On Static and Dynamic Partitioning Behavior of Large-Scale Networks

Derek Leonard, Zhongmei Yao, Xiaoming Wang, and Dmitri Loguinov*
Department of Computer Science
Texas A&M University, College Station, TX 77843
{dleonard, mayyao, xmwang, dmitri}@cs.tamu.edu

Abstract

In this paper, we analyze the problem of network disconnection in the context of large-scale P2P networks and understand how both static and dynamic patterns of node failure affect the resilience of such graphs. We start by applying classical results from random graph theory to show that a large variety of deterministic and random P2P graphs almost surely (i.e., with probability $1 - o(1)$) remain connected under random failure if and only if they have no isolated nodes. This simple, yet powerful, result subsequently allows us to derive in closed-form the probability that a P2P network develops isolated nodes, and therefore partitions, under both types of node failure. We finish the paper by demonstrating that our models match simulations very well and that dynamic P2P systems are extremely resilient under node churn as long as the neighbor replacement delay is much smaller than the average user lifetime.

1. Introduction

During the recent explosion of P2P research, network resilience has become an important issue [17], [19], [29], [38]. The primary interest in this line of study is to understand how dynamic user arrivals and abrupt departures affect the connectivity (and sometimes other metrics) of the system. The original thrust [20], [19], [38] in this direction focused on *static* node failure, where a fully-populated network experienced simultaneous node failures with independent probability p. While analytical results on the exact probability of disconnection under static failure are currently unavailable in the literature, prior analysis suggests that P2P networks are highly resilient to node faults and can survive the failure of up to 50% of the graph without significant degradation in performance [38].

Since users in P2P networks rarely fail simultaneously [5], a different approach [23], [26], [32] is to examine disconnection in *dynamic* systems, where users continuously join and leave the network according to some arrival/departure processes. The only analytical results available on the dynamic resilience of generic P2P networks correlate the rate of churn with user notification frequency [26] and examine how stabilization delays affect the consistency of Chord's finger table [23].

In this paper, we bridge the gap between static and dynamic disconnection analysis and show that the problem of graph partitioning under both types of failure can be reduced to computation of the probability that a P2P network develops at least one isolated node during the failure. Under the umbrella of this unifying model, we then derive a closed-form model for static resilience and examine the same issue in dynamic networks where users depart the system after spending random amounts of time online. Our results show that under p-percent static failure, almost every sufficiently large k-regular P2P graph G of n nodes remains connected with probability:

$$P(G \text{ is connected}) = e^{-n(1-p)p^k}. \tag{1}$$

Using Chord's degree $k = \log_2 n$ and the commonly used failure probability $p = 1/2$ [20], [38], it immediately follows that Chord remains connected after 50%-percent failure with probability $e^{-0.5} \approx 0.6$. Also notice that for $p < 1/2$, this probability converges to 1 (i.e., almost every graph is connected) as $n \to \infty$ and for $p > 1/2$, it converges to 0 (i.e., almost every graph is disconnected).

Outside of static resilience, our second result is the derivation of disconnection probabilities for dynamic systems, which frequently exhibit high levels of churn [5], [26] and are more mathematically elusive. To capture user behavior in such systems, we propose a simple node-failure model in which users stay in the system for random periods of time before deterministically failing at the end of their lifetime. To maintain a resilient system, we assume that each node monitors its neighbors and randomly re-

* Supported by NSF grants CCR-0306246, ANI-0312461, and CNS-0434940.

places[1] them upon detecting their failure. Replacement delays S_i and lifetimes L_i are drawn from some (possibly heavy-tailed) distributions and generally determine the resilience of the system. Our main result demonstrates that dynamic k-regular P2P systems can survive N user joins without partitioning with probability at least:

$$P(G \text{ survives } N \text{ joins}) \geq \left(1 - \frac{\rho k}{(1+\rho)^k + \rho k - 1}\right)^N,$$

where $\rho = E[L_i]/E[S_i]$ is the ratio of the mean user lifetime to the mean neighbor replacement delay. To understand this result, consider the following example. Given a system with 5 million users that join the network once a day, $k = 12$ neighbors per node, mean user lifetime of 0.5 hours, and 1-minute search delay (i.e., $\rho = 30$), the probability that the network survives for $10,000$ years without disconnecting is 99.2%.

This paper is organized as follows. Section 2 examines previous work. Section 3 discusses how isolated nodes affect graph connectivity under both static and dynamic node failure. Section 4 focuses on static resilience and Section 5 discusses the dynamic case. Section 6 describes some implications of our results to real-world systems. Section 7 concludes the paper.

2. Background

2.1. Random Graph Theory

One of the first approaches to network reliability stems from random graph theory. The issue of partitioning and disconnection of random graphs $G(n,p)$ has a long history [14]. It is well-known that, as with any other monotone property, connectivity of $G(n,p)$ experiences a sharp transition from "almost never" to "almost always" at the threshold $p = \log n/n$; however, a more powerful result states that $G(n,p)$ and all of its derivatives [8], [33] are almost surely connected *if and only if they have no isolated nodes*. Defining $\Phi(G)$ to be the probability that a random graph remains connected under node or edge failure and assuming X is the number of isolated nodes in the graph, the following holds with probability $1 - o(1)$ as the size of the graph $n \to \infty$:

$$\Phi(G) = P(X = 0). \tag{2}$$

2.2. Deterministic Graphs

After some technical manipulation, a result similar to (2) can be shown to hold for certain deterministic networks as well. For example, Burtin [9] and later Bollobas [7] prove that under independent uniform failure, hypercubes are almost surely connected if and only if they have no isolated nodes. Intuitively, this result means that the conditional probability that a hypercube partitions along a set boundary[2] ∂S, for some non-trivial set S, while having no isolated nodes is $o(1)$ as $n \to \infty$. We leverage these observations later in the paper.

Connectivity of *generic* deterministic graphs $G = (V,E)$ under independent node failure has also received significant attention in the literature [6], [16], [21]. In this line of work, $\Phi(G)$ is called *residual node connectivity* and can be written as:

$$\Phi(G) = \sum_{i=1}^{n} S_i(G) p^{n-i}(1-p)^i,$$

where p is the failure probability of each node and $S_i(G)$ is the number of connected induced subgraphs of G with exactly i nodes [6]. While this closed-form expansion is beneficial for simple graphs (such as trees), computation of $\Phi(G)$ for a generic graph requires the knowledge of an NP-complete [39] metric $S_i(G)$, whose expression is unknown even for the basic hypercube.

Najjar and Gaudiot [30], however, noticed that several non-hypercube deterministic networks frequently develop disconnections around individual nodes rather than along boundaries of larger sets $S, |S| \geq 2$. This lead to the following model for the probability that an n-node, k-regular graph partitions under p-percent node failure [30]:

$$\Phi(G) = \sum_{i=0}^{n} Q_i \binom{n}{i} p^i (1-p)^{n-i}, \tag{3}$$

where

$$Q_i = \prod_{j=1}^{i} \left[1 - \frac{k(n-k-1)!(j-1)!(n-j)}{(n-1)!(j-k)!}\right]. \tag{4}$$

Other approaches that study disconnection of hypercubes include [12], [15], [18], [24]; however, none of them provide a practically usable model that is both accurate and simple to evaluate.

2.3. P2P Resilience

Given the wide variety of recently developed P2P systems, several techniques have been employed to evaluate the resilience of such graphs. One commonly-used method is to monitor several performance metrics (e.g., percentage of successful queries, graph connectivity, consistency of links) under node failure and show how they change depending

[1] Replacement in DHTs is simply the predecessor taking over the failed zone, while that in unstructured systems may rely on a variety of active neighbor selection strategies not essential to our analysis.

