provider
2. Uniform interest distribution: every node searches for content c with equal probability

Under these assumptions, availability is simply: one over total number of nodes carrying the query at the time when the first content provider is discovered

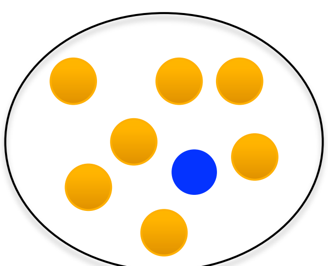


availability = 1/8

Assumptions

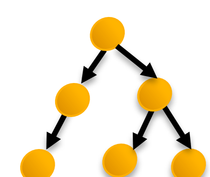
1. Uniform distribution of content: every node is equally likely to be a provider
2. Uniform interest distribution: every node searches for content c with equal probability

Under these assumptions, availability is simply: one over total number of nodes carrying the query at the time when the first content provider is discovered



availability = 1/8

Remember: query replicas follow different branches of the distribution tree



Assumptions

1. Uniform distribution of content: every node is equally likely to be a provider
2. Uniform interest distribution: every node searches for content c with equal probability

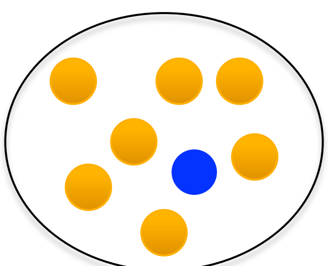
Under these assumptions, availability is simply: one over total number of nodes carrying the query at the time when the first content provider is discovered

Query

hop count	

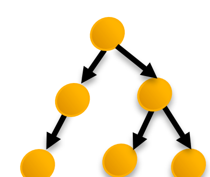
Response

query-hop	



availability = 1/8

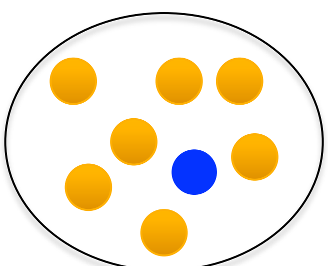
Remember: query replicas follow different branches of the distribution tree



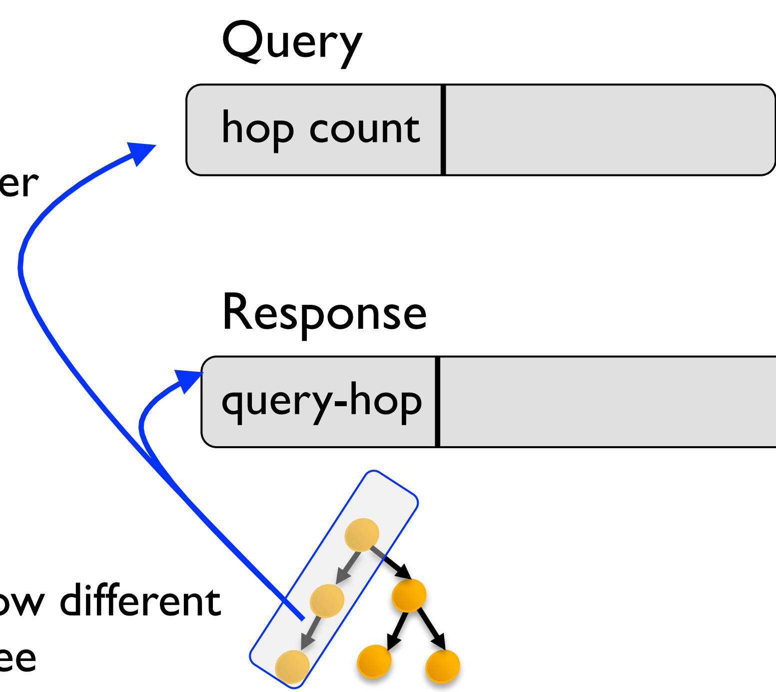
Assumptions

1. Uniform distribution of content: every node is equally likely to be a provider
2. Uniform interest distribution: every node searches for content c with equal probability

Under these assumptions, availability is simply: one over total number of nodes carrying the query at the time when the first content provider is discovered



availability = 1/8

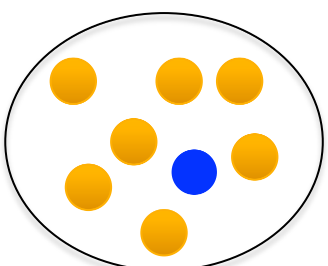


Remember: query replicas follow different branches of the distribution tree

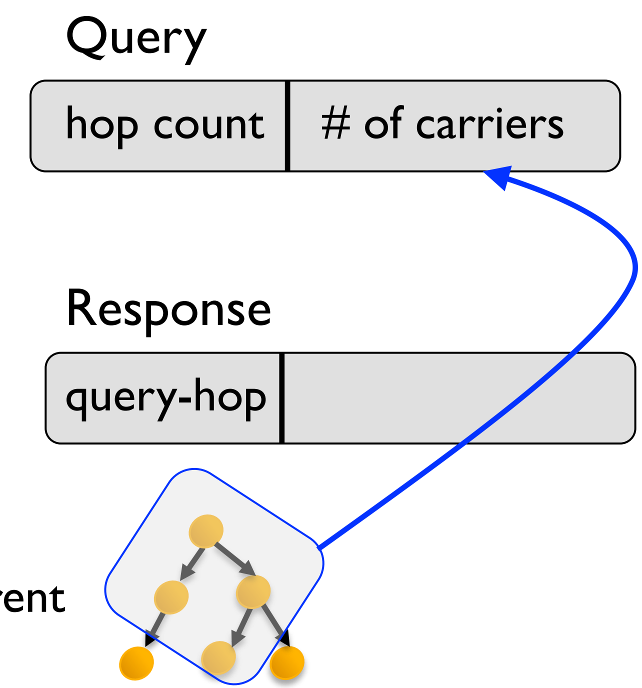
Assumptions

1. Uniform distribution of content: every node is equally likely to be a provider
2. Uniform interest distribution: every node searches for content c with equal probability

Under these assumptions, availability is simply: one over total number of nodes carrying the query at the time when the first content provider is discovered



availability = 1/8



Remember: query replicas follow different branches of the distribution tree

Assumptions

1. Uniform distribution of content: every node is equally likely to be a provider
2. Uniform interest distribution: every node searches for content c with equal probability

