iCluster: a Self-Organising Overlay Network for P2P Information Retrieval

Paraskevi Raftopoulou,
Euripides G.M. Petrakis

Dept. of Electronic and Computer Engineering
Technical University of Crete (TUC)
Chania, Crete, Greece

ECIR 2008

Motivation and Objectives

- Information sharing in P2P networks can be slow or inaccurate
  - large & distributed data collections
  - random overlay networks & blind query forwarding
- We want to provide real-time, low-overhead, dynamic sharing of available information
- Query processing can be speeded-up by restricting the search to peers similar to the query

Outline

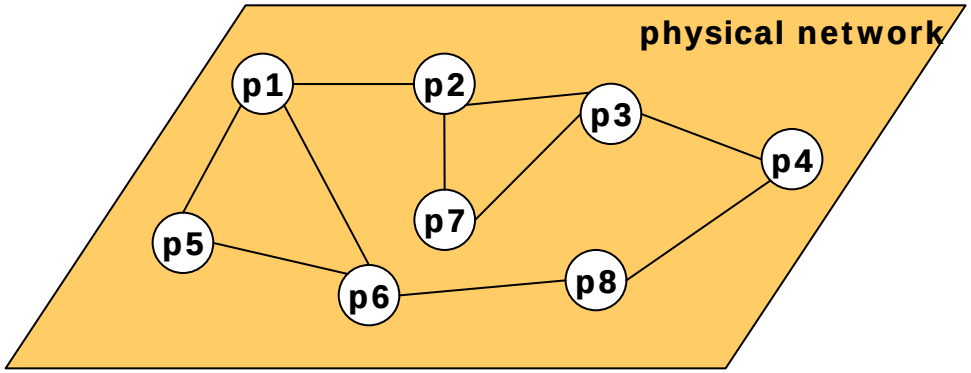
- Background
  - Semantic overlay networks
  - Rewiring strategies
- iCluster
  - Overview
  - Protocols
  - Evaluation
- Extensions

P2P Indexing

- Distributed Hash Tables (DHTs)
  - ✓ provide fast lookup mechanisms
  - × lay aside peer autonomy
  - × strictly coupled with the data model and the query language
- Semantic Overlay Networks (SONs)
  - ✓ support peer autonomy
  - ✓ loose coupling of peers
  - × specialisation assumption