[2] All nodes $u \in V \backslash S$ such that $(u,v) \in E, v \in S$.

on system parameters. A seminal paper in this genre written by Gummadi *et al.* [19] explores the impact of different routing geometries on the static resilience of the graph, which is defined as the ability of the graph to route messages *before* the designed recovery algorithm repairs the graph. Other papers that examine static resilience in a similar fashion are [27], [34], and [36]. A more recent study by Chun *et al.* [13] uses simulations to analyze the impact of different types of neighbor-selection algorithms on static resilience of P2P graphs under both random node failures and targeted attacks. The paper demonstrates that there is a distinct tradeoff between resilience and system performance.

The second approach is more analytical in nature. Chord [38] and Koorde [20] show that under independent uniform node failure, k-regular graphs require degree $k \geq \log_{1/p} n$ in order to upper-bound the probability of individual node isolation by $1/n$. Massoulie *et al.* [17], [29] develop a new P2P system based on random graphs and derive the probability that it remains connected under p-percent failure. Liben-Nowell *et al.* in [26] study the dynamic nature of P2P systems in regards to joins and unexpected departures and their impact on routing efficiency. The authors derive a lower bound on the number of users a node must be notified about in order for the system to avoid disconnection. In a more recent paper, Krishnamurthy *et al.* [23] focus on predicting the state of each finger pointer in a Chord system under dynamic failure conditions. They derive a probabilistic characterization of each neighbor and successor pointer, which allows them to obtain models for the percentage of failed queries in the system under user churn.

3. Unifying Model of Disconnection

In this section, we discuss how connectivity of P2P systems under static and dynamic node-failure patterns can be reduced to the problem of node isolation.

3.1. Generic Disconnection Model

We first turn to the question of what properties a graph G must possess in order to satisfy (2) under random edge and node failure. Interestingly, the property that makes hypercubes (and classical random graphs) very unlikely to partition into non-trivial subgraphs *without* developing isolated nodes is that the number of edges leaving *each* set S is an *increasing* function of set size $|S|$. Burtin [9] showed that for each set S in a hypercube, the size of its edge boundary[3] is at least:

$$|\{(u,v) \in E : u \in S, v \in V \backslash S\}| \geq |S|(k - \log_2 |S|), \quad (5)$$

where $k = \log_2 n$ is the degree of the graph. Condition (5) states that larger sets S are *always* better connected than smaller sets (up to half the graph in size) and ensures that the probability that any large subgraph disconnects after node failure is negligible compared to that of individual node isolation.

While the necessary conditions on G for (2) to hold are generally unknown, one can formulate a simple sufficient condition as stated below.

Proposition 1. *If a graph G has node expansion properties no worse than those of hypercubes or random graphs (as defined in [8]) of the same size, it will remain almost surely connected under random node failure if it has no isolated nodes.*

The statement of Proposition 1 is purposely generic so as to apply to as many types of graphs as possible. This result clearly holds for all DHTs that can be reduced to the hypercube, which includes Chord [38], logarithmic CAN with $d = \Theta(\log n)$ [34], randomized Chord [28], Tapestry [41], and Pastry [36]. It also holds for graphs (e.g., de Bruijn [27]) that have better expansion than hypercubes as long as $k = \Omega(\log n)$ and all types of random Gnutella-style networks where each user relies on random selection of neighbors during join. Even though Proposition 1 refers to graphs of asymptotically large size, extensive simulations below demonstrate the application and exceptional accuracy of (2) in graphs of *finite* size.

3.2. Static Resilience

Recall that static resilience alludes to the connectivity of a graph G after each node is removed from the graph independently with probability p. In this section we examine the accuracy of (2) in a wide array of networks that satisfy Proposition 1. In order to enhance the understanding of how graphs disconnect, we introduce another metric that captures the percentage of disconnections that contain at least one isolated node, which we denote by $q(G)$:

$$q(G) = P(X > 0 | G \text{ is disconnected}) = \frac{P(X > 0)}{1 - \Phi(G)},$$

where X is the number of isolated nodes as before. Interpreting this metric in the context of Proposition 1, it follows that $q(G)$ in almost all well-connected graphs must tend to 1 as $n \to \infty$.

We computed $\Phi(G)$, $P(X = 0)$, and $q(G)$ for a number of degree-regular and irregular P2P networks using $100,000$ node-failure patterns for each value of p. To deal with directed graphs, we assumed that each node's in-degree and out-degree neighbors contributed to its resilience and that isolation happened when a node lost

[3] For node failure, a similar condition must hold for the *node* boundary of each set S, i.e., $\{v : (u,v) \in E, u \in S, v \in V \backslash S\}$.

p	Chord $n = 16384$, $k = 27$			CAN $n = 16384$, $k = 14$			de Bruijn $n = 20736$, $k = 24$			Pastry $n = 15625$, $k = 24$		
	$\Phi(G)$	$P(X = 0)$	$q(G)$	$\Phi(G)$	$P(X = 0)$	$q(G)$	$\Phi(G)$	$P(X = 0)$	$q(G)$	$\Phi(G)$	$P(X = 0)$	$q(G)$
.4	.99999	.99999	1	.97321	.97321	1	.99999	.99999	1	1	1	N/A
.45	.99999	.99999	1	.88093	.88098	.9996	.99995	.99995	1	1	1	N/A
.5	.99996	.99996	1	.60704	.60735	.9992	.99930	.99930	1	.99950	.99950	1
.55	.99918	.99918	1	.18308	.18372	.9992	.99444	.99444	1	.99535	.99535	1
.6	.99354	.99354	1	.00645	.00661	.9998	.96181	.96194	.9966	.97105	.97105	1
.65	.95001	.95004	.9994	0	0	.9999	.79535	.79556	.9989	.83755	.83760	.9997
.7	.72619	.72650	.9988	0	0	1	.31999	.32119	.9982	.41305	.41395	.9985
.75	.17877	.18047	.9979	0	0	1	.00792	.00816	.9998	.02045	.02140	.9990
.8	.00040	.00043	.9999	0	0	1	0	0	1	0	0	1

Table 1. Simulations with degree-regular DHTs.

p	Symphony $k_{out} = 14$			Gnutella $k_{out} = 14$			Randomized Chord $k_{out} = 14$			Random-Zone Chord $k_{out} = 14$		
	$\Phi(G)$	$P(X = 0)$	$q(G)$	$\Phi(G)$	$P(X = 0)$	$q(G)$	$\Phi(G)$	$P(X = 0)$	$q(G)$	$\Phi(G)$	$P(X = 0)$	$q(G)$
.4	.99999	.99999	1	.99316	.99316	1	.99999	.99999	1	.9444	.9455	.9802
.45	.99998	.99998	1	.96609	.96609	1	.99999	.99999	1	.9057	.9089	.9661
.5	.99768	.99768	1	.86257	.86260	.9998	.99971	.99971	1	.8186	.8243	.9686
.55	.98750	.98750	1	.58042	.58064	.9995	.99747	.99747	1	.6248	.6367	.9683
.6	.93914	.93917	.9995	.17081	.17148	.9992	.98443	.98443	1	.3193	.3370	.9739
.65	.75520	.75527	.9997	.00547	.00560	.9998	.91624	.91625	.9999	.0585	.0673	.9907
.7	.31153	.31205	.9992	0	0	1	.63749	.63772	.9994	.0006	.0009	.9997
.75	.01269	.01296	.9997	0	0	1	.12993	.13076	.9990	0	0	1
.8	0	0	1	0	0	1	.00028	.00029	.9999	0	0	1

Table 2. Simulations with degree-irregular graphs for $n = 16384$.

all of its in- *and* out-degree neighbors. Similarly, a directed P2P network was considered partitioned (disconnected) when its *undirected* version was, which is a measure of weak connectivity of directed graphs.

For each directed P2P system, denote by k_{out} its out-degree. Then, after some manipulation, it is not hard to obtain that Chord's total node degree is $k = 2k_{out} - 1 = 2\log_2 n - 1$ and de Bruijn's degree is $k \approx 2k_{out}$. Table 1 shows the above three metrics for degree-regular DHTs Chord [38] with $k_{out} = 14$ and $k = 27$, CAN [34] with $k = 14$, de Bruin [20] with $k_{out} = 12$ and $k = 24$, and undirected Pastry [36] with $k = 24$, each populated with the maximum number of users. As shown in the table, $\Phi(G)$ is very close to $P(X = 0)$ for all graphs and all values of p. Further notice that $q(G)$ ranges between .9966 and 1, which confirms that almost every disconnection in this family of graphs occurs with at least one isolated node.

