Price of Structured Routing and Its Mitigation in P2P Systems under Churn

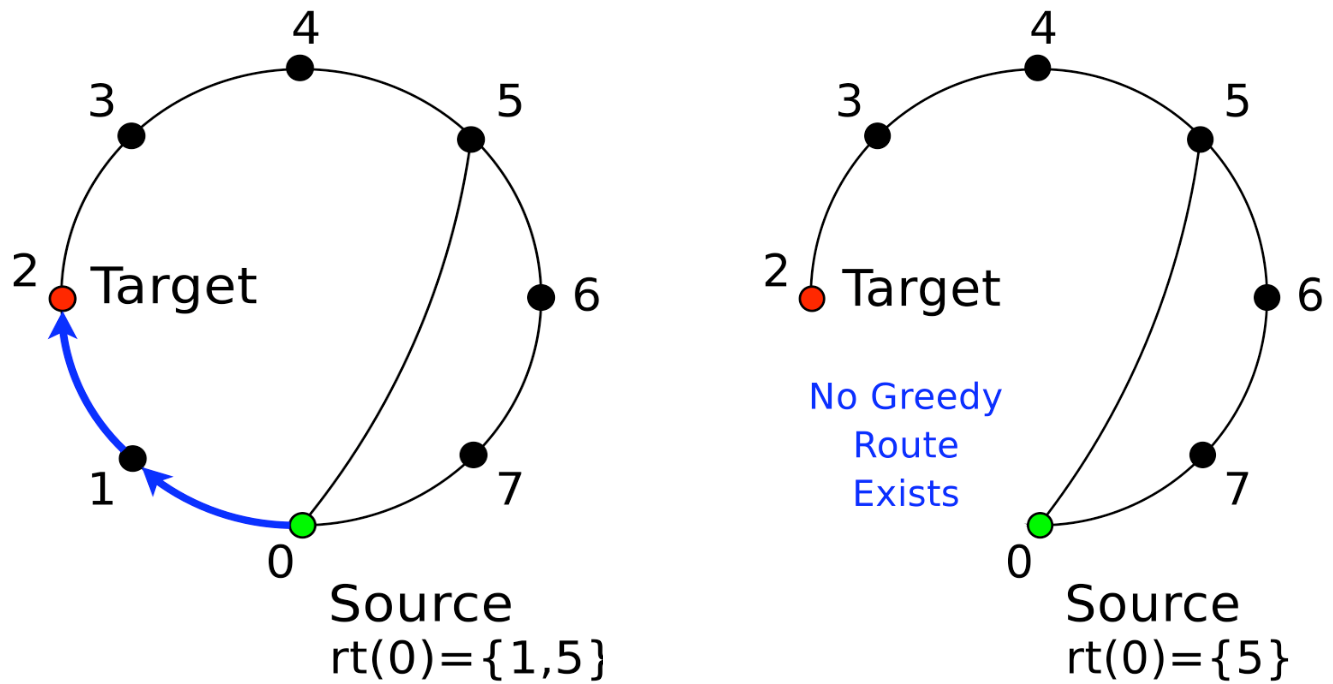
Joseph S. Kong

Department of Electrical Engineering, UCLA

Joint work with:
Prof. Vwani Roychowdhury

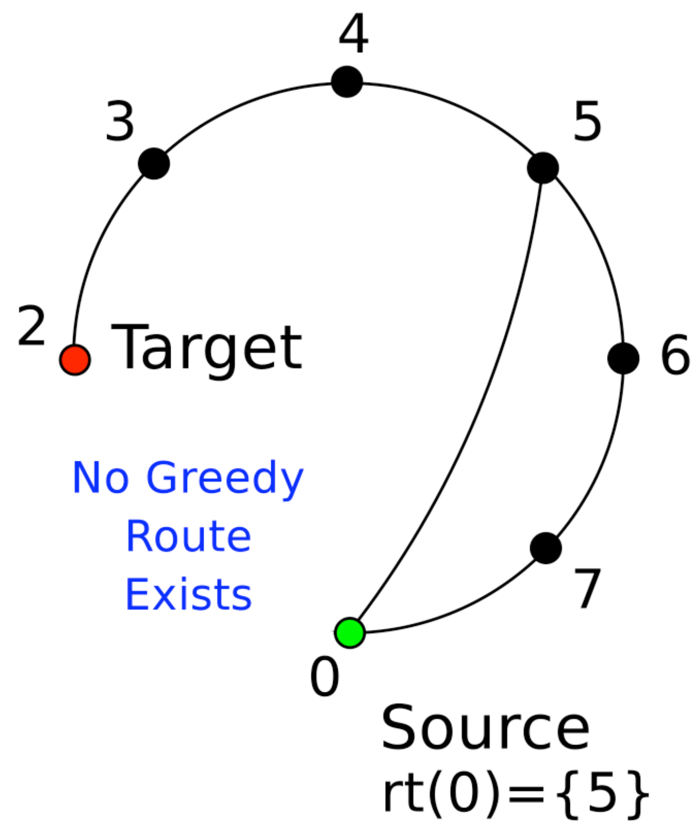
Introduction

- When a structured P2P system experiences failure, a node cannot always reach all other nodes in the same connected component.



Connected vs Reachable Component

- *Reachable* component = the set of nodes that a given node can reach;
- *Connected* component = the set of nodes that a given node is connected to;
- From node 0's perspective: node 2 is in the connected component, but *not* in the reachable component.



Price of Structured Routing

- For a given node in a P2P network, we examine the size of the gap between the *connected component* (CC) and the *reachable component (RC)* under different failure conditions.
- We will interpret this gap as the *price of structured routing* for a given node:

$$\textit{price of structured routing} = |CC| - |RC|$$

- The price of structured routing for a P2P network is obtained by taking the expectation over all nodes;
- Not a one-sided criticism of structured P2P systems.

Overview of the Presentation

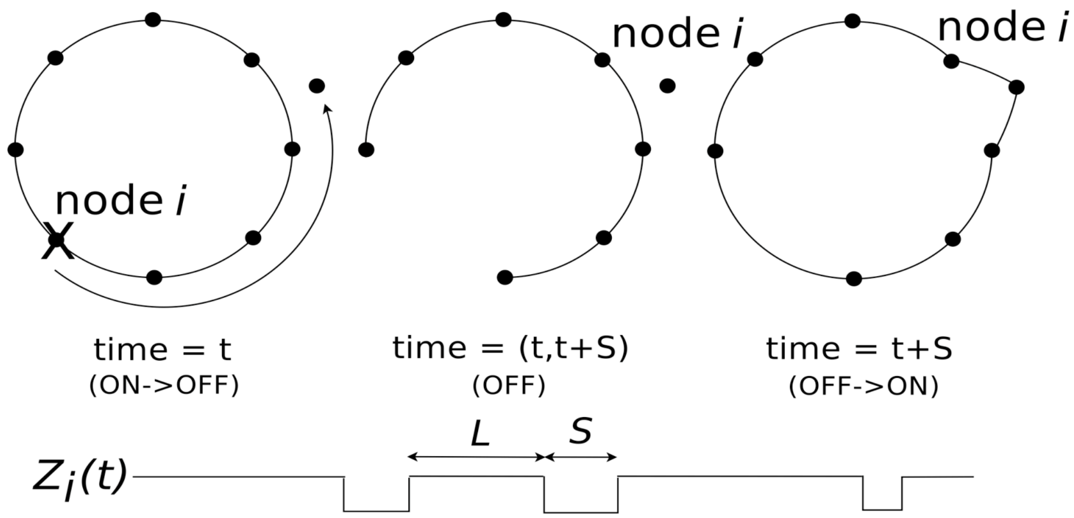
- Models of failure in P2P networks: the churn model and the uniform failure model.
- Simulation results on the price of structured routing for Symphony and Chord;
- Analytical derivation on the price of structured routing;
- Mitigation techniques.

Churn vs. Uniform Failure

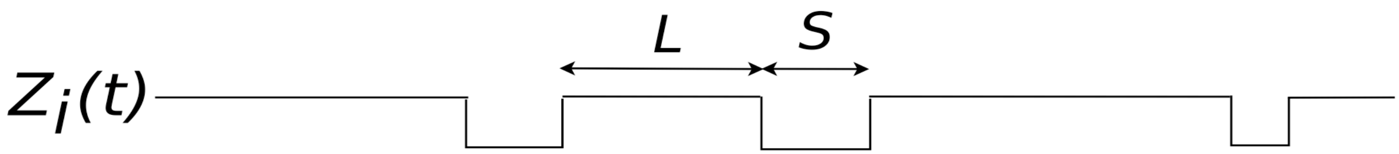
- **Lifetime-based churn model (Leonard et al.)**:
  - every node stays ON for the lifetime L (r.v.)*;*
  - after a node's neighbor has departed, the node spends the search-time S (r.v.) looking for replacement;
  - by the same token, a joining node spends S looking for the required connections.
- **Uniform failure model**: each node in network fails with probability $q = (1-p)$.

