UNIVERSITY OF STIRLING

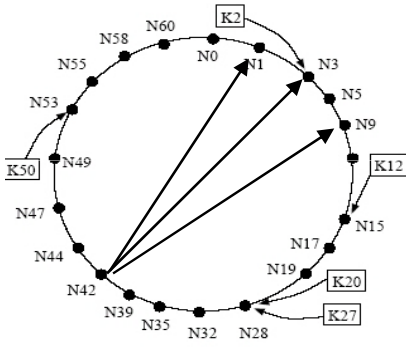
DEPARTMENT OF COMPUTING SCIENCE AND MATHEMATICS

Simulation and Modelling of Large-scale Structured P2P Overlays

Mario Kolberg & Jamie Furness

University of Stirling

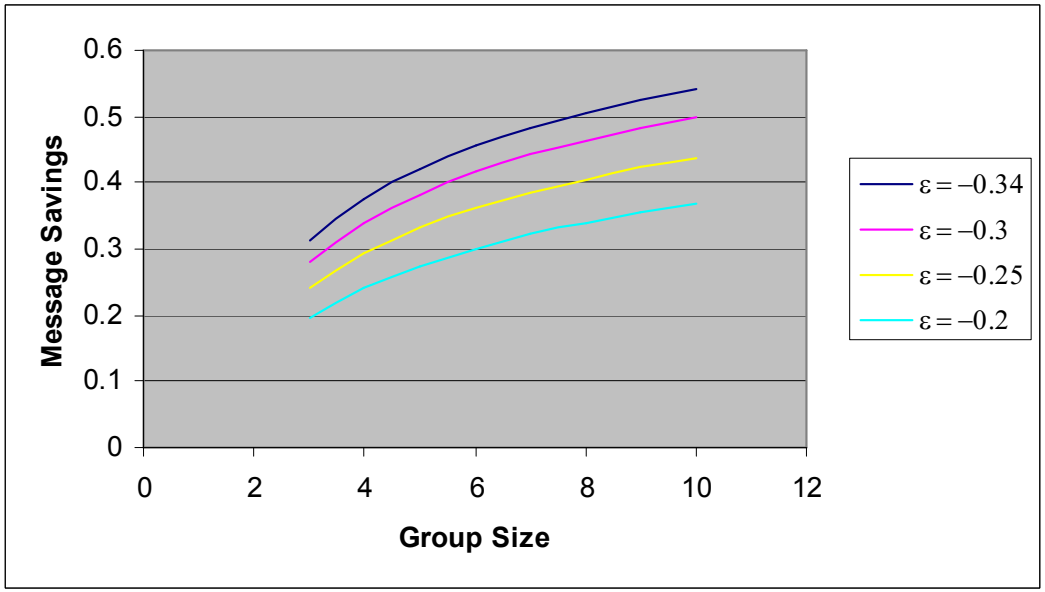
- Peer-to-Peer (P2P)
  - Overlay – build on top of the IP network
  - Nodes in the overlay are connected by virtual or logical links corresponding to a path (possibly through many physical links) in the underlying network.
  - Concentrated on one-hop structured P2P overlays
  - use a DHT for data indexing and discovery
  - (near) single hop from source node to destination node
  - Full routing table, maintenance traffic
  - EpiChord, D1HT, OneHop



