Know thy Neighbor's Neighbor: the Power of Lookahead in Randomized P2P Networks*

Gurmeet Singh Manku
Stanford University
California USA
manku@cs.stanford.edu

Moni Naor†
Weizmann Institute of Science
Rehovot Israel
moni.naor@weizmann.ac.il

Udi Wieder
Weizmann Institute of Science
Rehovot Israel
udi.wieder@weizmann.ac.il

ABSTRACT

Several peer-to-peer networks are based upon randomized graph topologies that permit efficient GREEDY routing, e.g., randomized hypercubes, randomized Chord, skip-graphs and constructions based upon small-world percolation networks. In each of these networks, a node has out-degree $\Theta(\log n)$, where n denotes the total number of nodes, and GREEDY routing is known to take $O(\log n)$ hops on average. We establish lower-bounds for GREEDY routing for these networks, and analyze Neighbor-of-Neighbor (NoN)-GREEDY routing. The idea behind NoN, as the name suggests, is to take a neighbor's neighbors into account for making better routing decisions.

The following picture emerges: Deterministic routing networks like hypercubes and Chord have diameter $\Theta(\log n)$ and GREEDY routing is optimal. Randomized routing networks like randomized hypercubes, randomized Chord, and constructions based on small-world percolation networks, have diameter $\Theta(\log n/\log\log n)$ with high probability. The expected diameter of Skip graphs is also $\Theta(\log n/\log\log n)$. In all of these networks, GREEDY routing fails to find short routes, requiring $\Omega(\log n)$ hops with high probability. Surprisingly, the NoN-GREEDY routing algorithm is able to diminish route-lengths to $\Theta(\log n/\log\log n)$ hops, which is asymptotically optimal.

Categories and Subject Descriptors

C.2.1 [**Computer Systems**]: computer-communication networks;

General Terms

Algorithms

Keywords

Greedy Routing, Peer to Peer Networks, Random Structures

*Research supported in part by the RAND/APX grant from the EU Program IST, NSF Grants IIS-0118173, EIA-0137761, and an SNRC grant.

†Incumbent of the Judith Kleeman Professorial Chair.

STOC'04, June 13–15, 2004, Chicago, Illinois, USA.

1. INTRODUCTION

Randomized network constructions that model the *Small-World Phenomenon* have recently received considerable attention. A widely-held belief pertaining to social networks is that any two people in the world are connected via a chain of six acquaintances (*six-degrees of separation*)[1]. The quantitative study of the phenomenon started with Milgram's [24] experiments in 1960's, asking people to send letters to unfamiliar targets only through acquaintances. Milgram's experiments, and a work by Pool and Kochen [29] confirmed that random pairs of individuals are indeed connected by short chains. As was noticed by Kleinberg [19], Milgram's experiments also demonstrated that individuals are able to *route* messages to unknown targets..

To model the routing aspects of the Small-World Phenomenon, Kleinberg constructed a family of random graphs. The graphs not only have small diameter (to model the "six degrees of separation") but also allow short routes to be discovered on the basis of local information alone (to model Milgram's observation that messages can be "routed to unknown individuals efficiently"). In particular, Kleinberg considered a 2D $n \times n$ grid with n^2 nodes. Each node is equipped with a small set of "local" contacts and one "long-range" contact drawn from a harmonic distribution. With GREEDY routing, the path-length between any pair of nodes is $O(\log^2 n)$ hops, w.h.p. Local knowledge available to a node suffices for GREEDY routing – a message is forwarded along that out-going link which takes it *closest* to the destination. Barrière *et al* [6] showed that GREEDY routing requires $\Omega(\log^2 n)$ hops for Kleinberg's construction.

Randomized Peer-to-Peer Networks

Symphony [23] is a successful adaptation of Kleinberg's construction [19] to arrive at a randomized P2P routing network. The idea is to place nodes in a ring (instead of a 2D grid) and to equip each node with multiple "long-distance" links (instead of one). The average length of GREEDY routes has been shown to be $O(\frac{\log^2 n}{k})$ both by Manku *et al* [23] and by Aspnes *et al* [3]. Recently, three more randomized P2P networks have been devised, all of which use GREEDY routing: randomized-hypercube [15, 8], randomized-Chord [15, 36], and skip-graphs [4] (also known as SkipNet [17]). Randomized-Chord is a variation on a deterministic P2P routing network called Chord [34, 14].

[1]According to Barabási [5] this idea may have its origins in a short story "Chains" by the Hungarian writer Frigyes Karinthy from 1929; this idea has been retold and recast many times since then, in the literature, popular press as well as scientific studies.

Skip-graphs build upon the intuition inherent in the skip-lists data structure [30]. All of these networks have $\Theta(\log n)$ out-going links per node. GREEDY routing is known to take $O(\log n)$ hops on average.

Among the various P2P routing networks, skip-graphs are unique in that node identifiers (or "keys" associated with nodes) can be drawn from an arbitrary ordered domain, e.g., the set of character strings. This property makes skip-graphs the only P2P routing network that naturally supports prefix-*search*. Other P2P routing networks assume that nodes are assigned identifiers that are drawn uniformly from the unit interval $[0, 1)$.

Many P2P networks share structural similarities with a network in which nodes are associated with a $d-$dimensional torus, and an edge (i, j) is established with probability $\frac{1}{||i-j||^d}$. We call this network a *small-world percolation network*. The small-world percolation network has its antecedents in classical "long range percolation" models. We outline a brief history at the beginning of Section 2.

Our work addresses two questions: (a) *Is* GREEDY *routing optimal?* (b) *What is the role of look-ahead upon* GREEDY *routing?* The idea underlying "look-ahead" is to allow a node to gain knowledge of its neighbor's neighbors for assistance in making better routing decisions. In a network with k out-going links per node, the average length of shortest paths is $\Omega(\log n/\log k)$. Therefore, with $\Theta(\log n)$ links per node, it *might* be possible to route in $O(\log n/\log\log n)$ hops. The upper bound for GREEDY routing for various randomized P2P routing networks is known to be $O(\log n)$. We furnish a matching lower bound, thereby showing that GREEDY is sub-optimal. We also show that NoN-GREEDY is in fact, optimal.

1.1 Our Contributions

The main contribution of this work is to show that in many cases GREEDY routing is asymptotically sub-optimal, while an algorithm which uses just one level of look-ahead is asymptotically optimal.

Upper bounds: We show that NoN-GREEDY routing, which fixes two hops of a route (by taking the neighbors of neighbors of a node into account), is optimal for small-world percolation networks. The NoN-GREEDY routing algorithm requires $\Theta(\log n/\log\log n)$ hops, w.h.p. (Section 2). We establish the same bound on the expected path length for randomized-hypercubes and randomized-Chord (Section 3). In Section 3 we analyze Symphony and show that the NoN algorithm is asymptotically better than GREEDY yet not optimal. We show that for skip-graphs the NoN-GREEDY routing algorithm requires an expected $\Theta(\log n/\log\log n)$ hops (Section 4). Thus skip-graphs are the only degree-optimal P2P network that supports *prefix search*.

Simulations show that for network sizes ranging from 2^{12} to 2^{24} nodes, NoN-GREEDY routes are 40% to 48% shorter than GREEDY routes in all of these topologies [23, 26].

Lower bounds We show that GREEDY routing requires $\Omega(\log n)$ hops on average in each of the following randomized P2P networks: skip-graphs, randomized-Chord, randomized-hypercube, and Symphony with $k = \Theta(\log n)$ per node.

In Section 5, we introduce a probing model for establishing lower bounds on algorithms that rely solely on local information for making routing decisions. We generalize the idea of GREEDY routing to 1-*local* and 2-*local* algorithms. We then establish that for skip-graphs and small-world percolation networks, *any* 1-local routing algorithm (of which GREEDY is a special case) requires $\Omega(\log n)$ hops on average.

1.2 Related Work

For decades, scientists have been devising random graph models that possess statistical properties of graphs that occur in nature. Examples of such graphs include social acquaintanceship networks, electric power grids, telephone call graphs, neural wiring of worms and influence networks. Models such as those by Watts and Strogatz [35] are characterized by a successful mixture of regularity and randomness to faithfully reproduce three statistical properties: the "characteristic path length", the "average vertex degree" and the "clustering coefficient" [28]. An important property ignored by these models is the existence of short routes, i.e., the *small-world phenomenon*. Kleinberg's construction [19] aims to incorporate routing properties into random graph models.