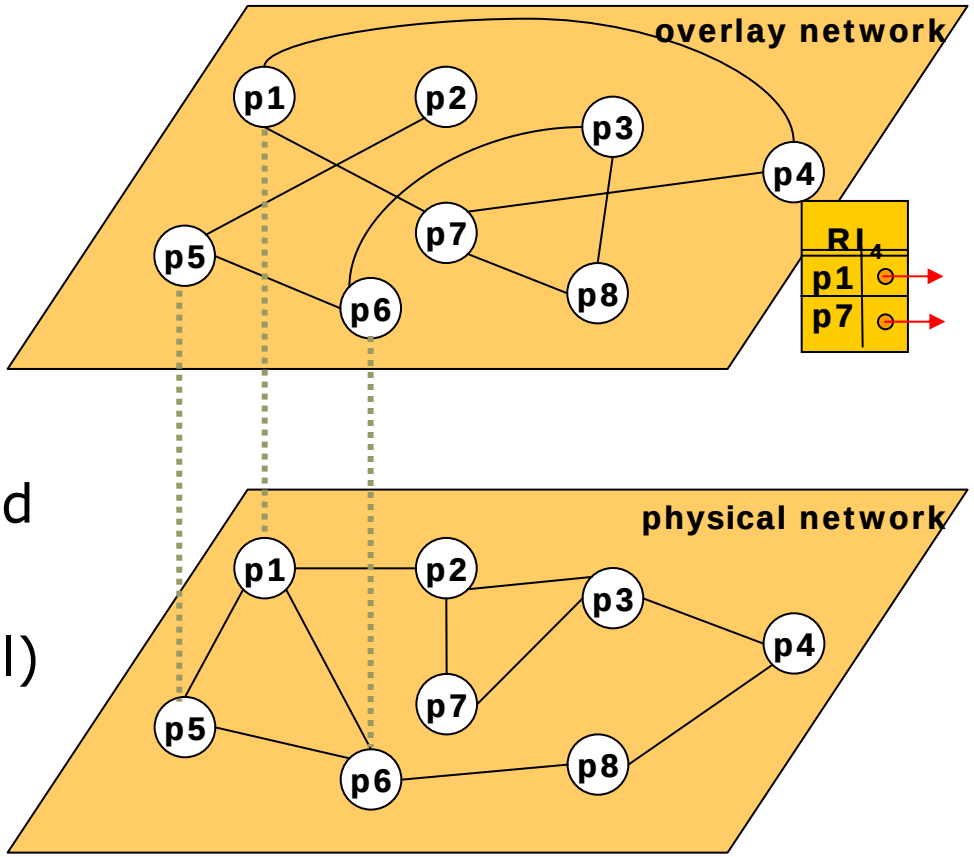
Semantic Overlay Networks

- virtually connected peers based on their content
- routing indices with links to other peers
- peers connected to each other are called neighbors
- support rich data models and expressive query languages
- provide semantic (and social) information about peers



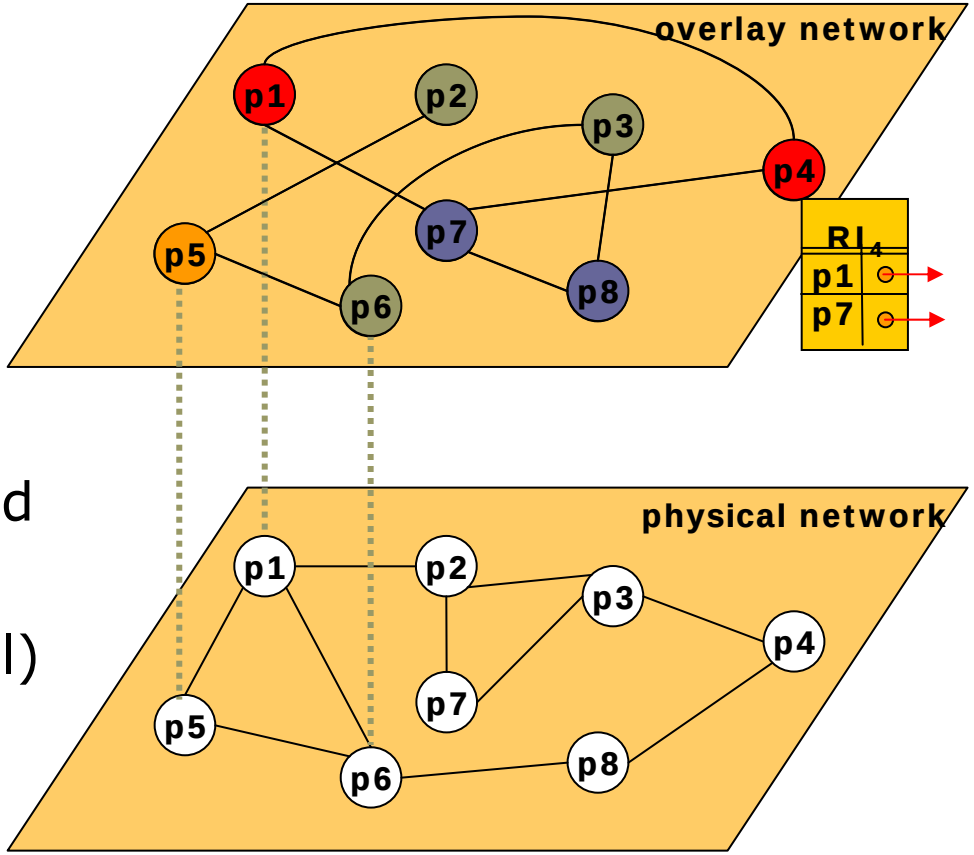
Semantic Overlay Networks

- virtually connected peers based on their content
- routing indices with links to other peers
- peers connected to each other are called neighbors
- support rich data models and expressive query languages
- provide semantic (and social) information about peers