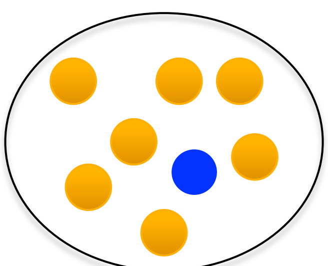
Under these assumptions, availability is simply: one over total number of nodes carrying the query at the time when the first content provider is discovered

Query

hop count	# of carriers

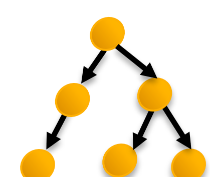
Response

query-hop	content provider

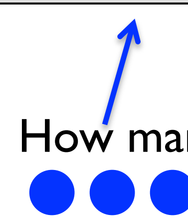


availability = 1/8

Remember: query replicas follow different branches of the distribution tree



How many?



Naïve estimation schemes

Query

hop count	# of carriers

Response

query-hop	content provider

- Q-HC: 1/(hop+1)
- Q-NC: 1/(# of carriers +1)

- R-HC: 1/(hop+1)
- R-CP: # of content providers/ network size

Naïve estimation schemes

Query

hop count	# of carriers

- Q-HC: 1/(hop+1)
- Q-NC: 1/(# of carriers +1)

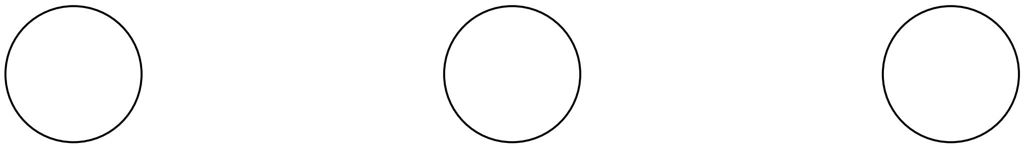
Response

query-hop	content provider

- R-HC: 1/(hop+1)
- R-CP: # of content providers/ network size

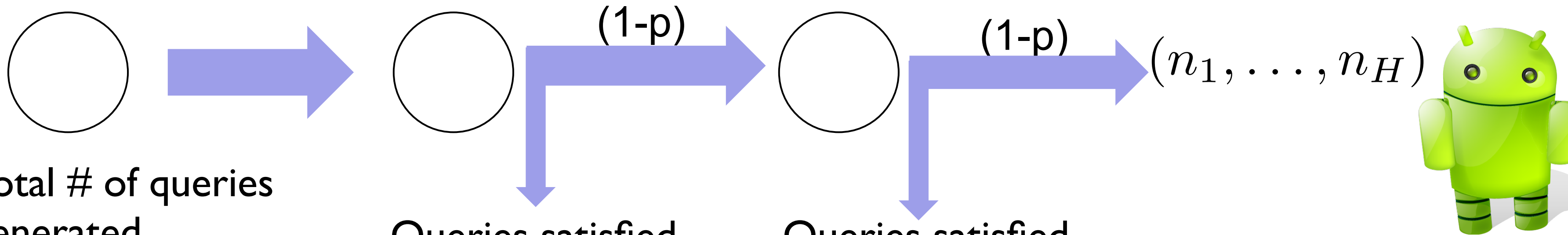
Can we have better schemes?

Least-squares (LS) based estimation, Q-LS



i-hop neighbors (i-1) hop neighbors 1-hop neighbors

Least-squares (LS) based estimation, Q-LS

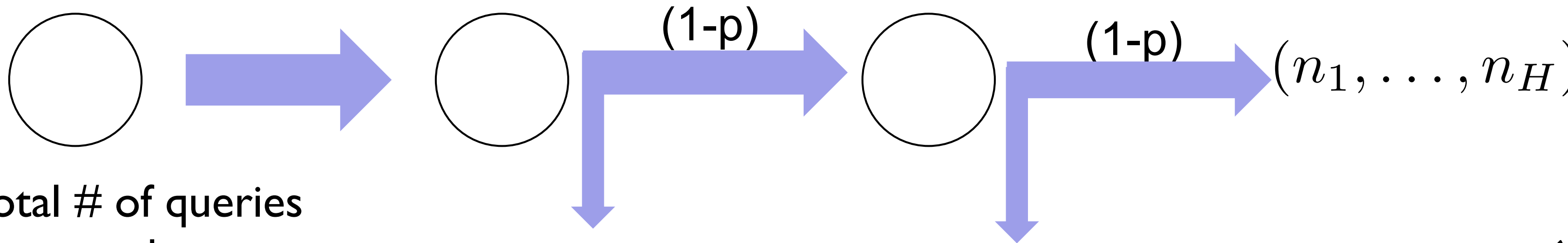


Total # of queries generated from i-hop neighbors: Mi

Queries satisfied with probability p 2-hop neighbors

Queries satisfied with probability p 1-hop neighbors

Least-squares (LS) based estimation, Q-LS



Total # of queries generated from i-hop neighbors: Mi

Queries satisfied with probability p 2-hop neighbors

Queries satisfied with probability p 1-hop neighbors

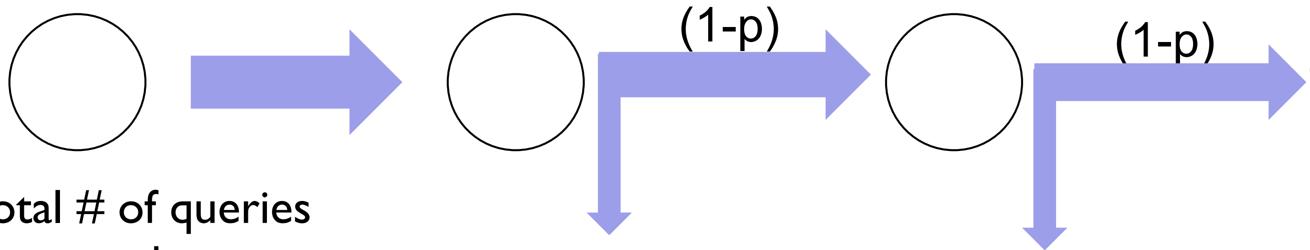
Observation:
# of queries with a certain hop count (ni)

