"Data Dissemination in Mobile Peer-to-Peer Networks"

Thomas Repantis, Vana Kalogeraki
Dept. of Computer Science and Engineering
University of California, Riverside

trep@cs.ucr.edu, vana@cs.ucr.edu
http://www.cs.ucr.edu/~vana

UNIVERSITY OF CALIFORNIA RIVERSIDE

Outline

1. Introduction and Background
2. System Description
   - Bloom Filters
   - Content-Driven Routing
   - Adaptive Data Dissemination Strategies
     - Immediate Local
     - Adaptive Local
     - Adaptive Local Remote
3. Experimental Evaluation
4. Conclusions and Future Work

Mobile Ad-hoc Networks

- Formed by mobile devices (e.g. laptops, PDAs, cellphones) with limited communication power and transmission range
- Devices communicate through various communication protocols in a peer-to-peer fashion
- Networks are ad hoc, unstructured, dynamic and self-organizing
- Mobile nodes that are in the transmission range can communicate directly
- Otherwise, messages may need to be propagated across multiple hops in the network

Mobile Applications

Users of a mobile network may run different queries:

- Find other users nearby with certain interests (e.g. tennis-fans)
- Be notified about local events pertinent to their interests (e.g. traffic jams)
- Explore and use services offered in their local environment by specifying meta-data attributes (e.g. cheap hotels within 10mile range)
- Retrieve media data available in their vicinity (e.g. movies, music files)

Challenges in Data Retrieval

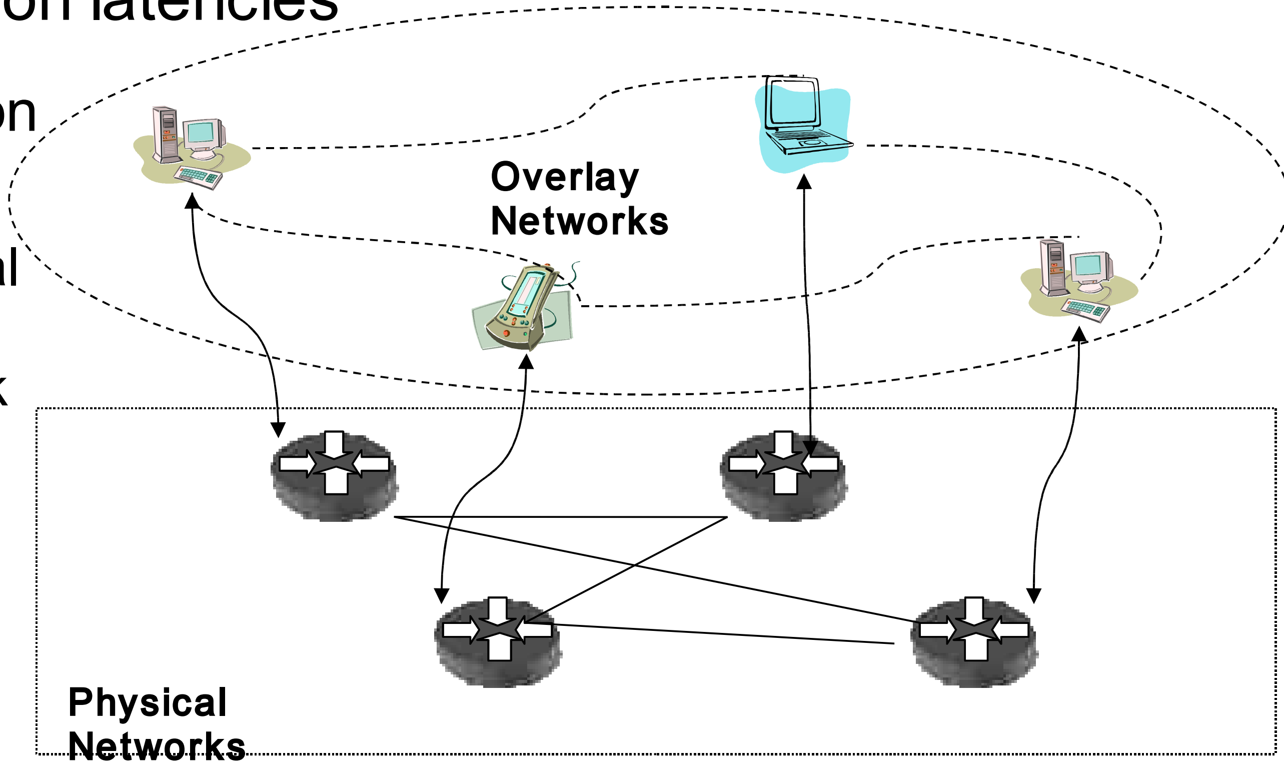
How to efficiently retrieve data objects in a mobile environment?