P2P routing networks have witnessed a flurry of research activity recently. Broadly, these networks can be classified into two categories – deterministic and randomized. Deterministic P2P networks are based upon classical parallel inter-connection networks like hypercubes [32, 37], its variants [34], multi-dimensional meshes [31] and de Bruijn graphs [18, 12, 25, 1, 20]. Randomized P2P networks include Viceroy [21] (a randomized emulation of butterfly networks), Symphony [23] (an adaptation of Kleinberg's construction [19]), randomized-hypercubes [15, 8], randomized-Chord [36, 15], and a combination of Kleinberg's construction with butterfly networks [22].

The tradeoff between the average path length and the out-degree of nodes, is of fundamental interest to designers of P2P routing networks. Hypercubes and Chord offer average paths of length $\Theta(\log n)$ with $\Theta(\log n)$ links per node with GREEDY routing (optimal routes in Chord were identified by Ganesan and Manku [14]). For the same number of links per node, de Bruijn graphs offer routes of length $\Theta(\log n/\log\log n)$. The routing protocol in de Bruijn graphs is not GREEDY – it is based on numeric computations on labels of nodes. Among the randomized P2P networks, Viceroy offers routes of length $\Theta(\log n)$ w.h.p. with only $O(1)$ links per node. A randomized construction in [22] combines ideas from Viceroy with Kleinberg's construction to arrive at a network that routes in $\Theta(\log n/\log k)$ hops w.h.p., with k links per node. Randomized-hypercubes and randomized-Chord are known to offer routes of length $O(\log n)$ with GREEDY routing. Both of these networks are significantly simpler than Viceroy and the construction in [22].

Overall, three classes of networks are known to route in $\Theta(\log n/\log\log n)$ hops with $\Theta(\log n)$ links per node: de Bruijn networks, deterministic butterflies and randomized butterflies. The P2P implementation of these networks requires that keys are *random*, thus unlike skip-graphs there is no natural way for keys to carry *semantic* meaning. The results of this paper add a fourth class – "randomized small-world networks". We hope that our results inspire further investigations into the general properties of these networks.

The basic idea of the NoN-GREEDY approach is drawn from two sources. A paper by Coppersmith *et al* [10] uses the neighbors-of-neighbors approach, though not in an algorithmic perspective. They use the idea to establish that the diameter of small-world percolation networks on n nodes is $O(\frac{\log n}{\log\log n})$ w.h.p. NoN-GREEDY *routing* was first used (under the name "GREEDY with 1-LOOKAHEAD") by Manku

et al [23] as a heuristic for Symphony, a randomized P2P network. Fraigniaud *et al* [13] recently analyzed GREEDY algorithms in Kleinberg's model, when each node is aware of the long-range contacts of the $\log n$ nodes which are closest to it. They show that a variant of NoN-GREEDY routes in expected $\Theta(\log^{1+\frac{1}{d}} n)$ hops (when d is the dimension of the mesh). Aspnes *et al* [3] established lower bounds for GREEDY over a general family of randomized networks under the assumption that each "long-range" link is drawn from the same probability distribution.

1.3 The NoN-GREEDY Routing Algorithm

We introduce the main object of our investigation, the NoN-GREEDY Routing Algorithm, in Figure 1. We assume the existence of a metric on the labels of nodes.

Algorithm for routing a message to node t.

1. Assume the message is currently at node $u \neq t$. Let $w_1, w_2, \ldots, w_k$ be the neighbors of u.
2. For each w_i, $1 \leq i \leq k$, find z_i - the closest neighbor to t. Let j be such that z_j is the closest to t among $z_1, z_2, \ldots, z_k$.
3. Route the message from u via w_j to z_j.

Figure 1: The NoN-GREEDY Algorithm. Some metric over the labels of nodes is assumed.

In the NoN-GREEDY algorithm , w_j may not be the neighbor of u which is closest to t. The algorithm could be viewed as a greedy algorithm on the *square* of the graph – a message gets routed to the best possible node among those at distance two.

2. SMALL-WORLD PERCOLATION

DEFINITION 2.1. *A "small-world percolation network" of dimension d is a graph whose vertex set is associated with the $d-$dimensional mesh. The probability that an edge (u, v) exists is $\frac{1}{||u-v||^d}$, where $||u - v||$ stands for the L_1 distance between u and v.*

Small-world percolation networks originate from a classical percolation model called "long range percolation". In that model, nodes lie on a lattice and an edge exists between a pair of nodes with some positive probability. The question of existence of infinite components was considered by Schulman [33], Aizenman and Newman [2] and Newman and Schulman [27], where the one dimensional lattice $\mathcal{Z}$ is studied and edges (i, j) are selected with probability $\beta/\|i - j\|^s$ for some values β, s.

Benjamini and Berger [7] proposed and studied a finite percolation model: a cycle graph over n nodes where an edge between nodes i and j exists with probability 1 if $\|i-j\| = 1$, otherwise, it exists with probability $\exp(-\beta/\|i - j\|^s)$, for some values β, s. Coppersmith *et al* [10] extended the model to multiple dimensions: a $d-$dimensional mesh where an edge (u, v) is selected independently with probability $1/\|u-v\|^d$. Coppersmith *et al* established that the *diameter* of the resulting graph is $\Theta(\log n/ \log\log n)$ w.h.p. Their proof used the neighbor-of-neighbor approach for part of the way, and a non-constructive argument for the rest of the way. Thus their proof does not immediately suggest a *routing algorithm*. We now show that Non-GREEDY routing results in paths of length $\Theta(\log n/ \log\log n)$ w.h.p.

THEOREM 2.2. *Using the NoN-GREEDY routing algorithm, a message is routed between any two nodes in the small-world percolation network over n nodes, in $O(\frac{\log n}{\log\log n})$ hops, with probability at least $1 - \frac{1}{n}$ (the probability is taken over the configuration of the graph).*

PROOF. The L_1 distance between any two nodes is at most n. So we assume the worst case - that the distance between the source and target is n. We partition the routing into two phases. In the first phase, the message is routed so that the remaining distance to the target diminishes to $e^{\sqrt{\log n}}$ or less. In the second phase, the message covers the remaining distance. We show that each phase takes $O(\log n/ \log\log n)$ w.h.p., thus proving the theorem. The first phase was handled in Lemma (6.1) from [10].

LEMMA 2.3 ([10]). *If $m = (2d+2)\cdot 2^{d+1} \log n/ \log\log n$, then after m NoN-GREEDY routing steps, the message would reach a node that lies at distance $e^{\sqrt{\log n}}$ or less from the destination, with probability at least $1 - \frac{1}{n^{2d}}$.*

The second phase of the routing could in fact be performed by plain GREEDY routing.

LEMMA 2.4. *Assume a message is at distance $e^{\sqrt{\log n}}$ from its destination. With probability at least $1 - \frac{1}{n^{2d}}$, the message would reach its destination within $O(\log n/ \log\log n)$ GREEDY steps.*

First we show the following:

CLAIM 2.5. *Assume the message is at distance δ from the destination and after performing one greedy step the message is at distance δ'. There is an $\epsilon(d) = \epsilon > 0$ independent of δ, such that*
$\Pr[\delta' \leq \lceil(1 - \frac{1}{k})\delta\rceil] \geq 1 - \frac{1}{k^\epsilon}$.