Key idea: {ni} should be a decreasing sequence, exploit its distribution to find availability

Assumes:

- knowledge of i-hop neighbors**,** uniform query distribution, uniform content distribution

Least-squares (LS) based estimation, Q-LS



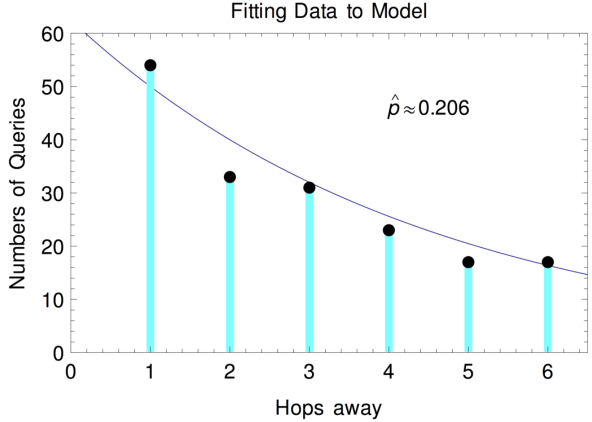
$(n_1, \ldots, n_H)$

Total # of queries generated from i-hop neighbors: Mi

Queries satisfied with probability p
2-hop neighbors

Queries satisfied with probability p
1-hop neighbors

$$\mathrm{E}[N_i] = (1-p)^{i-1}\mathrm{E}[M_i]$$



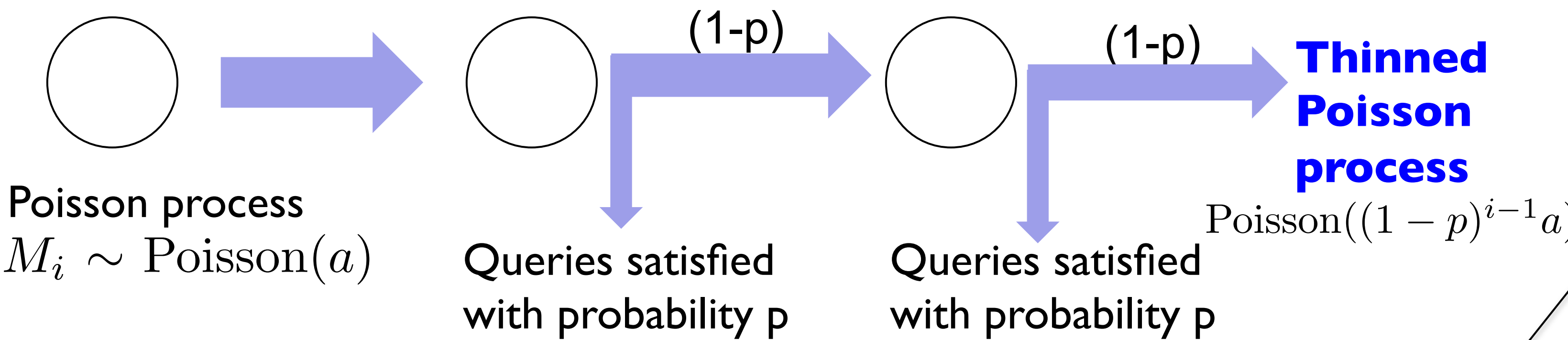
Observation:
# of queries with a certain hop count (ni)

Key idea: {ni} should be a decreasing sequence, exploit its distribution to find availability

Assumes:

- knowledge of i-hop neighbors**,** uniform query distribution, uniform content distribution

MLE based estimation, Q-ML



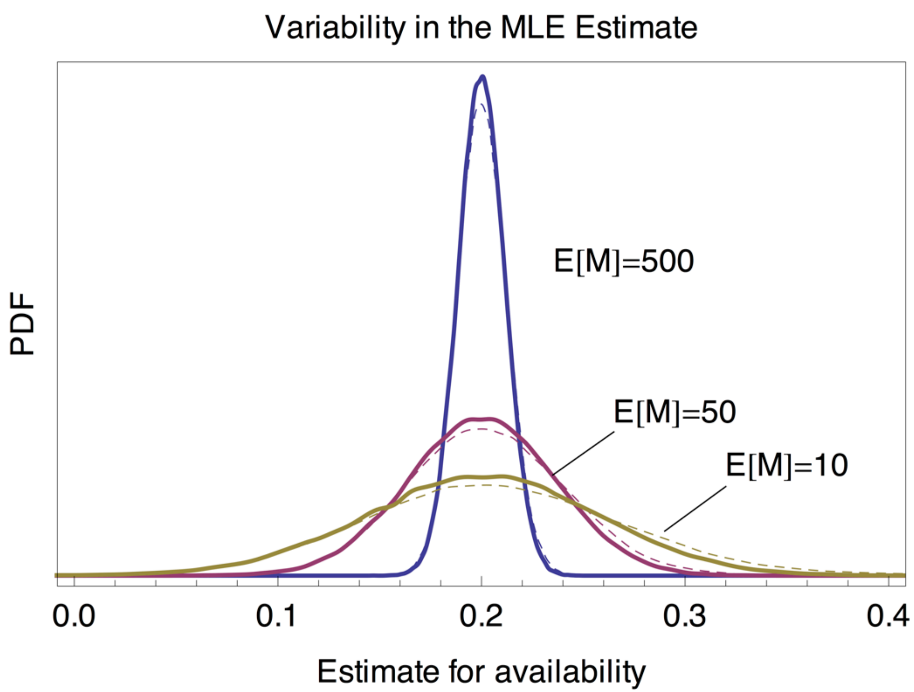
Observation: Thinned Poisson process

Assumes:
- knowledge of i-hop neighbors
- uniform query distribution
- uniform content distribution

Solve the MLE and find p

$$L = \prod_{i=1}^{H} \frac{\left(a(1-p)^{i-1}\right)^{n_i}}{n_i!} e^{-a(1-p)^{i-1}}$$

Q-ML vs. Q-LS (dashed lines)