- Transient nature of mobile connections
- Heterogeneity of mobile devices
- Low processing and communication capabilities
- Power restrictions (e.g., run out of batteries)
- Amount of messages (event notifications, service advertisements, forwarded queries) can be overwhelming

Peer-to-Peer (P2P) Model

- The P2P Computing paradigm became a powerful model for developing large-scale systems
  - Create virtual networks of heterogeneous processors with different resource capabilities and variable communication latencies

- Resource aggregation
- Improved scalability
  - No need for central coordinator, self-organizing network
- Increased autonomy
- Improved reliability
  - No single point of failure



UNIVERSITY OF CALIFORNIA RIVERSIDE

Vana Kalogeraki

Peer-to-Peer Applications

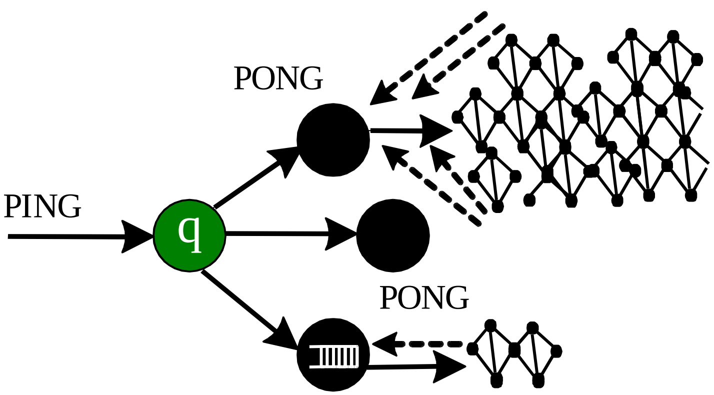
- The P2P Model found fascinating applications in **Internet-Scale Systems** which rely on the computing power of individual nodes:
  - **File-sharing** (e.g. Kazaa, Gnutella, Chord, Oceanstore..)
  - **Web Caching** (e.g. SQUIRREL based on Pastry)
  - **Distributed Computing** (Seti@, Folding@)
  - **Online Gaming** (e.g. Sony Playstation)
  - **Spam Detection Networks** (e.g. SpamNet)

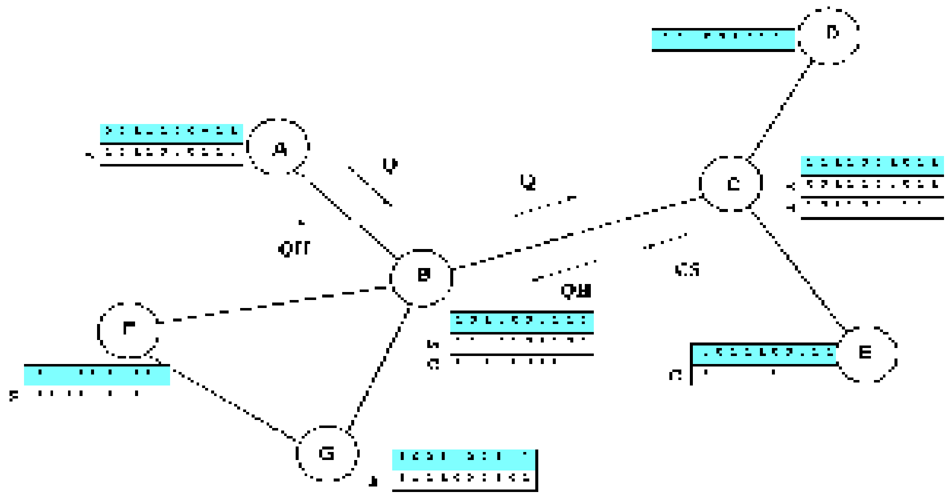
Problem Statement

- How to build a system to locate content in a fast and efficient manner in a mobile P2P network
- Our approach:
  - Adaptive data dissemination and content-driven routing to intelligently route queries in the mobile P2P environment
  - The goal is to reach the users with the content while keeping the number of propagated messages small