Lifetime-based Churn Model

- An assumption: the average size of the network is constant; a failed node is replaced by an entering node.
- This is equivalent to the failed node *rejoining* the network at a new randomized location.
- State of a node is an ON/OFF process.



A Simple Yet Important Lemma

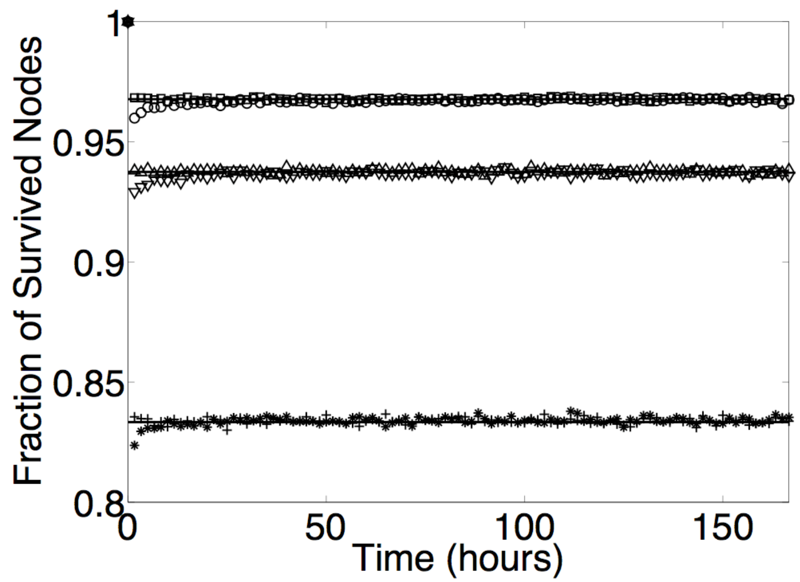


- **Lemma 1**: Let $Z_i(t)$ denote the ON/OFF process for node i in a structured P2P system. The steady-state probability of finding the process in the ON state is:

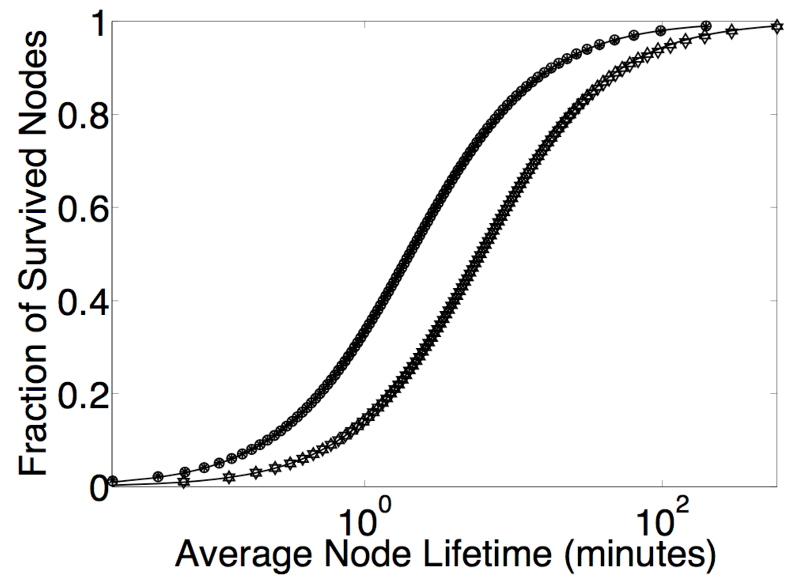
$$p = \lim_{t \to \infty} P(Z_i(t) = 1) = \frac{\bar{L}}{\bar{L} + \bar{S}}$$

- We can now reduce the lifetime-based churn model to an equivalent uniform failure model by *Lemma 1*.

Simulations: Lemma 1



$S = 2$ min, $\bar{L} = \{10,30,60\}$ min



$S = \{2,6\}$ min

Randomized-Chord

- In randomized-Chord, nodes are placed in numerical order around a ring, with each node keeping 1 predecessor/successor.
- Each node maintains $\log_2(N)$ connections or fingers, with the *i*th finger at a distance uniformly chosen from an interval $[2^{i-1}, 2^{i}]$.
- For example, when simulating a network of 2^{20} nodes, each node keeps exactly 20 connections.
- Routing in Chord is done greedily.

