On Lifetime-Based Node Failure and Stochastic Resilience of Decentralized Peer-to-Peer Networks

Derek Leonard, Vivek Rai, and Dmitri Loguinov
Presented by Xiaoming Wang

Department of Computer Science
Texas A&M University
College Station, TX 77843

8th June 2005
Outline

1. Background
   - Motivation

2. Lifetime-Based Resilience
   - Expected Time to Isolation
   - Probability of Isolation
   - Varying Node Degree

3. Global P2P Resilience
   - Classical Result
   - Static Failure
   - Lifetime-Based Extension
Outline

1. Background
   - Motivation

2. Lifetime-Based Resilience
   - Expected Time to Isolation
   - Probability of Isolation
   - Varying Node Degree

3. Global P2P Resilience
   - Classical Result
   - Static Failure
   - Lifetime-Based Extension
Previous Techniques

- Traditional study of P2P resilience centers around uniform, independent, simultaneous node failure
  - Nodes fail with independent probability $p$
Motivation I

Previous Techniques

- Traditional study of P2P resilience centers around uniform, independent, simultaneous node failure
  - Nodes fail with independent probability $p$
- The analysis of Chord is a typical example of this
  - Using $p = 0.5$, the paper determines what node degree is necessary to ensure that each node stays connected (i.e., is not isolated) with high probability after the failure
Previous Techniques

- Traditional study of P2P resilience centers around uniform, independent, simultaneous node failure
  - Nodes fail with independent probability $p$
- The analysis of Chord is a typical example of this
  - Using $p = 0.5$, the paper determines what node degree is necessary to ensure that each node stays connected (i.e., is not isolated) with high probability after the failure
- For Chord, we have:

\[ P(\text{isolated}) = p^{\text{degree}} \leq \frac{1}{n} \]
Previous Techniques

- Traditional study of P2P resilience centers around uniform, independent, simultaneous node failure
  - Nodes fail with independent probability $p$
- The analysis of Chord is a typical example of this
  - Using $p = 0.5$, the paper determines what node degree is necessary to ensure that each node stays connected (i.e., is not isolated) with high probability after the failure
- For Chord, we have:

  \[ P(\text{isolated}) = p^{\text{degree}} \leq \frac{1}{n} \]

- Example: $n = 100$ billion, $k$ must be at least 37
What can be said about node failure in real-world P2P systems?
Motivation II

**Lifetime-based Node Failure**

- What can be said about node failure in real-world P2P systems?
  - The $p$-percent model may be useful in some cases; however, there is no evidence that such failure patterns occur in real P2P networks.
What can be said about node failure in real-world P2P systems?

- The \( p \)-percent model may be useful in some cases; however, there is no evidence that such failure patterns occur in real P2P networks.
- Nodes arrive/depart dynamically instead of remaining static.
Motivation II

Lifetime-based Node Failure

- What can be said about node failure in real-world P2P systems?
  - The $p$-percent model may be useful in some cases; however, there is no evidence that such failure patterns occur in real P2P networks
  - Nodes arrive/depart dynamically instead of remaining static
- Model: we assign each user a random lifetime $L_i$ from a distribution $F(x)$ that reflects the behavior of the user and represents the duration of his/her service (e.g., sharing files) to the system
Model Assumptions

- **Arrival**: nodes arrive randomly according to any process; however, their arrival times are uncorrelated with lifetimes of existing nodes.
Overview of Lifetime Model I

Model Assumptions

- Arrival: nodes arrive randomly according to any process; however, their arrival times are uncorrelated with lifetimes of existing nodes.
- Departure: nodes deterministically die (fail) after spending $L_i$ time units in the system.
Overview of Lifetime Model I

Model Assumptions

- **Arrival**: nodes arrive randomly according to any process; however, their arrival times are uncorrelated with lifetimes of existing nodes.

- **Departure**: nodes deterministically die (fail) after spending $L_i$ time units in the system.