For degree-irregular graphs, simulations in Table 2 demonstrate that Symphony [28], Gnutella (a random graph with a fixed out-degree k_{out}), Randomized Chord [28], and "Random-zone" Chord (i.e., Chord with a random partitioning of the circle) also follow the classical result well. Besides the fact that $\Phi(G)$ is very close to $P(X = 0)$, notice in Table 2 that the performance of Chord with random zone sizes is inferior to load-balanced (i.e., complete) Chord since there is more possibility for nodes with smaller-than-average degree to disconnect the graph.

3.3. Dynamic Resilience

While the use of p-percent uniform node failure provides an accurate approximation of actual network behavior in some cases, it has been noted that it has questionable applicability to real P2P networks [5], [26], where users join and leave the system asynchronously based on their individual browsing habits. One approach to modeling such systems is to assign each joining user a random lifetime L_i, which determines the duration that node i stays in the system before abruptly (i.e., without graceful notification of its neighbors) departing from the network and represents the amount of time a user spends in the network browsing for content and/or providing services to other peers.

Most structured P2P systems [28], [38], [34] use DHT-specific neighbor-replacement algorithms to repair the zones of failed nodes and maintain consistency of routing. Certain unstructured systems [11] also explicitly perform replacement of failed neighbors to achieve the desired level of routing and search performance. In addition to maintaining consistency of routing [38] and avoiding congestion in the graph [11], neighbor replacement serves the purpose of keeping the system resilient to dis-

Search	CAN, $N = 10^6$		Chord, $N = 50,000$	
delay	Simulations	Model (7)	Simulations	Model (6)
6 min	.9732	.9728	.6295	.6251
7.5 min	.8118	.8124	.3284	.3184
8.5 min	.5669	.5659	.2189	.2206
9 min	.4065	.4028	.1460	.1483
9.5 min	.2613	.2645	.1211	.1274
10.5 min	.0482	.0471	.0493	.0493

Table 3. Lifetime simulations of the probability $P(Z > N)$ that the network survives at least N user joins (fixed search delays).

connection. We next examine the question of how quickly failed neighbors should be replaced and what levels of resilience one should expect from churn-based P2P networks.

Throughout the paper, we assume that each node performs a "search" to find new neighbors as soon as it detects the failure. At this stage, we are not concerned with how this is accomplished and combine both failure detection and repair into a generic random variable S_i that measures the total delay required to perform these operations. Given this new paradigm of node-failure, we now define the probability ϕ that a given user i becomes isolated *during its lifetime* because its neighbors are failing at a faster rate than i is able to obtain their replacements from among the remaining nodes. We derive ϕ in the following sections; however, we now show how the knowledge of this *local* metric can be used to study *global* resilience of lifetime-based P2P networks.

Define Z to be the random time (in terms of user joins) when graph G disconnects for the first time. Then assuming that G satisfies Proposition 1 and each joining node i is assigned a Bernoulli random variable X_i that determines whether the user is isolated from the network during its lifetime, the probability that the graph stays connected for more than N user joins is almost surely:

$$P(Z > N) = P\left(\bigcap_{i=1}^{N}[X_i = 0]\right) = \prod_{i=1}^{N}(1 - E[X_i]). \quad (6)$$

For k-regular graphs, each user has the same probability of isolation (i.e., $E[X_i] = P(X_i = 1) = \phi$) and the above reduces to:

$$P(Z > N) = (1 - \phi)^N. \quad (7)$$

We next verify this evolution model and applicability of (7) using simulations, where both $E[X_i]$ and ϕ are computed empirically. The simulations use two types of DHTs and two distributions of lifetimes: exponential with CDF $1 - e^{-\lambda x}$ and shifted Pareto with CDF $1 - (1 + x/\beta)^{-\alpha}$. The first system under study is a 12-regular fully-populated CAN with exponential lifetimes, $\lambda = 2$ (mean lifetime 30 minutes), $n = 4096$ users, and $N = 10^6$. The second system is a random-zone *degree-irregular* Chord with Pareto lifetimes, $\alpha = 3$, $\beta = 1$ (mean lifetime also 30 minutes), $n = 128$ users, $k \approx 13$ (out-degree 7), and $N = 50,000$. Simulation results are shown in Table 3, where both models (6)-(7) match $P(Z > N)$ well. Observe in the table that zone-balanced CAN is significantly more resilient that random-zone Chord since the latter frequently develops isolation around nodes with smaller-than-average degree.[4] In fact, the resilience of CAN is quite impressive as it can survive 1 million user joins with probability 0.97 using 6-minute replacement delays.

Next, notice that while the node-failure scenario of this section is different from that in the static case, the previous conclusions about graph disconnection through isolated nodes still hold. Table 3 already confirms this fact; however, additional analysis of the disconnection pattern observed in simulations demonstrates that in cases when disconnection does occur, the largest connected component of dynamic systems almost always contains exactly $n - 1$ nodes. This implies a much stronger result: *for reasonably small search delays, network partitioning in lifetime-based systems almost surely effects only one node in the system.*

4. Static Resilience

This section develops a simple closed-form model for $P(X = 0)$, i.e., the probability that the graph contains at least one isolated node, under static node failure and compares this result to simulations of $\Phi(G)$. In the next section, we address the issue of dynamic node failure and derive a model for ϕ.

4.1. Isolated Nodes

Assume that each node i has k_i neighbors in some graph G and again define X_i to be a Bernoulli indicator variable of whether node i is isolated or not after each node is removed from the system with independent probability p:

$$X_i = \begin{cases} 1 & \text{isolated and alive} \\ 0 & \text{otherwise} \end{cases}.$$

Denote by $p_i = P(X_i = 1) = (1-p)p^{k_i}$ the probability that i is isolated and alive after the failure. Next, notice that $\{X_i\}$ may be identically or non-identically distributed, but they are almost certainly *dependent*. However, as $n \to \infty$, this dependency in graphs satisfying Proposition 1 becomes negligible and $\{X_i\}$ asymptotically behave *as if* they were independent [4], [8]. This is a consequence of the fact that in the P2P graphs under study, any two nodes i and j have a

[4] More analysis of zone size distributions in DHTs can be found in [40].

p	Chord $n = 16384$, $k = 27$			de Bruijn $n = 20736$, $k = 24$		
	$\Phi(G)$	Model	Najjar	$\Phi(G)$	Model	Najjar
.4	.9999	1	.9986	.9999	.9999	.9955
.45	.9999	1	.9984	.9999	.9999	.9948
.5	.9999	.9999	.9982	.9993	.9994	.9940
.55	.9992	.9993	.9976	.9944	.9945	.9892
.6	.9935	.9933	.9916	.9618	.9615	.9550
.65	.9500	.9503	.9463	.7954	.7907	.7750
.7	.7262	.7239	.7055	.3199	.3037	.2737
.75	.1788	.1766	.1501	.0079	.0055	.0033
.8	.0004	.0004	.0002	0	10^{-9}	10^{-10}

Table 4. Simulation results and model (9) for two regular graphs.

p	Symphony		Gnutella		Randomized Chord	
	$\Phi(G)$	Model	$\Phi(G)$	Model	$\Phi(G)$	Model
.4	.9999	.9999	.9932	.9934	.9999	.9999
.45	.9998	.9996	.9661	.9666	.9999	.9999
.5	.9977	.9977	.8626	.8646	.9997	.9997
.55	.9875	.9875	.5804	.5829	.9975	.9976
.6	.9391	.9394	.1708	.1700	.9844	.9845
.65	.7552	.7535	.0055	.0053	.9162	.9151
.7	.3115	.3107	0	10^{-7}	.6375	.6372
.75	.0127	.0122	0	10^{-15}	.1299	.1282
.8	0	10^{-7}	0	10^{-34}	.0003	.0002

Table 5. Simulation results and model (10) for three irregular graphs.

fixed number of common neighbors, which becomes negligible compared to the total degree $k = \Omega(\log n)$ as $n \to \infty$.

