max planck institut informatik

Distributed Statistics for Peer-to-Peer Web Search

Gerhard Weikum (weikum@mpi-inf.mpg.de)

In collaboration with:
Matthias Bender, Sebastian Michel, Nikos Ntarmos,
Josiane Parreira, Peter Triantafillou, Christian Zimmer

„An engineer is someone who can do for a dime what any fool can do for a dollar" [anonymous]

Peer-to-Peer (P2P) Systems

Decentralized, self-organizing, highly dynamic loose coupling of many autonomous computers

→ unstructured overlay networks
with epidemic dissemination (flooding)

→ structured overlay networks
based on distributed hash tables (DHTs)

Applications:

- Large-scale computation (SETI@home, etc.)
- File sharing (Napster, Gnutella, KaZaA, BitTorrent, etc.)
- Publish-Subscribe (Blogs, Marketplaces, etc.)
- Collaborative work (Games, etc.)
- IP telephony (Skype)

Any applications that are useful *and* legal ?
And scientifically challenging?

Peer-to-Peer Web Search

Vision: Self-organizing P2P Web Search Engine
with Google-or-better functionality

- **Scalable & Self-Organizing** Data Structures and Algorithms
  (DHTs, Semantic Overlay Networks, Epidemic Spreading, Distr. Link Analysis, etc.)

- Better **Search Result Quality** (Precision, Recall, etc.)
  - Powerful Search Methods for Each Peer
    (Concept-based Search, Query Expansion, XML IR, Personalization, etc.)
  - Leverage User/Community Input („Wisdom of Crowds“)
    (Bookmarks, Feedback, Query Logs, Click Streams, Evolving Web, etc.)
  - Collaboration among Peers
    (Query Routing, Incentives, Fairness, Anonymity, etc.)

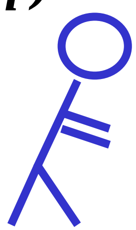
- Benefit of Large-scale **Social Networks**:
  - Small-World Phenomenon
  - Breaking Information Monopolies

Exploiting Query Logs and Click Streams

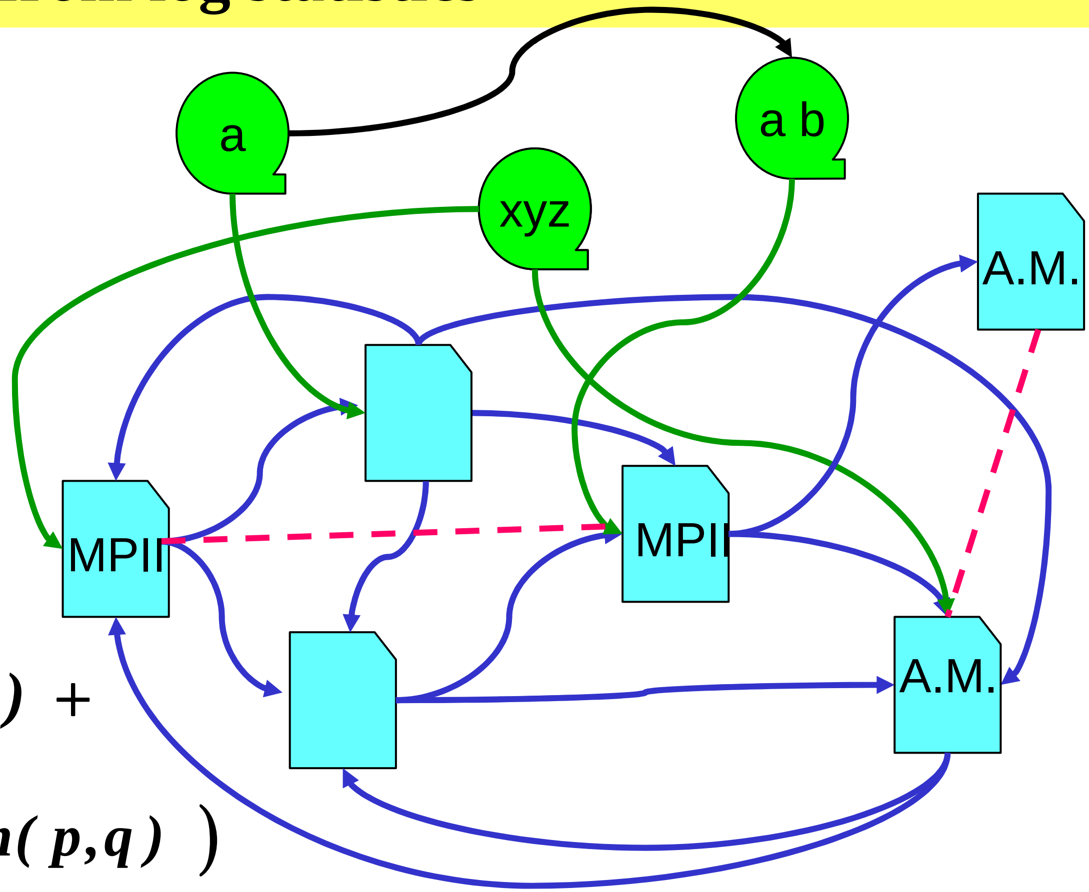
from PageRank: uniformly random choice of links + random jumps

to QRank: + query-doc transitions + query-query transitions
+ doc-doc transitions on implicit links (w/ thesaurus)
with probabilities estimated from log statistics

$$PR(q) = \varepsilon \cdot j(q) + (1-\varepsilon) \cdot \sum_{p \in IN(q)} PR(p) \cdot t(p,q)$$



$$QR(q) = \varepsilon \cdot j(q) + (1-\varepsilon) \cdot \Big(\alpha \sum_{p \in explicitIN(q)} PR(p) \cdot t(p,q) \; + \; (1-\alpha) \sum_{p \in implicitIN(q)} PR(p) \cdot sim(p,q) \Big)$$



Preliminary Experiments

Setup:

70 000 Wikipedia docs, 18 volunteers posing Trivial-Pursuit queries
ca. 500 queries, ca. 300 refinements, ca. 1000 positive clicks
ca. 15 000 implicit links based on doc-doc similarity

Results (assessment by blind-test users):

- **QRank top-10 result preferred over PageRank in 81% of all cases**
- **QRank has 50.3% precision@10, PageRank has 33.9%**

Untrained example query „philosophy“:

	PageRank	QRank
1.	Philosophy	Philosophy
2.	GNU free doc. license	GNU free doc. license
3.	Free software foundation	Early modern philosophy
4.	Richard Stallman	Mysticism
5.	Debian	Aristotle

Outline

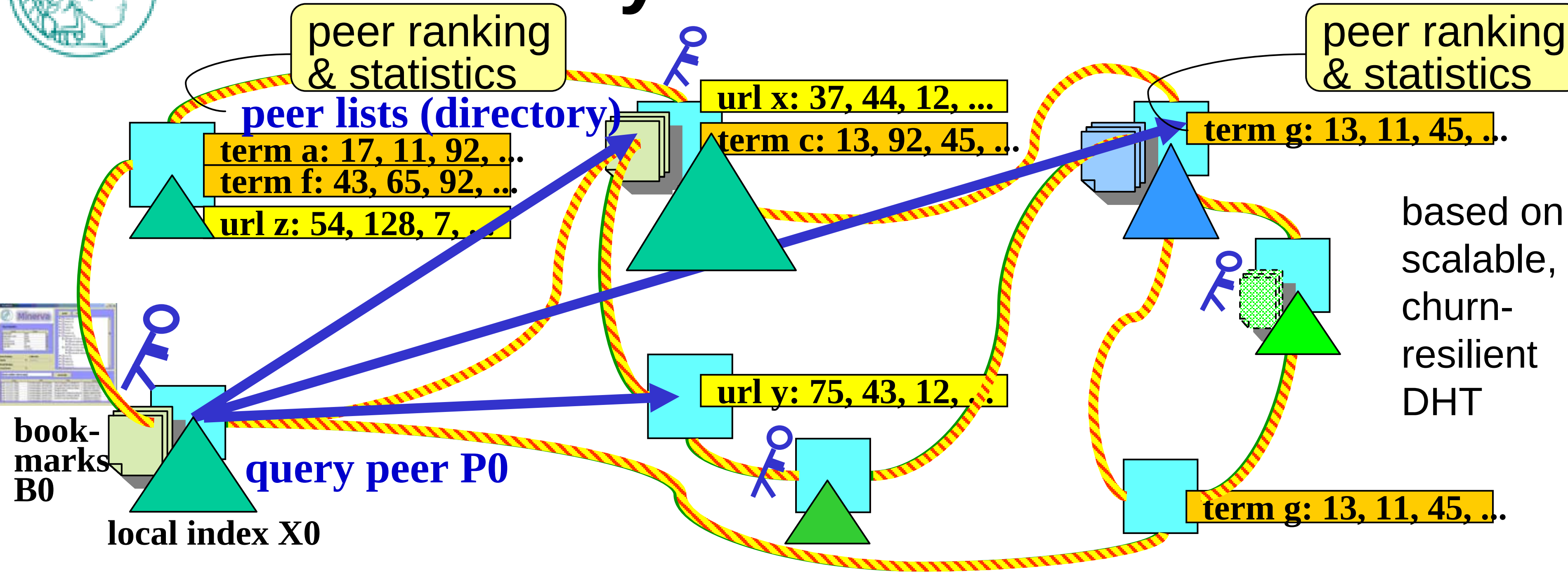
✓ **Motivation and Research Directions**