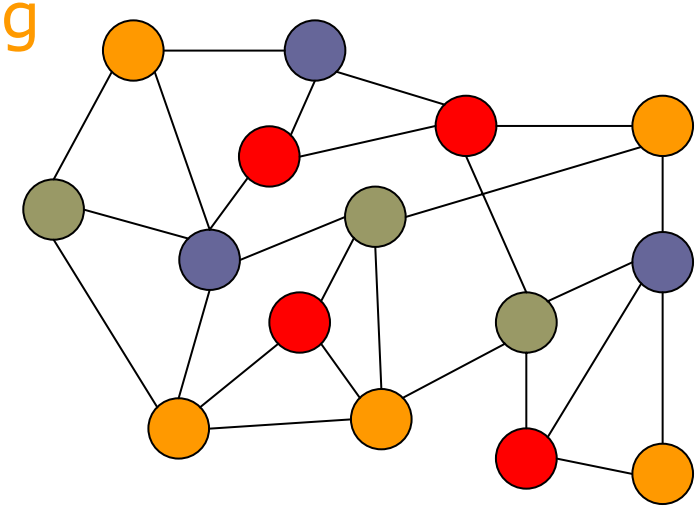
Semantic Overlay Networks

- virtually connected peers based on their content
- routing indices with links to other peers
- peers connected to each other are called neighbors
- support rich data models and expressive query languages
- provide semantic (and social) information about peers



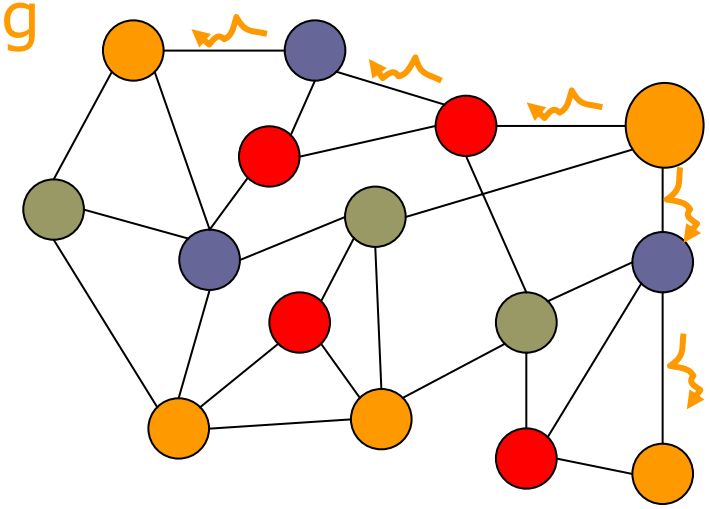
Rewiring strategies

- Techniques for self-organising peers:
  - abandon old connections and create new ones
  - periodic process
- Inspired by the 'small world effect'
  - reach anybody in a small number of routing hops



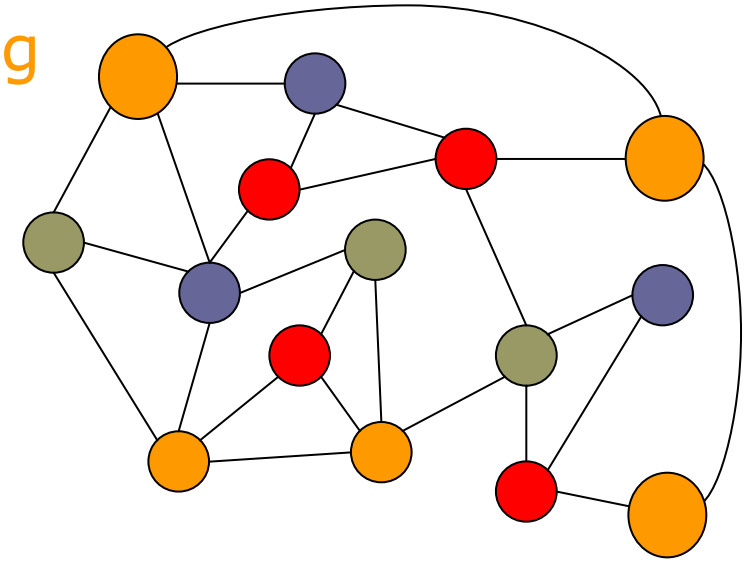
Rewiring strategies

- Techniques for self-organising peers:
  - abandon old connections and create new ones
  - periodic process
- Inspired by the 'small world effect'
  - reach anybody in a small number of routing hops