- **Neighbor selection**: neighbors are picked from among the existing nodes using any rules that do not involve node lifetimes or age (e.g., based on random walks, DHT space assignment, topological locality, content interests, etc.).
Overview of Lifetime Model I

Model Assumptions

- Arrival: nodes arrive randomly according to any process; however, their arrival times are uncorrelated with lifetimes of existing nodes.
- Departure: nodes deterministically die (fail) after spending $L_i$ time units in the system.
- Neighbor selection: neighbors are picked from among the existing nodes using any rules that do not involve node lifetimes or age (e.g., based on random walks, DHT space assignment, topological locality, content interests, etc.).
- Neighbor replacement: once a failed neighbor is detected, a replacement search is performed.
Overview of Lifetime Model II

Definition

A node becomes isolated when all of the neighbors in its table are in the failed state.
Definition
A node becomes isolated when all of the neighbors in its table are in the failed state

Node Departure
- All departures are considered to be abrupt, requiring each node to search for a replacement upon failure of its neighbor
Overview of Lifetime Model III

Lifetimes of Neighbors

- Node $v$ enters at time $t_v$ with random lifetime $L_v$
- The $k$ neighbors of $v$ are represented by residual lifetimes
Overview of Lifetime Model III

Lifetimes of Neighbors

- Node $v$ enters at time $t_v$ with random lifetime $L_v$
- The $k$ neighbors of $v$ are represented by residual lifetimes

![Diagram showing lifetimes of neighbors with $t_v$, $R_1$, and $R_4$.]
Overview of Lifetime Model III

Lifetimes of Neighbors

- Node $v$ enters at time $t_v$ with random lifetime $L_v$
- The $k$ neighbors of $v$ are represented by residual lifetimes $R_i$

Definition

Let $R_i$ be the remaining lifetime of neighbor $i$ when $v$ joined the system
Overview of Lifetime Model IV

Formalizing Search Time

- How do nodes replace neighbors?
Overview of Lifetime Model IV

Formalizing Search Time

- How do nodes replace neighbors?
  - There is usually some mechanism for detecting that a neighbor has failed (e.g. periodic probing, etc.)
  - Systems often repair the failed zone of a DHT or find a random replacement neighbor in unstructured systems
Overview of Lifetime Model IV

### Formalizing Search Time

- **How do nodes replace neighbors?**
  - There is usually some mechanism for detecting that a neighbor has failed (e.g. periodic probing, etc.)
  - Systems often repair the failed zone of a DHT or find a random replacement neighbor in unstructured systems

- **We allow this process to be arbitrary as the technique employed has no effect on our results**
How do nodes replace neighbors?

- There is usually some mechanism for detecting that a neighbor has failed (e.g. periodic probing, etc.)
- Systems often repair the failed zone of a DHT or find a random replacement neighbor in unstructured systems

We allow this process to be arbitrary as the technique employed has no effect on our results

Let $S_i$ be a random variable describing the total search time for the $i$-th replacement in the system
Overview of Lifetime Model V

Example

- Reconsider the same Chord system given before:
  - $n = 100$ billion nodes
  - $E[L_i] = 30$ minutes
  - $E[S_i] = 1$ minute
Reconsider the same Chord system given before:

- $n = 100$ billion nodes
- $E[L_i] = 30$ minutes
- $E[S_i] = 1$ minute

Classical analysis requires $k = 37$ to ensure that a given node remains connected with high probability.
Example

- Reconsider the same Chord system given before:
  - $n = 100$ billion nodes
  - $E[L_i] = 30$ minutes
  - $E[S_i] = 1$ minute
- Classical analysis requires $k = 37$ to ensure that a given node remains connected with high probability
- Using the lifetime model we find that the same bound can be achieved with $k = 9$
Overview of Lifetime Model V

Example

- Reconsider the same Chord system given before:
  - \( n = 100 \) billion nodes
  - \( E[L_i] = 30 \) minutes
  - \( E[S_i] = 1 \) minute

- Classical analysis requires \( k = 37 \) to ensure that a given node remains connected with high probability

- Using the lifetime model we find that the same bound can be achieved with \( k = 9 \)