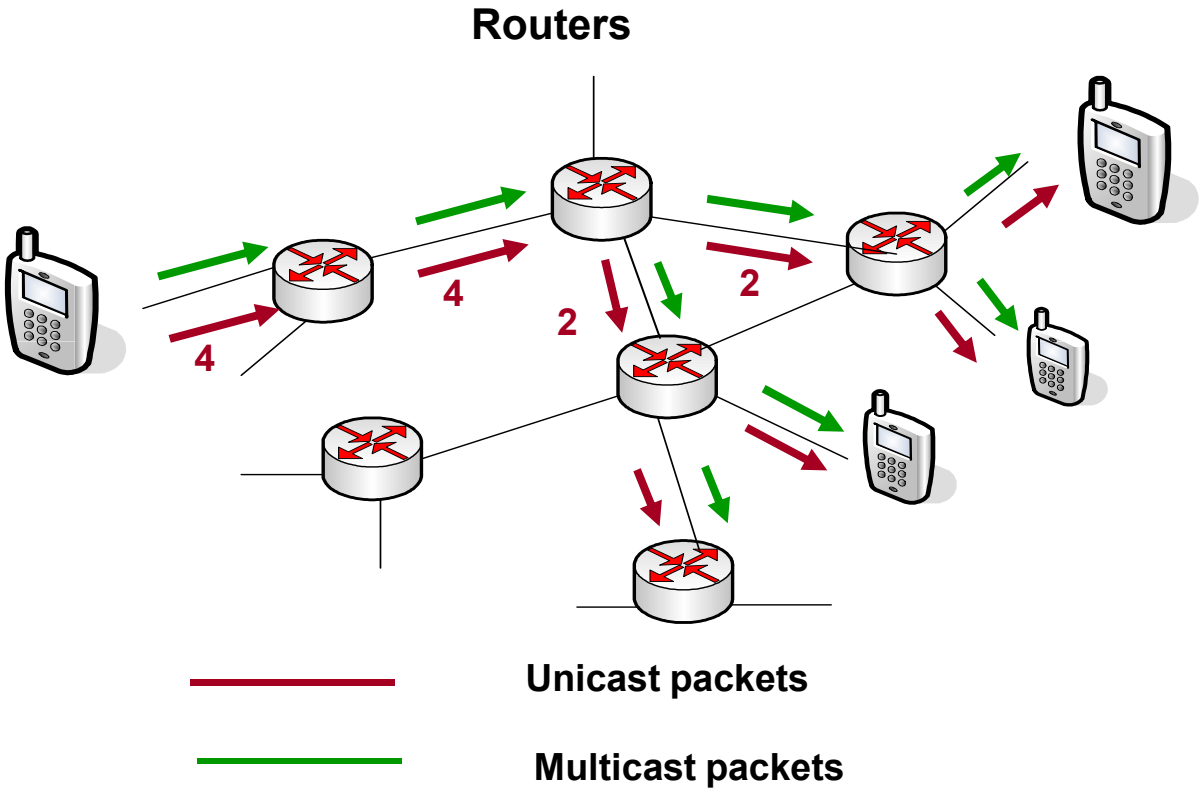
- DHTs are the indexing mechanism for P2P systems
  - DHT - Node IDs and Data Keys
  - O(1)-hop overlays have better latency characteristic than multi-hop overlays, but require more maintenance traffic
  - How to obtain best performance in a large-scale wide area context for DHT operations is an important question.

- Issues:
- Algorithms are hard to validate
  - Complex algorithms
  - Large networks (up to millions of nodes)
  - Simulations are resource hungry
    - Very dynamic behaviour (nodes joining and leaving)
    - Large amount of state (routing table) for each node
    - The state of a particular node at a certain point in time is very hard to ascertain
  - Looked at two problems:
    - Multicast efficiency gains in overlays
    - Efficient broadcast algorithms for wildcard searches
  - There are a number of “simple” models of P2P but often they neglect the issue of churn

- How to make P2P Overlays more efficient? → Multicast
- Why multicast?
  - Chuang-Sirbu multicast scaling law states message savings are related to group size: $1 - m^{-\varepsilon}$, $-0.34 < \varepsilon < -.2$
  - 5-way: 28% to 42%, 10-way: 37% to 54%
  - Host group multicast vs. multidestination multicast
    - Overhead, goup size, group numbers, life time of a group