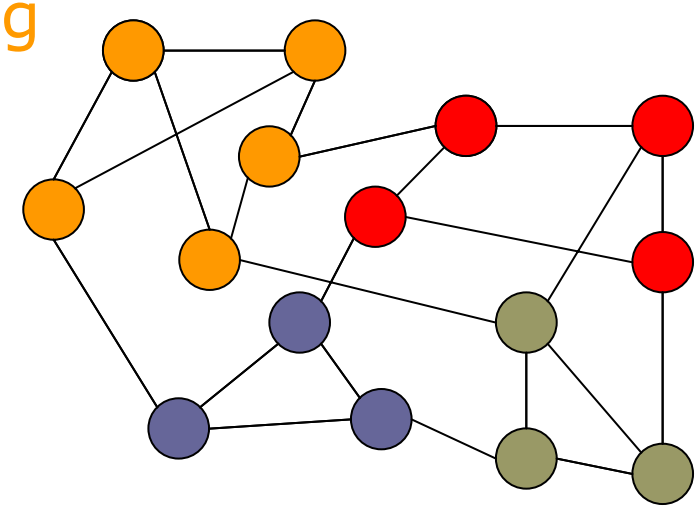
Rewiring strategies

- Techniques for self-organising peers:
  - abandon old connections and create new ones
  - periodic process
- Inspired by the 'small world effect'
  - reach anybody in a small number of routing hops



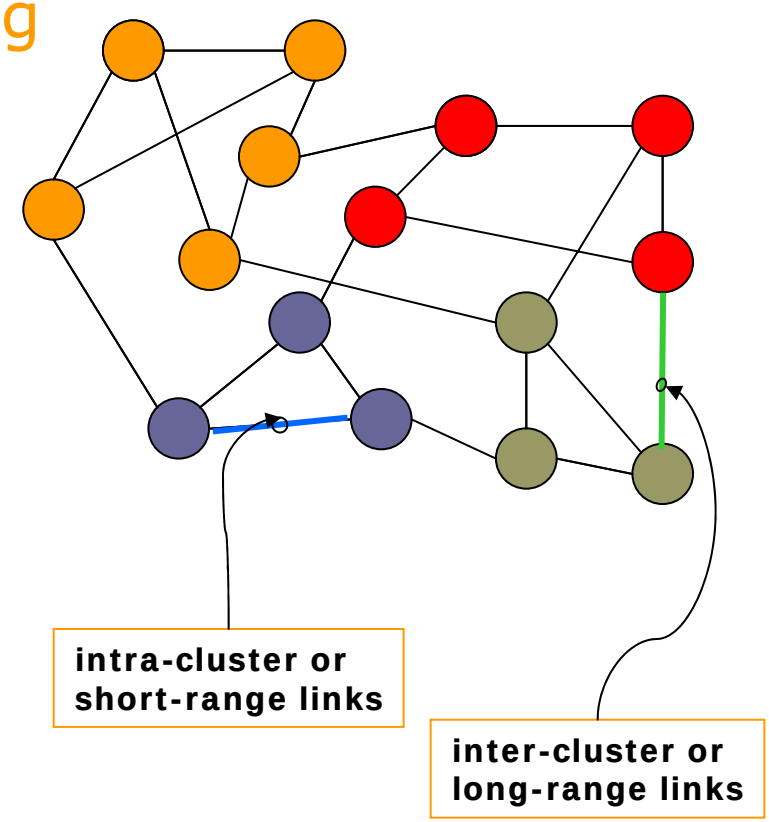
Rewiring strategies

- Techniques for self-organising peers:
  - abandon old connections and create new ones
  - periodic process
- Inspired by the 'small world effect'
  - reach anybody in a small number of routing hops