- P2P systems are more resilient than we thought!
Overview of Lifetime Model VI

Pertinent Questions

- What questions can we now address given this lifetime node-failure model for P2P networks?
Overview of Lifetime Model VI

Pertinent Questions

- What questions can we now address given this lifetime node-failure model for P2P networks?
  - What is the average amount of time a node will spend in the system before becoming isolated?
Overview of Lifetime Model VI

Pertinent Questions

- What questions can we now address given this lifetime node-failure model for P2P networks?
  - What is the average amount of time a node will spend in the system before becoming isolated?
  - What is the probability that a node will become isolated from the network within its lifetime?
Pertinent Questions

- What questions can we now address given this lifetime node-failure model for P2P networks?
  - What is the average amount of time a node will spend in the system before becoming isolated?
  - What is the probability that a node will become isolated from the network within its lifetime?
  - How does varying node degree between users improve/degree resilience?
Overview of Lifetime Model VI

Pertinent Questions

- What questions can we now address given this lifetime node-failure model for P2P networks?
  - What is the average amount of time a node will spend in the system before becoming isolated?
  - What is the probability that a node will become isolated from the network within its lifetime?
  - How does varying node degree between users improve/degree resilience?
  - How does the absence of isolated vertices affect global resilience of the network (i.e., its connectivity)?
Outline

1 Background
   - Motivation

2 Lifetime-Based Resilience
   - Expected Time to Isolation
   - Probability of Isolation
   - Varying Node Degree

3 Global P2P Resilience
   - Classical Result
   - Static Failure
   - Lifetime-Based Extension
Let $T$ be a random variable describing the amount of time a node can spend in the system before becoming isolated.
Expected Time to Isolation

Let $T$ be a random variable describing the amount of time a node can spend in the system before becoming isolated.

Assuming relatively small search delays, we use renewal process theory to derive the following:

$$E[T] \approx \frac{E[S_i]}{k} \left[ \left( 1 + \frac{E[R_i]}{E[S_i]} \right)^k - 1 \right]$$
Expected Time to Isolation

Let $T$ be a random variable describing the amount of time a node can spend in the system before becoming isolated.

Assuming relatively small search delays, we use renewal process theory to derive the following:

$$E[T] \approx \frac{E[S_i]}{k} \left[ \left(1 + \frac{E[R_i]}{E[S_i]}\right)^k - 1 \right]$$

Despite the approximation, simulations show that the model is very accurate and not sensitive to lifetime or search delay distribution.
Simulations

Simulations were run with average lifetime 30 minutes and $k = 10$ for a 1000 node system. Four distributions of $S_i$ were used.

(a) uniform $S_i$

(b) binomial $S_i$
Expected Time to Isolation III

(c) exponential $S_i$

(d) Pareto $S_i$ with $\alpha = 3$
Example

- Consider an example Chord system
  - \( n = 1 \) million (average distance of 10 hops)
  - keep-alive timeout \( \delta \)
  - Average inter-peer delay \( d = 200 \) ms
  - \( E[R_i] = 1 \) hour
Expected Time to Isolation IV

Example

- Consider an example Chord system
  - \( n = 1 \) million (average distance of 10 hops)
  - keep-alive timeout \( \delta \)
  - Average inter-peer delay \( d = 200 \) ms
  - \( E[R_i] = 1 \) hour

- We immediately obtain from the main model:

\[
E[T] = \frac{\delta + d \log_2 n}{2k} \left(1 + \frac{2E[R_i]}{\delta + d \log_2 n}\right)^k
\]
Expected Time to Isolation V

<table>
<thead>
<tr>
<th>Timeout $\delta$</th>
<th>$k = 20$</th>
<th>$k = 10$</th>
<th>$k = 5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 sec</td>
<td>$10^{41}$ years</td>
<td>$10^{17}$ years</td>
<td>188,034 years</td>
</tr>
<tr>
<td>2 min</td>
<td>$10^{28}$ years</td>
<td>$10^{11}$ years</td>
<td>282 years</td>
</tr>
<tr>
<td>45 min</td>
<td>404,779 years</td>
<td>680 days</td>
<td>49 hours</td>
</tr>
</tbody>
</table>