Multi-Destination Routing

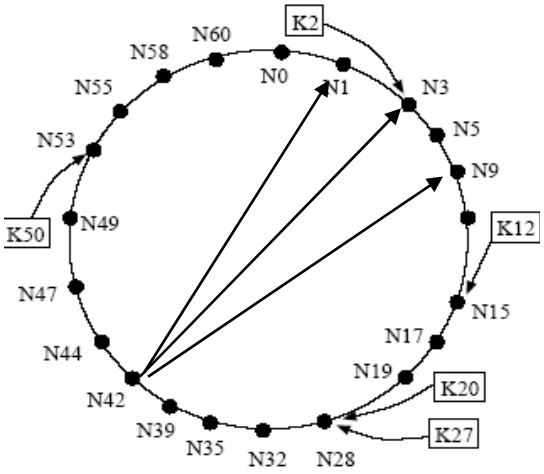


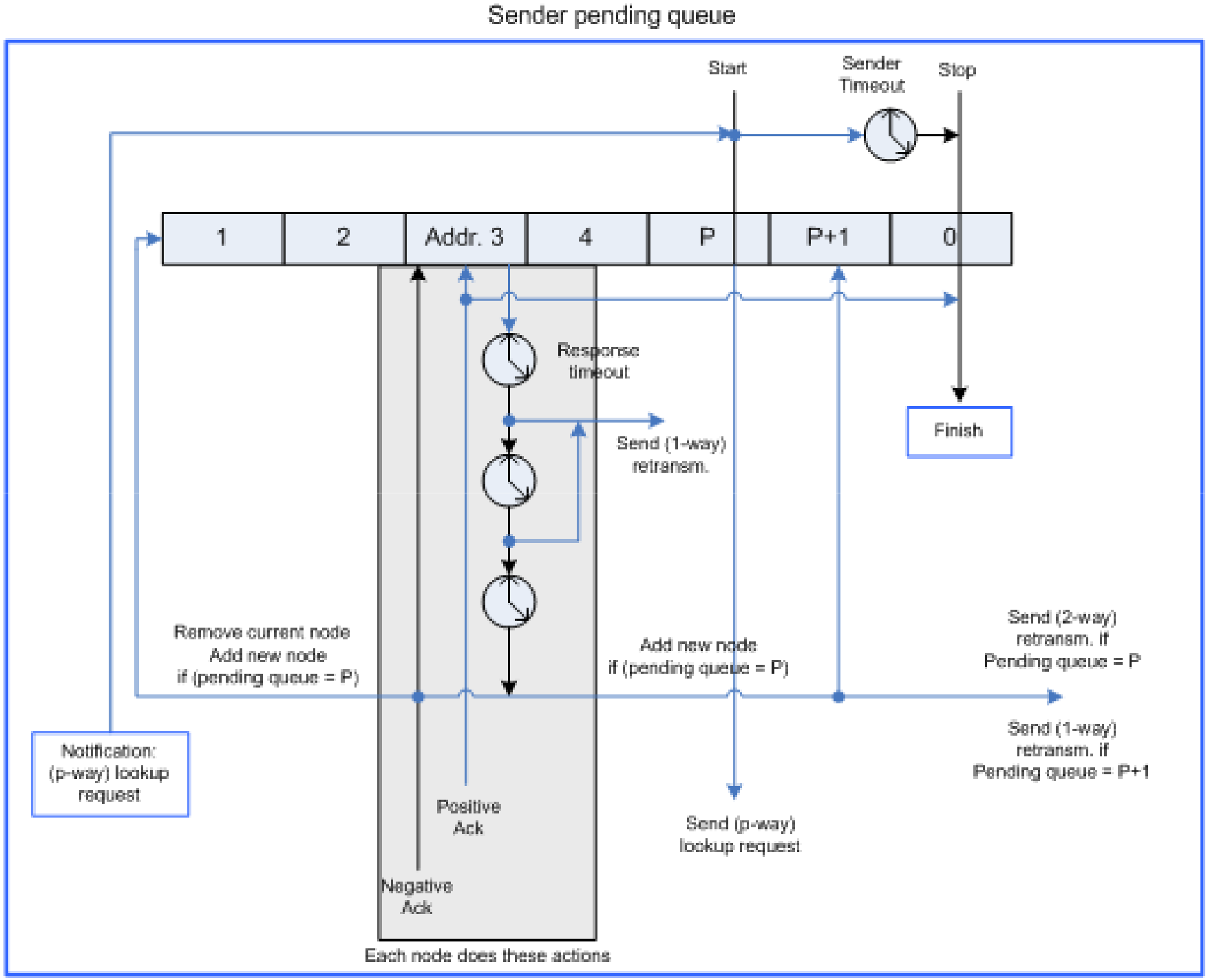
XCAST = Experimental Multi-Destination Routing Protocol

Experimentation

- To determine whether multi-destination routing is applicable to Overlay systems, we used simulation and modelling:
  - EpiChord (simulation).
  - Markov Model(s)
- Simulations were carried out using a 10,450 node network in the SSFNet simulation environment. Overlay sizes varied from 1k to 9k nodes.