Construction of P2P Overlay

- Each node has a unique global id and is responsible for keeping a small number of connections to other peers
  - The network is unstructured and self-organizing; peers join the overlay by establishing a connection with any peer in the overlay; no explicit membership is required
  - Number of connections depends on available resources
  - Each node monitors the type and number of messages sent by its neighbors and builds connection profiles for its neighbors



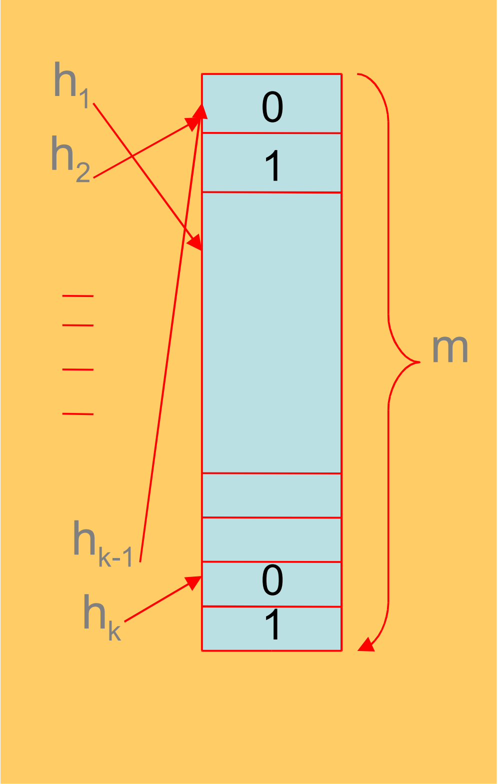


System Model

- Each peer p has n objects $S_p = a_1, a_2 .... a_n$
- A Bloom Filter BF_p of length m is used to represent the set of objects S_p
  - We assume k hash functions $h_1(s), h_2(s), .. h_k(s)$
  - Reduce memory and bandwidth demands – depends on the Bloom filter size
  - Tradeoff between consistency and resource usage
- Each peer stores 2 types of Bloom Filters: a *local filter* for objects available locally at the peer and *remote filters* for objects stored in remote peers

Bloom Filters

- An efficient, lossy way of describing sets of data
- A Bloom Filter data structure represents the set of data by using an array of bits; each bit takes a binary one or a zero value
- To check if an object is in the set
  - Object is hashed
  - Corresponding bits of the filter are checked
    - If at least one of the bits is not set, object not in the set
    - If all bits are set, object may be in the set (false positive)