- **P2P Query Routing**
  - **Peer Content Quality**
  - **Overlap Awareness**
  - **Challenges in Query Routing & P2P Statistics**
- **P2P Link Analysis**
  - **JXP Authority Scoring**
  - **Challenge: Beyond Link Analysis**
- **Conclusion**

Computational Model

- Peers connected by **overlay network** (e.g. DHT, random graph) and IP
- Each peer has a **full-fledged local search engine** with crawler/importer, indexer, query processor
- Each peer has **autonomously compiled** (e.g. crawled) its **own content** according to the user‘s thematic interests $\rightarrow$ peer-specific collections
- When a query is issued by a peer, it is first **executed locally** and then possibly **routed** to carefully selected **other peers**
- Peers can **post** summaries / synopses / metadata / QoS info to (distr.) **network-wide directory** **(space O(#terms * #peers))** with efficient per-key **lookup**

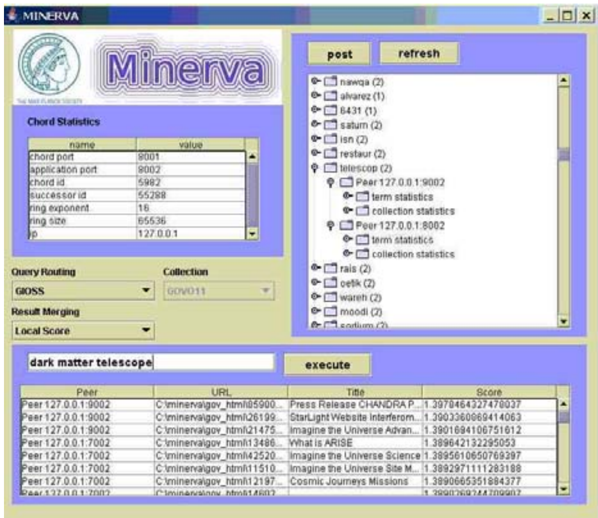
Minerva System Architecture



Query routing (QR) aims to optimize **benefit/cost**
driven by distributed statistics on
peers‘ content quality, content overlap,
freshness, authority, trust, performability etc.

Dynamically precompute „good peers“
to maintain a **Semantic Overlay Network (SON)**

Exploit **community input** (bookmarks, etc.)



P2P Query Routing (QR) (Resource Selection)

Related to distributed IR & metasearch engines, but:
larger scale, high dynamics, local execution first

State-of-the-art representative:
CORI [Callan et al.] – prob-IR weights with BM25-like tuning

score (peer p, query q) ~

$$\sum_{t \in q} \frac{cdf(t,p)}{cdf(t,p) + 50 + 150 \cdot \frac{|terms(p)|}{avg_{peers\ v} |terms(v)|}} \cdot \frac{\log((|peers| + 0.5) / cf(t))}{\log(|peers| + 1)}$$

collection document frequency cdf(t,p): #docs at peer p that contain term t
collection frequency cf(t): #peers that contain term t

- **Precompute** per-term peer-quality scores & keep in directory
- QR **aggregates PeerLists** for query terms & selects top-k peers

Peer Quality based on Peer LM

Multinomial model for peer p generating query q = {t_1, t_2, …$t_{|q|}$}:

$$P[q \mid p] = \binom{|q|}{f(t_1)\, f(t_2) \ldots f(t_{|q|})} \Pi_{t \in q} \theta_t(p)^{f_t(q)}$$

neg. Kullback-Leibler (KL) divergence of f(q) and θ(p)

$$\sim \Sigma_{t \in q} f_t(q) \log_2 \theta_t(p) \quad = -H(f(q), \theta(p))$$

$$\sim -H(f(q), \theta(p)) + H(f(q)) \quad = -\sum_t f_t(q) \log_2 \frac{f_t(q)}{\theta_t(p)}$$

LMs effective but expensive and tuning-intensive

Lifting doc-query LM to peer-query LM is attractive:

- **larger granularity → better cost amortization**
- **naturally smoothed peer LM → better parameter estimation**
- **can use local query result or bookmarks for smoothed query LM**

Showstoppers (for now):

- **KL is not a metric, no transitive pruning**
- **Peer-peer similarity overfits to content quality and ignores overlap**

Overlap Awareness [Bender et al.: SIGIR'05]

Estimate overlap(p_0, p_j) between
query initiator peer p_0 and QR candidate p_j
using Bloom filters on the URLs in the collections of p_0 and p_j

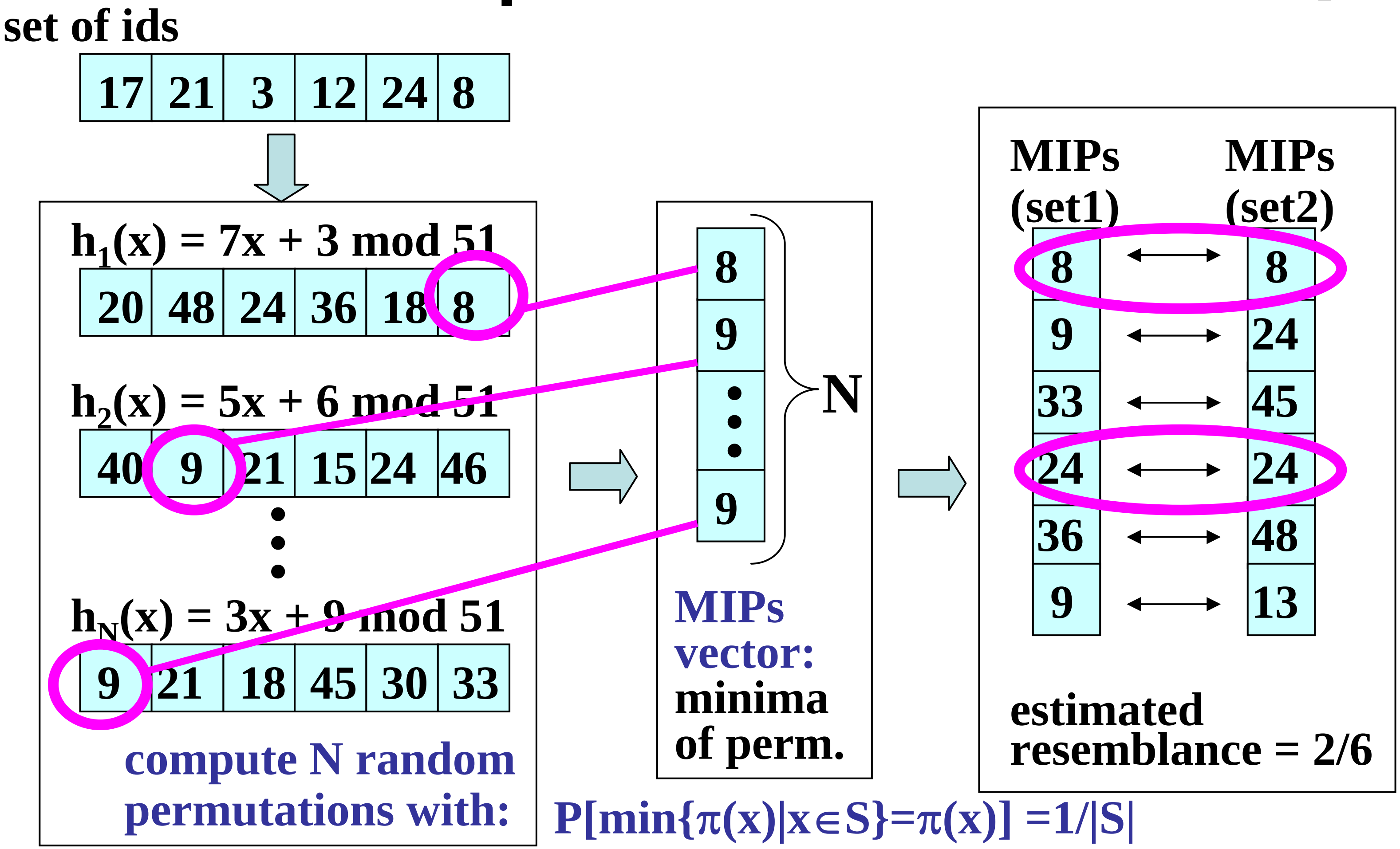
Consider candidates p_j in desc. order of est. quality for q
and re-rank by:

$$benefit(p_j, q) = \alpha \sum_{t \in q} quality(p_j, t) + (1-\alpha) \cdot overlap(p_j, p_0)$$