Next, let $X = \sum_{i=1}^{n} X_i$ be the total number of isolated nodes in G. Applying Markov's inequality $P(X \geq 1) \leq E[X]$, we directly obtain the next lower bound on the connectivity of the system.

Proposition 2. *For graphs satisfying Proposition 1, the following lower bound holds almost surely:*

$$\Phi(G) \geq 1 - \sum_{i=1}^{n} p_i. \tag{8}$$

While this bound is very tight for small p and is better than those shown in [12] for all values of p, it produces negative values for sufficiently high failure rates. To overcome this limitation, an alternative approach is to notice that X is in fact a sum of a large number of Bernoulli random variables with certain well-know asymptotic properties. Due to the diminishing dependency between $\{X_i\}$ as $n \to \infty$, we can applying the Chen-Stein method [4] to X and immediately obtain a much tighter result on $\Phi(G)$.

Proposition 3. *For graphs satisfying Proposition 1 and $n \to \infty$, the number of isolated vertices X tends to a Poisson distribution with mean $\lambda = \sum_i p_i$ and the probability $\Phi(G)$ of having a connected graph converges to $e^{-\lambda}$ with probability* 1.

In the next two sections, we use this generic result to obtain static disconnection models for both degree-regular and irregular graphs.

4.2. Degree-Regular Graphs

For degree-regular networks, the previous result simplifies to a trivial closed-form expression:

$$\Phi(G) = e^{-n(1-p)p^k}. \tag{9}$$

To verify (9), we compare $\Phi(G)$ calculated in simulations over $100,000$ node failure patters to that of the model in Table 4 for Chord [38] with $k = 27$ ($n = 16384$) and de Bruijn graphs [20] with $k = 24$ ($n = 20736$). As the table shows, simulations follow the model quite well for each graph over all values of p. For comparison purposes, the table also plots Najjar's model (3), which is surprisingly less accurate than (9) and significantly more complex to compute.

4.3. Degree-Irregular Graphs

While many ideal DHTs are degree-regular, their instances under random node join and departure often exhibit degree irregularity that depends on random partitioning of the DHT space (e.g., zone-size distribution in Chord). Additional degree-irregular graphs include DHTs in which the in-degree is random (e.g., Symphony, Randomized Chord [28]) and unstructured P2P systems such as Gnutella. For such graphs, we obtain the probability of disconnection under static failure:

$$\Phi(G) = e^{-(1-p)\sum_i p^{k_i}} \approx e^{-n(1-p)E[p^{k_i}]}, \tag{10}$$

where $\sum_i p^{k_i}$ is approximated by $nE[p^{k_i}]$ treating k_i as a random variable.

To compute this model, we first use simulations to obtain $E[p^{k_i}]$ and then utilize this value in (10). Simulations of $\Phi(G)$ for Gnutella, Randomized Chord [28], and Symphony [28], all with degree $k_{out} = 14$ and 16384 nodes, are shown in Table 5, which demonstrates that the model follows simulation results very accurately for all values of p.

To our knowledge there are no results on this topic for degree-irregular graphs with which to compare our model. As Najjar's result (3) is based on a complicated combinatorial argument that only applies to k-regular graphs, it cannot be easily extended to degree-irregular networks.

4.4. Summary

The results of this section have confirmed that large-scale P2P networks generally disconnect through isolated nodes, both in degree-regular and irregular cases. Metric $q(G)$ in all studied simulations has remained between 0.968 and 1, where deviation from 1 was more apparent in smaller graphs and cases when the degree of certain nodes was allowed to become much smaller than average (e.g., in Random-Zone Chord). For larger graphs (hundreds of thousands or millions of nodes), the agreement between $\Phi(G)$ and $P(X = 0)$ will become even stronger.

5. Dynamic Resilience

Using lifetime-based concepts developed in Section 3, we next derive the probability ϕ that all k neighbors of a given node v are simultaneously in the failed state before the lifetime of node v expires. We start with formalizing churn-based P2P systems and explaining our assumptions.

5.1. Lifetime Model

Previous research suggests that the distribution of user lifetimes in real systems is often heavy-tailed (i.e., Pareto) [10], [37], where most users spend very little time browsing the network, while a small group of other peers remain logged in for weeks at a time providing services to other peers. Thus, to allow arbitrarily small lifetimes, we use a shifted Pareto distribution $F(x) = 1 - (1 + x/\beta)^{-\alpha}, x > 0, \alpha > 1$ to represent heavy-tailed user lifetimes, where scale parameter $\beta > 0$ can change the mean of the distribution without affecting its range $(0, \infty]$. Note that the mean of this distribution $E[L_i] = \beta/(\alpha - 1)$ is finite only if $\alpha > 1$, which we assume holds throughout the paper.

In addition to node v selecting k original neighbors when it joins the graph, most current P2P systems repair broken routes and increase resilience by replacing neighbors that have failed by nodes that are still present in the graph. Failure detection can be easily performed through transport or application-layer keep-alive mechanisms, which may include periodic probing, retransmission of lost messages, and timeout-based decisions to search for a replacement. Once a failure is detected, a repair algorithm is initiated to replace the failed neighbor. Since the delays required to carry out these actions are usually random, we use variable S_i to denote the replacement (or search) time of the i-failure in the system.

5.2. Assumptions

We impose the following restrictions on the systems we study to maintain tractability. We only consider those networks that have evolved enough to allow asymptotic results from renewal process theory to hold (this usually applies in practice since real P2P systems continuously evolve and seldom or never restart). We also require certain stationarity of lifetime L_i, which means that all users joining the system have the same lifetime distribution $F(x)$. While it may be argued that users joining late at night browse the network longer (or shorter) than those joining in the morning, our results below can be easily extended to non-stationary environments and used to derive upper/lower bounds on the performance of such systems. Finally, we allow the number of nodes n in the system to vary with time according to any arrival/departure process as long as n remains sufficiently large.

We also impose some conditions on neighbor selection, where we assume that selection of a node i is independent of i's lifetime L_i and its current age A_i. The first assumption holds in practice since each node does not generally know how long the user plans to browse the network. The second assumption also holds in most current P2P systems [20], [34], [36], [38], [11] since neighbor selection is performed based on a uniform hashing function in the case of DHTs or other methods (e.g. random walks) in the case of unstructured P2P graphs. An important consequence of these assumptions is that we can model the instance when v selects a neighbor to be *uniformly random* within the neighbor's lifetime (i.e., its presence online).

5.3. Modeling Neighbors

Next, we formalize the notion of residual lifetimes and understand how to model neighbor evolution. Define R_i to be the remaining lifetime of node i when it was selected by a joining user v to be its neighbor. As before, let $F(x)$ be the CDF of lifetime L_i. Assuming that n is large and the system has reached stationarity, the CDF of residual lifetimes is given by [35]:

$$F_R(x) = P(R_i < x) = \frac{1}{E[L_i]} \int_0^x (1 - F(z))dz. \quad (11)$$

For exponential lifetimes, which we study in this section for comparison purposes, the residuals are trivially exponential using the memoryless property of $F(x)$: $F_R(x) = 1 - e^{-\lambda x}$; however, the residuals of Pareto distributions with shape α are *more* heavy-tailed and exhibit shape parameter $\alpha - 1$:

$$F_R(x) = 1 - \left(1 + \frac{x}{\beta}\right)^{1-\alpha}. \quad (12)$$

This means that Pareto-lifetime systems under churn are *more* resilient than the corresponding exponential systems for a given average lifetime since each user in the former case acquires neighbors with *larger* remaining lifetimes than those in the latter case. This can be explained by the fact that $E[R_i] = \beta/(\alpha-2)$ is larger than $E[L_i] = \beta/(\alpha-1)$ for all values of α and that residual lifetimes R_i in the Pareto case are stochastically larger than the corresponding lifetimes.