PROOF. Assume the message is at node $\vec{0}$, and the target note t is such that $||t|| = \delta$. For each integer k define B_k to be all nodes with distance at most $(1 - \frac{1}{k})\delta$ from t (for convenience we remove the ceilings and floors). We calculate the probability there is an edge from $\vec{0}$ to the ball B_k. Define ℓ_i to be the number of vertices x such that $||x|| = i$ and x is in B_k. We have:

$$\Pr[\vec{0} \text{ is not connected to } B_k] = \prod_{i=\delta/k}^{\delta} (1 - i^{-d})^{\ell_i}$$

$$\leq \prod_{i=2\delta/k}^{\delta} (1 - i^{-d})^{\ell_i} \leq \prod_{i=2\delta/k}^{\delta} e^{\ell_i/i^d} = \exp(\sum_{i=2\delta/k}^{\delta} \frac{\ell_i}{i^d})$$

Now assuming that ℓ_i is $\Theta(i^{d-1})$ for $\frac{2\delta}{k} \leq i \leq \delta$, for some constant ϵ it holds that

$$\exp(\sum_{i=2\delta/k}^{n} \frac{\ell_i}{i^d}) \leq \frac{1}{k^\epsilon}$$

which proves the claim. It remains to show that indeed $\ell_i = \Theta(i^{d-1})$ for $\frac{2\delta}{k} \leq i \leq \delta$. There are $\Theta(i^{d-1})$ nodes at

distance i from $\vec{0}$. we need to show that a constant fraction of them are in B_k. Let i take some value $2\delta/k \leq i \leq \delta$. Now let x be a point on a shortest path from $\vec{0}$ to u such that $||x|| = \frac{3}{4}i$; i.e. x is at distance $\delta - \frac{3}{4}i$ from u. There are $\Theta((\frac{1}{4}i)^{d-1}) = \Theta(i^{d-1})$ points that are of distance $\frac{1}{4}i$ from x. How many of them are of distance i from $\vec{0}$? The points at distance $\frac{1}{4}i$ from x are evenly divided between the 2^d quadrants of the ball around x. It follows that a $2^{-d} = \Theta(1)$ fraction of them are at distance i from $\vec{0}$. The distance of each of these points from u is at most

$$(\delta - \frac{3}{4}i) + \frac{1}{4}i = \delta - \frac{1}{2}i \leq \delta(1 - \frac{1}{k}).$$

Which concludes the proof of Claim 2.5. □

PROOF OF LEMMA 2.4. According to the previous claim, the probability the distance is reduced by a factor of $1 - \frac{1}{(\log n)^{1/4}}$ is $1 - \frac{1}{\log^{\epsilon} n}$. This means that $o(\log n/\log\log n)$ steps, each reduces the distance by $1 - \frac{1}{(\log n)^{1/4}}$, would route the message to the destination. We prove this occurs with probability $1 - \frac{1}{n^{2d}}$ using the following argument: Let X_i be the random Bernoulli variable indicating whether the i^{th} NoN-hop have failed in reducing the distance by a factor of $1 - \frac{1}{(\log n)^{1/4}}$. We know that $\Pr[X_i = 1] \leq \frac{1}{\log^{\epsilon} n}$. Now assume that the variable X_i is simulated by tossing $\epsilon \log\log n$ fair coins and setting $X_i = 1$ if all coins turned up to be 1. Now we have $c \log n$ fair coins, and if less than $\frac{3}{4}$ of the coins turned up to be 1 the algorithm will not fail. The standard Chernoff bound [9] shows there is a constant c such that this happens with probability at least $1 - \frac{1}{n^{2d}}$. □

The proof of Theorem 2.2 is now completed by combining Lemma 2.3 which handled the first phase of the routing, with Lemma 2.4 which handled the second phase of the routing.

3. SMALL-WORLD P2P NETWORKS

In this section, we analyze GREEDY and NoN-GREEDY routing for various randomized P2P routing networks which are related to the small world model. Skip Graphs, which are of a different flavor, are analyzed in Section 4. We begin by defining these networks formally. For each of the following we assume there are $n = 2^{\ell}$ nodes arranged on a circle.

- **Randomized-Hypercube** [8, 15]: The out-degree of each node is ℓ. For each $1 \leq i \leq \ell$, node $\mathbf{x}$ makes a connection with node $\mathbf{y}$ defined as follows: The top $i-1$ bits of $\mathbf{y}$ are identical to those of $\mathbf{x}$. The i^{th} bit is flipped. Each of the remaining $\ell - i$ bits is chosen uniformly at random. Edges are directed.
- **Randomized-Chord** [36, 15]: Node $\mathbf{x}$ makes ℓ connections as follows: Let $r(i)$ denote an integer chosen uniformly at random from the interval $[0, 2^i)$. Then for each $0 \leq i < \ell$, node $\mathbf{x}$ creates an edge with node $(\mathbf{x} + 2^i + r(i)) \bmod n$. Edges are directed. Each node has out-degree ℓ.
- **Symphony** [23]: Node $\mathbf{x}$ establishes a *short-distance* edge with node $(\mathbf{x}+1) \bmod n$. Node $\mathbf{x}$ also establishes $k \geq 1$ *long-distance* edges as follows: For each edge, node $\mathbf{x}$ first draws a random number r from the probability distribution $p(x) = 1/(x \ln n)$ where $x \in [1, n]$ and then establishes a link with node $\lceil \mathbf{x} + r \rceil \bmod n$. Edges are directed. The resulting graph is thus a multi-graph since two $\mathbf{x}$ could be connected to $\mathbf{y}$ by more than one edge.
- **Symphony***: Node $\mathbf{x}$ establishes a *short-distance* edge with node $(\mathbf{x}+1) \bmod n$. Let δ denote a real number satisfying $\ln \delta = (\ln n)/k$. Let $\mathcal{I}_1 = [1, \delta]$. For $1 < i \leq k$, let $\mathcal{I}_i = (\delta^{i-1}, \delta^i]$. For interval $\mathcal{I}_i$, let ϕ^i denote a probability distribution over integers in $\mathcal{I}_i$ such that the probability at integer d is proportional to $1/d$. For each $1 \leq i \leq k$, an edge is established with a node lying clockwise distance d away, where d is an integer drawn from ϕ^i. Edges are directed. The out-degree of each node is k.

Symphony* with $k = 1$ is identical to Kleinberg's construction [19] in one dimension. For larger k, it is akin to chopping the probability distribution into k equal pieces and carrying out *stratified sampling*. Experimental results indicate that Symphony* is slightly superior to Symphony. Moreover, all networks are structurally similar to small-world percolation networks with $d = 1$ (see Definition 2.1). An important distinction is that the out-degree for each of the P2P routing networks is fixed.

Some easy adaptations of Lemma 2.3 and 2.4 could be used to prove the following theorem:

THEOREM 3.1. *For randomized-Chord and for randomized-hypercube, NoN-*GREEDY *routing from x to y requires only $\Theta(\log n/\log\log n)$ steps, with probability at least $1 - \frac{1}{n}$, for an arbitrary pair of nodes x, y.*

The next theorem presents lower bounds for GREEDY. We believe that a high probability result can be derived from the theory we develop in Section 5. However, we include the proofs below because of their relative simplicity.

THEOREM 3.2. *The expected number of hops required by* GREEDY *routing in randomized-hypercube, randomized-Chord and Symphony* with $O(\log n)$ links per node, is $\Omega(\log n)$. The expectation is over the choice source, target and the formation of the graph.*

PROOF. *Randomized-hypercube*: Consider the route from node $\mathbf{x}$ to node $\mathbf{y}$. Successive hops correspond to *fixing* the top bit of $\mathbf{z} \oplus \mathbf{y}$, where $\mathbf{z}$ is the current node. The probability that a specific bit requires fixing is half. It follows that the expected number of hops is $\ell/2 = \Theta(\log n)$.

Randomized-Chord: We first prove a lemma related to the the following process: A particle starts at position m where $m > 0$ is an integer. At successive time-steps, the particle moves to a new position. When the current position is p, then the new position is a random variable X^p that ranges over the integers $0, \ldots, p$. We assume that for all $i \in [0, p-1]$, $Pr[X^p = i] > 0$. The process terminates when the particle reaches position 0. Let $T(m)$ denote the number of steps required for the process to terminate.