Caveats:

- **Per-term Bloom filters precomputable and composable by ∪ and ∩, with fixed-sized bit vectors only, but designed for membership testing, not for cardinality estimation**
- **Overlap not only critical between p_0 one candidate p_j, but also among multiple candidate peers**

Min-Wise Independent Permutations [Broder 97]



MIPs are **unbiased estimator of resemblance**:

$$P\,[\min\{h(x) \mid x \in A\} = \min\{h(y) \mid y \in B\}] = |A \cap B| \,/\, |A \cup B|$$

MIPs can be viewed as repeated sampling of x, y from A, B

Integrated Quality-Novelty QR [Bender et al.: EDBT'06]

Precompute MIPs vector & df values for each peer and term (possibly with variable #permutations)

IQN QR (q, p0):

fetch **PeerLists** from directory for all $t \in q$ and aggregate by quality

repeat

fetch **MIPs vectors** for new candidates from directory

- **Select best peer:**

  choose p_j, given already selected set $P=\{p_1, \ldots, p_{j-1}\}$,

  based on weighted sum of quality and $\sum_{t \in q} novelty(p_j(t) \mid P(t))$

- **Aggregate synopses:**

  compute **union of MIPs** synopses for P and p_j by min of minima

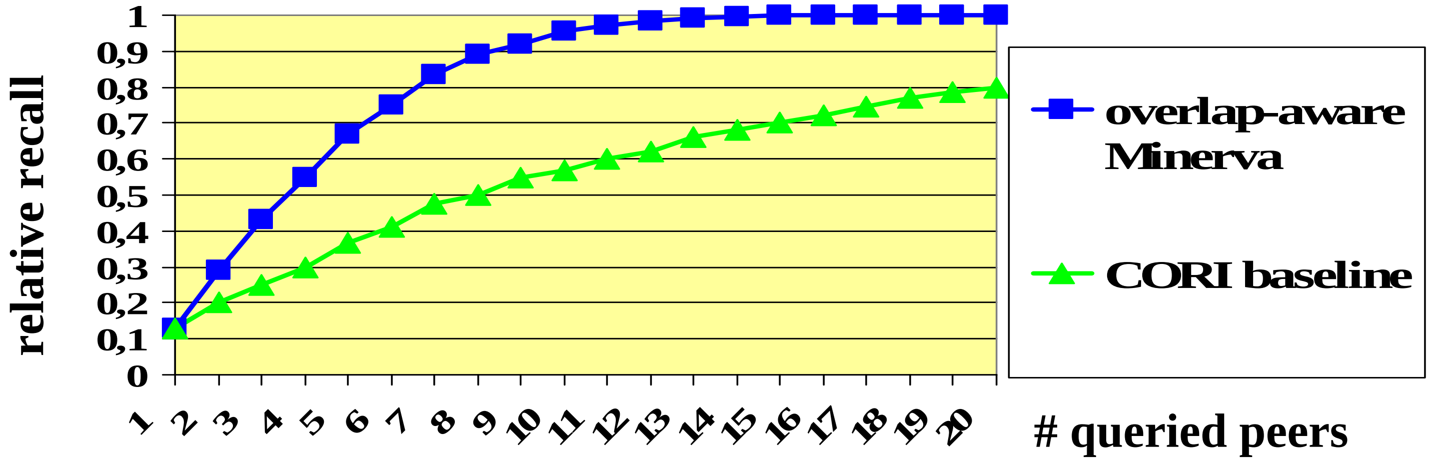
until sufficient #peers or expected recall above threshold

$$novelty(A \mid B) = |B| - |A \cap B| = |B| - \frac{resemblance(A,B) \cdot (|A| + |B|)}{resemblance(A,B) + 1}$$

IQN Experimental Results

Experiment:

based on 100 .Gov partitions (1.25 Mio. docs), assigned to 50 peers,
with each peer holding 10 partitions and 80% overlap for peers P_i, P_{i+1}
with 50 TREC-2003 Web queries,
e.g.: „pest safety control“ „juvenile delinquency“, „Marijuana legalization“, etc.

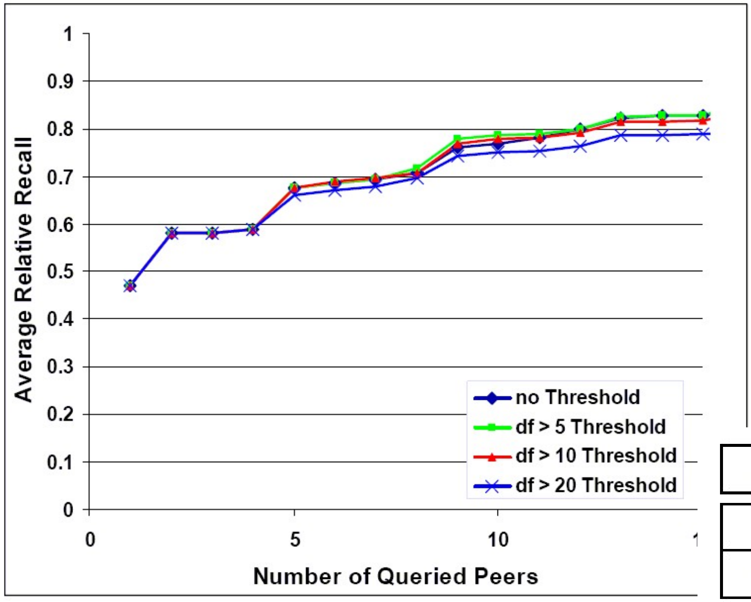


For more experiments see our papers (Bender et al.: SIGIR’05, EDBT’06, WIRI’06)

Discriminative Posting

peer p_j posts a term only if p_j has term-specific content **above average** (or above quantile) of quality measure
→ reduces load on P2P directory
→ may ease decision on good query routing
→ requires **global statistics on quality measures !**

e.g. peer posts only if **local df > λ*(global df)** with λ < 1



Experiment:
250 000 Web pages on 40 peers
popular Google queries
(e.g. „national hurricane center“)

Absolute Threshold	0	5	10	20
Total # of Posts	4,747,517	964,274	651,437	430,080
Percent	100.00%	20.37%	13.72%	9.06%

Efficiently Capturing Global Statistics

gdf (global doc. freq.) of a term is interesting key measure,
- for discriminative posting or
- for P2P result merging,

but overlap among peers makes simple distr. counting infeasible

hash sketches [Flajolet/Martin 85]:

duplicate-sensitive cardinality estimator for multisets

- hash each multiset element *x* onto *m*-bit bitvector and remember **ls 1 bit ρ(h(x))**
- **$\max_{x \in S}$ ρ(h(x))** estimates ≈ $\log_2 0.77351 |S|$ with std.dev. / |S| = $O(1/\sqrt{m})$
- rough intuition: $|S| << \log_2 k \Leftrightarrow \max \rho < k$
- average multiple iid sketches

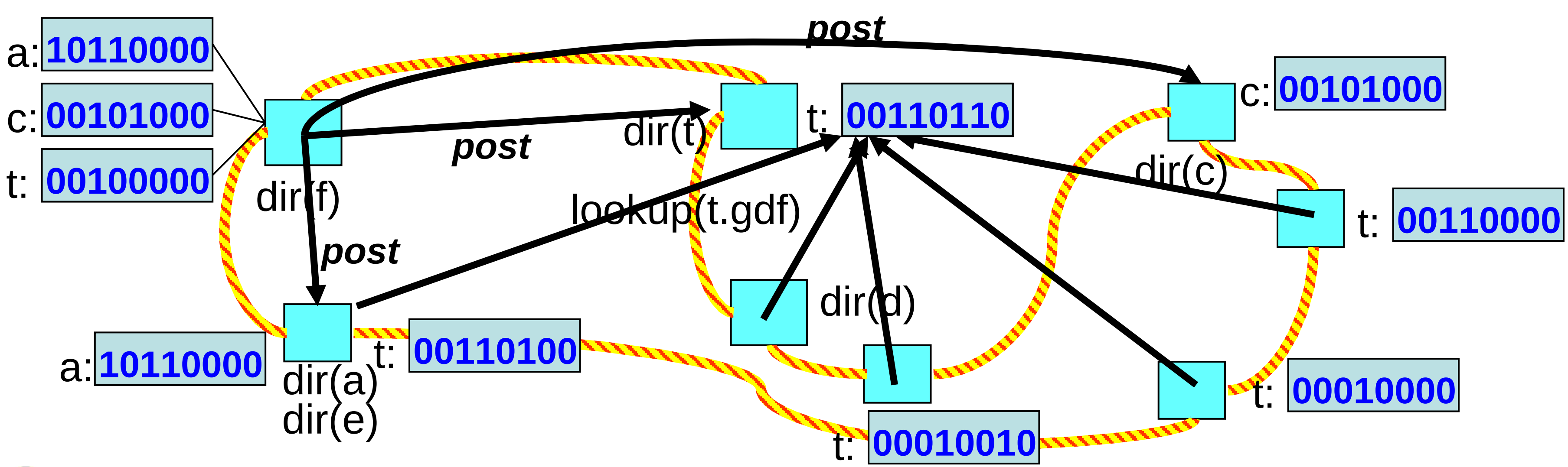
Efficient & Accurate gdf Estimation

Hash sketches of different peers collected at directory peer

distributivity is free: $\cup_i \{\rho(h(x)) \mid x \in S_i\} = \{\rho(h(x)) \mid x \in \cup_i S_i\}$

gdf estimation algorithm:

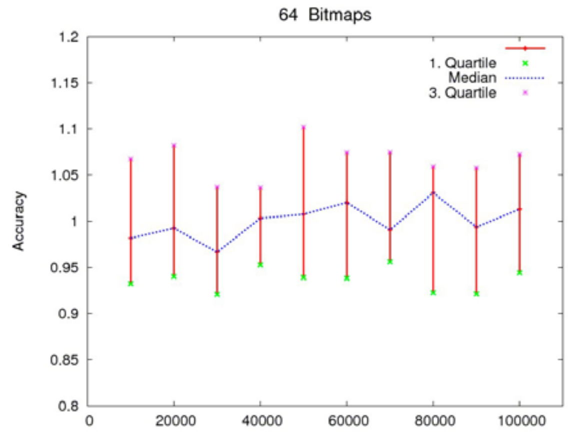
- each peer p posts hash sketch for each (discriminative) term t to directory
- directory peer for term t forms union of incoming hash sketches
- when a peer needs to know gdf(t), simply ask directory peer for t
- sliding-window techniques for dynamic adjustment

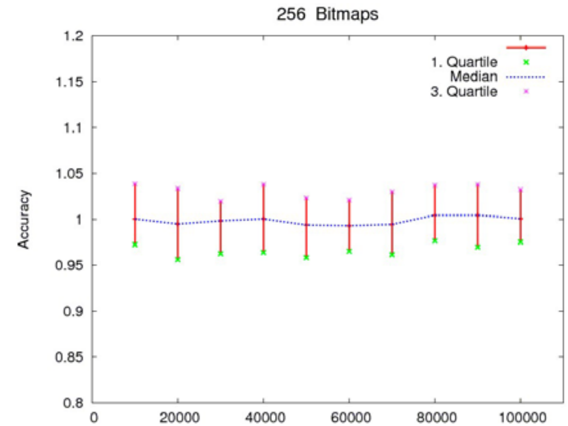


gdf Estimation Experiments

Experiment with steady-state P2P system:

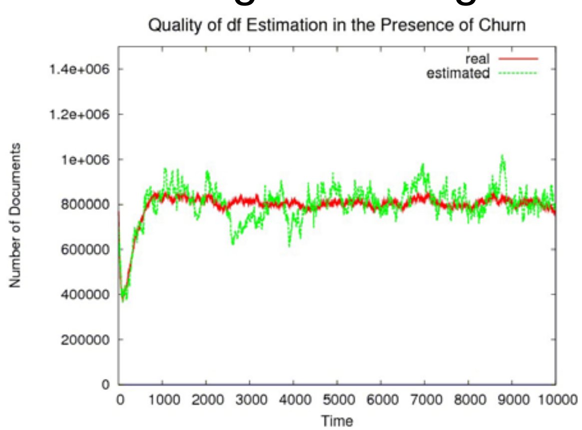
1000 peers, each with 1000 randomly chosen docs from 1 Mio. docs

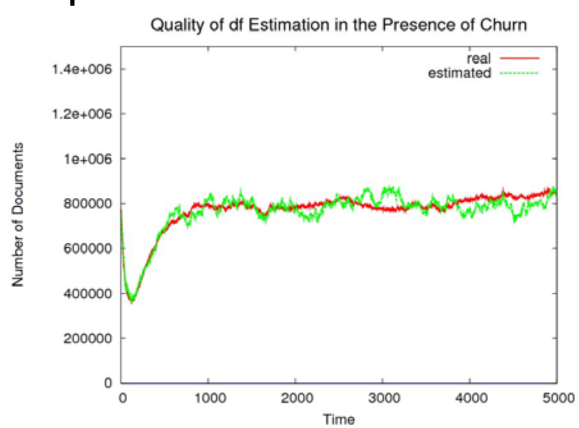




Experiment with churn:

Peers joining and leaving according to Poisson processes





Challenges in QR & P2P Statistics

★ Low-overhead distributed management of statistical synopses with precomputation and composability

? Efficient P2P implementation of KL-based peer selection

? Deeper understanding of precomputation (directory maintenance) vs. online QR, especially in presence of correlation

? Complete framework for composable synopses

? Distributed statistics with strong (probabilistic) guarantees

? Use of order statistics (ranks instead of values)

Outline

✓ **Motivation and Research Directions**

✓ **P2P Query Routing**

- **Peer Content Quality**
- **Overlap Awareness**
- **Challenges in Query Routing & P2P Statistics**

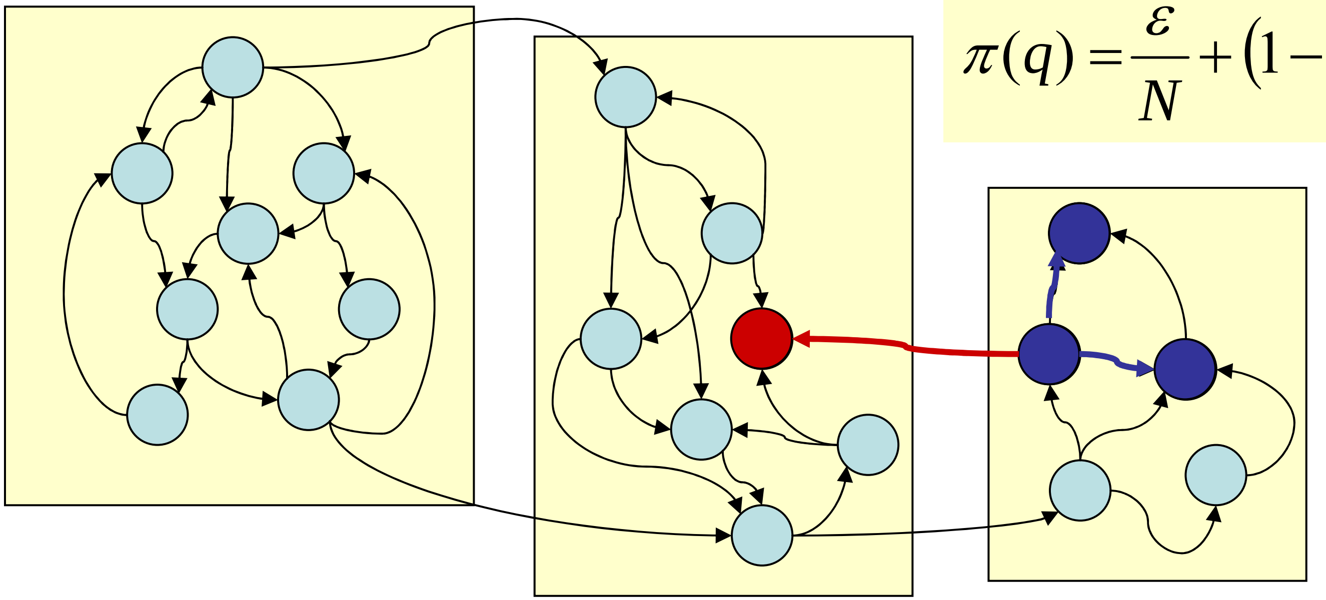
- **P2P Link Analysis**
  - **JXP Authority Scoring**
  - **Challenge: Beyond Link Analysis**
- **Conclusion**

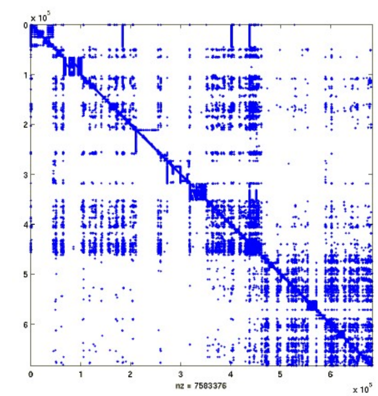
Distributed PageRank (PR)