Next, assume that each neighbor j $(1 \le j \le k)$ of node v is either alive at any time t or v is searching for its replacement. Thus, neighbor j can be considered in the *on* state at time t if it is alive or in the *off* state otherwise. This neighbor failure/replacement procedure can be modeled as an alternating renewal process $Y_j(t)$:

$$Y_j(t) = \begin{cases} 1 & \text{neighbor } j \text{ alive at } t \\ 0 & \text{otherwise} \end{cases}. \tag{13}$$

Note that the average *on* delay of each process $Y_j(t)$ is $E[R_i]$ and the average *off* delay is $E[S_i]$. Using this notation, the degree of node v at time t is equal to $W(t) = \sum_{j=1}^{k} Y_j(t)$. Denote by T the time at which a node is isolated when all of its neighbors are simultaneously in the *off* state. Thus, the maximum time a node can spend in the system before it is isolated can be written as the *first hitting time* of process $W(t)$ on level 0:

$$T = \inf(t > 0 : W(t) = 0 | W(0) = k). \tag{14}$$

Notice that for exponential L_i and S_i, process $W(t)$ is a birth-death chain with an absorbing state 0. We thus first develop a model for T assuming Markovian behavior of $W(t)$ and then extend it to non-exponential cases.

5.4. Probability of Isolation

In this section, we analyze the probability that a node v becomes isolated due to all of its neighbors simultaneously reaching the failed state during v's lifetime. Assuming L_v is the random lifetime of node v, notice that ϕ is simply $P(T < L_v)$. To obtain this metric, we start with deriving the stationary distribution of $W(t)$.

Proposition 4. *For exponential lifetimes and exponential search delays, the stationary distribution of $W(t)$ is given by:*

$$\pi_j = \lim_{t\to\infty} P(W(t) = j) = \binom{k}{j} \frac{\rho^j}{(1+\rho)^k}, \tag{15}$$

where $\rho = E[L_i]/E[S_i]$.

Proof. Denote by $\mu = 1/E[L_i]$ the node-failure rate and by $\lambda = 1/E[S_i]$ the node-recovery rate. Then, the rate of transitions from state $j < k$ to state $j+1$ is $q_{j,j+1} = (k-j)\lambda$ and from state $j > 0$ to state $j-1$ is $q_{j,j-1} = j\mu$. Treating $W(t)$ as a Markov chain, the balance equations assume the following shape:

$$\pi_j = \pi_{j-1} \frac{(k-j+1)\lambda}{j\mu} = \pi_0 \rho^j \frac{k!}{j!(k-j)!}, \tag{16}$$

where $\rho = \lambda/\mu$. Summing up all probabilities, we have:

$$\pi_0 \sum_{i=0}^{k} \binom{k}{i} \rho^i = 1. \tag{17}$$

Noticing that the above is a binomial expansion of $(1+\rho)^k$, we get $\pi_0 = 1/(1+\rho)^k$ and directly obtain (15). □

Before proceeding to the next result, we define Q_0 to be the rate matrix that corresponds to states $1, \ldots, k$ of $W(t)$ (i.e., without the absorbing state 0). Therefore, assuming Q is the rate matrix of the entire chain, we can write:

$$Q = \begin{pmatrix} 0 & 0 \\ \mathbf{r} & Q_0 \end{pmatrix}, \tag{18}$$

where $\mathbf{r}$ is a column vector of transition rates into state 0. Furthermore, define a diagonal matrix $\Pi = \text{diag}(\pi_j)$ of the stationary states of $W(t)$, a scaled rate matrix $R = \Pi^{1/2} Q_0 \Pi^{-1/2}$, and the j-th orthonormal eigenvector $\mathbf{x}_j$ of R. Then we have the CDF of hitting time T as follows.

Proposition 5. *For exponential lifetimes and exponential search delays, the CDF of T is:*

$$P(T < t) = \sum_{j=1}^{k} \frac{(\delta \mathbf{v}_j)(\mathbf{u}_j^T \mathbf{r})(1 - e^{-\xi_j t})}{\xi_j}, \tag{19}$$

where $-\xi_j$ is the j-th eigenvalue of R, $\delta = (0, 0, \ldots, 1)$ is a $1 \times k$ vector, $\mathbf{v}_j = \Pi^{-1/2}\mathbf{x}_j$, and $\mathbf{u}_j = \Pi^{1/2}\mathbf{x}_j$.

Proof. Since $W(t)$ is a reversible Markov chain, the PDF of its first hitting time to state 0 starting from state k can be written as a mixture of exponential distributions with rates ξ_j [22]:

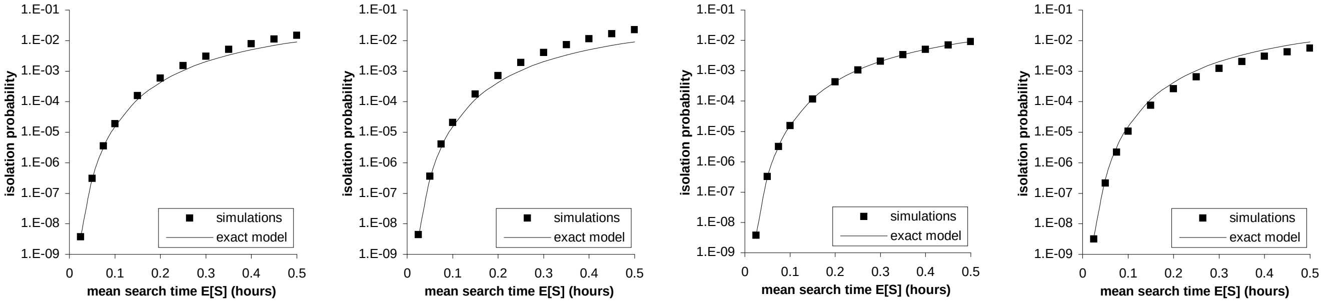
$$f_T(t) = \sum_{j=1}^{k} (\delta \mathbf{v}_j)(\mathbf{u}_j^T \mathbf{r}) e^{-\xi_j t}. \tag{20}$$

Integrating (20) with respect to t, we obtain (19). □

Proposition 6. *For exponential lifetimes and exponential search delays, the probability of isolation is:*

$$\phi = \sum_{j=1}^{k} \frac{(\delta \mathbf{v}_j)(\mathbf{u}_j^T \mathbf{r})}{\mu + \xi_j}, \tag{21}$$

where $\mu = 1/E[L_i]$ and the remaining variables are the same as in the previous proposition.



(a) uniform S_i (b) constant S_i (c) exponential S_i (d) Pareto S_i with $\alpha = 3$

Figure 1. Comparison of model (21) to simulations with exponential lifetimes and $E[L_i] = 0.5, k = 8$.

Proof. Setting $\beta_j = (\delta \mathbf{v}_j)(\mathbf{u}_j^T \mathbf{r})$ and integrating (19) using the PDF $f(t)$ of user lifetimes, we obtain:

$$\begin{aligned} \phi &= P(T < L_v) = \int_0^\infty P(T < t) f(t) dt \\ &= \int_0^\infty \sum_{j=1}^k \frac{\beta_j (1 - e^{-\xi_j t})}{\xi_j} \mu e^{-\mu t} dt \\ &= \sum_{j=1}^k \frac{\beta_j}{\xi_j} \int_0^\infty \mu (e^{-\mu t} - e^{-(\xi_j + \mu) t}) dt, \quad (22) \end{aligned}$$

which directly leads to (21). □

We next verify (21) in simulations and show that it is very accurate for non-exponential search delays as well. Figure 1 shows ϕ obtained in simulations using four distributions of search time for a graph with $k = 8$ and mean lifetime $E[L_i] = 0.5$ hours. Denoting by s the mean search delay, the first distribution is uniform in $[0, 2s]$, the second is constant equal to s, the third is exponential with rate $1/s$, and the fourth is Pareto with $\alpha = 3$ and $\beta = s(\alpha - 1)$. As the figure indicates, all four cases are very close to the values predicted by (21), which can be explained by the quickly-mixing properties of $W(t)$ and relatively small values of search delays S_i [1]. Simulations with other values of k and $E[L_i]$ demonstrate that as search delays become small (i.e., $E[S_i] \to 0$), the above model is accurate for *any* distribution of search delays as long as lifetimes are exponential.