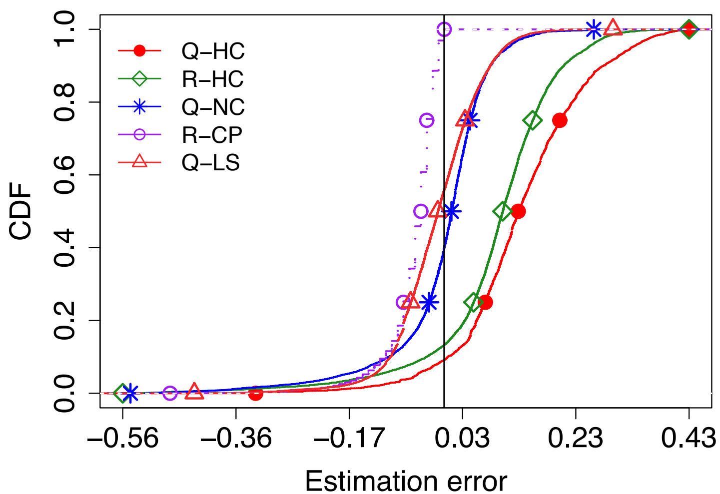
Assess variability in estimates using Monte-Carlo simulations, real availability = 0.2

Q-ML slightly better than Q-LS, but comes with higher complexity

Remember: we look for simple solutions, Q-LS

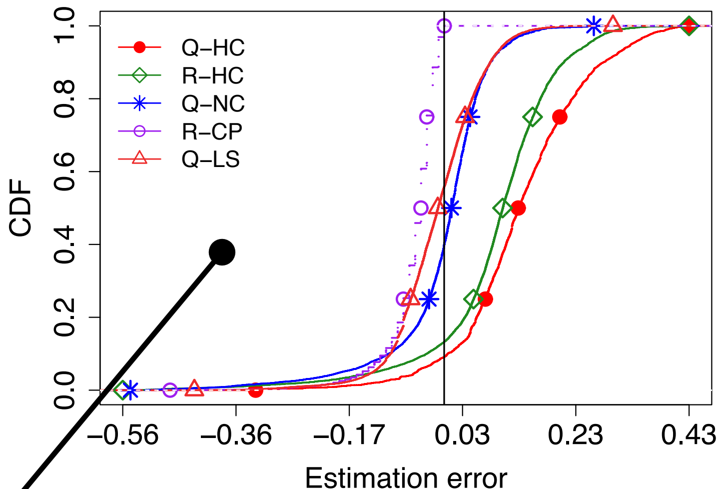
Infocom06: 98 users

- 100 content items, around 3200 queries generated, Zipf availability distribution 0.6, Weibull popularity distribution with k = 0.513
- Error = Estimated availability – Real availability



Infocom06: 98 users

- 100 content items, around 3200 queries generated, Zipf availability distribution 0.6, Weibull popularity distribution with k = 0.513
- Error = Estimated availability – Real availability

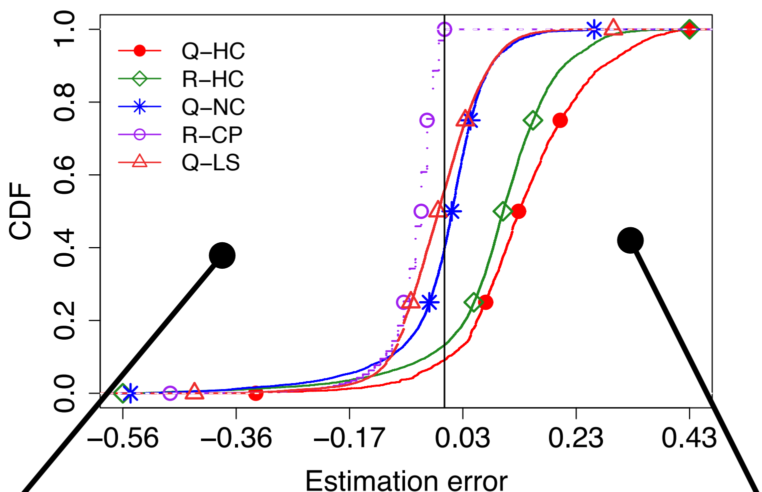


Under-estimation

R-CP: Not-all content providers can be discovered

Infocom06: 98 users

- 100 content items, around 3200 queries generated, Zipf availability distribution 0.6, Weibull popularity distribution with k = 0.513
- Error = Estimated availability – Real availability



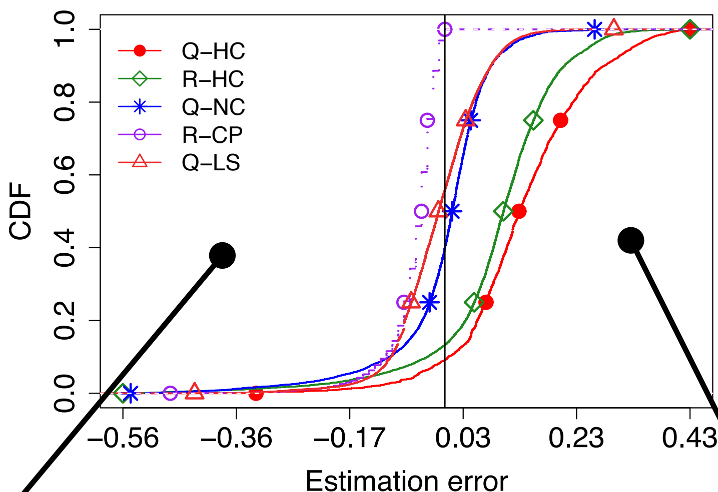
Under-estimation

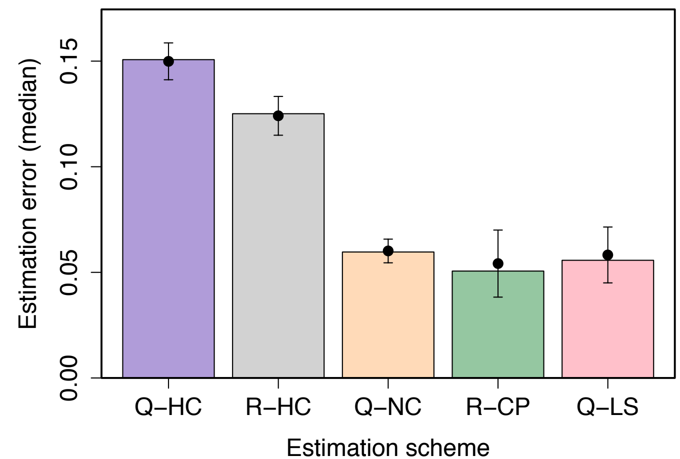
R-CP: Not-all content providers can be discovered