- DHT lookups and routing table maintenance use parallel unicast requests
- Failed responses are used iteratively to update routing table and narrow the search
- Opportunistic maintenance of routing table

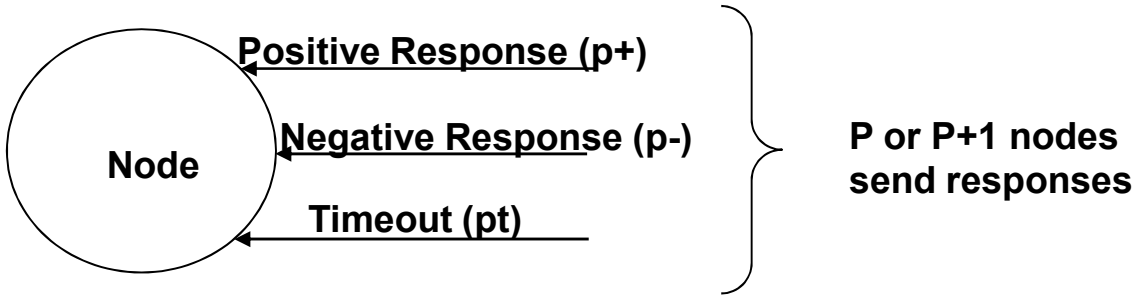


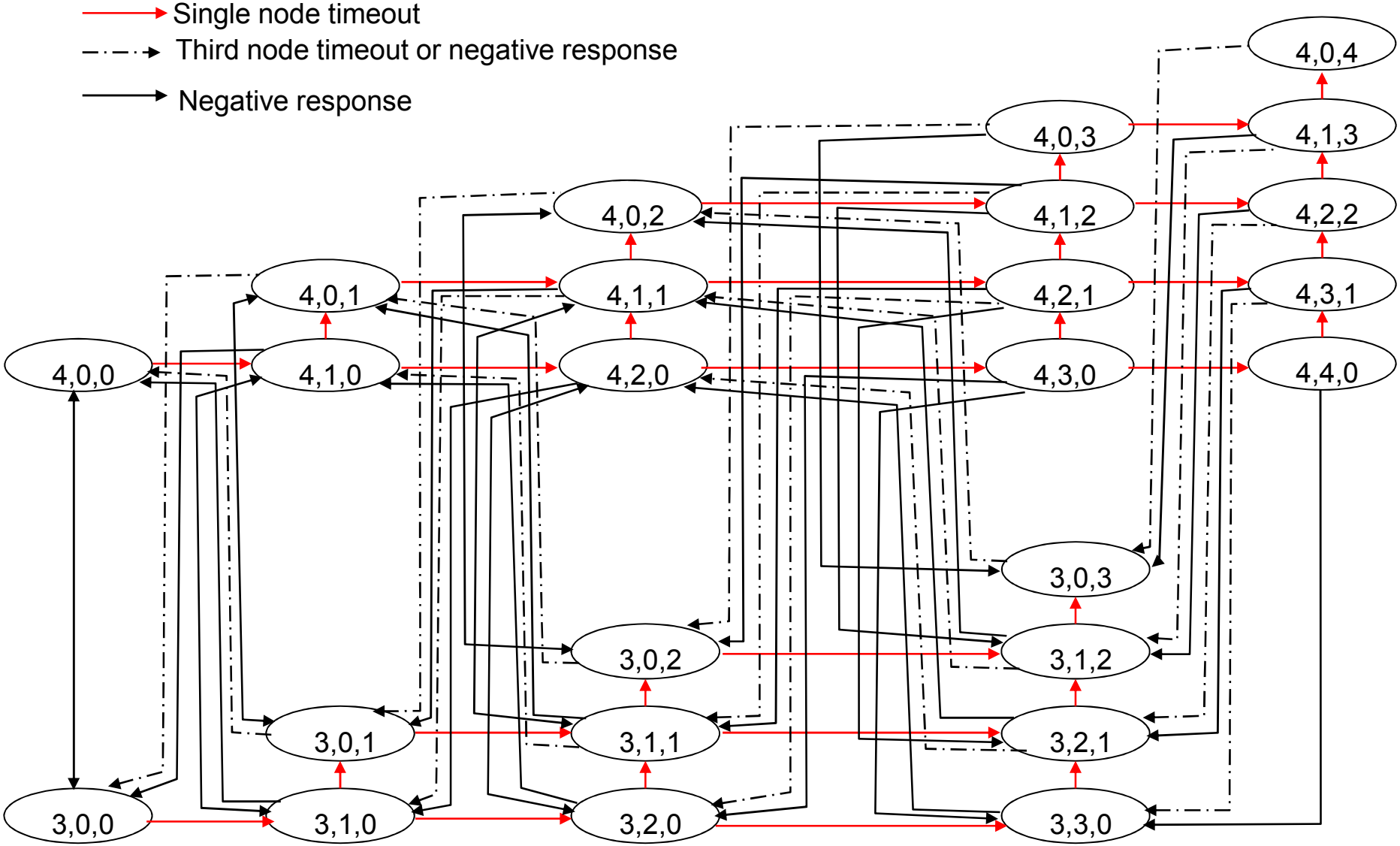

Sender pending queue
Start
Sender Timeout
Stop
1
2
Addr. 3
4
P
P+1
0
Response timeout
Send (1-way) retransm.
Finish
Remove current node
Add new node
if (pending queue = P)
Add new node
if (pending queue = P)
Send (2-way) retransm. if Pending queue = P
Send (1-way) retransm. if Pending queue = P+1
Notification: (p-way) lookup request
Positive Ack
Negative Ack
Send (p-way) lookup request
Each node does these actions

Analytical Model of XCAST enabled EpiChord

- Chuang Sirbu predict saving of 1 - m $^{-\varepsilon}$, with $\varepsilon = -0.2$
- Does not take into account EpiChord retransmissions and timeouts
- A model will allow for more flexible and scalable analysis of the expected savings than simulation.
- Comparing results of the model with simulation

- The size of the pending queue changes depending on the type of response received
- Know Probabilities of receiving a certain response from simulations
- Hence pending queue size can be calculated, and so the average # of 2-way and 1-way retransmissions
- Pending queue has been modelled as a DTMC, transition matrix




Single node timeout
Third node timeout or negative response
Negative response
4,0,4
4,0,3
4,1,3
4,0,2
4,1,2
4,2,2
4,0,1
4,1,1
4,2,1
4,3,1
4,0,0
4,1,0
4,2,0
4,3,0
4,4,0
3,0,3
3,0,2
3,1,2
3,0,1
3,1,1
3,2,1
3,0,0
3,1,0
3,2,0
3,3,0