LEMMA 3.3. *If for all $p \geq 2$, $Pr[X^p = i] \geq 2^{i-1} Pr[X^p = 0]$ for $i \in [0, p-2]$, then $\mathbf{E}T(m) = \Omega(m)$.*

PROOF. Consider the same process but with each probability distribution X^p replaced by Y^p where $Pr[Y^p = i] = 2^{i-1} Pr[Y^p = 0]$ for $1 \leq i \leq n-2$ and $Pr[Y^p = p-1] = Pr[Y^p = p] = 0$. If $U(m)$ represents the number

of steps required for the new process to terminate, then $\mathbf{E}T(m) \geq \mathbf{E}U(m)$. For each p, $Pr[Y^p = 0] = 1/2^{p-1}$ and $Pr[Y^p = i] = 2^{i-1}/2^{p-1}$ for $1 \leq i \leq p-2$. Using induction, it can be shown that for $m > 0$, $\mathbf{E}U(m) \geq cm$ for some constant $c < 1$. □

For a GREEDY route in randomized-Chord, we define *phases* as follows: Phase 0 consists of one integer, namely 0. For $p \geq 1$, phase p consists of all integers in the interval $[2^{p-1}, 2^p - 1)$. Consider a message in phase $p \geq 2$, i.e., its remaining distance is d such that d belongs to phase p. For $0 \leq p' \leq p-2$, let $\phi(p \to p')$ denote the probability that the next phase is p'. By the definition of R-Chord, only two links at the current node decide the next hop for forwarding the message. For $d' \in [0, d - 2^p]$, the probability that the remaining distance is d' is exactly $1/2^{p-1}$. For $d' \in [d - 2^p + 1, d - 2^p + 2^{p-1}]$, the probability is exactly $(2^p - d - 1)/(2^{p-1}2^{p-2})$. The latter probability is larger iff $d \leq 3 \cdot 2^{p-2} - 1$. In any case, $\phi(p \to p') \geq 2^{p'-1}\phi(p \to 0)$ for $0 \leq p' \leq p-2$. In fact, the equality holds if $d > 3 \cdot 2^{p-2}$. There are $\log_2 d$ different phases if the initial distance is d. By applying Lemma 3.3, we deduce that the expected number of routing steps for distance d is $\Omega(\log d)$. Averaged over all possible values of d, we get that the average length of GREEDY routes is $\Omega(\log n)$.

Symphony* can be handled along the same lines. □

THEOREM 3.4. *The expected number of hops taken by NoN-GREEDY to route between any two nodes in Symphony* is $O\left(\frac{\log^2 n}{k \log k}\right)$, when $1 \leq k \leq \log n$ and the expectation is over the formation of the graph.*

PROOF. Consider node $\mathbf{x}$ that holds a message destined for node $\mathbf{y}$ lying clockwise distance d away. It is proven in [23] that GREEDY routing takes $O((\log n \log d)/k)$ hops. Therefore, if $\log d \leq \log n/\log k$, then the remaining distance can be covered by NoN (which is faster than plain GREEDY) in $O(\log^2 n/(k \log k))$ hops.

We now consider large d satisfying $\log n/\log k < \log d \leq \log n$. Let $r(d) = (ck \log d)/\log n$ where d is the clockwise distance currently remaining and c is a constant that we will shortly fix. Since $\log n/\log k < d \leq \log n$, we deduce that $ck/\log k < r \leq ck$. Let $\mathcal{E}$ denote the event that the current node is able to diminish the remaining distance from d to at most $d/r(d)$ in (at most) two hops. Let $\phi(\mathcal{E})$ denote the probability that event $\mathcal{E}$ occurs. We will shortly prove that $\phi(\mathcal{E}) = \Omega(k/\log n)$, independent of d. Thus the expected number of nodes encountered before a successful event $\mathcal{E}$ occurs is $O((\log n)/k)$. Since $ck/\log k < r$, there can be at most $O(\log n/\log k)$ such events for a total of $O(\log^2 n/(k \log k))$ hops. When d becomes small enough to satisfy $\log d < \log n/\log k$, plain GREEDY routing will take at most $O(\log^2 n/(k \log k))$ hops. Summing the two, the total number of hops is $O(\log^2 n/(k \log k))$.

Proof of $\phi(\mathcal{E}) = \Omega(k/\log n)$: Recall that $\mathcal{E}$ is the event that the current node is able to diminish the remaining distance d to at most $d/r(d)$ in (at most) two hops. Let $d' = \lceil d(1 - 1/r(d)) \rceil$. Let ψ denote the probability that node $\mathbf{x}$ has a link in $[d', d]$. There are at least $k \log d/\log n$ nodes (including $\mathbf{x}$ itself) reachable from $\mathbf{x}$ in zero or one hop, such that the node is at most clockwise distance d away. Let ψ denote the probability that such a node has a link in $[d', d]$. Overall, the probability that one or more of these nodes has a link in $[d', d]$ is $\phi(\mathcal{E}) \geq 1-(1-\psi)^{(k \log d/\log n)}$. We will shortly show that $\psi \geq c'k/(r(d) \log n)$ where c' is some constant. We had defined $r(d) = (ck \log d)/\log n$. We set $c = c'$. This ensures that $(c'k/(r(d) \log n))(k \log d/\log n) = k/\log n \leq 1$. Using the fact that $1-(1-x)^t \geq xt/2$ if $x \in (0,1)$ and $xt \leq 1$, we deduce that $\phi(\mathcal{E}) \geq (c'k^2 \log d)/(2r(d) \log^2 n)$. Substituting $r(d) = (c'k \log d)/\log n$, we get $\phi(\mathcal{E}) \geq k/(2 \log n)$.

Proof of $\psi = \Omega(k/(r(d) \log n))$: Recall that ψ denotes the probability that node $\mathbf{x}$ has a link in $[d', d]$ where $d' = \lceil d(1 - 1/r(d)) \rceil$. From the definition of δ for Symphony*, $\ln \delta = (\ln n)/k$. If $[d', d]$ is completely contained in some interval $\mathcal{I}_i$ (for definition of $\mathcal{I}_i$, see the definition of Symphony*), then $\psi = s_i^{-1} \sum_{i \in [d', d]} 1/i$. Now $\sum_{i \in [d', d]} 1/i = \Omega(\ln 1/(1-1/r(d)) = \Omega(1/r(d))$. Substituting $s_i^{-1} = \Omega(k/\log n)$, we get $\psi = \Omega(k/(r(d) \log n))$. If $[d', d]$ spans two intervals $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$, then $\psi = \psi_1 + \psi_2 - \psi_1\psi_2$, where $\psi_1 = s_i^{-1} \sum_{i \in [d', \delta^i]} 1/i$ and $\psi_2 = s_{i+1}^{-1} \sum_{i \in (\delta^i, d]} 1/i$. Using the fact that $a + b - ab \geq \frac{3}{4}(a + b)$ if $a + b \leq 1$, we deduce that $\psi \geq \frac{3}{4}(\psi_1 + \psi_2)$. Since both s_i^{-1} and s_{i+1}^{-1} are $\Omega(k/\log n)$, we get $\psi \geq \frac{3}{4}(ck/\log n) \sum_{i \in [d', d]} 1/i$, where c is some constant. It follows that $\psi = \Omega((k/\log n) \ln 1/(1 - 1/r(d))) = \Omega(k/(r(d) \log n))$. □

We believe that Theorem 3.4 extends to all other P2P randomized networks we defined earlier.

4. SKIP GRAPHS

In this section, we analyze NoN-GREEDY routing in skip-graphs [4], which adapt skip-lists [30] for creating a randomized P2P routing network. SkipNet [17] is another network along the same lines. We follow the description in [4].

4.1 Definitions

In a skip-graph, a node x possesses a key $k(x)$ and a set of out-going edges. For node x, the out-going edges are determined by $m(x)$, its *membership vector*, which is an infinite string of *random bits*. Membership vectors are chosen independently by different nodes. We think of $|m(x)|$ as infinite for convenience even though $O(\log n)$ bits suffice with overwhelming probability, in a network of n nodes.

For notational convenience, we assume that the set of keys corresponds to the set of integers $\{0, 1, 2, \ldots, n-1\}$. Nodes are ordered by their keys and placed on a circle, ordered by their keys. Node i is then connected by edges to nodes $i-1$ and $i+1$, where arithmetic is done modulo n. The remainder of the edges are determined by the membership vectors. Denote by $m_k(x)$, the first k bits of $m(x)$. Let (x, y) denote the set of integers lying *between* x and y, going clockwise along the circle from x to y. Then nodes x, y are connected by an edge if there exists some k such that $m_k(x) = m_k(y)$, and $\forall z \in (x, y)$: $m_k(z) \neq m_k(x)$. In other words, two nodes are connected by an edge if their corresponding membership vectors share some prefix which is not shared by any of the nodes between them. The cycle edges could be viewed as corresponding to the empty prefix. It is easy to see that w.h.p., all nodes have a logarithmic degree.

Note that since the edges do not depend on the keys themselves but rather on their ordering and the random vectors, the keys may be arbitrary and carry semantic meaning. This is in contrast with the other networks we discuss, which requires that keys be random.