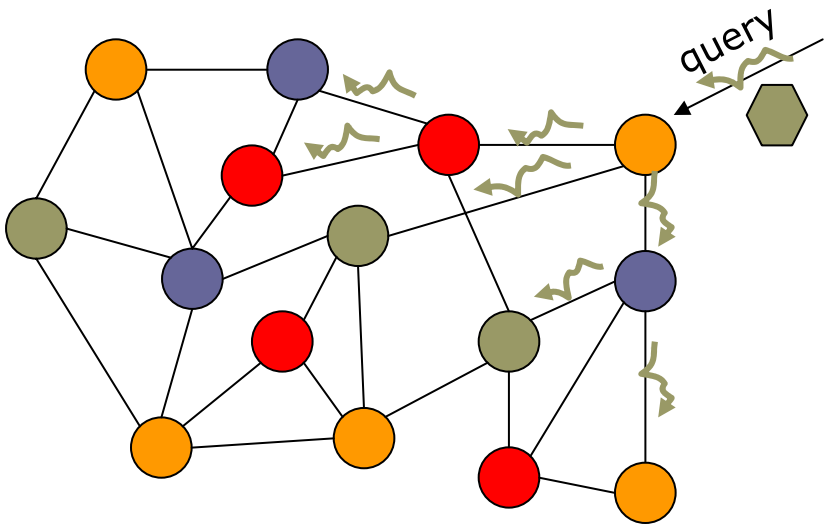
Rewiring strategies

- Techniques for self-organising peers:
  - abandon old connections and create new ones
  - periodic process
- Inspired by the 'small world effect'
  - reach anybody in a small number of routing hops



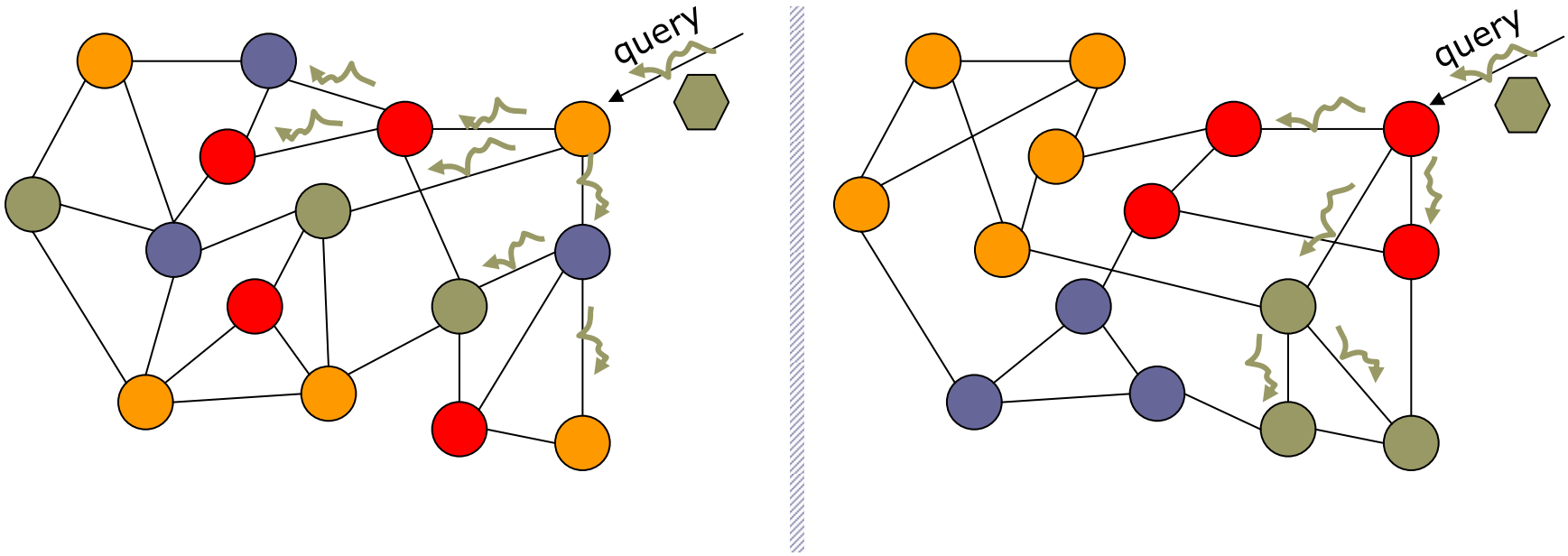
Searching

- Queries are propagated to the regions of the network having more chances to match the query



Searching

- Queries are propagated to the regions of the network having more chances to match the query



What has been proposed?