Assumptions

- Assumption 1:
  - probabilities do not change over time
  - The time the queue is in a certain state is ignored
- Assumption 2:
  - A transition occurs after one and only one response is received
  - Considers only a single node
- Assumption 3:
  - It is equally likely for a node to time out once, twice or three times
  - Probabilities of timing out is independent of the state

Results

P	Neg Resp. per lookup	Timeouts per lookup	Xcast (model)	Xcast (simul)	unicast (model)	unicast (simul)
3	1.44	1.3	0.77	0.75	0.87	0.9
4	1.98	1.54	1.06	1	1.02	1.05
5	2.54	1.77	1.35	1.22	1.18	1.2

P	Neg Resp. per lookup	Timeouts per lookup	Xcast (model)	Xcast (simul)	unicast (model)	unicast (simul)
3	6.1	3.16	3.19	3.05	2.2	2.22
4	7.27	3.67	3.81	3.45	2.52	2.6
5	8.49	4.23	4.49	3.92	2.88	3.03

- Use Pepa to model the system to get closer results…

PEPA

- Two models
- Communicating model
  - Pending queue process
  - Processes for each process in the pending queue
- “Simple model” based on the states of the DTMC

- Expected results to be closer to simulation values
- Results show too many retransmissions (actually quite a bit worse than DTMC)

UNIVERSITY OF STIRLING

DEPARTMENT OF COMPUTING SCIENCE AND MATHEMATICS

Complex Search Techniques

- Structured P2P networks don't tend to support all types of complex queries.
- Unstructured networks do, and hence are more popular. However, they are inefficient.
- Using efficient broadcasting it is possible to support all types of complex queries over structured P2P.
- We investigate the effects of churn on broadcast search over Chord and Pastry.