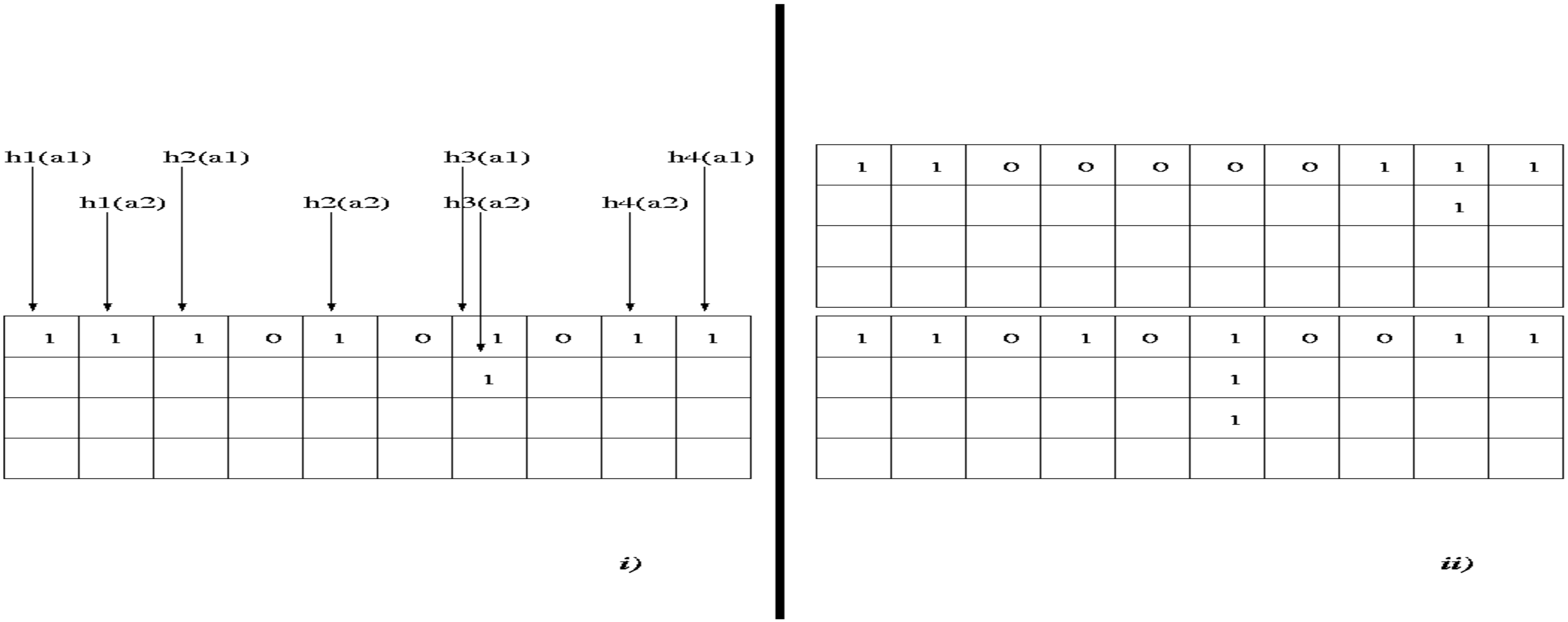


$$p_{err} = (1 - p_0)^k = \left(1 - \left(1 - \frac{1}{m}\right)^{kn}\right)^k \approx \left(1 - e^{-\frac{kn}{m}}\right)^k$$

$$(1/2)^k = (0.6185)^{\frac{m}{n}}$$

n: number of objects,
m: Bloom Filter width
k: number of hash functions

Counting and Multi-level Bloom Filters



- Counters keep track of the number of objects corresponding to each bit of the filter
- Multi-level bloom filters: the filter of each level is appended to that of the previous level

Content-Driven Routing

- Each peer stores its own content synopses and also the content synopses of other peers
- When a peer receives a query, it performs a local search
- If results are not found, then it searches the contents of its peers
- The results are ranked based on the goodness of the comparison
- The query message is propagated to those peers with the highest ranks
- These are the peers with highest probability of containing the object
- Otherwise, the query is forwarded to a random set of immediate peers

Adaptive Dissemination Strategies

Nodes employ adaptive content dissemination strategies to propagate content synopses to other peers

- IL: propagates only its local synopsis to all peers one hop away.
- AL: propagates its local synopsis to selected immediate and remote peers.
- ALR: propagates both its local and stored remote synopses to selected immediate and remote peers.

Disseminate Local Content Synopsis to Immediate Peers (IL)

- Each peer sends its local content synopsis to all its immediate peers
- The queries are routed based on the content synopses of the immediate peers
- If no matches are found, the protocol resorts to randomly choosing peers
- Strategy is simple, but of limited use; only a small number of content synopses is examined

Disseminate Local Content Synopsis to Peers Selected Adaptively (AL)

- Each peer sends its local content synopsis to a selection of peers
  - The recipients of content synopses are among immediate and remote peers whose queries have been answered in the past
- The queries are routed based on the local content synopses of peers
- Local content synopses not only of immediate peers, but also of peers one has previously interacted with
- This strategy aims to make content synopses available to peers that have high probability of using them in the future