Symphony

- A small-world routing network in the 1-D case;
- Has a ring-like address space;
- Each node is connected to a constant number of short-range connections (k_S) to its nearest neighbors;
- And a constant number of long-range connections (k_L) to distant nodes that have a 1/d distance distribution (d is the distance).
- Routing is performed greedily.

Price of Structured Routing: Simulation

Simulation setup under the *uniform failure model*:

- Each node fails with probability $q=1-p$;
- The failed node is deleted along with all its edges;
- A random survived node is picked and we find its |CC| and |RC|;
- This procedure is repeated a fixed number of times to obtain the sample averages for |CC| and |RC|, thus the simulated price of structured routing by the difference;

Simulation Results (Uniform Model)

UCLA

Size (N)	$p = 0.8$ (Symphony) % of N	$p = 0.8$ (Symphony) # of nodes (k)	$p = 0.96$ (Symphony) % of N	$p = 0.96$ (Symphony) # of nodes (k)
2^{16}	41.0	26.9	2.5	1.6
2^{18}	46.0	121	2.6	6.8
2^{20}	52.9	555	**3.7**	39.2

Size (N)	$p = 0.8$ (Chord) % of N	$p = 0.8$ (Chord) # of nodes (k)	$p = 0.96$ (Chord) % of N	$p = 0.96$ (Chord) # of nodes (k)
2^{16}	6.14	4.0	1.00	0.6
2^{18}	6.17	16.1	1.09	2.8
2^{20}	6.13	64.4	**1.09**	11.5

Table 1: Price of structured routing: simulation results. The number of nodes *affected* is expressed in thousands.

Price of Structured Routing: Simulation

Simulation setup under the *churn model*:

- A joining node is assigned a lifetime L according to a distribution;
- A node fails at the end of its lifetime, loses all of its connections and rejoins the structured network;
- We record the time average size of the CC and RC (hence the price of structured routing by taking the difference);
- We repeat the simulation for the Pareto and exponential lifetime distribution with different mean lifetimes; the search-time S is fixed to be 1 minute.

Simulation Results (Churn Model)

$\bar{L}$ (min.)	Pareto (Symphony) % of N	Pareto (Symphony) # of nodes (k)	Exponential (Symphony) % of N	Exponential (Symphony) # of nodes (k)
4	50.9	534	53.1	557
9	19.7	207	20.7	217
24	**4.0**	41.9	**3.6**	37.8

$\bar{L}$ (min.)	Pareto (Chord) % of N	Pareto (Chord) # of nodes (k)	Exponential (Chord) % of N	Exponential (Chord) # of nodes (k)
4	5.89	61.8	6.32	66.3
9	3.06	32.1	2.96	31.0
24	**1.17**	12.3	**1.17**	12.3

Table 1: Price of structured routing: simulation results for the churn model. The search delay is 1 minute. Size of the network is 2^{20} nodes. The symbol $\bar{L}$ denotes the mean node lifetime.

Analytical Derivation Roadmap

- Use site percolation theory to derive the size of the GCC, $x(p)$, under failure probability $(1\text{-}p)$;
- The expected size of the CC of a randomly chosen surviving node, S_c, is: $S_c = x^2(p)/p$;
- Employ the *reachable component method* to derived the size of the RC of a random surviving node;
- Invoke Lemma 1 for results under the churn model:

$$p = \frac{\bar{L}}{\bar{L} + \bar{S}}$$

Site Percolation Theory

- Given a graph, site percolation theory deals with the emergence of a giant cluster that spans the graph, when nodes (i.e. sites) of the graph randomly fail with probability q (i.e. q=1-p, where p is the node *survival* probability).
- Each graph has a percolation threshold p_C;
- For $p > p_C$, a giant connected component (GCC) appears; for $p < p_C$, only small disconnected components exist.

Derivation of the Size of GCC

- Primary quantity of interest is *P(n),* which is the probability that a randomly selected node belongs to a CC of *n* nodes;
- *P(n)* has the probability generating function $G(z) = \sum_{n=0}^{\infty} P(n) z^n$
- We find the self-consistent expression for *G(z);*
- We compute the mean connected component size and the percolation threshold;
- Above the threshold, *G(z)* is redefined for the connected components that *do not* belong to the GCC; thus, the size of the GCC is given by *1-G(1)*.