- SONs created using
  - similar documents [Klampanos et al. 04]
  - concepts from ontologies [Schmitz 04]
- Hierarchies of clusters [Doulkeridis et al. 06]
  - bottlenecks at cluster gateways
  - complex structure to maintain
- Peer specialisation assumption (one interest/peer)
  - not a realistic assumption
  - too general / too specific content

iCluster

An approach towards efficient peer organisation into SONs to support IR functionality

iCluster is

- automatic (no user intervention)
- general (no previous knowledge & all data types)
- adaptive (adjusts to dynamic content changes)
- efficient (fast query processing)
- accurate (high recall)

Contributions

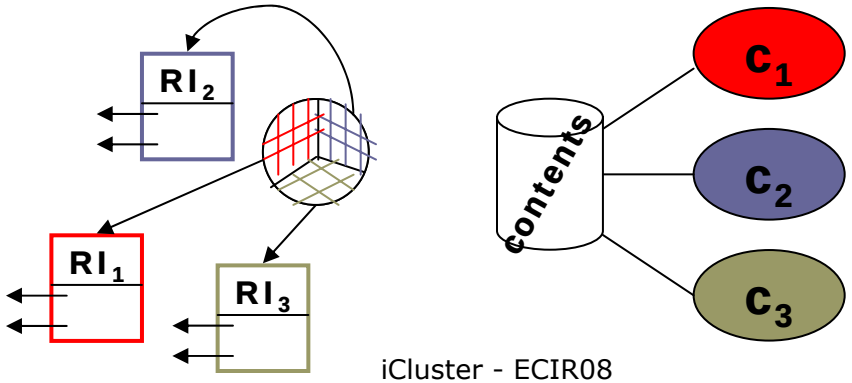
- Drops specialisation assumption
  - multiple peer interests
  - evolving peer interests
  - better representation of peer contents
- Novel strategy for self-organising peers
  - any peer may exploit a rewiring message
  - reserve a portion of the RI for long-range links
- Supports IR in a dynamic environment
  - data model independent

Peer join

When a peer *p* joins the network

1. applies a local clustering algorithm
2. computes its interests $\{c_1, c_2, \ldots, c_L\}$
3. connects to λ peers for each interest c_k

These (randomly selected) links will be refined using peer rewiring



Peer rewiring

A peer *p*

1. computes its intra-cluster similarity
   (average similarity with its short-range links)
2. initiates rewiring if similarity < threshold θ
3. sends a message (*msg*) with its interest to *m* neighbors

- All peers receiving *msg* append their interest and forward *msg* to *m* neighbors
- The message is sent back to *p* when *TTL = 0*

Peer rewiring (insights)

- Each peer starts the rewiring process
  - independently of other peers
  - based on its local view of the network
- Message is forwarded non-deterministically:
  - to *m* peers most similar to *p*

    or
  - to *m* randomly selected peers
- All peers receiving *msg* update both short- & long-range links

Query processing (1/2)

1. Locate a cluster of peers similar to the query

A peer *p* issuing the query *q*

1. initiates a message *(msg)*
2. compares *q* against its interests
3. if $sim(q,p) \geq \theta$

   broadcast *msg* in *p*'s neighborhood

   if $sim(q,p) < \theta$

   forward *msg* to the *m* peers most similar to *q*

Query processing (2/2)

2. Find the documents similar to the query

Each peer *p* receiving a *msg*

1. compares *q* against its interests
2. if $sim(q,p) \geq \theta$
   - *p* matches *q* against its locally stored content
   - *p* retrieves similar documents
3. Pointers to similar documents are sent to the initiator peer *p*
4. Candidate answers are ordered by similarity to *q* and returned to the user