5.5. Asymptotic Expansion

Since (21) requires the spectrum of matrix R, our next task is to simplify this model and obtain a simple closed-form expression for ϕ that does not involve any numerical manipulation. The following result holds assuming asymptotically small search delays.

Proposition 7. *For exponential lifetimes and exponential search delays, the probability of isolation is given by the following as $E[S_i] \to 0$:*

$$\phi = \frac{\rho k}{(1+\rho)^k + \rho k - 1} + o(1), \quad (23)$$

where $\rho = E[L_i]/E[S_i]$ is the ratio of the mean user lifetime to the mean search delay.

Proof. The proof proceeds in two steps. We start by deriving the expected time $E[T]$ before the first visit to state 0 and then use an exponential approximation to the density of T to obtain an asymptotic expansion of ϕ.

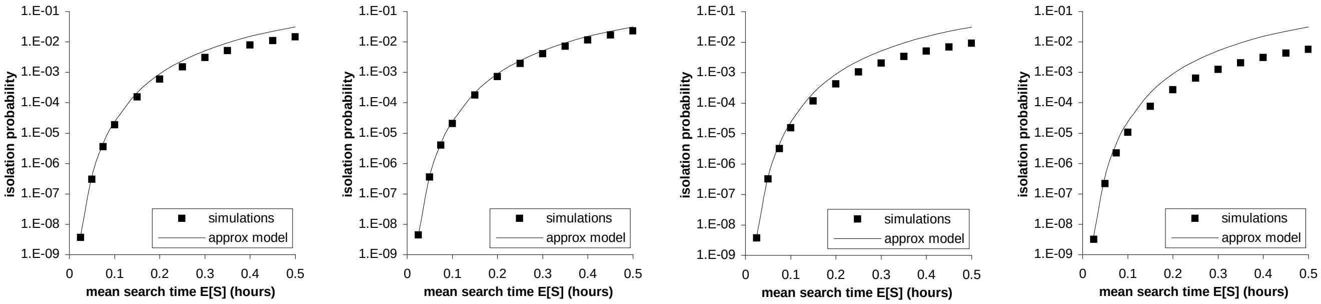
We begin by deriving $E[T]$ in closed-form assuming that search delays are reasonably small. Treating the chain as non-absorbing throughout this proof and denoting by T_{00} the delay between the visits to state 0, we get using the stationary distribution π derived in Proposition 4:

$$E[T_{00}] = \frac{1}{\pi_0 q_0} = \frac{E[S_i]}{k} (1+\rho)^k, \quad (24)$$

where $q_0 = k\lambda = k/E[S_i]$ is the rate of transition in the non-absorbing chain from state 0 to itself. Using the fact that the relaxation time of the chain is asymptotically small compared to $E[T_{00}]$ (see below) and assuming that the chain starts in its stationary state, the expected delay before the *first* visit to state 0 converges to the mean delay between the subsequent visits to the same state [2]. Thus, subtracting from $E[T_{00}]$ the average time spent in state 0, we get:

$$\begin{aligned} E_\pi[T] &= E[T_{00}] - \frac{1}{q_0} + o(1) \\ &= \frac{E[S_i]}{k} \Big((1+\rho)^k - 1 \Big) + o(1), \quad (25) \end{aligned}$$

where $E_\pi[T]$ denotes the mean first hitting time on state 0 assuming that the initial distribution of $W(0)$ is the stationary distribution π of the chain. Notice, however, that as $E[S_i] \to 0$, the stationary distribution π given in (16) converges to the actual initial distribution of the chain (i.e., $\pi \to (0, 0, \ldots, 1)$ and $W(0) = k$ with probability 1), which leads to $E[T] = E_\pi[T] + o(1)$.



(a) uniform S_i (b) constant S_i (c) exponential S_i (d) Pareto S_i with $\alpha = 3$

Figure 2. Comparison of model (23) to simulations with exponential lifetimes with $E[L_i] = 0.5, k = 8$.

Next, observe that for small search delays S_i, $E[T]$ is large and state 0 is visited rarely. This allows the application of Aldous' inequality [3] for rare events in Markov chains, which states that T asymptotically behaves as an exponential random variable with mean $E[T]$:

$$|P(T > t) - e^{-t/E[T]}| \le \frac{\tau}{E[T]}, \qquad (26)$$

where τ is the relaxation time of the chain. For the birth-death chain of our case, it can be shown [25] that $\tau = 1/(\lambda + \mu) \approx E[S_i]$, where the last approximation holds assuming that $E[S_i] \ll E[L_i]$. Hence as $E[S_i] \to 0$, relaxation time $\tau = o(1)$ and the bounds in (26) reduce to:

$$P(T > t) = e^{-t/E[T]} + o(1). \qquad (27)$$

Integrating (27) with respect to the PDF $f(t)$ of user lifetimes, we get:

$$\begin{aligned} \phi &= P(T < L_v) = \int_0^\infty P(T < t) f(t) dt \\ &= \int_0^\infty (1 + o(1) - e^{-t/E[T]}) \mu e^{-\mu t} dt \\ &= \frac{1}{\mu E[T] + 1} + o(1). \end{aligned} \qquad (28)$$

Using (25) and recalling that $\mu = 1/E[L_i]$, we obtain (23) as the asymptotic shape of ϕ when $E[S_i] \to 0$. □

This model is verified in Figure 2 for the same four cases of search delay S_i. Notice that the asymptotic model is less accurate for the exponential search delays, but provides an almost exact match to the constant delay case (part (b) in the figure). Also observe that as $E[S_i]$ becomes smaller, all four cases indeed converge to (23) and achieve isolation probability $\phi \approx 4.2 \times 10^{-9}$ when the expected search time reduces to 1.5 minutes

Also note that constant search delays provide the *worst-case* scenario for isolation, while highly-variable distributions of S_i are the best. This immediately follows from the non-negative nature of search times and the fact that for a given $E[S_i]$ higher variance of S_i implies that more probability mass is concentrated at values well below $E[S_i]$. We thus obtain that random search delays can only *improve* the resilience of the system compared to the worst-case scenario (i.e., constant S_i). This can be observed in Figure 2 where ϕ in part (b) is the largest among the four cases. Since constant search delays produce an almost ideal match to the approximate model, the result in (23) can be treated as an upper bound on ϕ for all cases with exponential lifetimes.

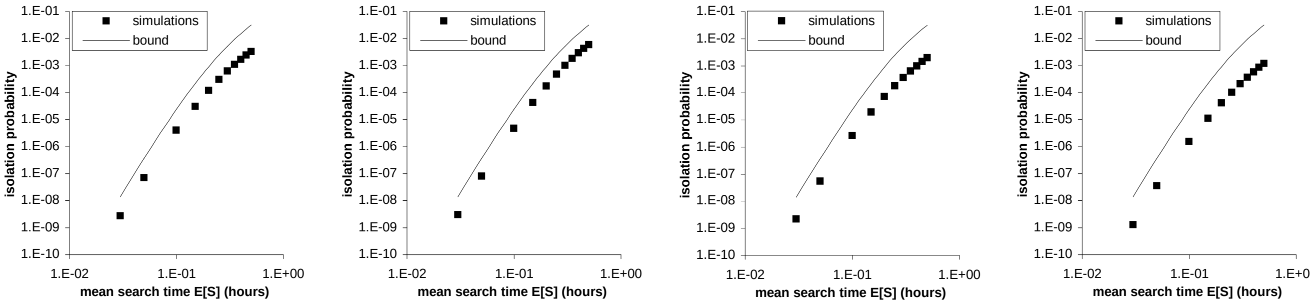
$E[S_i]$	Model (21)	Model (23)	Ratio
1 hour	3.2480×10^{-2}	1.3971×10^{-1}	4.3017
6 min	1.5379×10^{-5}	2.3814×10^{-5}	1.5485
36 sec	8.2856×10^{-12}	8.7397×10^{-12}	1.0548
3.6 sec	1.0023×10^{-18}	1.0078×10^{-18}	1.0054
360 ms	1.0218×10^{-25}	1.0224×10^{-25}	1.0006

Table 6. Convergence of (23) to (21) for exponential search delays and $E[L_i] = 0.5, k = 8$.