4.2 Routing in Skip Graphs

The routing algorithm suggested in [4] and [17] is GREEDY: a node x routes the message along the link corresponding to the longest-possible prefix of $m(x)$, without overshooting the target. Such GREEDY routing takes $O(\log n)$ hops on average. We improve by showing in Theorem 4.1 that NoN-GREEDY routing takes only $\Theta(\log n/\log\log n)$ hops on average, and by showing in Theorem 5.1 that GREEDY routing needs $\Omega(\log n)$ hops.

THEOREM 4.1. *The expected length of a path between node $n-1$ and node 0 is $O(\log n/\log\log n)$, where the expectation is taken over the choices of membership vectors.*

The idea of the proof is to show that with fixed probability there exists a length 2 path which reduces the distance to the target by a logarithmic factor. Assume the message is at node d (i.e. is at distance d from the destination 0). We say a NoN step *succeeds* if there is a path of length 2 originating from d which reduces the distance to the target by a factor $\frac{1}{\alpha}\log d$, i.e., the path leads to the segment $[0, \frac{\alpha d}{\log d}]$, where α is some constant to be chosen later.

LEMMA 4.2. *Let D be the event that the algorithm reached node d. There exists a constant α independent of n, d such that $\Pr[$NoN step succeeds $\mid D] \geq 0.3$.*

First we show Lemma 4.2 is sufficient to prove Theorem 4.1. Since at each step the success probability is ≥ 0.3, it would take the algorithm on average less than $10\log n/\log\log n$ steps to succeed $3\log n/\log\log n$ times. As long as the distance to the target is larger than $2^{\sqrt{\log n}}$, each success would reduce the distance by a factor of $\frac{\alpha}{\sqrt{\log n}}$. Since

$$n\cdot\left(\frac{\alpha}{\sqrt{\log n}}\right)^{\frac{3\log n}{\log\log n}} \leq 2^{\sqrt{\log n}}$$

the message would reach a distance of $O(2^{\sqrt{\log n}})$ from the target. From here on, the distance to the target is reduced by a factor of $\frac{1}{2}$ with probability at least $\frac{1}{4}$. Therefore the message routes through the remaining $2^{\sqrt{\log n}}$ in less than $4\sqrt{\log n}$ hops on expectation. We conclude that the expected number of hops it takes a message to route from 0 to n is at most $20\log n/\log\log n + 4\sqrt{\log n} = O(\log n/\log\log n)$.

PROOF OF LEMMA 4.2. Let X denote the number of paths of length 2 that connect node d with the segment $[0, \frac{\alpha d}{\log d}]$. Limit the first of the two steps to be of length at most $d/4$ and the second of the two steps to correspond to a prefix of length $\lfloor\log d\rfloor$. Recall that D is the event that the algorithm reached node d. Our goal is to show that $\Pr[X > 0 \mid D] \geq 0.3$. We do this by bounding the expectation of $X|D$ from below (Lemma 4.4) and the variance of it from above (Lemma 4.5). We begin with a technical lemma. Let $i, j \in [0, d]$ and A_{ij} be the event: "there is an edge between i and j, and this edge corresponds to a prefix of length $\lfloor\log|i-j|\rfloor$ ". We may write A_{ijk} meaning $A_{ij} \wedge A_{jk}$.

LEMMA 4.3. *For each $0 \leq i, j \leq d$ we have $\Pr[A_{ij}|D] = \Theta(\frac{1}{|i-j|})$.*

PROOF. Ignore for the moment the conditioning on D and consider the probability i, j share a prefix of length k while the $|i-j|-1$ nodes between them do not share it. This probability is $2^{-k}\cdot(1-2^{-k})^{|i-j|-1}$. Now if $k = \lfloor\log(|i-j|)\rfloor$ then there exists two constants c_1, c_2 such that

$$\frac{c_1}{|i-j|} \leq 2^{-k}\cdot(1-2^{-k})^{|i-j|-1} \leq \frac{c_2}{|i-j|}$$

In order to handle the conditioning, we make the following minor change in the algorithm: The algorithm will ignore the line edges, i.e., the edges of length 1 which correspond to the empty prefix. If the algorithm is stuck and must use a line edge, then it continues by proceeding along the line edges all the way to the target. The algorithm is stuck only if it encounters a node whose only edge towards 0 is the line edge. The probability node i does not have an edge towards 0 corresponding to a prefix of length 1 is at least $\frac{1}{2^i}$. Therefore the average number of hops added due to the change in the algorithm is at most $\sum\frac{1}{2^i}i$ which is $O(1)$. Assume that nodes encountered by the algorithm prior to d were $v_1, v_2, \ldots, v_m$. Now, as long as line edges are not used, the first bit of the membership vector is unchanged, i.e. all the nodes $v_1, v_2, \ldots, v_m$ share the same first bit in their membership vector. Assume the membership vectors of nodes in $[0, d-1]$ are sampled *after* the vectors of $[d, n]$. The conditioning on D implies that when sampling membership vector for nodes in $[0, d-1]$, we are prohibited from sampling vectors that would create an edge to $v_1, v_2, \ldots, v_m$. Now consider a node u in $[i+1, j-1]$. There are at least 2^{k-1} vectors of length k which differ from $m(v_1), m(v_2), \ldots$ on the first bit. We conclude that the probability $m(u)_k \neq m(i)_k$ conditioned on D is $1-\Theta(2^{-k})$. Similarly $\Pr[m(i)_k = m(j)_k|D] = \Theta(2^{-k})$. Now as in the unconditioned case, if $k = \lfloor log(|i-j|)\rfloor$ then there exists constants c_1, c_2 such that $\frac{c_1}{|i-j|} \leq \Pr[A_{i,j}|D] \leq \frac{c_2}{|i-j|}$. □

Our goal is to show that with a fixed probability, X is greater than zero. We start by calculating its expectation.

LEMMA 4.4. *Let X denote the number of paths of length 2 that connect node d with the segment $[0, \frac{\alpha d}{\log d}]$. For a sufficiently large constant α, it holds that $\mathbb{E}[X|D] \geq 5$.*

PROOF. Let S denote the set of nodes in the segment $[0, \frac{\alpha d}{\log d}]$. By linearity of expectation we have

$$\begin{aligned}\mathbb{E}[X|D] &= \sum_{\frac{d}{4}<i<d}\sum_{k\in S}\Pr[A_{dik}|D]\\ &= \sum_{\frac{d}{4}<i<d}\sum_{k\in S}\Pr[(A_{di}|D)\wedge(A_{ik}|D)]\\ &= \sum_{\frac{d}{4}<i<d}\sum_{k\in S}\Pr[A_{di}|D]\cdot\Pr[A_{ik}|D]\end{aligned}$$

where the last equality holds since $A_{di}|D$ and $A_{ik}|D$ are independent. For a fixed $k \in S$ Lemma 4.3 shows that there exists constant c_1, c_2, c_3 such that

$$\begin{aligned}\sum_{\frac{d}{4}<i<d}\Pr[A_{di}|D]\cdot\Pr[A_{ik}|D] &\geq c_1\sum_{\frac{d}{4}<i<d}\frac{1}{i(k-i)}\\ &\geq c_2\sum_{\frac{d}{4}<i<d}\frac{1}{i(d-i)}\\ &\geq c_3\frac{\log d}{d}\end{aligned}$$

Figure 2: The events A_{dix}, A_{dij}.

Now $\mathbb{E}[X|D] \geq \alpha c_3$ and the lemma holds for α a large enough constant. □

By Chebyschev's inequality, $\Pr[X = 0|D] \leq \frac{var[X|D]}{\mathbb{E}^2[X|D]}$. We show that $\Pr[X = 0|D] \leq 0.7$ by bounding the size of the covariance of each pair $A_{0ix}|D, A_{0jy}|D$.