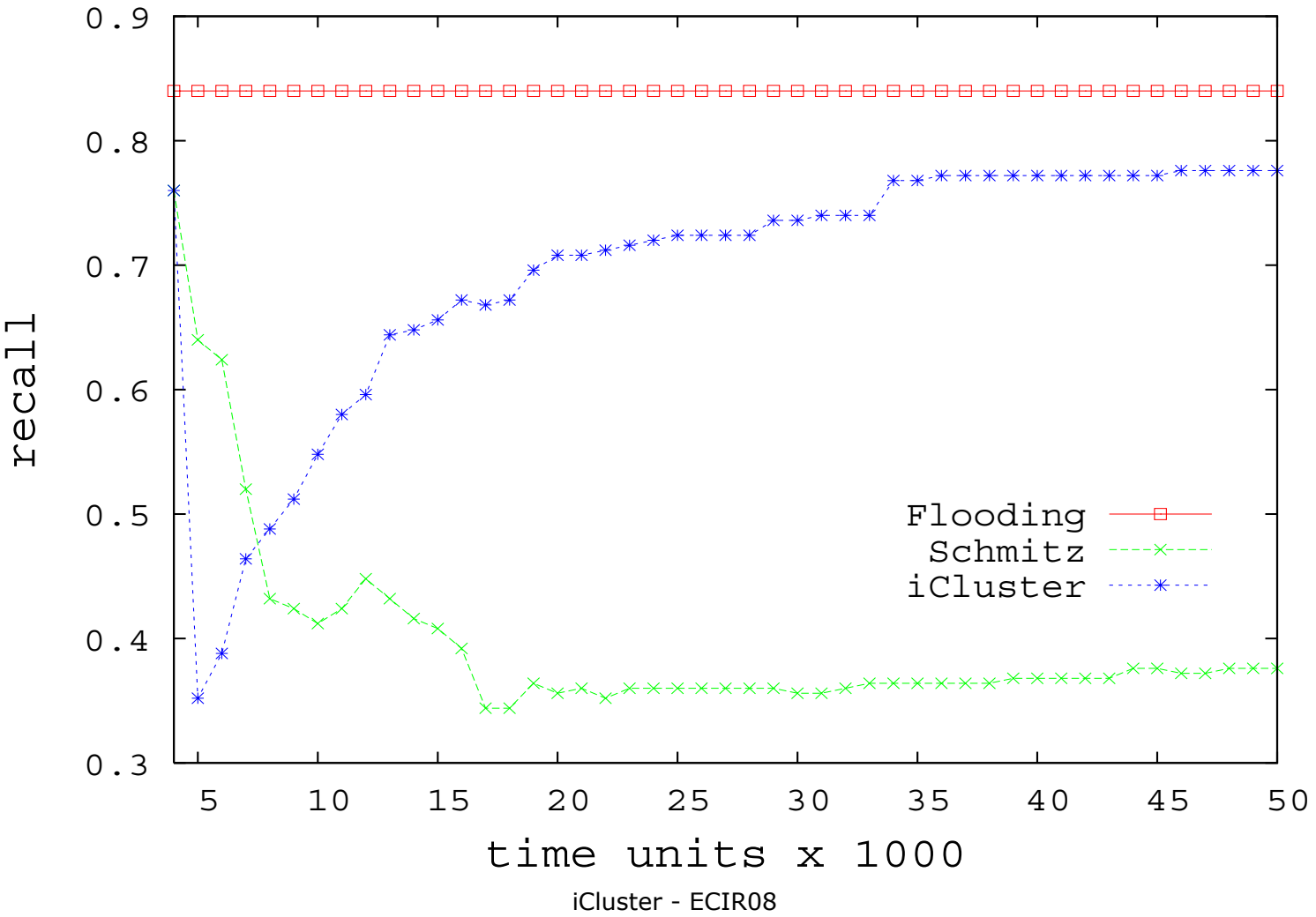
Experimental setting

- Data sets
  - a subset of *OHSUMED TREC:*
    32K medical documents, *10* classes
  - *TREC-6:*
    556K general interest documents, *100* classes
- Experimental set-up
  - *2K* peers
  - *10* links per peer (*10%* long-range links)
  - *5* initial network topologies (*5* runs for each topology)
- Experiments with different parameter values (θ, TTL, …)
- Compare against
  - the [Schmitz 04] approach
  - exhaustive search by flooding