Over-estimation

Infocom06: 98 users

- 100 content items, around 3200 queries generated, Zipf availability distribution 0.6, Weibull popularity distribution with k = 0.513
- Error = Estimated availability – Real availability





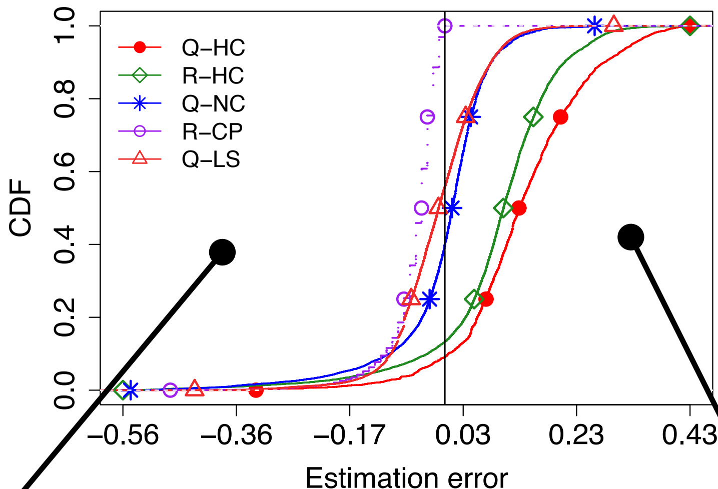
Under-estimation

R-CP: Not-all content providers can be discovered

Over-estimation

Infocom06: 98 users

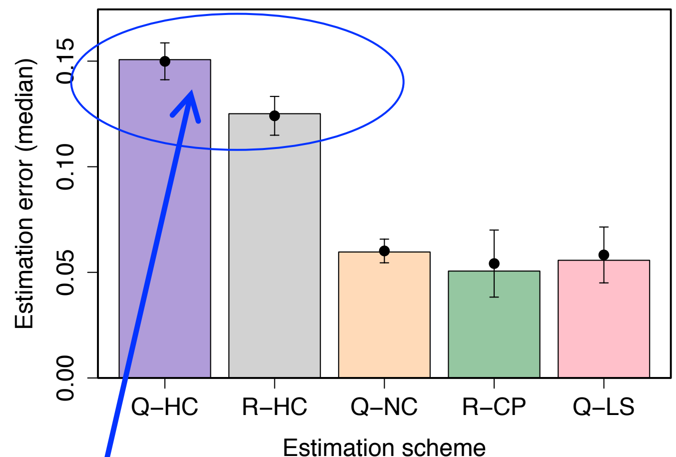
- 100 content items, around 3200 queries generated, Zipf availability distribution 0.6, Weibull popularity distribution with k = 0.513
- Error = Estimated availability – Real availability



Under-estimation

R-CP: Not-all content providers can be discovered

Over-estimation



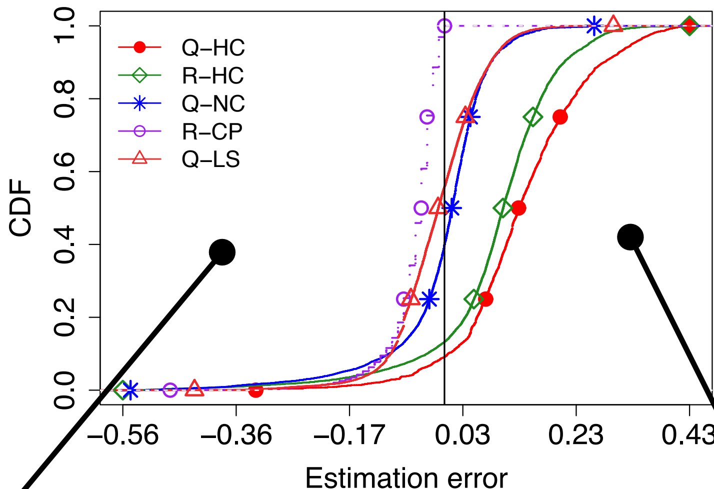
Not surprisingly, hop-based schemes overestimate

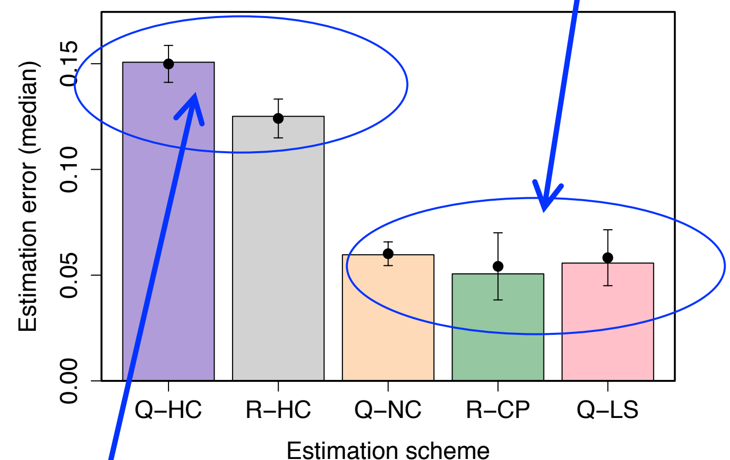
Small-world networks

Infocom06: 98 users

- 100 content items, around 3200 queries generated, Zipf availability distribution 0.6, Weibull popularity distribution with k = 0.513
- Error = Estimated availability – Real availability

Similar performance





Under-estimation

R-CP: Not-all content providers can be discovered

Over-estimation

Not surprisingly, hop-based schemes overestimate

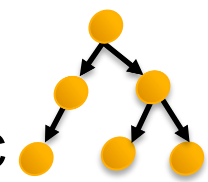
Small-world networks

Skewed distribution: can schemes detect it?