Page authority important for final result scoring

Exploit locality in Web link graph: construct block structure (disjoint graph partitioning) based on sites or domains

$$\pi(q) = \frac{\varepsilon}{N} + (1-\varepsilon)\sum_{p \in IN(q)} \frac{\pi(p)}{out(p)}$$



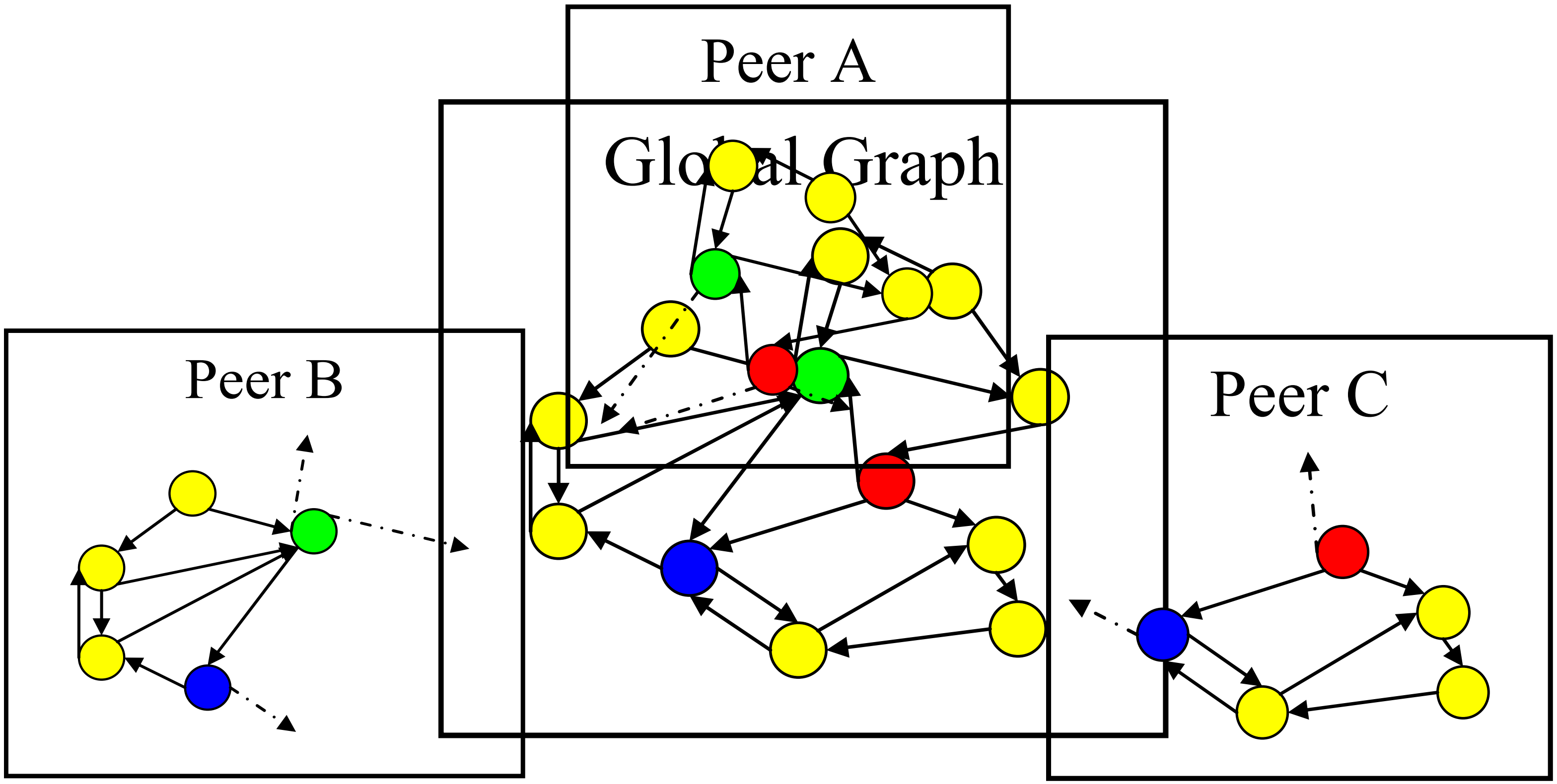


Compute page PR within site/domain & site/domain weights,

- combine page scores with site/domain scores
  [Kamvar03, Lee03, Broder04, Wang04, Wu05] or
- communicate PR mass propagation across sites
  [Abiteboul00, Sankaralingam03, Shi03, Jelasity05]

PageRank (PR) in a P2P Network

Every peer crawls Web fragments at its discretion
and has its own local & personalized search engine
→ **overlaps between peers' graphs** may occur



JXP (Juxtaposed Approximate PageRank)

[J.X. Parreira et al.: WebDB 05, VLDB 06]

based on **Markov-chain aggregation (state lumping)**

each peer represents external, a priori unknown part of the global graph by one superstate, a **„world node“**

peers meet randomly

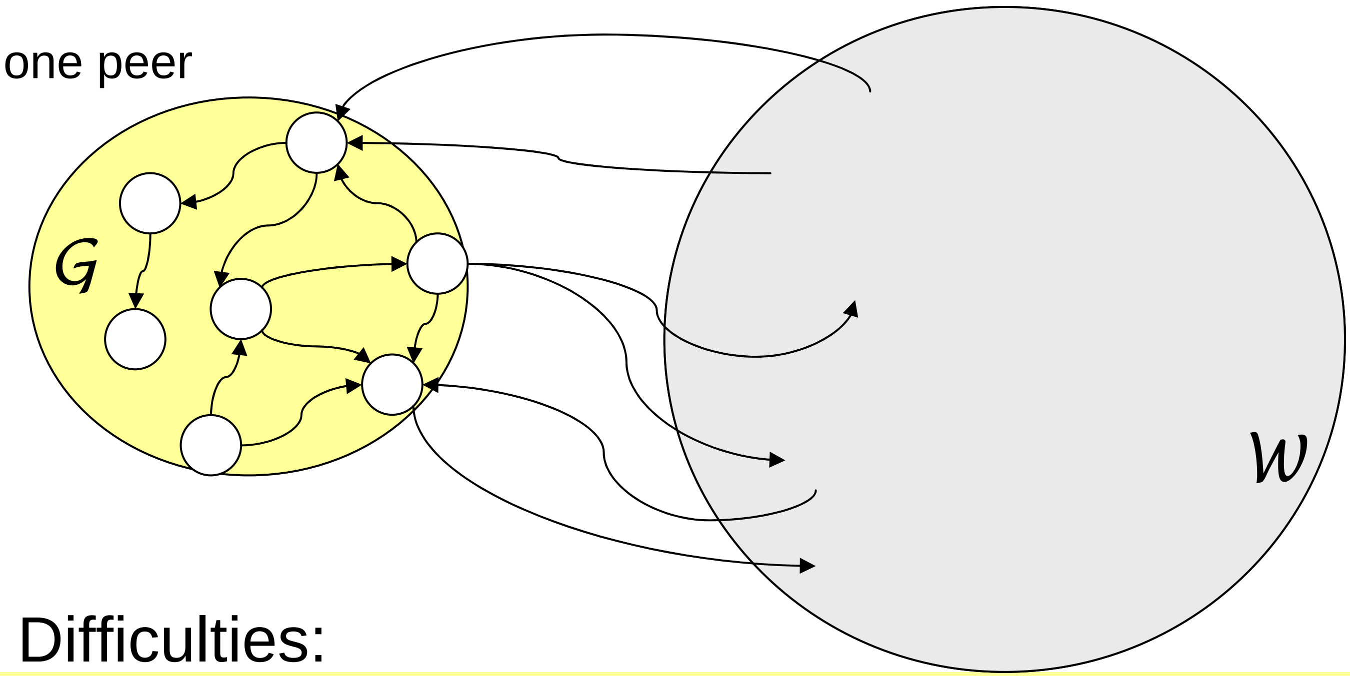
- **exchange their local graph fragments and PR vectors**
- **learn about incoming edges to nodes of local graph**
- **compute local PR on merged graphs or enhanced local graph**
- **keep only improved PR and own local graph**
- **don‘t keep other peers‘ graph fragments**

converges to global PR (experiments + theoretical arguments)

convergence sped up by **biased p2pDating** strategy:
prefer peers whose nodeset of outgoing links
has high overlaps with our nodeset (use MIPs as synopses)