Table: Expected time $E[T]$ to isolation

Example Continued

Notice that for small keep-alive delays, even $k = 5$ provides longer expected time to isolation than the lifetime of any human.
Outline

1 Background
   - Motivation

2 Lifetime-Based Resilience
   - Expected Time to Isolation
   - Probability of Isolation
   - Varying Node Degree

3 Global P2P Resilience
   - Classical Result
   - Static Failure
   - Lifetime-Based Extension
Questions to Answer

What is the probability $\pi$ that a node will become isolated from the network during its lifetime?
Questions to Answer

- What is the probability $\pi$ that a node will become isolated from the network during its lifetime?
  - Let $\pi = P(T < L_v)$
### Probability of Isolation I

#### Questions to Answer

- What is the probability $\pi$ that a node will become isolated from the network during its lifetime?
  - Let $\pi = P(T < L_v)$

- The exact distribution of $T$ is difficult to develop in closed-form for non-exponential lifetimes
Questions to Answer

- What is the probability \( \pi \) that a node will become isolated from the network during its lifetime?
  - Let \( \pi = P(T < L_v) \)
- The exact distribution of \( T \) is difficult to develop in closed-form for non-exponential lifetimes
- We model the neighbor failure/replacement procedure as an on/off process \( Y_i(t) \)

\[
\begin{align*}
Y_1(t) & \quad \text{on} \quad \text{off} \\
\vdots & \quad R_i \quad S_i \\
Y_k(t) & \quad \text{on} \quad \text{off}
\end{align*}
\]
Then the degree of node \( v \) at time \( t \) is:

\[
W(t) = \sum_{i=1}^{k} Y_i(t)
\]
Result

Using Markov Chain arguments based on $W(t)$ for exponential lifetimes and $E[S_i] \ll E[L_i]$, the probability of isolation $\pi$ converges to:

$$\pi = \frac{E[L_i]}{E[T]}$$
Result

- Using Markov Chain arguments based on $W(t)$ for exponential lifetimes and $E[S_i] \ll E[L_i]$, the probability of isolation $\pi$ converges to:

$$\pi = \frac{E[L_i]}{E[T]}$$

- Simulations match the model remarkably well and the results are not sensitive to the distribution of search delay
Simulations

We simulated a system with $E[L_i] = 0.5$ and $k = 10$ using four search distributions to verify the model.

(e) exponential $S_i$

(f) constant $S_i$
Probability of Isolation V

(g) uniform $S_i$

(h) Pareto $S_i$ with $\alpha = 3$

Simulations
As $E[S_i]$ becomes small the simulations converge to the model
Application to Pareto Lifetimes

We use the exponential result to derive an upper bound for any lifetime distribution with an exponential or heavier tail:

$$\pi \leq \frac{kE[L_i]E[S_i]^{k-1}}{(E[L_i] + E[S_i])^k - E[S_i]^k}$$
Application to Pareto Lifetimes

We use the exponential result to derive an upper bound for any lifetime distribution with an exponential or heavier tail:

$$\pi \leq \frac{k E[L_i] E[S_i]^{k-1}}{(E[L_i] + E[S_i])^k - E[S_i]^k}$$

<table>
<thead>
<tr>
<th>$\pi$</th>
<th>Uniform lifetime $p = 1/2$</th>
<th>Lifetime P2P Simulations</th>
<th>Mean search time $E[S_i]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-6}$</td>
<td>20 Bound $\pi$ simulations</td>
<td>10 7 5</td>
<td>6 min 2 min 20 sec</td>
</tr>
<tr>
<td>$10^{-9}$</td>
<td>30 Bound $\pi$ simulations</td>
<td>14 9 6</td>
<td></td>
</tr>
</tbody>
</table>
Outline

1. Background
   - Motivation

2. Lifetime-Based Resilience
   - Expected Time to Isolation
   - Probability of Isolation
   - Varying Node Degree

3. Global P2P Resilience
   - Classical Result
   - Static Failure
   - Lifetime-Based Extension
Effect of Degree Regularity on Resilience