ER Graph Example

- The ER random graph has a Poisson degree distribution with the generating function as: $H(z) = e^{\bar{k}(z-1)}$, where $\bar{k}$ is the average degree.
- We can write down the self-consistency equation:

$$G(z) = z \sum p_k [G(z)]^k = zH(G(z))$$

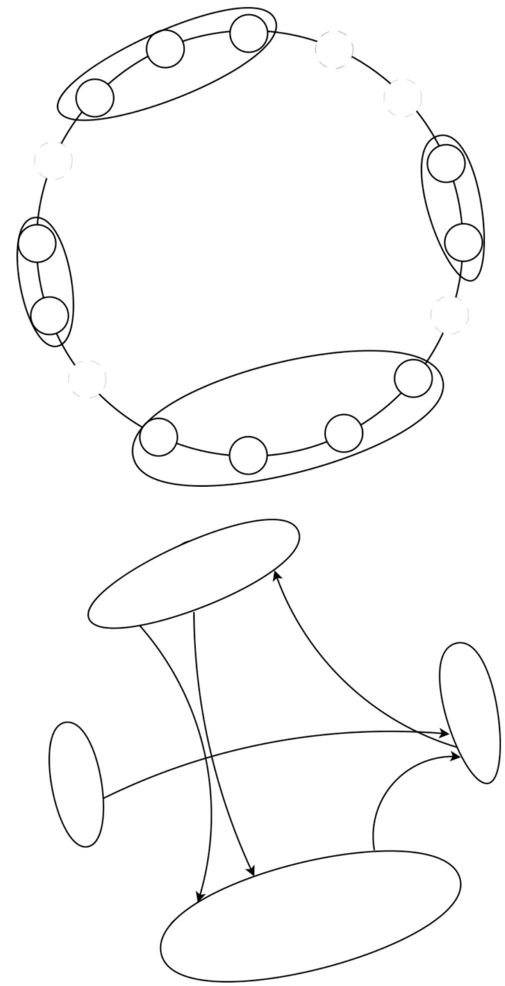
- Thus, the recursive equation describing the size of the GCC, x, is given as: $x = 1 - G(1) = 1 - e^{-\bar{k}x}$
- The size of the GCC is obtained by finding the solution to the above equation.

Derivation for Symphony

UCLA

- We have *k* local clusters or "islands” form the bottom layer.

- These local clusters will be joined by the long-range connections to form larger connected components making up the top layer.



Derivation for Symp. (bottom layer)

- Let us use $P_0(n)$ to denote the probability that a randomly chosen node belongs to a local cluster of n nodes:

$$P_0(n) = \begin{cases} 1-p, & n=0 \\ npr^{n-1}(1-r)^2, & n \geq 1 \end{cases}$$

where the variable r, which can be viewed as the probability of extending the local cluster in one direction, is defined as: $r=1-(1-p)^{k_S}$.

- Generating function of $P_0(n)$ is $G_0(z)$.

Derivation for Symp. (top layer)

- Every connected component that consists of a local cluster of n nodes has nk_L outgoing long-range connections from the local cluster and m incoming long-range connections attached to the local cluster:

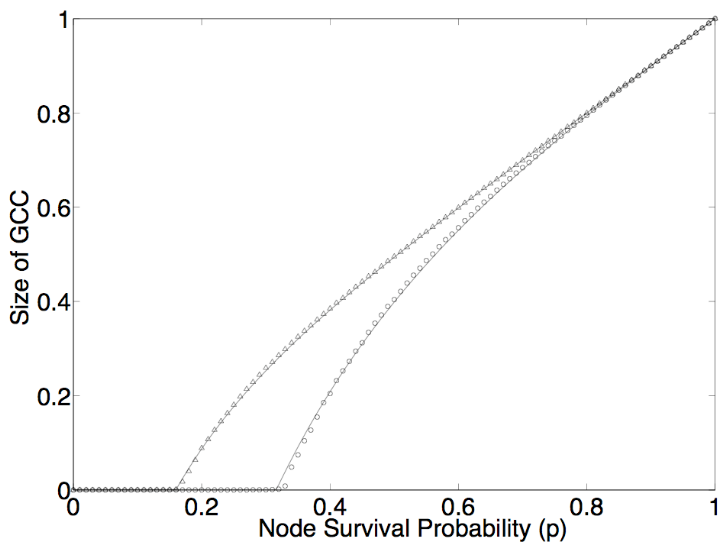
$$G(z) = \sum_{n=0}^{\infty} P_0(n) z^n \sum_{m=0}^{\infty} P(m|n)[G(z)]^{c(m+nk_L)}$$

- where c is the fraction of effective shortcuts and $P(m|n)$ denotes the probability of having m incoming long-range connections to a local cluster of size n.