LEMMA 4.5. *For each $\frac{d}{4} < i, j \leq d$ and each $x, y \in [0, \frac{\alpha d}{\log d}]$ it holds that $cov[A_{dix}|D, A_{djy}|D] \leq \frac{1}{2}\Pr[A_{dix}|D] \cdot \Pr[A_{djy}|D]$.*

PROOF. From here on we drop for notational convenience the "$|D$" symbol after each random variable. Consider a pair of events as depicted in figure 2. We have:

$$\Pr[A_{dix} \wedge A_{dij}] = \Pr[A_{di}] \cdot \Pr[A_{dj}|A_{di}] \cdot \Pr[A_{jy}|A_{di}, A_{dj}] \cdot \Pr[A_{ix}|A_{jy}, A_{di}, A_{dj}]$$

We bound the size of each of the elements in the expression. The occurrence of the event A_{di} means that $m(i)$ could not be used in order to establish an edge (d, j), therefore $\Pr[A_{dj}|A_{di}] \leq \Pr[A_{dj}]$. The event A_{jy} is independent of the membership vectors in the segment $[d, j]$ and therefore is independent of A_{di}, A_{dj}. Let $\ell = \lfloor \log d \rfloor$ and $m_\ell(i)$ denote the first ℓ bits of $m(i)$. Consider, as before, only the case in which the second hop corresponds to prefixes of length ℓ. Now we have that $\Pr[A_{ix}|A_{di}, A_{dj}, A_{jy}]$ is the probability $m_\ell(x) = m_\ell(i)$, and that for each node k between i and x it holds that $m_\ell(k) \neq m_\ell(i)$, conditioned on the existence of the edges $(d, i), (d, j), (j, y)$. The existence of the edge (j, y) means that for all nodes $j < k \leq x$ it holds that $m_\ell(k) \neq m_\ell(j)$; i.e. it excludes the existence of one prefix in the segment $[j, x]$ and therefore has only a small effect on the probability of A_{ix}. The existence of the edge (d, j) is independent of membership vectors in the segment $[j+1, x]$. Denote by L the number of prefixes of length ℓ that may be sampled under the conditioning on D. The proof of Lemma 4.3 showed that $L = \Theta(2^\ell)$. We have:

$$\begin{aligned}\Pr[A_{ix}|A_{di}, A_{dj}, A_{jy}] &\leq \frac{1}{L-1} \cdot \left(1 - \frac{1}{L-1}\right)^{(x-j-1)} \\ &\leq 1.1(1 - L^{-1})^{i-j} \cdot L^{-1}(1 - L^{-1})^{x-i+1} \\ &\leq 1.1(1 - L^{-1})^{-d/4} \cdot \Pr[A_{ix}] \\ &\leq 1.5 \Pr[A_{ix}]\end{aligned}$$

We have

$$\begin{aligned}\Pr[A_{dix} \wedge A_{dij}] &\leq 1.5 \Pr[A_{di}] \Pr[A_{ix}] \Pr[A_{dj}] \Pr[A_{jy}] \\ &= 1.5 \Pr[A_{dix}] \cdot \Pr[A_{djy}]\end{aligned}$$

therefore

$cov[A_{dix}, A_{djy}] = \Pr[A_{dix} \wedge A_{djy}] - \Pr[A_{dix}] \cdot \Pr[A_{djy}] \leq \frac{1}{2}\Pr[A_{dix}] \cdot \Pr[A_{djy}]$. The cases in which $x > y, x = y, i = j$ are done in an analogous way. □

Now calculate the variance

$$\begin{aligned}var[X] &= \sum_{i,x} var[A_{d,i,x}] + \sum_{i,j,x,y} cov[A_{dix}, A_{djy}] \\ &\leq \mathbb{E}[X] + \frac{1}{2}(\sum_{i,x} \Pr[A_{dix}] \cdot \sum_{j,y} \Pr[A_{dix}]) \\ &= \mathbb{E}[X] + \frac{1}{2}\mathbb{E}^2[X]\end{aligned}$$

Now we have

$$\Pr[X = 0] \leq \frac{\mathbb{E}[X] + \frac{1}{2}\mathbb{E}^2[X]}{\mathbb{E}^2[X]} \leq 0.7$$

where the last inequality follows since $\mathbb{E}[X] \geq 5$. This completes the proof of Lemma 4.2.

While the constant derived from the proof Theorem 4.1 is rather large, simulations in [26] have shown that for $n = 2^{17}$, NoN-GREEDY improves over GREEDY by a factor of about 1.9, suggesting that the real constant is quite small.

5. LOWER BOUNDS

In this section we prove that in order to find a path between nodes at distance n, a routing algorithm must either run in $\Omega(\log n)$ time w.h.p (i.e. $\Omega(\log n)$ hops), or must use additional knowledge about the neighbor's neighbors of a node. The lower bound holds for a model which generalizes the GREEDY algorithm, thus it applies for a larger family of algorithms which includes GREEDY. It holds both for small-world percolation networks and skip graphs.

A logarithmic lower bound of $\Omega(\log^2 n)$ for GREEDY routing in Kleinberg's construction [19] in one dimension was proved by Barrière *et al* [6]. Aspnes *et al* [3] extended the result to a larger family of random graphs. They show that if the average degree is $O(\log n)$ then GREEDY routing would take $\Omega(\log n)$ hops on average. The proof however is limited to the case where the nodes are set on a one dimensional line and the probability upon the edges has some symmetry assumptions that do not apply to skip graphs. We show lower bounds for small-world percolation networks and skip-graphs. We believe that randomized-Chord, randomized-hypercube and Symphony are quite similar to small-world percolation networks, and the proofs could be adapted for each of them.

5.1 A Probing Model

Assume that our goal is to find a path between two specific vertices distance n apart, say node 0 and node n. In order to do so, an algorithm must *probe* the vertices of the graph, where the probing of a vertex reveals all the edges connected to it. Our lower bounds apply in a *probing* model, where we bound the number of probes needed to find a path. Clearly, a lower bound on the number of probes needed by the algorithm is a lower bound on the (sequential) time complexity of a routing algorithm.

We define a 1−local algorithm to be a probing algorithm with the following properties:

1. The algorithm begins by probing the node 0.
2. The algorithm only probes nodes to which it has already established a path from 0.

The term *local* derives from the assumption that the algorithm starts at 0 and is only allowed to probe nodes it has

already reached. The term 1−local is used, since the probing of a node reveals its neighborhood of radius 1, i.e. its neighbors. If it is assumed that a probe reveals a neighborhood of radius k then the algorithm is termed *k-local*. Every routing algorithm which relies on local information only, corresponds to a 1−local probing algorithm. The GREEDY algorithm therefore is 1−local. The NoN-GREEDY algorithm could be viewed, following Theorems 2.2,4.1 as either a 2−local algorithm with $O(\log n/\log\log n)$ probes on average, or as a 1−local algorithm having an averge probing complexity of $O(\log^2 n/\log\log n)$.

5.2 Lower Bounds in the Probing Model

THEOREM 5.1. *(i) In a skip graph - any* 1-local *algorithm that outputs a path between two nodes at distance n, must probe $\Omega(\log n)$ probes, with probability at least $1-\frac{1}{n^\epsilon}$. In particular, the expected number of probes is $\Omega(\log n)$.*
(ii) In a $d-$dimensional small-world percolation network - any 1-local *algorithm that outputs a path between two nodes at distance n, must probe $\Omega(\log n)$ probes, with probability at least $1-\frac{1}{n^\epsilon}$. In particular, the expected number of probes is $\Omega(\log n)$.*

The theorem implies that if a node holds only its neighbors then *any* routing algorithm would need $\Omega(\log n)$ probes w.h.p. Thus the assumption that nodes have some knowledge of their neighbor's neighbors is essential.

We first argue that GREEDY dominates any other 1−local algorithm. The following lemma holds both for skip graphs and small-world percolation networks.

LEMMA 5.2. *Let A be a 1−local algorithm . Denote by F_d, G_d the random variables representing the number of probes it takes the algorithm A and the* GREEDY *algorithm respectively, to find a path between two nodes at distance d. For all $d > k > 0$ it holds that* $\Pr[G_d \le k] \ge \Pr[F_d \le k]$.

PROOF. We distinguish between the two cases.