- How does the varying node degree among users improve/degrade resilience?
Effect of Degree Regularity on Resilience

- How does the varying node degree among users improve/degrade resilience?
  - In particular, are DHTs more resilient than unstructured systems?
Varying Node Degree I

Effect of Degree Regularity on Resilience

- How does the varying node degree among users improve/degrade resilience?
  - In particular, are DHTs more resilient than unstructured systems?
  - Recall that average degree is constant and node lifetimes are independent of degree and are not used in the neighbor-selection process.
Effect of Degree Regularity on Resilience

- How does the varying node degree among users improve/degrade resilience?
  - In particular, are DHTs more resilient than unstructured systems?
  - Recall that average degree is constant and node lifetimes are independent of degree and are not used in the neighbor-selection process

Theorem

*Under the above assumptions, degree-regular graphs are the most resilient for a given average degree $E[k_i]$*
Simulations

We verify finding on four different systems with average degree $E[k_i] = 10$ and Pareto lifetimes with $E[L_i] = 0.5$ hours.
When degree is independent of user lifetimes, we find no evidence to suggest that unstructured P2P systems with a heavy-tailed (or other irregular) degree can provide better resilience than $k$-regular DHTs.
When degree is independent of user lifetimes, we find no evidence to suggest that unstructured P2P systems with a heavy-tailed (or other irregular) degree can provide better resilience than $k$-regular DHTs.

Varying node degree from peer to peer can have a positive impact on resilience *only* when decisions are correlated with lifetimes.
Outline

1 Background
   • Motivation

2 Lifetime-Based Resilience
   • Expected Time to Isolation
   • Probability of Isolation
   • Varying Node Degree

3 Global P2P Resilience
   • Classical Result
   • Static Failure
   • Lifetime-Based Extension
Effect of Isolated Nodes

- How does the absence of isolated vertices affect the network’s connectivity?
  - This topic has been researched extensively in random graph theory and interconnection networks.
Effect of Isolated Nodes

- How does the absence of isolated vertices affect the network’s connectivity?
  - This topic has been research extensively in random graph theory and interconnection networks
- Erdös and Rényi in the 1960s demonstrated that almost every (i.e., with probability $1 - o(1)$ as $n \to \infty$) random graph is connected if and only if it has no isolated vertices.

$$P(G \text{ is connected}) = P(G \text{ has no isolated nodes})$$
Effect of Isolated Nodes

- How does the absence of isolated vertices affect the network’s connectivity?
  - This topic has been research extensively in random graph theory and interconnection networks
- Erdös and Rényi in the 1960s demonstrated that almost every (i.e., with probability $1 - o(1)$ as $n \to \infty$) random graph is connected if and only if it has no isolated vertices.

$$P(G \text{ is connected}) = P(G \text{ has no isolated nodes})$$

- Almost every disconnection occurs with at least one isolation
Outline

1. Background
   - Motivation

2. Lifetime-Based Resilience
   - Expected Time to Isolation
   - Probability of Isolation
   - Varying Node Degree

3. Global P2P Resilience
   - Classical Result
   - Static Failure
   - Lifetime-Based Extension
Burtin (1977) and Bollobás (1983) showed that the same result applies to certain deterministic graphs such as hypercubes. This can be extended to any graph with similar or better node expansion properties (Chord, CAN, Pastry, etc.).

Table: Chord with $n = 16384$ under $p$-percent failure

<table>
<thead>
<tr>
<th>$p$</th>
<th>$P(G$ is connected)</th>
<th>$P$ (no isolated nodes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.99996</td>
<td>0.99996</td>
</tr>
<tr>
<td>0.6</td>
<td>0.99354</td>
<td>0.99354</td>
</tr>
<tr>
<td>0.7</td>
<td>0.72619</td>
<td>0.72650</td>
</tr>
<tr>
<td>0.8</td>
<td>0.00040</td>
<td>0.00043</td>
</tr>
</tbody>
</table>
Application to P2P graphs

The tested P2P graphs (Chord, Symphony, CAN, Pastry, Randomized Chord, de Bruijn, and several unstructured random graphs) remained connected almost surely as long as they did not have an isolated node.
Application to P2P graphs

- The tested P2P graphs (Chord, Symphony, CAN, Pastry, Randomized Chord, de Bruijn, and several unstructured random graphs) remained connected almost surely as long as they did not have an isolated node.