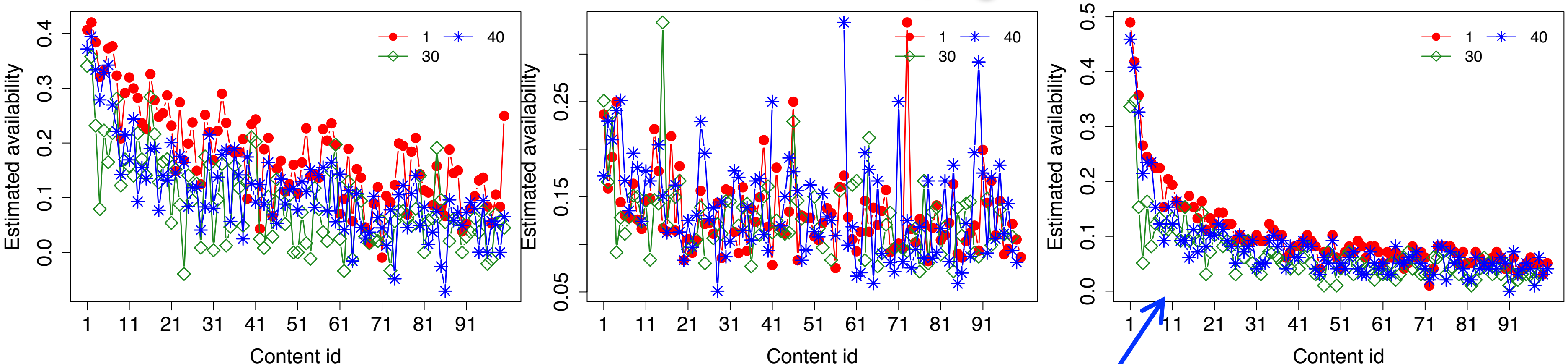
- Rather than availability value, availability group is more useful (e.g., which content to evict from cache) to know, e.g. content lies at the head or tail of the distribution.

Skewed distribution: can schemes detect it?

- Rather than availability value, availability group is more useful (e.g., which content to evict from cache) to know, e.g. content lies at the head or tail of the distribution.