Small-World Percolation Networks. For convenience, we label the target node as 0, and assume that the mesh is infinite. The trick is to give A some extra power. Assume that at some step, the closest node to 0 which A had found is at distance d from 0, where the distance is measured by the L_1 norm. At this point, we grant A access to all nodes outside a ball of radius d from 0. Now if $d_1 > d_2$ then for every configuration of edges, every move A can do in case the distance is d_1, is also available when the distance is d_2, i.e. for every k, $\Pr[F_{d_1} \le k] \le \Pr[F_{d_2} \le k]$. In other words, for every k, $\Pr[F_d \le k]$ is monotonically decreasing in d. The algorithm A samples some point v. Let f_i denote the event that the neighbor of v which is closest to 0 is at distance i. The greedy choice is to sample the point closest 0, call that point u. Let g_i denote the event that the closest neighbor of u to 0 is at distance i. Now

$$\Pr[F_d \le k] = \sum_i \Pr[f_i] \cdot \Pr[F_i \le k-1]$$

Since u is closer to 0 than v we know that for every i, $\sum_{j=0}^{i} \Pr[g_j] \ge \sum_{j=0}^{i} \Pr[f_j]$. Now since $\Pr[F_d \le k]$ is monotonically decreasing we have:

$$\sum_i \Pr[g_i] \Pr[F_i \le k-1] \ge \sum_i \Pr[f_i] \Pr[F_i \le k-1]$$

In other words, the best thing A can do is sample the greedy point, which implies that the GREEDY algorithm dominates any other 1−local algorithm.

Skip Graphs. We use the same technique as in the previous section. Now if at some step the closest node to 0 which A had found is at distance d from 0 we supply to A both the access and the membership vectors of all the nodes in the segment $[n, d]$. We need to handle some dependencies. Denote by M_d the membership vectors of this segment. Using the notation of the previous section we have

$$\Pr[F_d | M_d \le k] = \sum_i^{d-1} \Pr[f_i | M_d] \cdot \Pr[F_i | M_i \le k-1]$$

Now we proceed by induction on k. Assume that for every instance of M_i, we have $\Pr[F_i|M_i \le k-1] \le \Pr[G_i|M_i \le k-1]$. It is easy to see that for every instance of M_d, we have $\Pr[f_i|M_d] \le \Pr[g_i|M_d]$. Conclude that

$$\Pr[F_d | M_d \le k] = \sum_i^{d-1} \Pr[f_i | M_d] \cdot \Pr[F_i | M_i \le k-1]$$

$$\le \sum_i^{d-1} \Pr[g_i | M_d] \cdot \Pr[G_i | M_i \le k-1] = \Pr[G_d | M_d \le k]$$

which concludes the proof of the lemma. □

It remains to lower bound the number of hops taken by the GREEDY algorithm. Divide the nodes of the graph into sets $B_0, B_1, \ldots, B_{\log n}$ according to their distance from 0, such that $B_i = \{u | 2^{i-1} \le dist(u,0) < 2^i\}$. So $0 \in B_0$ and $n \in B_{\lceil \log n \rceil}$. We slightly change the GREEDY algorithm thus: if the algorithm reaches a node within a ball B_i it is granted access to all nodes with distance at least 2^{i-1} from 0, i.e. to all nodes in $B_i, B_{i+1}, ..., B_n$. When routing in a skip graph the algorithm is also given the membership vectors of these nodes. The reason for this change is to cancel the dependencies on previous hops, it may only reduce the number of hops GREEDY takes, since it allows the algorithm a 'free' hop to the edge of the ball B_i. For each $0 \le i \le \log n$ let X_i be the indicator of the event: "The path taken by GREEDY includes at least one vertex in B_i". Clearly the number of nodes in the path is at least $\sum_{i=o}^{\log n} X_i$.

LEMMA 5.3. *Both for skip graphs and for small world graphs and for each i, it holds that*

$$\Pr[X_i = 1 | X_{i+1}, X_{i+2}, \ldots, X_{\log n}] \ge c$$

for some constant c independent of n.

Before proving the lemma we show why it suffices to prove Theorem 5.1. Let Y_i be a Bernoulli variable with $\Pr[Y_i = 1] = c$. Now $\mathbb{E}[\sum Y_i] = c \log n \le \mathbb{E}[\sum X_i]$. Furthermore the random variable $\sum X_i$ dominates the random variable $\sum y_i$. We have

$$\Pr[\sum X_i \le \frac{1}{2} c \log n] \le \Pr[\sum Y_i \le \frac{1}{2} c \log n] \le \frac{1}{n^\epsilon}$$

according to Chernoff's bounds.

PROOF OF LEMMA 5.3. Let us assume that the values of $X_{i+1}, \ldots, X_{\log n}$ are already set and that j is the smallest index such that $X_j = 1$. Since we changed the algorithm such that when a ball B_i is reached all nodes in it are revealed, it

is clear that X_i is independent from $X_{j+1}, X_{j+2}, \ldots, X_{logn}$, it remains to analyze $\Pr[X_i = 1 | X_{i+1} = 0, X_{i+2} = 0, \ldots, X_j = 1]$. Let y be the node in B_j which is closest to 0, i.e. the node probed by GREEDY. The notation $y \sim B_i$ stands for the event - 'y is connected by an edge to B_i'. For convenience let $B = \cup_{j=0}^{i-1} B_j$. Now we distinguish between skip graphs and small world graphs:

Small-World Percolation Networks. All edges are independent of each other. Therefore $\Pr[X_i = 1 | X_{i+1} = 0, X_{i+2} = 0, \ldots, X_j = 1]$ is the probability y is connected to B_i and is not connected to $B_0, B_1, ..., B_{i-1}$, conditioned on it being connected to one of them. We need to compute:

$$\Pr[y \sim B_i \wedge y \not\sim B | y \sim B \cup B_i] = \frac{\Pr[y \sim B_1 \wedge y \not\sim B]}{\Pr[y \sim B \cup B_i]} = \frac{\Pr[y \sim B_1] \cdot \Pr[y \not\sim B]}{1 - Pr[y \not\sim B] \Pr[y \not\sim B_i]}$$

It is easy to verify that $\Pr[y \not\sim B] \geq \Pr[y \not\sim B_i]$ and that $\Pr[y \not\sim B] \geq \epsilon$ for some constant ϵ. We have:

$$\frac{\Pr[y \sim B_i] \cdot \Pr[y \not\sim B]}{1 - Pr[y \not\sim B] \Pr[y \not\sim B_i]} \geq \frac{\Pr[y \not\sim B](1 - \Pr[y \sim B_i])}{(1 - \Pr[y \not\sim B_i])(1 + \Pr[y \not\sim B])} \geq \frac{\epsilon}{1 + \epsilon}$$

Skip Graphs. Here we have to deal with some dependencies. Let D denote the event that the algorithm reached the node y (i.e. the segment B_j which contains y). As before we need to compute:

$$\frac{\Pr[(y \sim B_i \wedge y \not\sim B) | D]}{\Pr[y \sim B \cup B_i | D]}$$

As in the proof of Lemma 4.3, it could be seen that the conditioning on D changes the probabilities by a constant factor at most. Furthermore, the events $\{y \sim B_i\}$ and $\{y \not\sim B\}$ are positively correlates. So the calculation of the previous section applies here as well. □

6. DISCUSSION

Do People Use the NoN-GREEDY Algorithm in Social Networks? Since the original motivation of analyzing small-world graphs was the modeling of social networks, it is interesting to check whether people use the NoN-GREEDY algorithm when they navigate in a social network. Recently Dodds *et al* [11] repeated the famous experiment by Milgram [24] in which letters were passed between random nodes on a social network where edges corresponds to say, an acquaintance known by first name. In [11] people were given a target and were asked to forward an email to some person they were acquainted with. The goal of forwarding was to ensure that the email would reach its destination quickly. People were also asked to explain *why* they chose the person from among their set of acquaintances. It appears that in the first two steps of the "routing", which are most meaningful, about 25% of the people sent the message to a recipient for one of the following reasons:

1. The recipient was known to have traveled to the target's geographical region.
2. The recipient's family was known to have originated from the target's geographical region.

Both reasons suggest that the recipient received the message based on who his/her (possible) friends were, and not on the individual characteristics of just the recipient. Other reasons, such as – "the recipient has the same education as the target" – could be viewed both as greedy and NoN-GREEDY steps. We can conclude that at least some of the time, the NoN-GREEDY algorithm was used.

Randomization Reduces Latency: A common strategy in the design of P2P routing networks is to first identify a static graph which is known to possess good properties and then, to adapt the static graph topology to handle the dynamism (arrival/departure of nodes) and scale (changes in the average number of nodes). The resulting dynamic routing network resembles the underlying static graph closely. In the case of skip graphs, a 'perfect' skip graph has the i^{th} edge of each node cover a distance of 2^i, i.e., the lengths of edges of a node form a geometric series. The randomization involved in the dynamic construction is usually considered as a negative by-product and much effort is put in reducing it. For instance, a deterministic P2P routing network that guarantees that the skip graph is almost 'perfect' is presented in [16]. As was noticed by Harvey *et al* [17], a perfect skip graph is similar to Chord [34]. The average length of shortest paths in both Chord (studied in [14]) and hypercubes is $\Omega(\log n)$. This leads to the following counter-intuitive and surprising fact:

> Randomization of edges *reduces* the average length of shortest paths in the hypercubes, Chord and Skip Graphs.