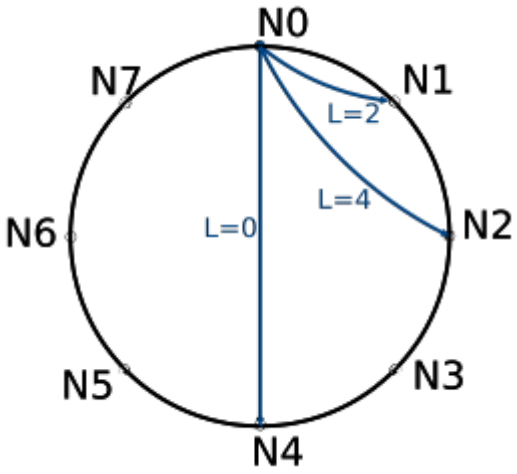
- Complex queries

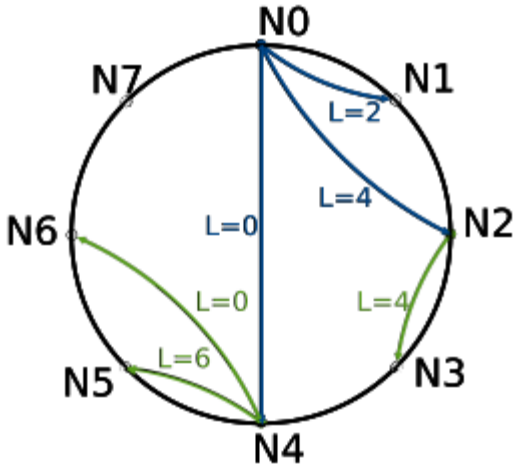
- Exact-match: nine inch nails - the slip (2008) - letting you [v0].mp3
- Keyword: nine inch nails, nin, the slip
- Range bit-rate: 256-320
- Wild-card: nine inch nails *
- Semantic: 9 inch nails
- Regex: ^nine inch nails .*\.(mp3|flac|alac)$

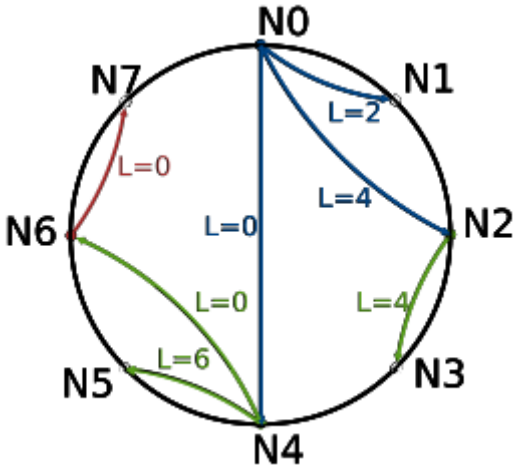
- Unstructured overlays
  - No structure, links established arbitrarily.
  - Flooding or random-walks used to retrieve data.
  - Easy to implement.
  - Inefficient, low success rate.
- Structured overlays
  - Nodes are assigned a key, often based on their IP address
  - Data is assigned a key, often based on its file-name.
  - Distributed Hash Table (DHT) interface can store data or retrieve data given its corresponding key.
  - Examples: Chord, Pastry...

- Structured networks make use of consistent hashing.
  - Both types of keys are generated using the same hash function, usually SHA-1.
  - Reduces arbitrary length keys to a fixed identifier space.
  - Balances load, relieving hot-spots.
- Example
  - track → 42aef171c1c0accaeee38c605d98ab5db51a13f5
  - track1 → ea6b175de80bd33899cdf4a0530059aabffb8f66
  - track2 → 08979fbae1fe1e5b06b3646138be36b27d583f34
- Not locality aware, patterns in keys are lost after hashing.

- Broadcasting supports all types of complex queries.
- Performed by forwarding the query to a few nodes, assigning each of them an area to cover.
- Queries are processed at each node.
- Many more messages than regular searches in structured networks... but many less than flooding in unstructured networks.