Towards P2P PageRank

Why not straightforward **MC aggregation** ? (state lumping, decomposable MC [Courtois 1977], stochastic complementation [Meyer 1988])
→ assumes knowledge of **complete MC**, aims for speed-up
→ in P2P, each **peer has only limited knowledge !**



Complete Web graph from peer's view:
$U = G \cup W$
$|U| = N$ pages overall

Difficulties:

1) G needs to re-scale local PR values, needs to estimate $\pi(G)$
2) G needs to learn its IN neighbors
3) G needs to estimate the stationary prob's π_p of nodes $p \in WIN(G)$

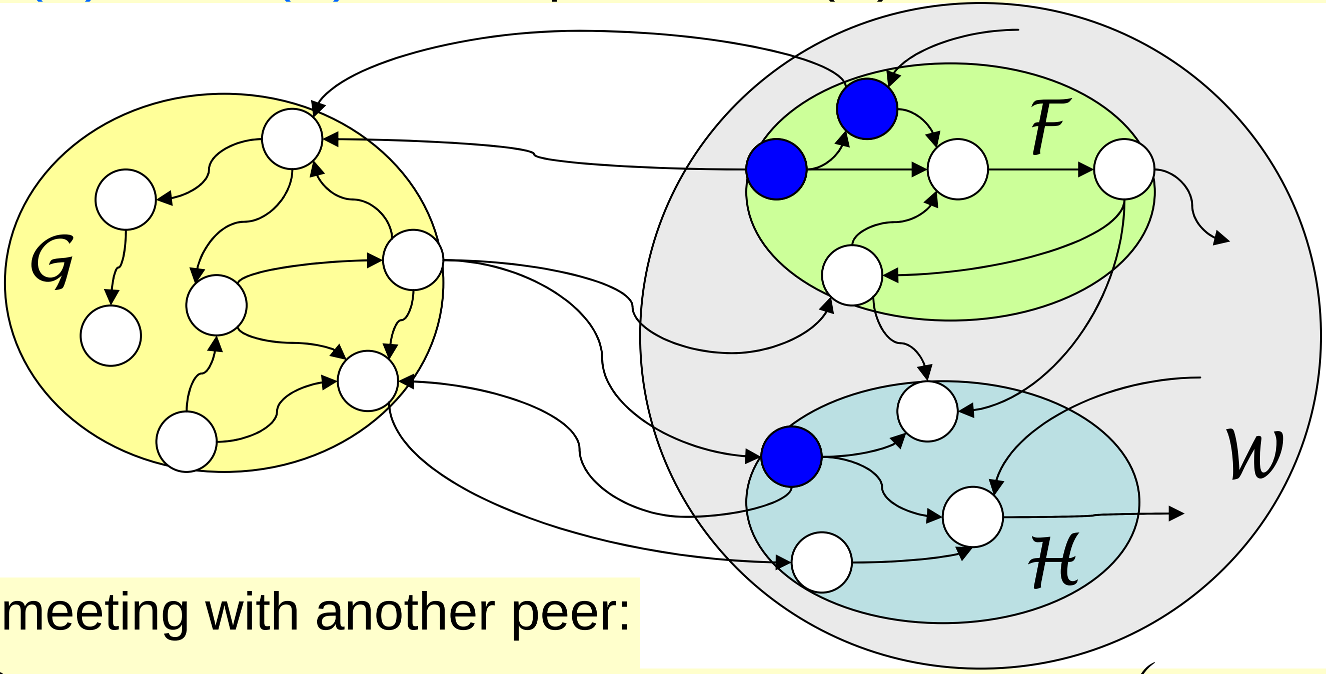
JXP Algorithm

Input:

- G: local graph
- GOUT: {q∈G | q→ s ∧ s∈W}
- n: #pages in G; N: #pages in U = G∪W
- WIN(G): {p∈W | p→ q ∧ q∈G}
- WIN*(G) ⊆ WIN(G): known part of WIN(G)

Output:

- π*(q) for q∈G: est. stationary prob's (PR)
- π*(G) = $\sum_{q \in G}$ π*(q)=1- π*(W) est. total mass of G



At each meeting with another peer:

compute

- for all q∈G: $t_{wq} = \pi^*(W)\ \varepsilon / N\ + \sum_{p \in WIN^*(G)} \left(\frac{\pi^*(p)}{\pi^*(W)} (1-\varepsilon)/out(p) \right)$
- world self-loop: $t_{ww} = 1 - \sum_{q \in G} t_{wq}$

compute all π* values for G∪{w}; remember WIN*(G) info

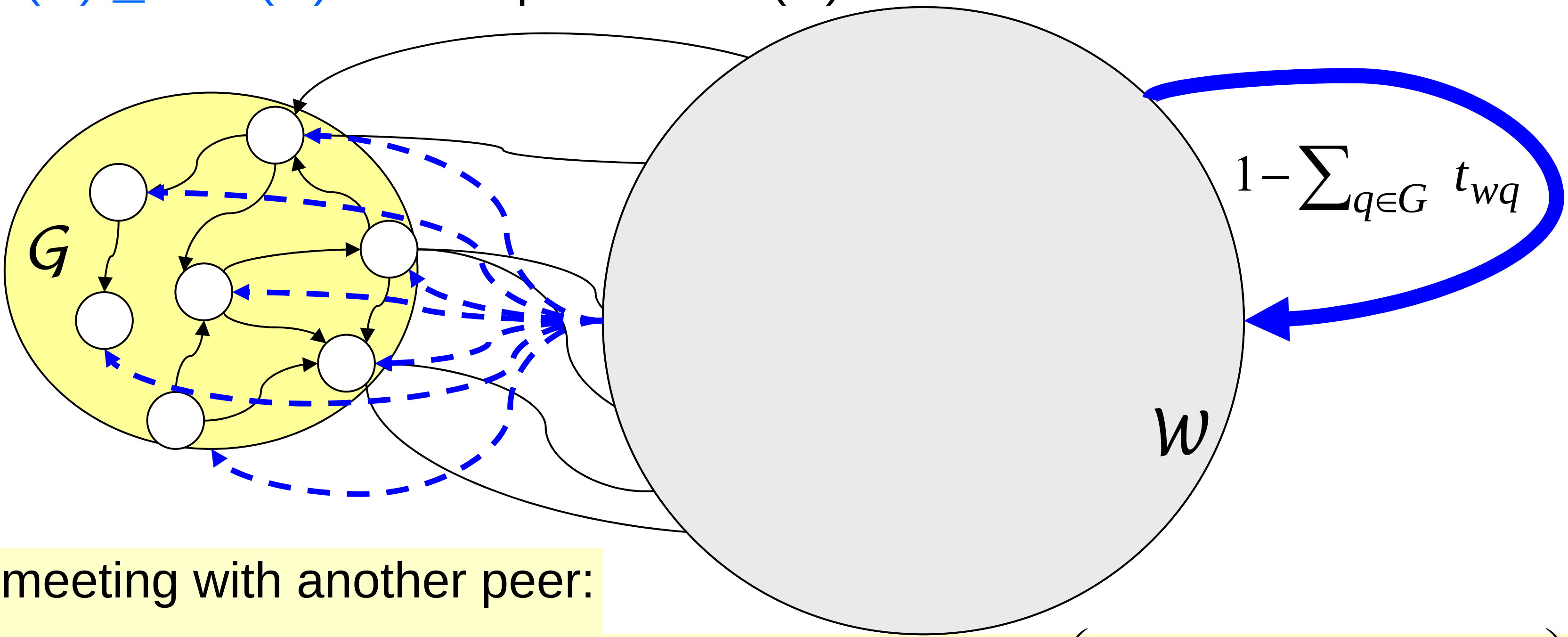
JXP Algorithm: Initial

Input:

- G: local graph
- GOUT: $\{q \in G \mid q \rightarrow s \wedge s \in W\}$
- n: #pages in G; N: #pages in U = G∪W
- WIN(G): $\{p \in W \mid p \rightarrow q \wedge q \in G\}$
- WIN*(G) ⊆ WIN(G): known part of WIN(G)

Output:

- $\pi^*(q)$ for $q \in G$: est. stationary prob's (PR)
- $\pi^*(G) = \sum_{q \in G} \pi^*(q) = 1 - \pi^*(W)$ est. total mass of G