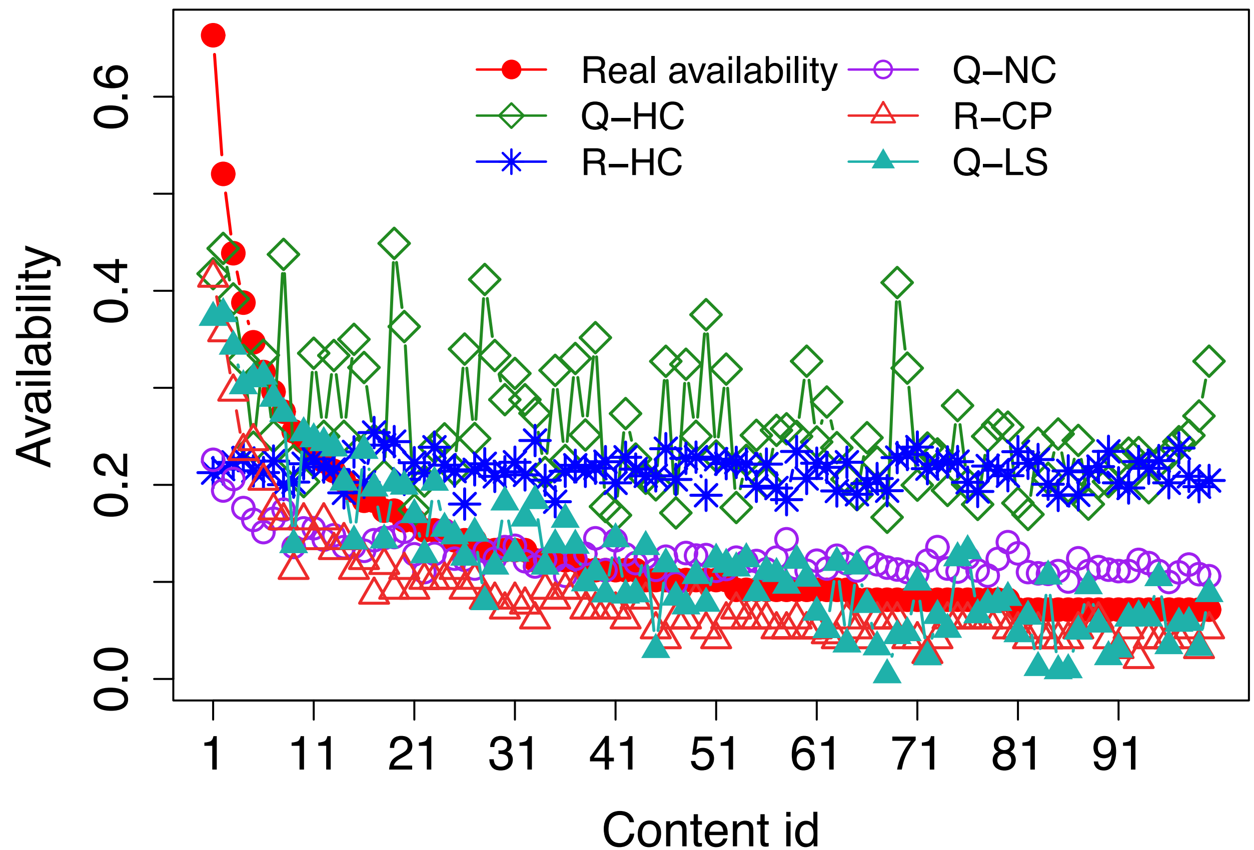
Q-LS Q-NC R-CP



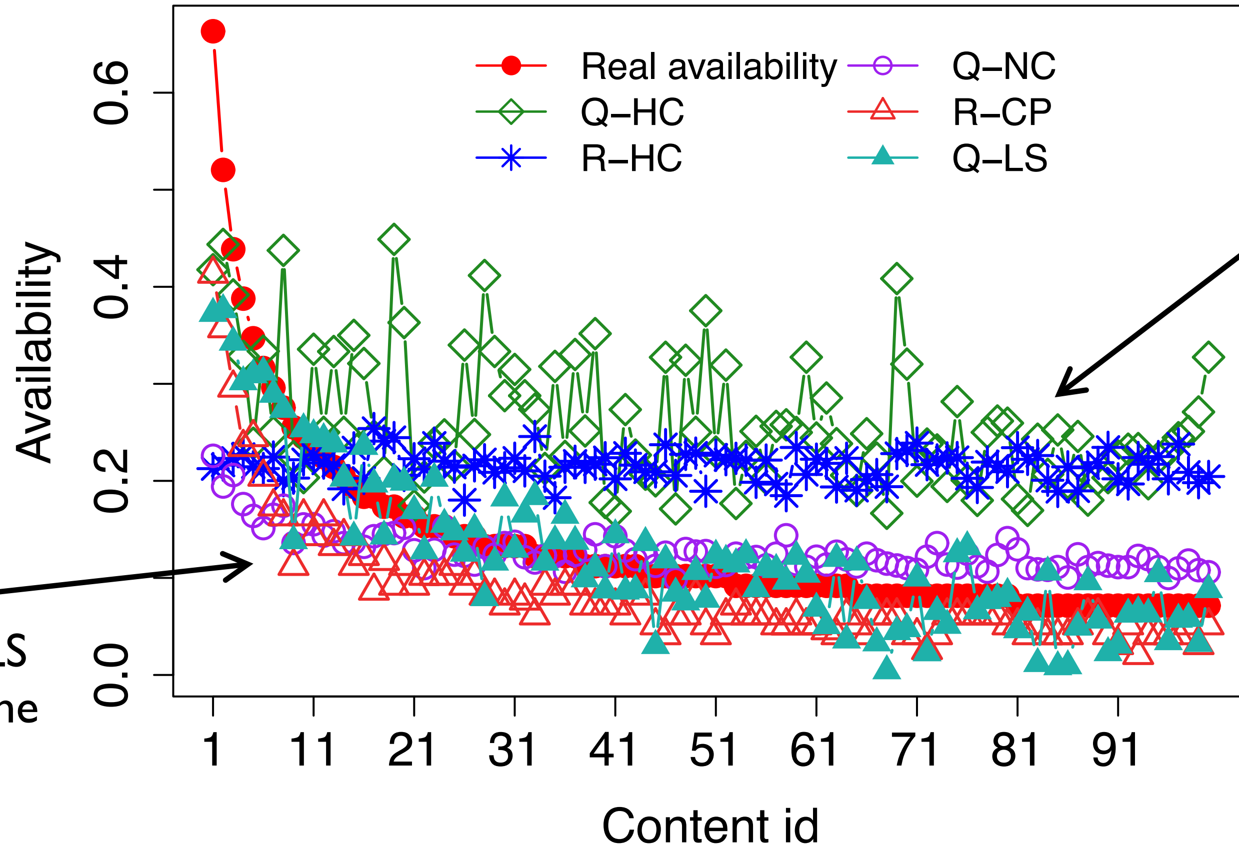
R-CP quite successful in catching the skew

Not a big difference among nodes

Collective estimates (nodes share their estimates)



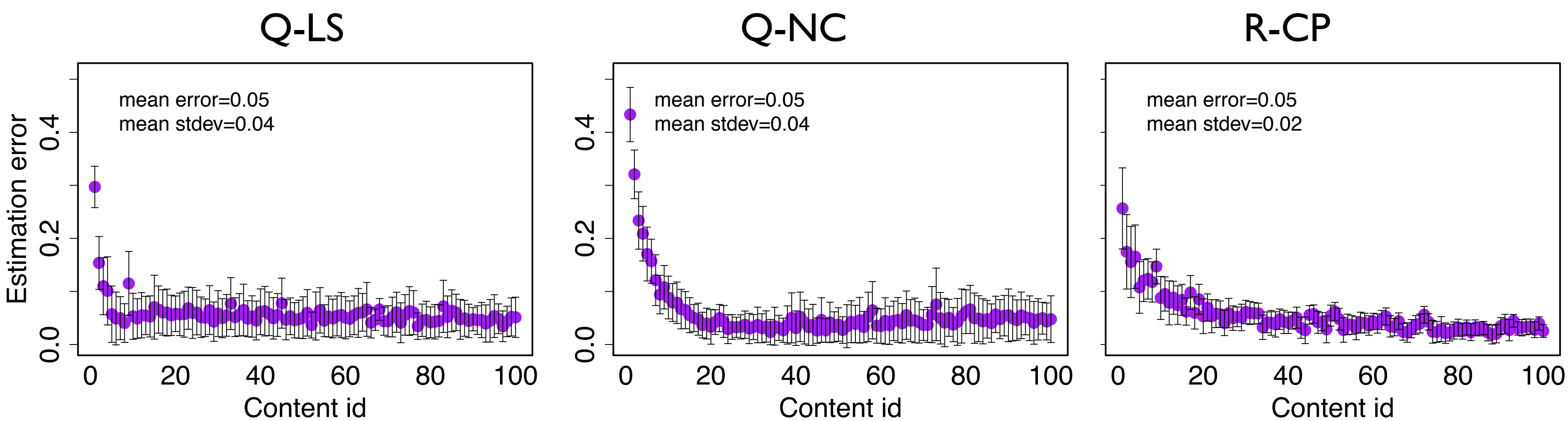
Collective estimates (nodes share their estimates)



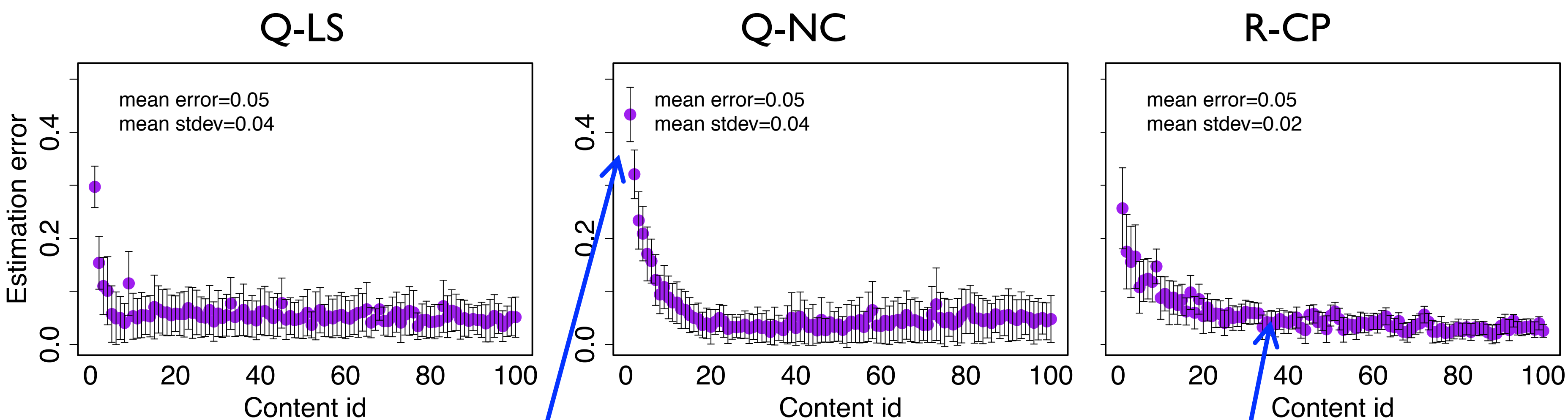
Small world! Query paths are short

R-CP and Q-LS can observe the skewness

Accuracy: the tail or the head?