- When they did disconnect, an isolated node almost surely existed.
Application to P2P graphs

- The tested P2P graphs (Chord, Symphony, CAN, Pastry, Randomized Chord, de Bruijn, and several unstructured random graphs) remained connected almost surely as long as they did not have an isolated node.

- When they did disconnect, an isolated node almost surely existed.

Implication

*Local resilience of popular P2P networks implies their global resilience*
We now apply this result to the lifetime-based model for node failure.

Recall that the probability of isolation $\pi_v = P(T < L_v)$ for node $v$.

Problem

What is the probability that a graph $G$ survives $N$ user joins without disconnecting?
We now apply this result to the lifetime-based model for node failure.

Instead of $p$-percent failure, we use the probability of isolation $\pi$ associated with each joining user $i$. 

Recall that the probability of isolation $\pi = P(T < L_v)$ for node $v$. 

Problem:
What is the probability that a graph $G$ survives $N$ user joins without disconnecting?
Application to Lifetime Model

- We now apply this result to the lifetime-based model for node failure.
  - Instead of $p$-percent failure, we use the probability of isolation $\pi$ associated with each joining user $i$.
- Recall that the probability of isolation $\pi = P(T < L_v)$ for node $v$. 
Application to Lifetime Model

- We now apply this result to the lifetime-based model for node failure.
  - Instead of $p$-percent failure, we use the probability of isolation $\pi$ associated with each joining user $i$.
- Recall that the probability of isolation $\pi = P(T < L_v)$ for node $v$.

Problem

What is the probability that a graph $G$ survives $N$ user joins without disconnecting?
Definition

Let $Y$ be a geometric random variable measuring the number of user joins before the first disconnection of the network.
Definition

Let $Y$ be a geometric random variable measuring the number of user joins before the first disconnection of the network.

Model

Then, for almost every sufficiently large graph:

$$P(Y > N) = (1 - \pi)^N$$
Definition

Let $Y$ be a geometric random variable measuring the number of user joins before the first disconnection of the network.

Model

Then, for almost every sufficiently large graph:

$$P(Y > N) = (1 - \pi)^N$$

We measured the probability that the graph disconnects with exactly one isolated node. We found this metric to be 1 for all simulations!
### Simulations

Consider $k$-regular CAN with exponential lifetimes of mean 30 minutes.

<table>
<thead>
<tr>
<th>Search time</th>
<th>Actual $P(Y &gt; N)$</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0.9732</td>
<td>0.9728</td>
</tr>
<tr>
<td>7.5</td>
<td>0.8118</td>
<td>0.8124</td>
</tr>
<tr>
<td>8.5</td>
<td>0.5669</td>
<td>0.5659</td>
</tr>
<tr>
<td>9</td>
<td>0.4065</td>
<td>0.4028</td>
</tr>
<tr>
<td>9.5</td>
<td>0.2613</td>
<td>0.2645</td>
</tr>
<tr>
<td>10.5</td>
<td>0.0482</td>
<td>0.0471</td>
</tr>
</tbody>
</table>

Table: Comparison of $P(Y > 10^6)$ in CAN to the model
Simulations

- Consider $k$-regular CAN with exponential lifetimes of mean 30 minutes.
- The graph has $d = 6$ dimensions and degree $k = 12$.

Table: Comparison of $P(Y > 10^6)$ in CAN to the model.