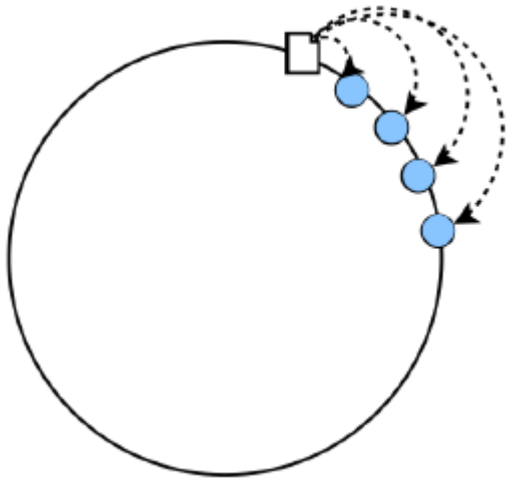




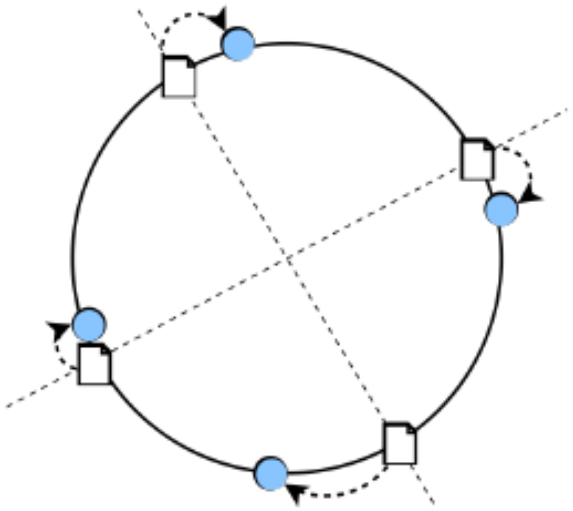
- Our aim was to compare the performance of broadcasting a search query over different overlays while the network is under churn, focusing on some specific areas:
  - Success rate
  - Bandwidth requirements
  - Data replication

- Simulations developed using OverSim.
- Network sizes of 1,000 and 10,000 nodes.
- Average node lifetime from 100 secs to 10,000 seconds.
- Replication rate from 1 to 32.

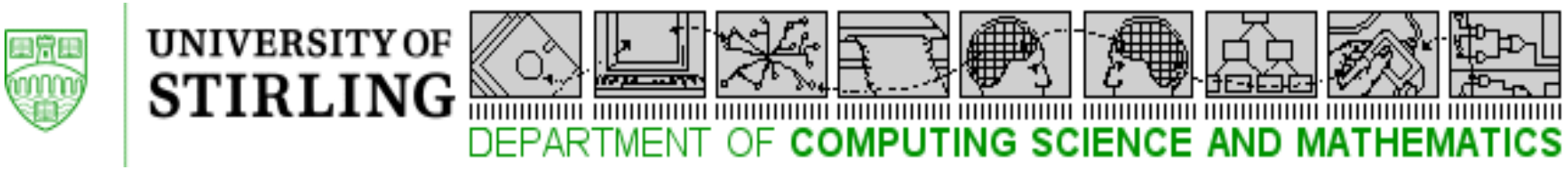
- Neighbour replication
  - Replicates data at neighbouring nodes.
  - Maintenance is cheap.
  - Commonly used.
  - Bad for broadcasting.



- Multi publication replication
  - Replicates data evenly around the network.
  - Maintenance is more expensive.
  - Good for broadcasting.