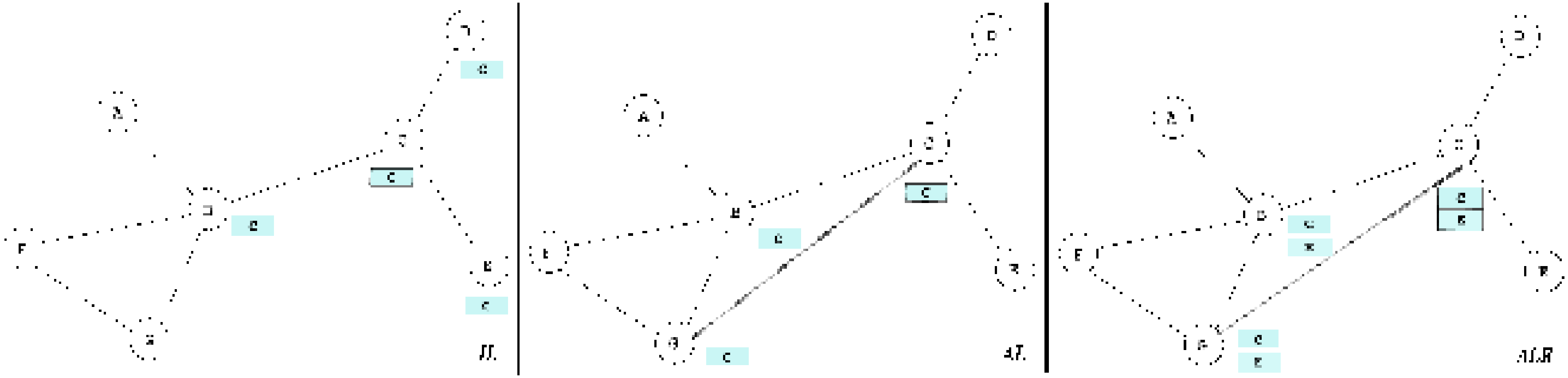
Disseminate Local and Remote Content Synopses to Peers Selected Adaptively (ALR)

- Each peer sends its local content synopsis together with remote content synopses it has stored to a selection of peers
- The queries are routed based on both local and remote content synopses
- This strategy enables the peers to examine more content synopses before routing queries
- The amount of information transferred between nodes is higher than the previous strategies

Adaptive Synopses Dissemination Parameters

- The AL and ALR dissemination strategies take into account the following parameters when disseminating the synopses
  - The number of queries a node has received by a peer, and their frequency (indication of peers that route a lot of traffic)
  - The number of replies a node has provided a peer with, and their frequency (indication of the popularity of the stored objects among specific peers)

Synopses Dissemination Strategies



- IL: C propagates only its local synopsis to all peers one hop away.
- AL: C propagates its local synopsis to selected immediate and remote peers.
- ALR: C propagates both its local and stored remote synopses to selected immediate and remote peers.

Implications of Dynamic Behavior

- In a mobile P2P system, nodes may disconnect or join independently

  Thus, content synopses must be updated when:

  – The content changes (objects inserted, or deleted or modified)

  – Connections are established or dropped.

- Peers can be static or dynamic

  – Dynamic (highly mobile) peers pull the content synopses of their current neighbors

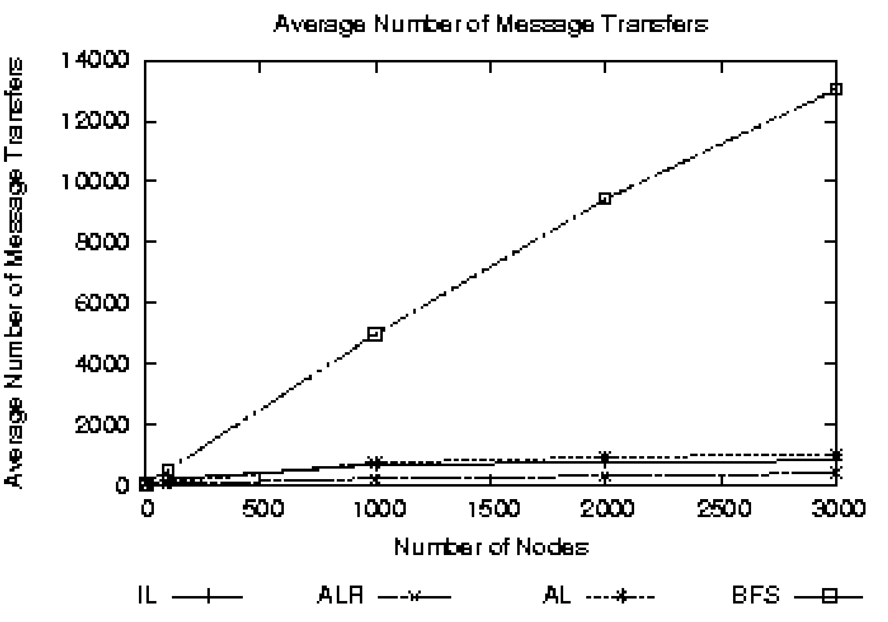
  – Neighbors of static peers push their content synopses to them