Derivation for Symphony

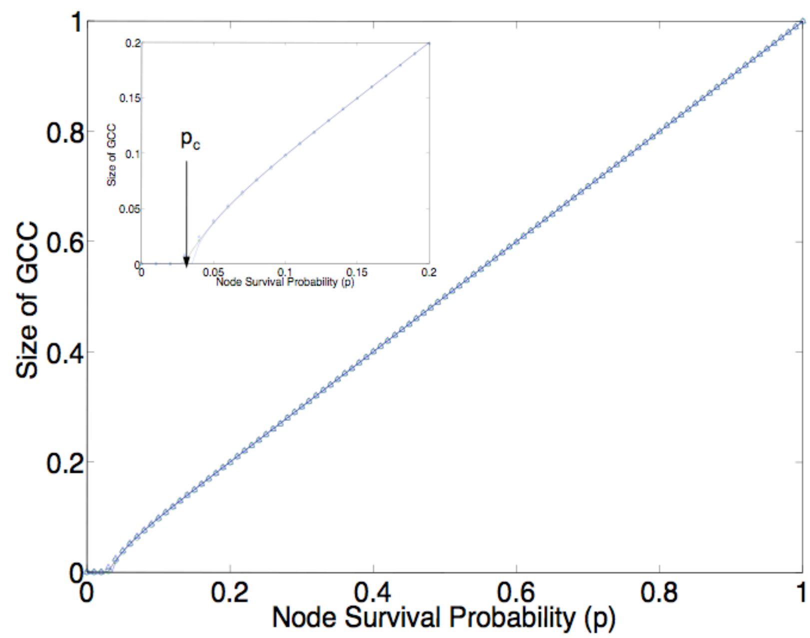
- After some computations, we arrive at the equation for finding the size of the GCC, x:

$$x = 1 - G_0((1-x)^{ck_L} e^{k_L((1-x)^c - 1)})$$



Derivation for Chord

- The derivation for Chord follows closely as the derivation for Symphony, although there are several key differences (details omitted).



Reachable Component Method

- Derived in a separate work (Kong et al. IEEE DSN06);
- For a system with $N=2^d$ nodes, the expected size of the RC for Chord is:

$$E[S] \approx \sum_{h=1}^{d} \binom{d}{h} \prod_{m=1}^{h} (1 - q^m)$$

- For Symphony:

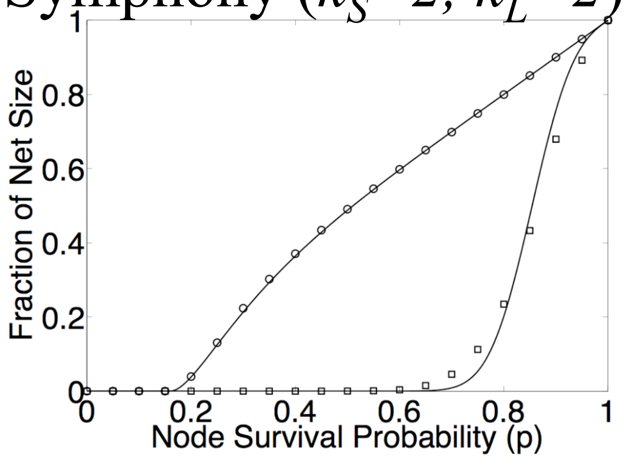
$$E[S] \approx \sum_{h=1}^{d} 2^{h-1} \prod_{m=1}^{h} (1 - Q_{sym})$$

where

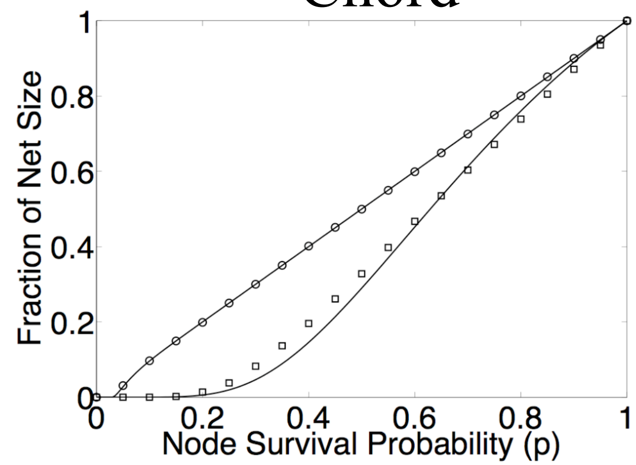
$$Q_{sym} \approx q^{1+k_L} \left(\frac{1 - (1 - \frac{k_L}{d} - q^{1+k_L})^{\frac{d}{1-q}+1}}{1 - (1 - \frac{k_L}{d} - q^{1+k_L})} \right)$$