At each meeting with another peer:

compute

- for all $q \in G$: $t_{wq} = \pi^*(W)\ \varepsilon / N\ + \sum_{p \in WIN^*(G)} \left(\frac{\pi^*(p)}{\pi^*(W)} (1-\varepsilon) / out(p) \right)$
- world self-loop: $t_{ww} = 1 - \sum_{q \in G} t_{wq}$

compute all π^* values for G∪{w}; remember WIN*(G) info

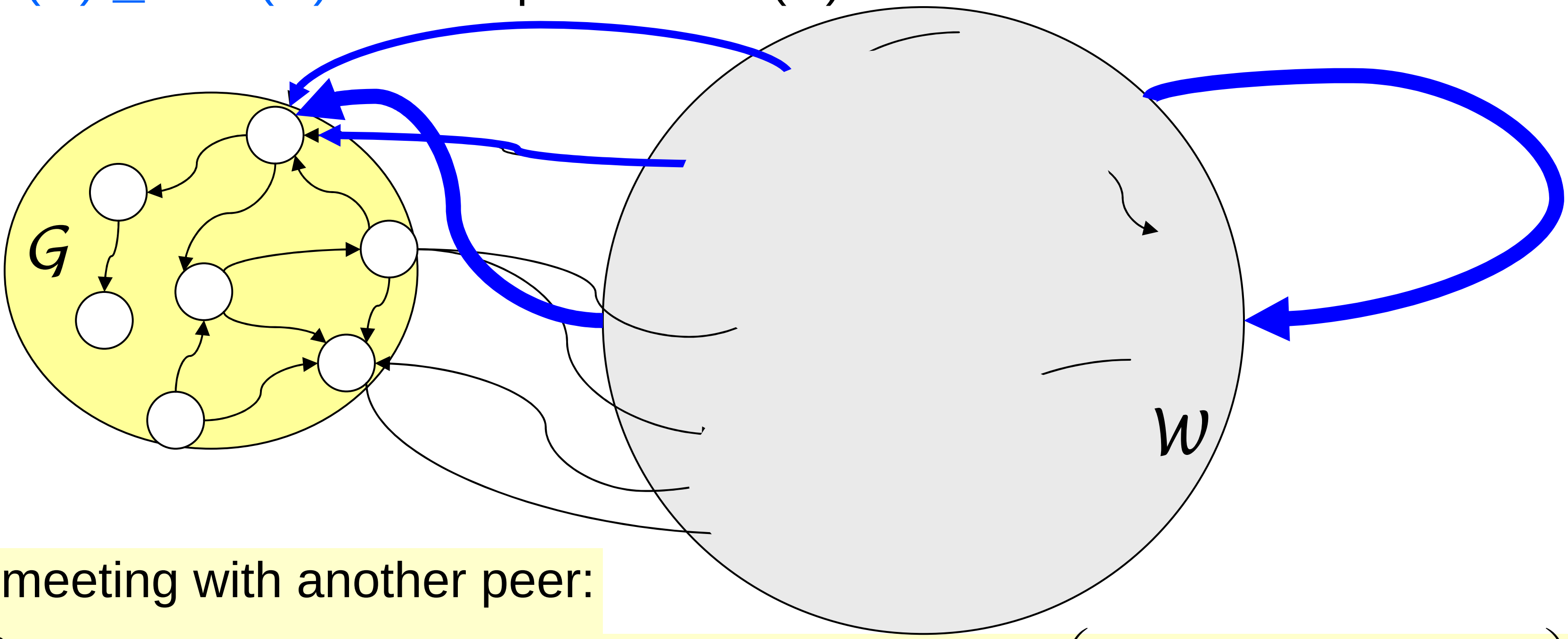
JXP Algorithm: 1st Meeting

Input:

G: local graph
GOUT: {q∈G | q→ s ∧ s∈W}
n: #pages in G; N: #pages in U = G∪W
WIN(G): {p∈W | p→ q ∧ q∈G}
WIN*(G) ⊆ WIN(G): known part of WIN(G)

Output:

π*(q) for q∈G:
est. stationary prob's (PR)
π*(G) = $\sum_{q \in G}$ π*(q)=1- π*(W)
est. total mass of G



At each meeting with another peer:

compute

- for all q∈G: $t_{wq} = \pi^*(W)\ \varepsilon / N\ +\ \sum_{p \in WIN^*(G)} \left(\frac{\pi^*(p)}{\pi^*(W)} (1-\varepsilon) / out(p) \right)$
- world self-loop: $t_{ww} = 1 - \sum_{q \in G}\ t_{wq}$

compute all π* values for G∪{w}; remember WIN*(G) info

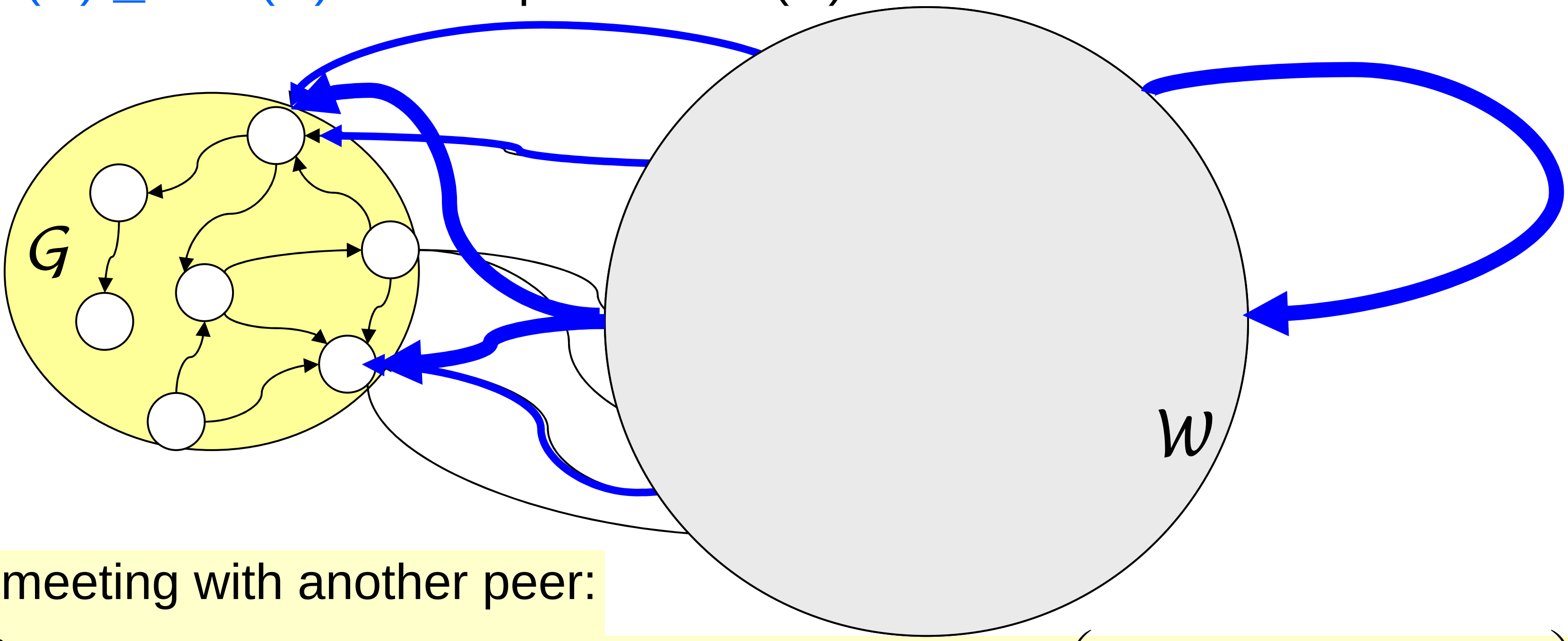
JXP Algorithm: 2nd Meeting

Input:

G: local graph
GOUT: {q∈G | q→ s ∧ s∈W}
n: #pages in G; N: #pages in U = G∪W
WIN(G): {p∈W | p→ q ∧ q∈G}
WIN*(G) ⊆ WIN(G): known part of WIN(G)

Output:

π*(q) for q∈G:
est. stationary prob's (PR)
π*(G) = $\sum_{q \in G}$ π*(q)=1- π*(W)
est. total mass of G