Accuracy: the tail or the head?



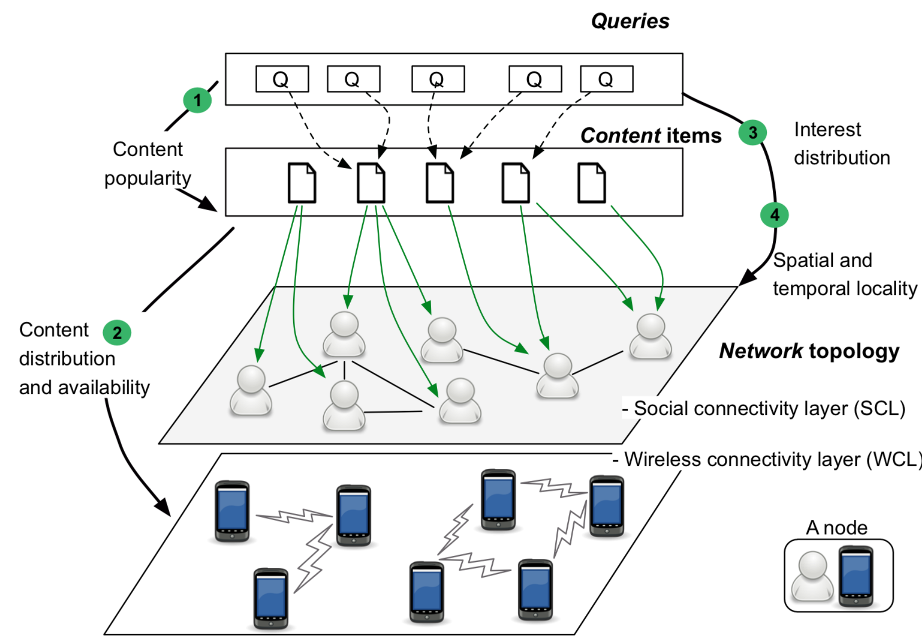
Lower accuracy at the head.
Available content: discovered easily, inaccurate estimation may not affect performance but may lead to resource inefficiency.

Lower variation among nodes

Three components of opportunistic search

A better representation of the three components of search

1. Users
   1. Wireless connectivity
   2. Social connectivity
2. Content items
   1. Popularity
   2. Availability
3. Queries
   4. Spatial and temporal locality



Can we capture all these layers and interactions?

Summary

- Three components of search: content availability, user's tolerated waiting time, network mobility (temporal distance to content)
- Optimal strategy depends on content availability (distribution) and cost metric
- Availability estimation: passive and naïve schemes based on number of replications, number of content providers, observed queries
    - Estimation in the wild with more realistic assumptions?
    - Dependency on number of observations, change in content availability/ popularity (how fast does it change?)
    - How to exploit this information in a complete search protocol design?

Thanks

http://www.netlab.tkk.fi/tutkimus/pdp/

supported by Academy of Finland

References

- E.Hyytiä, S.Bayhan, J. Ott and J. Kangasharju, Searching a Needle in (Linear) Opportunistic Networks, ACM MSWiM 2014.
- S.Bayhan, E. Hyytiä, J. Kangasharju, J. Ott, Two Hops or More: On Hop-Limited Search in Opportunistic Networks, ACM MsWiM 2015.
- E. Hyytiä, S. Bayhan, J. Ott, J. Kangasharju, On search and content availability in opportunistic networks, Computer Communications, 2015.
- S. Bayhan, E. Hyytiä, J. Kangasharju, J. Ott, Search in Digital Pockets: Retrieving Remote Content in Mobile Opportunistic Networks, under minor revision, 2015.
- J. Fan, J. Chen, Y. Du, P. Wang, and Y. Sun, "DelQue: A socially aware delegation query scheme in DTNs," IEEE Trans. on Vehicular Technology, vol. 60, no. 5, pp. 2181–2193, Jun 2011
- Y. Liu, Y. Han, Z. Yang, and H. Wu, "Efficient data query in intermittently-connected mobile ad hoc social networks," *IEEE Trans. on Parallel and Distributed Systems*, 2014.
- L. Wang, S. Bayhan, J. Ott, J. Kangasharju, A. Sathiaseelan, J. Crowcroft, Pro-Diluvian: Understanding Scoped-Flooding for Content Discovery in ICN, ACM ICN 2015.
- H. Shen, Z. Li, and K. Chen, "A scalable and mobility-resilient data search system for large-scale mobile wireless networks," IEEE Trans. on Parallel and Distributed Systems, vol. 25, no. 5, pp. 1124–34, 2013.
- M. Vojnović, and Alexandre Proutiere. "Hop limited flooding over dynamic networks." IEEE INFOCOM, 2011.
- E. Talipov, Y. Chon, and H. Cha, "Content sharing over smartphone- based DTNs," IEEE Trans. on Mobile Computing, vol. 12, no. 3, pp. 581–595, 2013.
- M. Pitkänen, T. Kärkkänen, J. Greifenberg, and J. Ott, "Searching for content in mobile DTNs," in IEEE PerCom, 2009.
- S. Bayhan, E. Hyytiä, J. Kangasharju and J. Ott. Seeker-Assisted Information Search in Mobile Clouds. ACM SIGCOMM Workshop on Mobile Cloud Computing (MCC), Hong Kong, 2013.

Estimation schemes

Actively collect information

Use existing information or minimal exchange