To finish this subsection, we examine the convergence of approximation (23) to the exact model (21) in more detail. Table 6 shows the values of ϕ produced by both models as $E[S_i]$ becomes very small. Observe in the table that both models indeed converge and that the relative difference diminishes to zero as $E[S_i]$ becomes small.

5.6. Pareto Lifetimes

Due to the non-Markovian nature of $W(t)$ under Pareto lifetimes and its slowly mixing properties, derivation of ϕ for this case is very complicated. Furthermore, the result is expected to be sensitive to the exact value of parameters α and β of the Pareto distribution, which are difficult to measure and may vary from system to system. We leave the exploration of Pareto ϕ for future work and instead utilize the exponential metric (23) as an upper bound on ϕ in systems with sufficiently heavy-tailed lifetime distributions. The re-



(a) uniform S_i (b) constant S_i (c) exponential S_i (d) Pareto S_i with $\alpha = 3$

Figure 3. Upper bound (29) and simulations for Pareto lifetimes with $E[L_i] = 0.5$ hours and $k = 8$.

ϕ	Uniform	Lifetime	Mean Search time $E[S_i]$		
	$p = 1/2$	P2P	6 min	2 min	20 sec
10^{-4}	14	Bound (29)	8	5	4
		Simulations	7	5	4
10^{-6}	20	Bound (29)	10	7	5
		Simulations	10	7	5
10^{-8}	27	Bound (29)	13	9	6
		Simulations	13	8	6

Table 7. Minimum degree needed for a certain ϕ in systems with Pareto lifetimes with $\alpha = 3, \beta = 1$ and $E[L_i] = 0.5$ hours.

sult below follows from the fact that heavy-tailed L_i imply stochastically larger residual lifetimes R_i and that (23) provides an upper bound for all search delay distributions.

Corollary 1. *For an arbitrary distribution of search delays and any lifetime distribution $F(x)$ with an exponential or heavier tail, which includes Pareto, lognormal, Weibull, and Cauchy distributions, the following upper bound holds:*

$$\phi \leq \frac{\rho k}{(1+\rho)^k + \rho k - 1}, \qquad (29)$$

where $\rho = E[L_i]/E[S_i]$ is the ratio of the mean user lifetime to the mean search delay.

For example, using 30-minute average lifetimes, 9 neighbors per node, and 1-minute average node replacement delay, the upper bound in (29) equals 1.02×10^{-11}, which allows each user in a 100-billion node network to stay connected to the graph for his/her entire lifespan with probability $1 - 1/n$. Using the uniform failure model of prior work and $p = 1/2$ [38], each user requires 37 neighbors to achieve the same ϕ *regardless of the actual dynamics of the system*.

To confirm that the upper bound (29) holds in practice, Figure 3 shows ϕ in simulations with Pareto lifetimes with $E[L_i] = 0.5$ and $k = 8$. Observe in the figure that Pareto systems are in fact more resilient than those with exponential lifetimes. Also notice that constant search delays once again provide the worst-case resilience for a given $E[S_i]$ and that the difference between the Pareto and exponential ϕ is by a constant factor (i.e., the two curves become parallel as $E[S_i] \to 0$).

Even though exponential ϕ is often several times larger than the Pareto ϕ (the exact ratio depends on shape α), it turns out that the difference in node degree needed to achieve a certain level of resilience is usually negligible. To illustrate this result, Table 7 shows the minimum degree k that ensures a given ϕ for different values of search time $E[S_i]$ and Pareto lifetimes with $\alpha = 3, \beta = 1$ ($E[L_i] = 0.5$ hours). The column "uniform $p = 1/2$" contains degree k that can be deduced from the p-percent failure model (for $p = 1/2$) discussed in previous studies [38]. Observe in the table that the exponential case in fact provides a tight upper bound on the actual minimum degree and that the difference between the two cases is at most 1 neighbor.

5.7. Graph Disconnection

We now apply the newly acquired model for the probability of isolation ϕ to (7) and examine its accuracy in simulations. Re-writing (7), the dynamic resilience of a graph G is lower-bounded by:

$$P(Z > N) \geq \left(1 - \frac{\rho k}{(1+\rho)^k + \rho k - 1}\right)^N, \qquad (30)$$

where Z is the number of user joins before the first disconnection of the system. Table 8 contains $P(Z > N)$ obtained in simulations of 12-regular CAN with exponential lifetimes, $E[L_i] = 0.5$ hours, $n = 4096$, and $N = 10^6$ user joins. The table also includes the value computed by model (7) using empirically measured ϕ along with the newly derived model (30) for comparison purposes. Note that even in the case of relatively large search delays (i.e., $S_i = 10.5$ minutes), the simulations still follow the model quite well

Fixed search time	Actual $P(Z > N)$	Model (7)	Model (30)	Metric $q(G)$
6 min	.9732	.9728	.9728	1
7.5 min	.8218	.8224	.8215	1
8.5 min	.5669	.5659	.5666	1
9 min	.4065	.4028	.4016	1
9.5 min	.2613	.2645	.2419	1
10.5 min	.0482	.0471	.0424	1

Table 8. Comparison of $P(Z > N)$ in CAN.

and that the graph never partitions without having at least one isolated node (i.e., $q(G) = 1$).

To further illustrate the gravity of (30) when used as a lower bound on the performance of lifetime-based P2P systems, consider the example first mentioned in the introduction. In a k-regular P2P system with $k = 12$ for each neighbor, search delay $E[S_i] = 1$ minute, and average lifetime $E[L_i] = 0.5$ hours, the probability of isolation is $\phi = 4.57 \times 10^{-16}$. When ϕ is applied to (30) in which 35 million users join and leave the system each week, the probability that the network survives for $10,000$ years without disconnecting is at least 99.2%. Model (30) further implies that the mean delay between disconnections is lower-bounded by $1/\phi$ user joins, or 1.2 million years.

Relatively small systems are also very resilient based on this analysis. A system with $k = 8$, a search delay of 30 seconds, average lifetime $E[L_i] = 0.5$ hours, and $50,000$ users join each day will survive for 100 years without disconnection with probability no less than 99.5%. These two examples show that both large and small-scale systems can easily achieve a high level of resilience.

6. Discussion

While the models described in this paper have shown that most current P2P systems are very resilient to node isolation and disconnection under many practical conditions, our results can also be exploited to develop even more resilient systems. As the average lifetime of users in the system cannot generally be influenced by system designers, they must focus on the elements within their control. The two basic ways to increase resilience without modifying the graph topology are to increase k or decrease $E[S_i]$. However, changes to these parameters often cause increased network overhead in terms of keep-alive messages, state kept at each node, and processing complexity.

A more cost-effective goal is to ensure that each node has a high probability of obtaining a neighbor with a large residual lifetime either upon join or during its stay in the system. Since the residual lifetime of each node is not known, the node's age (an easily obtainable metric) can be used instead. In fact, it can be shown that given a Pareto distribution of lifetimes, nodes with large age are expected to survive longer and possess stochastically larger residual lifetimes R_i than those with small age. We propose intentionally monitoring the age of each node and giving more preference during neighbor selection to the nodes with a larger value of this metric, which causes the system to achieve a twofold effect: short-lived nodes do not attract a large number (if any) neighbors and long-lived nodes are given a bigger responsibility over the structure of the graph. Preliminary simulation results of this method indicate that $E[R_i]$ of chosen neighbors increases by several times over uniformly random selection of neighbors and leads to much lower ϕ.