Simulation Setup

- We implemented our protocols on top of Gnutella using the Neurogrid simulator.
- 3000 nodes, 2000 total objects, objects replicated on the nodes
- Random network topology, uniform distribution of objects over nodes
- 10bits Bloom filter size, 4 hash (SHA-1) functions
- We compared IL, AL, ALR, and BFS (Breadth-First-Search)

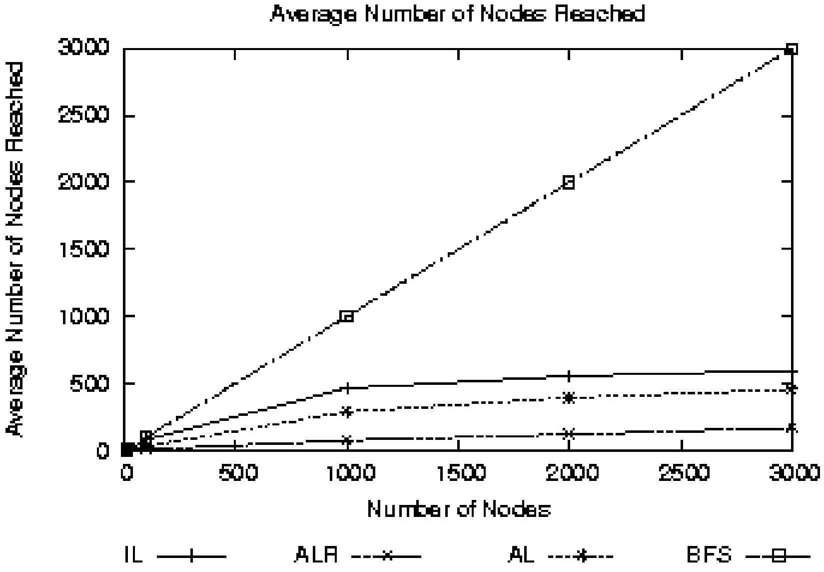
A) Average Message Transfers during Search

- Content-driven routing drastically decreases the number of query messages transferred during a search.
- In BFS, as the number of nodes increases, the number of message transfers grows dramatically.
- ALR achieves the minimum number of message transfers.



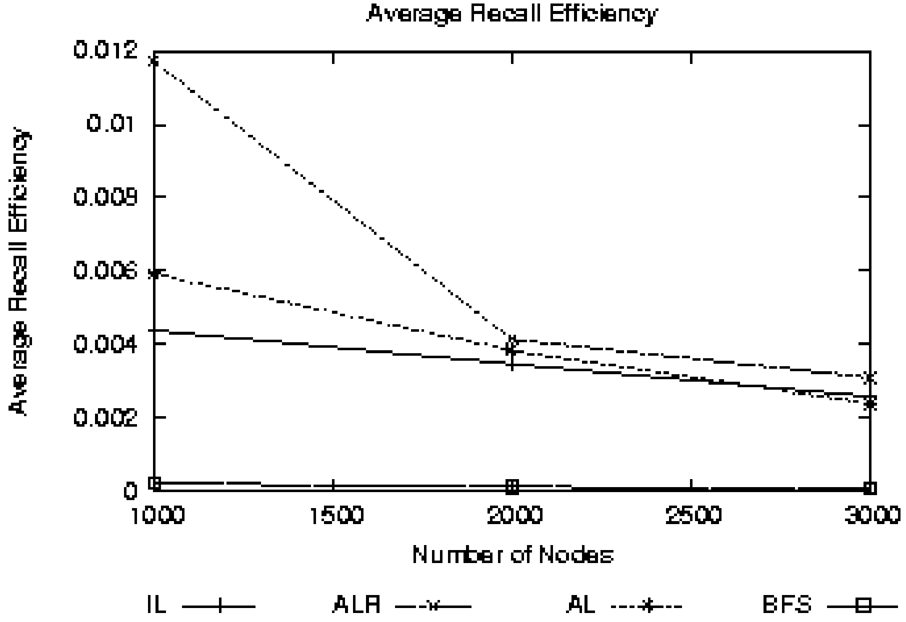
B) Average Number of Nodes Reached

- Content-driven routing techniques are able to provide query hits by contacting more than one order of magnitude less peers.
- The adaptive AL and ALR techniques guide queries more efficiently.