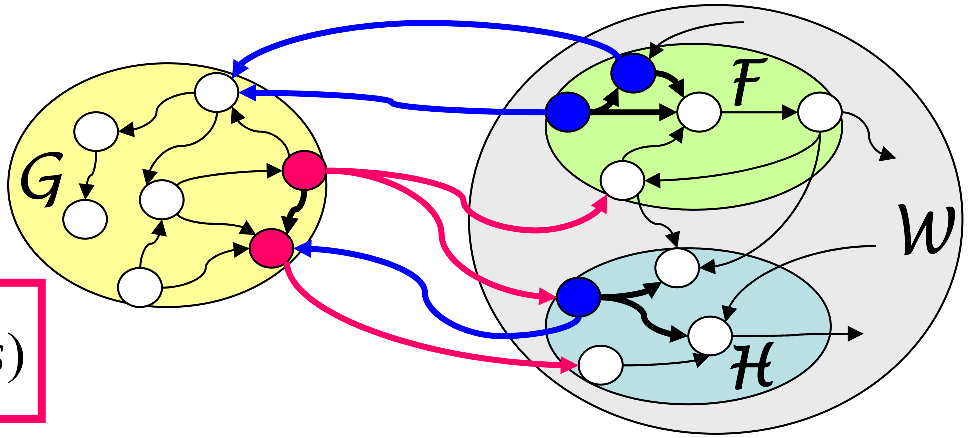


At each meeting with another peer:
compute

- for all q∈G: $t_{wq} = \pi^*(W)\ \varepsilon / N\ +\ \sum_{p \in WIN^*(G)} \left(\frac{\pi^*(p)}{\pi^*(W)} (1-\varepsilon) / out(p) \right)$
- world self-loop: $t_{ww} = 1 - \sum_{q \in G}\ t_{wq}$

compute all π* values for G∪{w}; remember WIN*(G) info

JXP Convergence – Proof Outline



Lemma 1: $\pi(\mathcal{G}) = \sum_{q \in \mathcal{G}} \pi(q) =$

$$\frac{|\mathcal{G}|}{N} + \frac{1-\varepsilon}{\varepsilon} \sum_{p \in WIN(\mathcal{G})} f_p \, \pi(p) - \frac{1-\varepsilon}{\varepsilon} \sum_{s \in GOUT(\mathcal{G})} (1 - f_s) \pi(s)$$

Lemma 2:

For errors ρ_q ($q \in G$), ρ_w in $\pi_q = \pi^*_q + \rho_q$ and $\pi_w = \pi^*_w + \rho_w$: $\sum_{q \in G} \rho_q = -\rho_w$

Lemma 3:

π^*_w and π^*_q (for $q \in G$) are non-increasing & non-decreasing resp.
$|\rho_q|$ and $|\rho_w|$ are decreasing whenever WIN*(G) is updated

Theorem:

With fair peer meetings, JXP scores for $q \in G$ converge to $\pi(q)$

p2pDating

Each peer p_j precomputes two MIPs synopses for

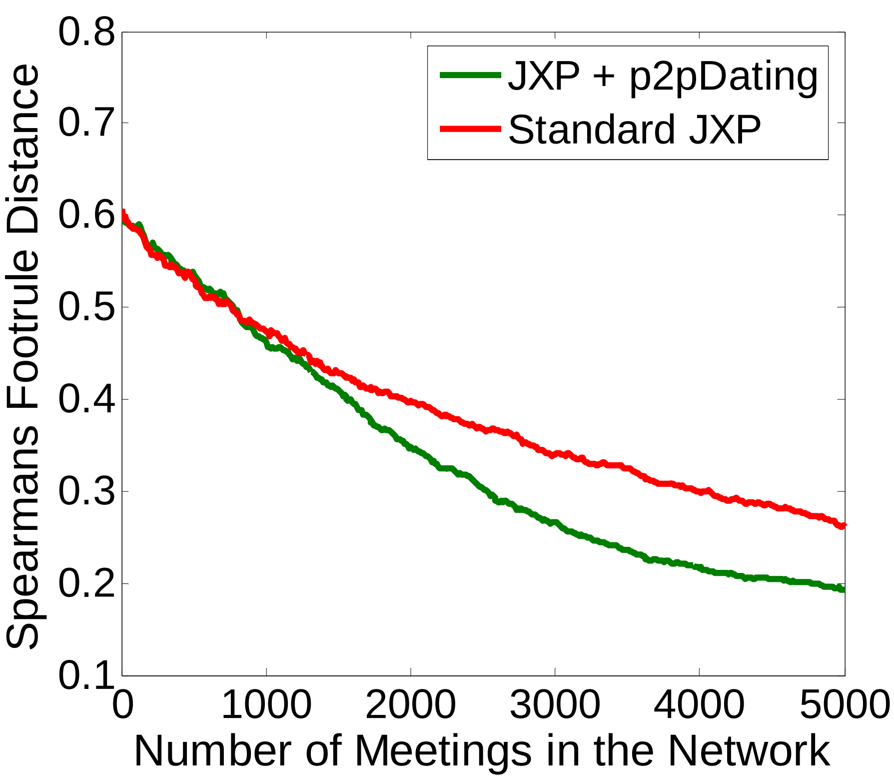
- **M(p_j):** URLs in the collection of p_j (the nodes of G) and
- **O(p_j):** URLs of the out-neighbors of pages of p_j (OUT(G))

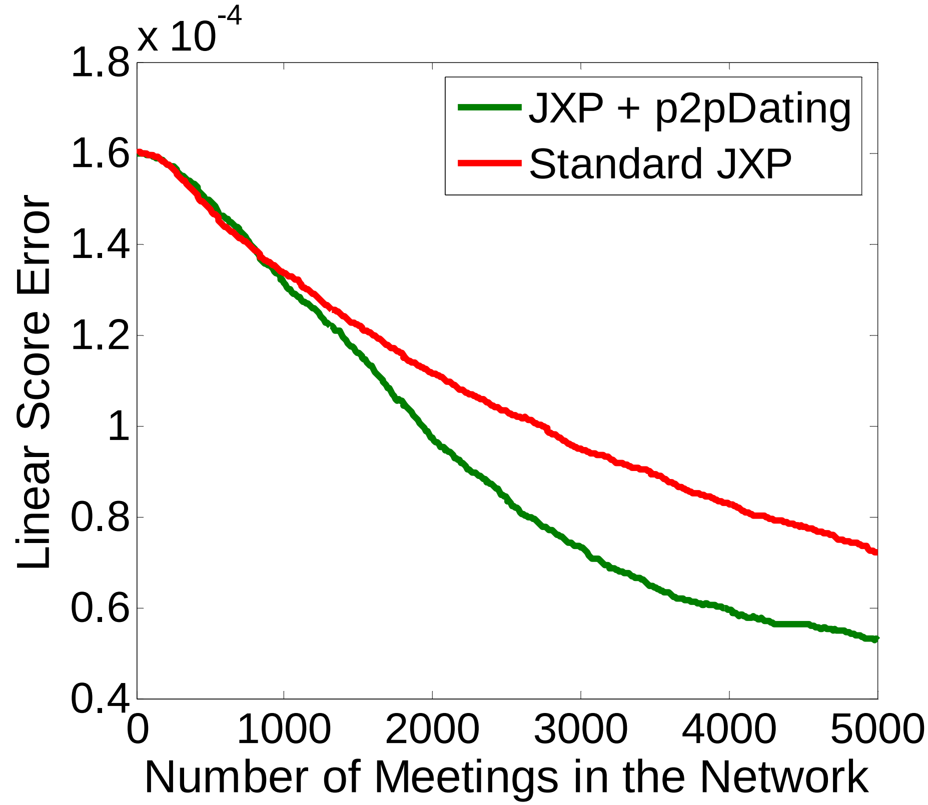
<u>repeat</u> <u>forever</u>

- peer p_j randomly picks „blind date“ candidate p_d:
  - p_j and p_k exchange their **O synopses**;
  - they may also recommend to each other a set of friends p_f and pass on their O synopses
- peer p_j maintains a list of dating candidates p_c ordered by **resemblance (M(p_j), O(p_c))**
- peer p_j chooses best candidate for next date (exchange of graphs, local PR computation, etc.)

JXP Experiments

100 peers with simulated crawls of **Amazon products** categories
(with recommended similar products as links)





similar and more results for real **Web data**

Link Analysis: Great Idea Whose Time Has Passed?

Links have been viewed as endorsements, but now:
- professional Web pages dynamically generated (incl. links)
- no more passion for „my favorite links" on personal home pages
- preferential-attachment model no longer valid

Nonetheless:
- recommendations, annotations & opinions in textual form
- Web still a wealth of „collective wisdom"

→ Employ large-scale text mining (NER etc.) for „soft links"
→ Combine link mining with query-log & click-stream analysis
→ Consider dynamic evolution of Web, social networks, etc. in generalized „link" analysis

Outline

✓ **Motivation and Research Directions**

✓ **P2P Query Routing**

- **Peer Content Quality**
- **Overlap Awareness**
- **Challenges in Query Routing & P2P Statistics**

✓ **P2P Link Analysis**

- **JXP Authority Scoring**
- **Challenge: Beyond Link Analysis**

- **Conclusion**

Conclusion: Challenges Remain Open

- **Distributed Statistics Management**
  - **Key to Query Routing, Quality/Overlap Estimation, Ranking (PR etc.)**
  - **Capturing Global Statistics in Decentralized Manner**
  - **Efficiently Disseminating Statistical Synopses**
- **Robustness to Churn and Cheating**
- **„Statistically Semantic" Overlay Networks**
- **Experimental Evaluation**
  - **Benchmarking Methodology**
  - **Large-scale P2P Testbed**
  - **Capturing User/Community Behavior**