7. Conclusion

This paper tackled the problem of P2P graph connectivity under both static and dynamic node-failure methods by establishing that almost every sufficiently large network remains connected if and only if it has no isolated nodes, a result from random graph theory that we confirm applies to P2P networks under both independent uniform node failure and lifetime-based node departure. We used this powerful result to derive models of graph connectivity for both the static and dynamic node failure cases that are much more accurate than previous efforts and are easily calculable. Our results show that most current P2P systems are extremely resilient to disconnections when the ratio of average lifetime to average search delay is non-trivial. Future work includes deriving an exact model for dynamic node failure using Pareto and other heavy-tailed lifetimes, extending the lifetime model to degree-irregular networks, and constructing more resilient P2P networks.

8. Acknowledgment

The authors are grateful to the anonymous reviewers for their helpful comments and Sambit Sahu for shepherding the paper.

References

[1] M. Abadi and A. Galves, "Inequalities for the occurrence times of rare events in mixing processes. The State of the Art," *Markov Proc. Relat. Fields*, vol. 7, no. 1, 2001.

[2] D. Aldous and M. Brown, "Inequalities for Rare Events in Time-Reversible Markov Chains I," *Stochastic Inequalities*, vol. 22, 1992.

[3] D.J. Aldous and M. Brown, "Inequalities for Rare Events in Time-Reversible Markov Chains II," *Stochastic Processes and their Applications*, vol. 44, 1993.

[4] R. Arratia, L. Goldstein, and L. Gordon, "Two Moments Suffice for Poisson Approximations: The Chen-Stein Method, *The Annals of Probability*, vol. 17, no. 1, Janurary 1989.

[5] R. Bhagwan, S. Savage, and G.M. Voelker, "Understanding Availability," *IPTPS*, February 2003.

[6] F. Boesch, D. Gross, C. Suffel, "A Coherent Model for Reliability of Multiprocessor Networks," *IEEE Trans. on Reliability*, vol. 45, no. 4, December 1996.

[7] B. Bollobás, "The Evolution of the Cube," *Combinatorial Mathematics*, 1983.

[8] B. Bollobás, *Random Graphs*. Cambridge Univ. Press, 2001.

[9] Yu.D. Burtin, "Connection Probability of a Random Subgraph of an n-Dimensional Cube, *Probl. Pered. Inf.*, vol. 13, no. 2, April-June 1977.

[10] F.E. Bustemante and Y. Qiao, "Friendships that Last: Peer Lifespan and its Role in P2P Protocols," *International Workshop on Web Caching and Distribution*, September 2003.

[11] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker, "Making Gnutella Like P2P Systems Scalable," *ACM SIGCOMM*, August 2003.

[12] J. Chen, I. A. Kanj, and G. Wang, "Hypercube Network Fault Tolerance: A Probabilistic Approach," *ICPP*, August 2002.

[13] B. Chun, B. Zhao, and J. Kubiatowicz, "Impact of Neighbor Selection on Performance and Resilience of Structured P2P Networks," *IPTPS*, February 2005.

[14] P. Erdös, and A. Rényi, "On the Evolution of Random Graphs", *Publications of the Math. Inst. of the Hungarian Academy of Sc.*, 1960.

[15] A. H. Esfahanian, "Generalized Measures of Fault Tolerance with Application to n-cube Networks," *IEEE Transactions on Computers*, vol. 38, 1989.

[16] H. Frank, "Maximally Reliable Node Weighted Graphs," *3rd Ann. Conf. Information Sciences and Systems*, March 1969.

[17] A. Ganesh and L. Massoulie, "Failure Resilience in Balanced Overlay Networks," *Allerton Conference on Communication, Control and Computing*, October 2003.

[18] Q.-P. Gu and S. Peng, "Unicast in Hypercubes with a Large Number of Faulty Nodes," *IEEE Transactions on Parallel and Distributed Systems*, vol. 10, 1999.

[19] K. Gummadi, R. Gummadi, S. Gribble, S. Ratnasamy, S. Shenker, and I. Stoica, " The Impact of DHT Routing Geometry on Resilience and Proximity," *ACM SIGCOMM*, August 2003.

[20] M.F. Kaashoek and D. Karger, "Koorde: A Simple Degree-optimal Distributed Hash Table, " *IPTPS*, February 2003.

[21] A.K. Kel'mans, "Connectivity of Probabilistic Networks," *Auto. Remote Contr.*, vol. 3, 1967.

[22] M. Kijima. *Markov Processes for Stochastic Modeling*. Chapman & Hall, 1997.

[23] S. Krishnamurthy, S. El-Ansarh, E. Aurell, and S. Haridi, "A Statistical Theory of Chord under Churn," *IPTPS*, February 2005.

[24] S. Latifi, "Combinatorial Analysis of the Fault Diameter of the n-cube," *IEEE Transactions on Computers*, vol. 42, 1993.

[25] R.B. Lenin and P.R. Parthasarathy, "Transient Analysis in Discrete Time of Markovian Queues with Quadratic Rates," *Southwest J. Pure and Appl. Math.*, July 2000.

[26] D. Liben-Nowell, H. Balakrishnan, and D. Karger, "Analysis of the Evolution of the Peer-to-Peer Systems," *ACM PODC*, 2002.

[27] D. Loguinov, A. Kumar, V. Rai, and S. Ganesh, "Graph-Theoretic Analysis of Structured Peer-to-Peer Systems: Routing Distances and Fault Resilience," *ACM SIGCOMM*, August 2003.

[28] G.S. Manku, M. Naor, and U. Weider, "Know thy Neighbor's Neighbor: the Power of Lookahead in Randomized P2P Networks," *ACM STOC*, June 2004.

[29] L. Massoulié, A.-M. Kermarrec, and A. Ganesh, "Network Awareness and Failure Resilience in Self-Organising Overlay Networks," *IEEE Symposium on Reliable Distributed Systems*, October 2003.

[30] W. Najjar and J.-L. Gaudiot, "Network Resilience: A Measure of Network Fault Tolerance, *IEEE Trans. on Computers*, vol. 39, no. 2, 1990.

[31] K. Palm, "Intensitätsschwankungen in Fernsprecherverkehr," *Ericsson Technics*, 44, 1943.

[32] G. Pandurangan, P. Raghavan, E. Upfal, "Building Low-Diameter Peer-to-Peer Networks, " *IEEE JSAC*, vol. 21, no. 6, August 2003.

[33] M.D. Penrose, "On k-connectivity for a Geometric Random Graph," *Random Structures & Algorithms*, vol. 15, no. 2, 1999.

[34] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network," *ACM SIGCOMM*, August 2001.

[35] S. Resnick. *Adventures in Stochastic Processes*. Birkhäuser, Boston, 2002.

[36] A. Rowstron and P. Druschel, "Pastry: Scalable, Decentralized, Object Location and Routing for Large-Scale Peer-to-Peer Systems," *IFIP/ACM Middleware*, November 2001.

[37] S. Saroiu, P.K. Gummadi, and S.D. Gribble, "A Measurement study of Peer-to-Peer File Sharing Systems," *MMCN*, 2002.

[38] I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer lookup Service for Internet Applications," *ACM SIGCOMM*, August 2001.

[39] K. Sutner, A. Satyanarayana, and C. Suffel, "The Complexity of the Residual Node Connectedness Reliability Problem," *SIAM J. Computing*, vol. 20, 1991.

[40] X. Wang, Y. Zhang, X. Li, and D. Loguinov, "On Zone-Balancing of Peer-to-Peer Networks: Analysis of Random Node Join," *ACM SIGMETRICS*, June 2004.

[41] B.Y. Zhao, J.D. Kubiatowicz, and A. Joseph, "Tapestry: An Infrastructure for Fault-Tolerant Wide-Area Location and Routing," *UC Berkeley Technical Report*, April 2001.