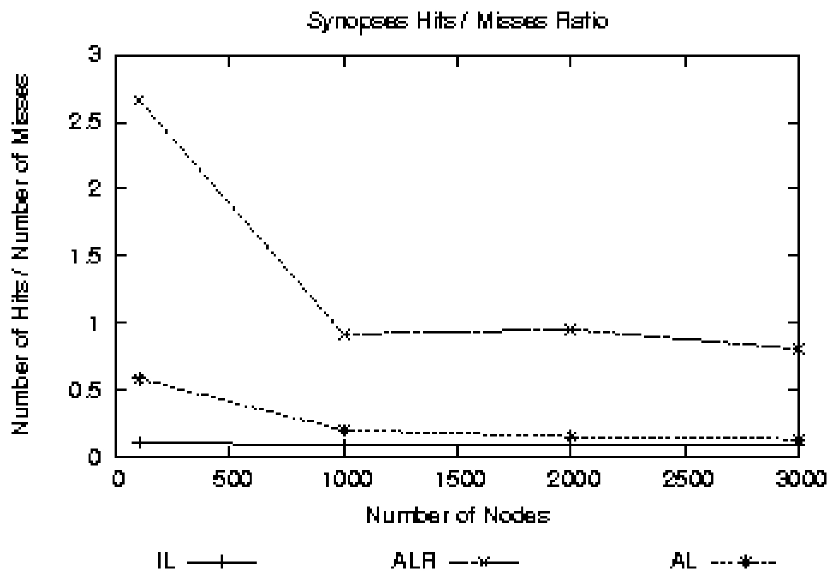
C) Average Recall Efficiency during Search

- Although the flooding of BFS is able to discover a lot of matches, the cost of query messages transferred results in low recall efficiency.
- ALR results in the highest recall efficiency.
- As the number of nodes grows, we notice that the proportion of the total matches discovered by the content-driven routing mechanisms decreases.



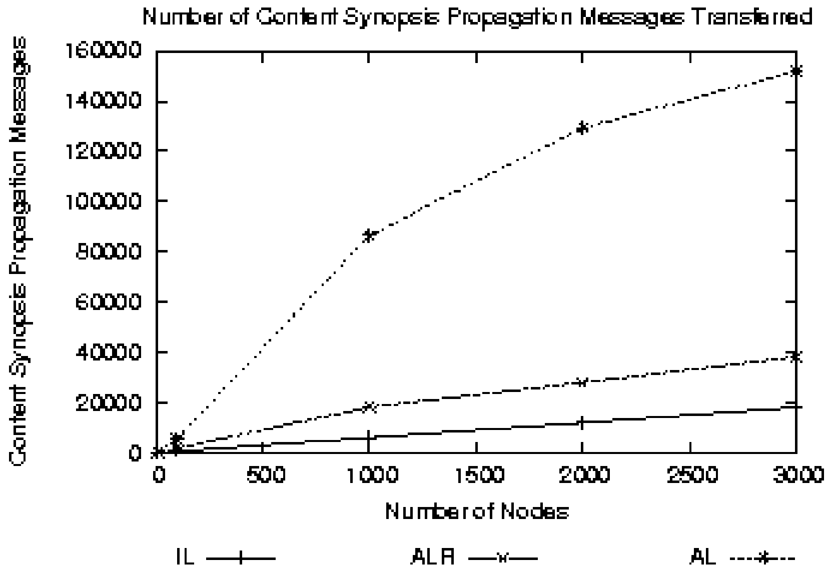
D) Content Synopses Hits over Misses

- Adaptively placing content synopses (AL and ALR) improves their usefulness.
- ALR is able to route approximately half of the queries based on the content synopses.