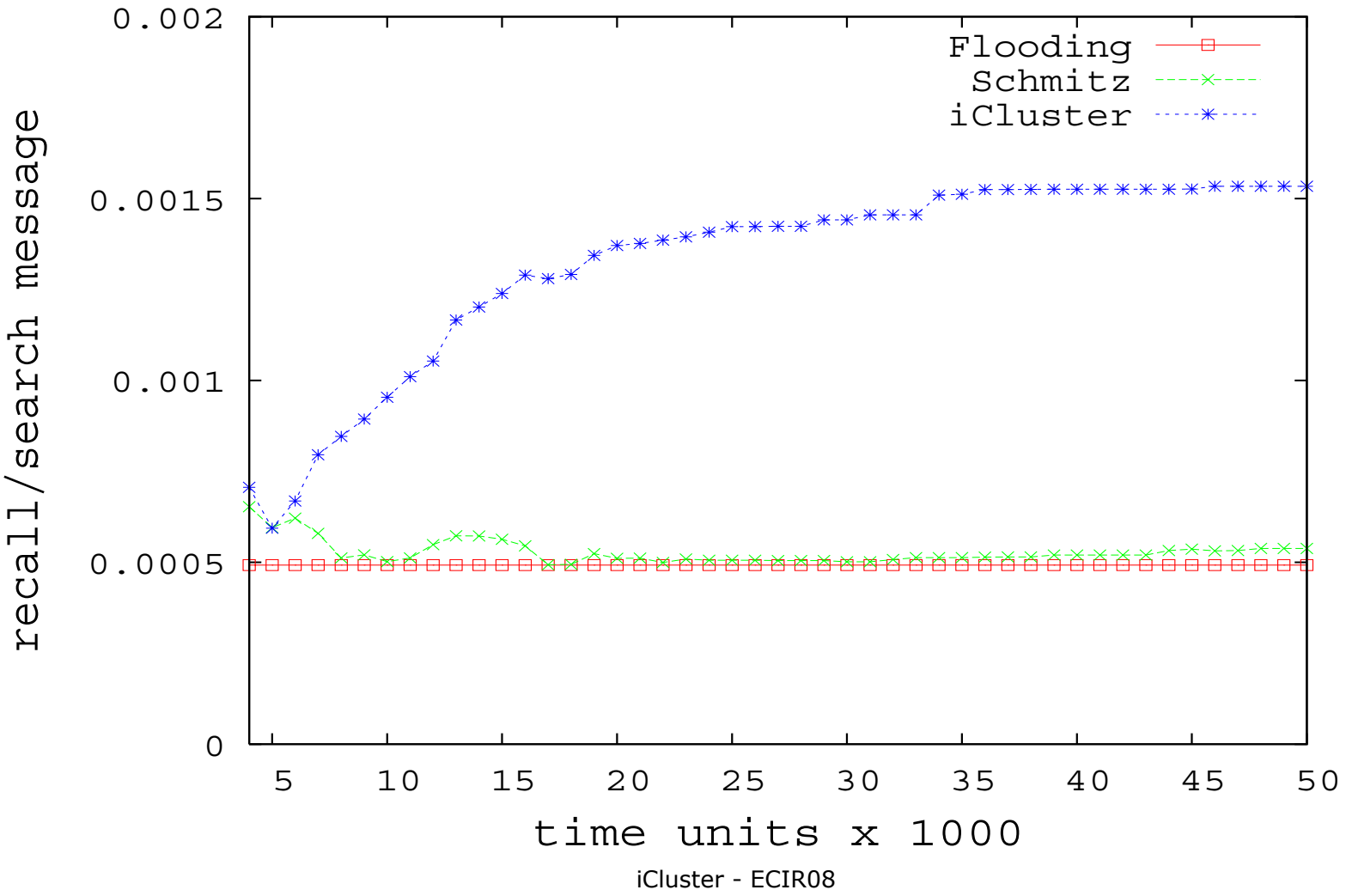
Performance measures

- (weighted) Clustering coefficient
  - measures cluster cohesion: is *p* linked with similar peers?
  - ratio of similarity between peers over total number of links in a peer's neighborhood
- Communication overhead
  - Organisation messages: to organise peers into clusters
  - Search messages: to answer a query
- Retrieval accuracy (recall)

Recall


0.9
0.8
0.7
0.6
0.5
0.4
0.3
recall
5
10
15
20
25
30
35
40
45
50
time units x 1000
Flooding
Schmitz
iCluster

Recall / Search message



Discussion / Observations

- More effective organisation and better recall for less messages (rewiring & search)
- Effects of different parameters
  - Rewiring similarity threshold θ affects clustering cohesion
  - Query effectiveness depends on rewiring similarity threshold θ and on query TTL
- IR can be highly improved by peer organisation
  - high clustering can raise 'caveman worlds'
  - long-range links ensure inter-cluster connectivity

Future Work

- Study the effect of churn
  - in terms of peers
  - in terms of content
- Assume different query distributions
  - more realistic
  - will improve query processing
- Extend the proposed protocols to support information filtering functionality

Thank U!