<table>
<thead>
<tr>
<th>Search time</th>
<th>Actual $P(Y &gt; N)$</th>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0.9732</td>
<td>0.9728</td>
</tr>
<tr>
<td>7.5</td>
<td>0.8118</td>
<td>0.8124</td>
</tr>
<tr>
<td>8.5</td>
<td>0.5669</td>
<td>0.5659</td>
</tr>
<tr>
<td>9</td>
<td>0.4065</td>
<td>0.4028</td>
</tr>
<tr>
<td>9.5</td>
<td>0.2613</td>
<td>0.2645</td>
</tr>
<tr>
<td>10.5</td>
<td>0.0482</td>
<td>0.0471</td>
</tr>
</tbody>
</table>

D. Leonard, V. Rai, D. Loguinov

Lifetime-Based Node Failure and Stochastic Resilience of P2P Networks
Simulations

- Consider $k$-regular CAN with exponential lifetimes of mean 30 minutes
- The graph has $d = 6$ dimensions and degree $k = 12$
- In this case we test $N = 10^6$

<table>
<thead>
<tr>
<th>Search time</th>
<th>Actual $P(Y &gt; N)$</th>
<th>Model $P(Y &gt; N)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0.9732</td>
<td>0.9728</td>
</tr>
<tr>
<td>7.5</td>
<td>0.8118</td>
<td>0.8124</td>
</tr>
<tr>
<td>8.5</td>
<td>0.5669</td>
<td>0.5659</td>
</tr>
<tr>
<td>9</td>
<td>0.4065</td>
<td>0.4028</td>
</tr>
<tr>
<td>9.5</td>
<td>0.2613</td>
<td>0.2645</td>
</tr>
<tr>
<td>10.5</td>
<td>0.0482</td>
<td>0.0471</td>
</tr>
</tbody>
</table>

Table: Comparison of $P(Y > 10^6)$ in CAN to the model
### Simulations

- Consider $k$-regular CAN with exponential lifetimes of mean 30 minutes.
- The graph has $d = 6$ dimensions and degree $k = 12$.
- In this case we test $N = 10^6$.
- The simulations match the model very well.

### Table: Comparison of $P(Y > 10^6)$ in CAN to the model

<table>
<thead>
<tr>
<th>Search time</th>
<th>Actual $P(Y &gt; N)$</th>
<th>Model $P(Y &gt; N)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0.9732</td>
<td>0.9728</td>
</tr>
<tr>
<td>7.5</td>
<td>0.8118</td>
<td>0.8124</td>
</tr>
<tr>
<td>8.5</td>
<td>0.5669</td>
<td>0.5659</td>
</tr>
<tr>
<td>9</td>
<td>0.4065</td>
<td>0.4028</td>
</tr>
<tr>
<td>9.5</td>
<td>0.2613</td>
<td>0.2645</td>
</tr>
<tr>
<td>10.5</td>
<td>0.0482</td>
<td>0.0471</td>
</tr>
</tbody>
</table>
Consider the same CAN system with 1-minute search delays with all $10^6$ users joining and leaving once each day.
Example

Consider the same CAN system with 1-minute search delays with all $10^6$ users joining and leaving once each day. The probability that the graph will survive for 2,700 years is 0.9956.
Example

- Consider the same CAN system with 1-minute search delays with all $10^6$ users joining and leaving once each day
- The probability that the graph will survive for 2,700 years is 0.9956

Implication

The mean delay to disconnection of the graph is 5.9 million years
Conclusion

Under all practical search times, $k$-regular graphs are much more resilient than traditionally implied P2P systems that endure churn will almost surely remain connected as long as no user suffers isolation from the system. Varying node degree from peer to peer can have a positive impact on resilience only when decisions are correlated with lifetimes.

Local resilience implies global resilience.

Findings
Under all practical search times, $k$-regular graphs are much more resilient than traditionally implied.
Conclusion

Findings

- Under all practical search times, $k$-regular graphs are much more resilient than traditionally implied.
- P2P systems that endure churn will almost surely remain connected as long as no user suffers isolation from the system.
Conclusion

Findings

- Under all practical search times, $k$-regular graphs are much more resilient than traditionally implied.
- P2P systems that endure churn will almost surely remain connected as long as no user suffers isolation from the system.
- Varying node degree from peer to peer can have a positive impact on resilience only when decisions are correlated with lifetimes.
- *Local resilience implies global resilience*