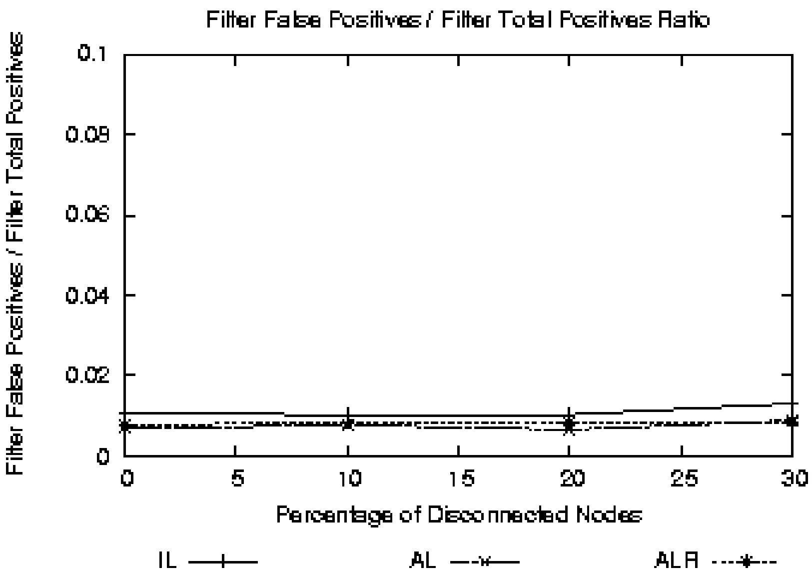


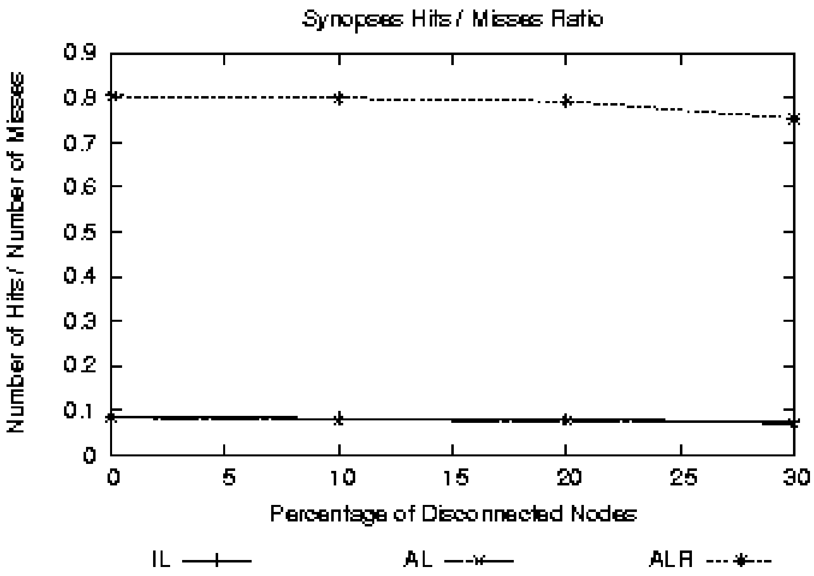
F) Total Content Synopses Messages

- By disseminating content synopses messages only to immediate peers IL keeps the overhead small.
- AL has to disseminate a lot of content synopses to be useful.
- ALR manages to route queries effectively and yet keep the overhead relatively small, even as the number of nodes increases.



G) Experiments on a Mobile Environment





- Content-driven routing remains useful even when peers disconnect. The immediate peers realize the disconnection and update their summaries.
- Peer disconnections do not affect the success of the synopses in query routing.

Conclusions and Future Work

- We have presented mechanisms for:
  - Content-driven routing and adaptive data dissemination, so that queries are forwarded where they will most probably be answered
  - Our mechanisms are scalable and make efficient usage of resources
- We plan to investigate more adaptive schemes for data dissemination
- Predict future locations based on the moving patterns of the users and intelligently propagate the queries
- Experimentally evaluate the mechanisms

“Data Dissemination in Mobile Peer-to-Peer Networks”

Thomas Repantis, Vana Kalogeraki
University of California, Riverside

trep@cs.ucr.edu, vana@cs.ucr.edu
http://www.cs.ucr.edu/~vana

Probabilistic Local Search

- Each node maintains a number of peers
- If the query cannot be satisfied locally, route the query to the most likely peer that has the highest probability to answer it
- This function in implemented by using attenuated Bloom filters

Related Work

- Mobile ad hoc networks: 7DS, Proem
- File Systems: Oceanstore, Squirrel, Pastry, Chord, Napster, Freenet, Gnutella
- Searching and data dissemination through gossipping in P2P networks: Planet-P, P-Grid, CUP
- Content Summarization: SummaryCache, Breadth and Depth Bloom Filters

E) False Positives over Total Positives

- Content-driven routing is extremely accurate.
- Only a very small percentage (around 1%) of the total queries is falsely routed.