Analysis vs. Simulation (Uniform)

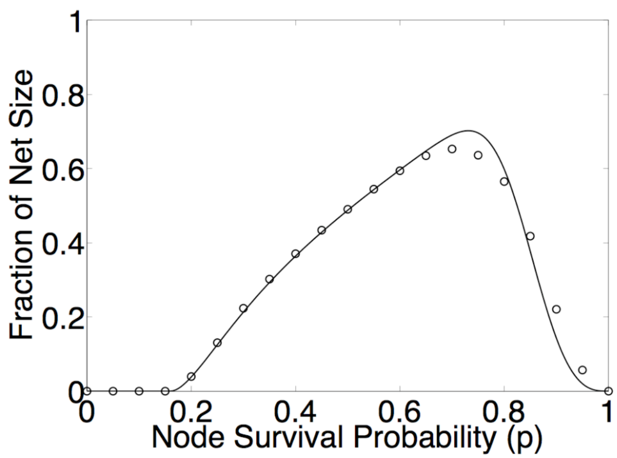
Symphony (k_S=2, k_L=2)

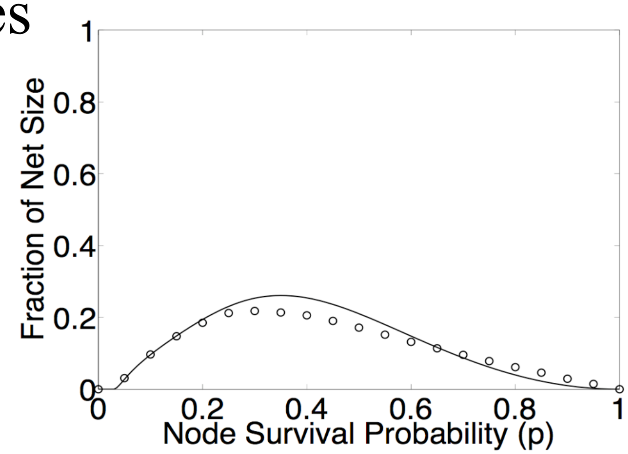


Chord



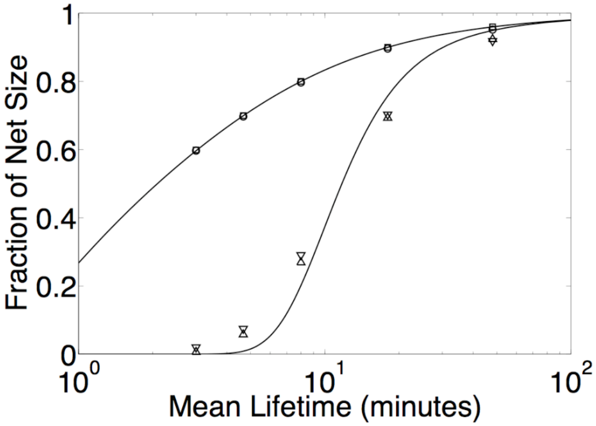
$N=2^{20}$ nodes

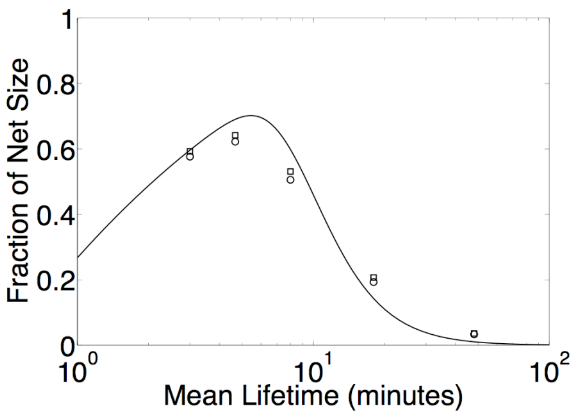




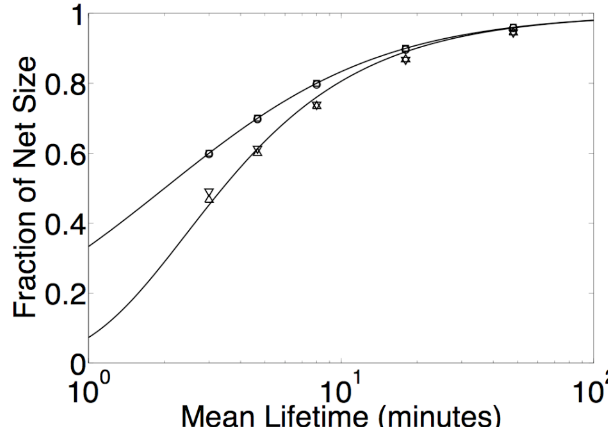
Analysis vs. Simulation (Churn)