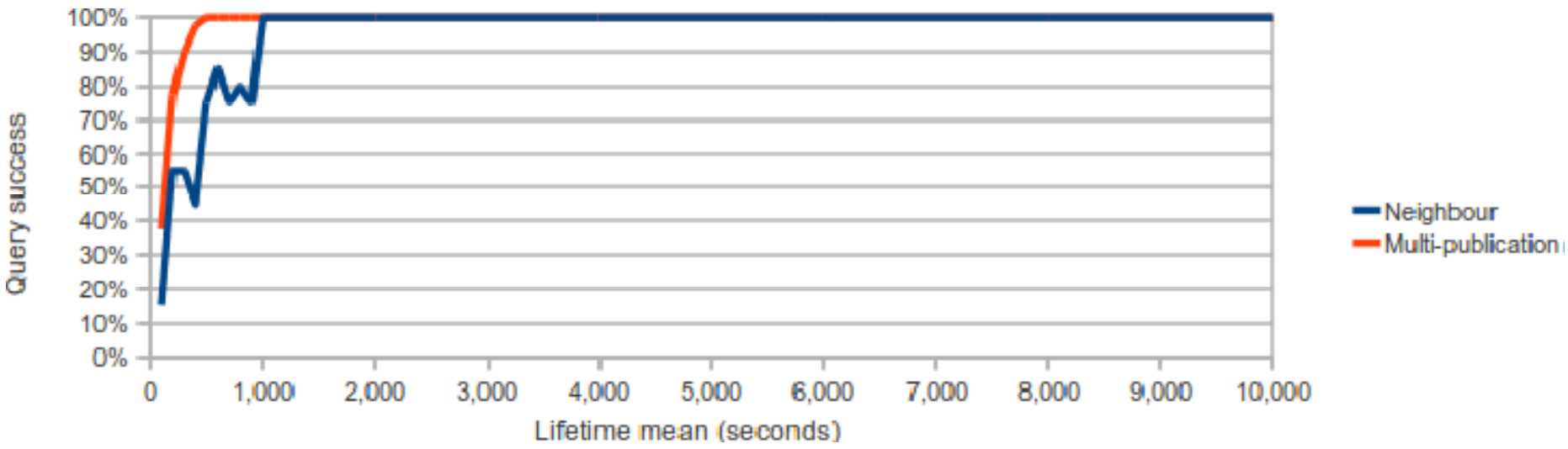


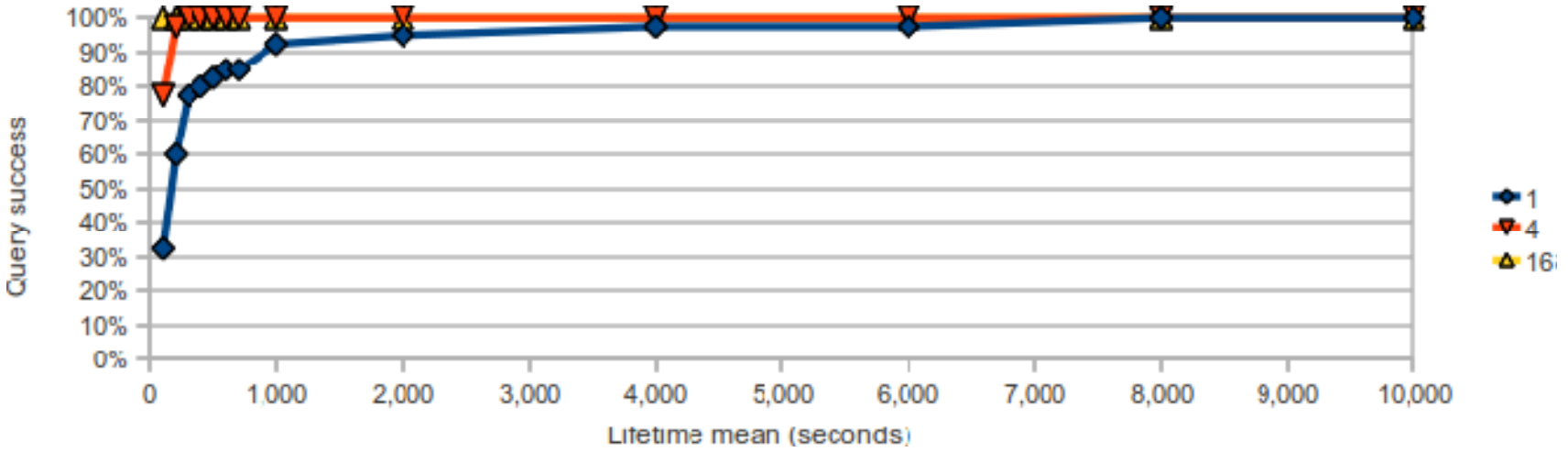
- Experimentation concentrated on bandwidth consumption and comparing replication strategies
  - Various overlays
  - Various levels of churn
  - Both replication strategies
  - What level of replication



UNIVERSITY OF STIRLING

DEPARTMENT OF COMPUTING SCIENCE AND MATHEMATICS





- Conclusions/Questions

- Simulations can help checking algorithms with P2P overlays
- Simulations are complex and limited: large amount of state, up to 10,000 nodes
- What kind of modelling approaches can help to verify the behaviour of algorithms?
- Can the problems be categorised and the appropriate modelling approaches are chosen?
- Can modelling approaches cope with the complexity, and help exploring larger networks?