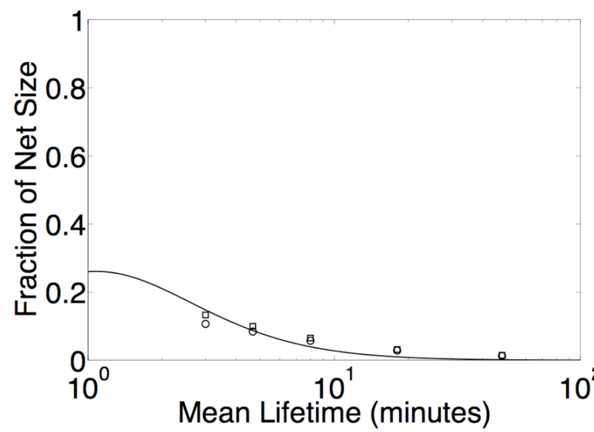
Symphony (k_S=2, k_L=2)





Chord



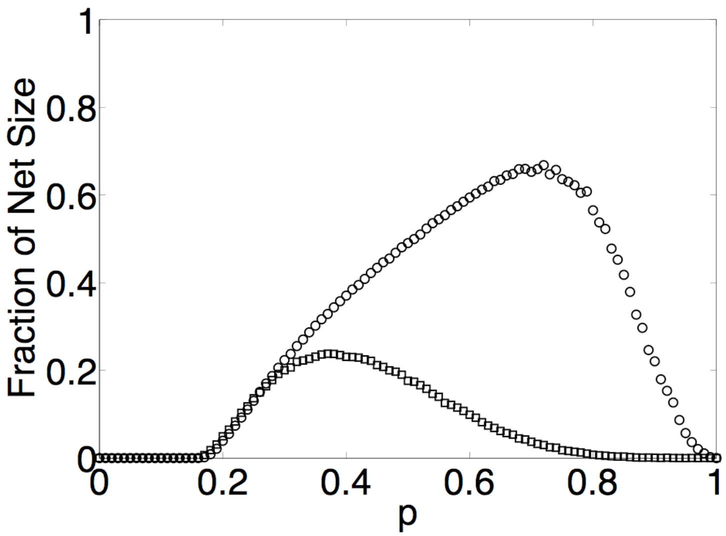


S=2 min

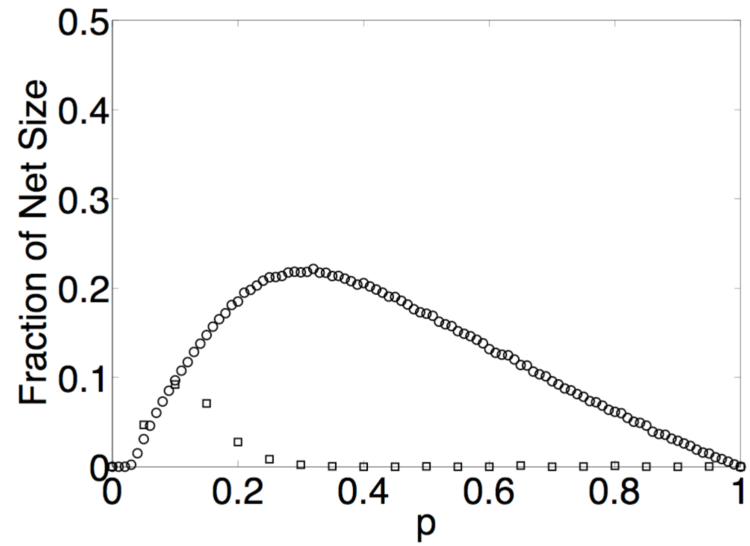
$N=2^{20}$ nodes

Mitigation Techniques

Symphony
(neighbor-of-neighbor)



Chord
(20 successors)



$N=2^{20}$ nodes

UCLA

Conclusion

- We showed that the churn model is reducible to the more tractable uniform failure model;
- We determined via analysis and simulation the *price of structured routing* for Symphony and Chord;
- Our work is a contribution to the further understanding of the resilience properties of P2P networks.