The reason is that the randomization enables a routing algorithm to use an 'exceptionally' long edge once in a while. The density of these long edges is just large enough so that the NoN-GREEDY algorithm finds them. In a 'perfect' skip graph, Chord, and in the hypercube - these long edges do not exist. Our results show that safety has a price: while these network topologies have guaranteed worst-case route-lengths, they enlarge the expected length of routes.

System Issues with NoN-GREEDY: An implementation of the NoN-GREEDY algorithm in a P2P network necessitates that each node acquire knowledge of its neighbor's neighbors. At first glance, it might appear that maintenance of such knowledge is problematic since it is tantamount to squaring the degree of the graph and therefore, squaring the size of the routing table at each node. However, it is important to note that the bottleneck in the system is actually the run-time cost of maintaining the TCP links between nodes. This cost remains unchanged, irrespective of which routing protocol we use: GREEDY or NoN-GREEDY. The primary concern in implementing NoN-GREEDY is the amount of communication-overhead needed to keep the neighbor-of-neighbor lists (reasonably) up-to-date. Updates could be piggy-backed on top of maintenance messages (the "keep-alive" messages). Moreover, the neighbor-of-neighbor information at a node does not have to be perfectly up-to-date at all times to derive the benefits of NoN-GREEDY routing. See [26] for further discussion of these issues and a detailed account of experimental results.

Acknowledgments

We thank James Aspnes and Gauri Shah for supplying us with their implementation of skip graphs, and for reading and commenting on an earlier version of the paper.

7. REFERENCES

[1] I. Abraham, B. Awerbuch, Y. Azar, Y. Bartal, D. Malkhi, and E. Pavlov. A generic scheme for building overlay networks in adversarial scenarios. In *Proc. Intl. Parallel and Distributed Processing Symposium (IPDPS 2003)*, Apr. 2003.

[2] M. Aizenman and C. M. Newman. Discontinuity of the percolation density in one dimensional $1/|x-y|^2$ percolation models. *Communications in Mathematical Physics*, 107:611–647, 1986.

[3] J. Aspnes, Z. Diamadi, and G. Shah. Fault-tolerant routing in peer-to-peer systems. In *Proc. 21st ACM Symp. on Principles of Distributed Computing (PODC 2002)*, pages 223–232, July 2002.

[4] J. Aspnes and G. Shah. Skip graphs. In *Proc. 14th ACM-SIAM Symp. on Discrete Algorithms (SODA 2003)*, pages 384–393, Jan. 2003.

[5] A. L. Barabási. *Linked: The New Science of Networks*. Perseus Publishing, 2002.

[6] L. Barrière, P. Fraigniaud, E. Kranakis, and D. Krizanc. Efficient routing in networks with long range contacts. In *Proc. 15th Intl. Symp. on Distributed Computing (DISC 2001)*, pages 270–284, Oct. 2001.

[7] I. Benjamini and N. Berger. The diameter of a longe-range percolation clusters on finite cycles. *Random Structures and Algorithms*, 19(2):102–111, 2001.

[8] M. Castro, P. Druschel, Y. C. Hu, and A. I. T. Rowstron. Topology-aware routing in structured peer-to-peer overlay networks. In *Proc. Intl. Workshop on Future Directions in Distrib. Computing (FuDiCo 2003)*, pages 103–107, 2003.

[9] H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Annals of Mathmematical Statistics*, 23:493–509, 1952.

[10] D. Coppersmith, D. Gamarnik, and M. Sviridenko. The diameter of a longe-range percolation graph. *Random Structures and Algorithms*, 21(1):1–13, 2002.

[11] P. S. Dodds, M. Roby, and D. J. Watts. An experimental study of search in global social networks. *Science*, 301:827–829, 2003.

[12] P. Fraigniaud and P. Gauron. (brief announcement) An overview of the content-addressable network D2B. In *Proc 22nd ACM Symposium on Principles of Distributed Computing (PODC 2003)*, pages 151–151, July 2003.

[13] P. Fraigniaud, C. Gavoille, and C. Paul. Eclecticism shrinks the world. Technical Report LRI-1376, University Paris-Sud, November 2003.

[14] P. Ganesan and G. S. Manku. Optimal routing in Chord. In *Proc. 15th ACM-SIAM Symp. on Discrete Algorithms (SODA 2004)*, pages 169–178, Jan. 2004.

[15] K. P. Gummadi, R. Gummadi, S. D. Gribble, S. Ratnasamy, S. Shenker, and I. Stoica. The impact of DHT routing geometry on resilience and proximity. In *Proc. ACM SIGCOMM 2003*, pages 381–394, 2003.

[16] N. Harvey and J. I. Munro. (brief announcement) deterministic Skipnet. In *Proc 22nd ACM Symposium on Principles of Distributed Computing (PODC 2003)*, pages 152–153, 2003.

[17] N. J. A. Harvey, M. Jones, S. Saroiu, M. Theimer, and A. Wolman. Skipnet: A scalable overlay network with practical locality properties. In *Proc. 4th USENIX Symposium on Internet Technologies and Systems (USITS 2003)*, 2003.

[18] M. F. Kaashoek and D. R. Karger. Koorde: A simple degree-optimal hash table. In *Proc. 2nd Intl. Workshop on Peer-to-Peer Systems (IPTPS 2003)*, pages 98–107, 2003.

[19] J. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *Proc. 32nd ACM Symposium on Theory of Computing (STOC 2000)*, pages 163–170, 2000.

[20] D. Loguinov, A. Kumar, V. Rai, and S. Ganesh. Graph-theoretic analysis of structured peer-to-peer systems: Routing distance and fault resilience. In *Proc. ACM SIGCOMM 2003*, pages 395–406, 2003.

[21] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: A scalable and dynamic emulation of the butterfly. In *Proc 21st ACM Symposium on Principles of Distributed Computing (PODC 2002)*, pages 183–192, 2002.

[22] G. S. Manku. Routing networks for distributed hash tables. In *Proc. 22nd ACM Symp. on Principles of Distributed Computing (PODC 2003)*, pages 133–142, July 2003.

[23] G. S. Manku, M. Bawa, and P. Raghavan. Symphony: Distributed hashing in a small world. In *Proc. 4th USENIX Symposium on Internet Technologies and Systems (USITS 2003)*, pages 127–140, 2003.

[24] S. Milgram. The small world problem. *Psychology Today*, 67(1):60–67, May 1967.

[25] M. Naor and U. Wieder. Novel architectures for P2P applications: The continuous-discrete approach. In *Proc. 15th ACM Symp. on Parallelism in Algorithms and Architectures (SPAA 2003)*, pages 50–59, June 2003.

[26] M. Naor and U. Wieder. Know thy neighbor's neighbor: Better routing for skip-graphs and small worlds. In *The Third International Workshop on Peer-to-Peer Systems (IPTPS)*, 2004.

[27] C. M. Newman and L. S. Schulman. One dimensional $1/|j-i|^s$ percolation models: The existence of a transition for $s \leq 2$. *Communications in Mathematical Physics*, 180:483–504, 1986.

[28] M. E. J. Newman, D. J. Watts, and S. H. Strogatz. Random graph models of social networks. *Proc. National Academy of Science, USA*, 99 (suppl 1):2566–2572, 2002.

[29] I. Pool and M. Kochen. Contacts and influence. *Social Networks*, 1:1–48, 1978.

[30] W. Pugh. Skip lists: A probabilistic alternative to balanced trees. *Communications of the ACM*, 33(6):668–676, June 1990.

[31] S. Ratnasamy, P. Francis, M. Handley, and R. M. Karp. A scalable Content-Addressable Network. In *Proc. ACM SIGCOMM 2001*, pages 161–172, 2001.

[32] A. I. T. Rowstron and P. Druschel. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware 2001)*, pages 329–350, 2001.

[33] L. S. Schulman. Long range percolation in one dimension. *Journal of Physics A*, 16(17):L639–L641, 1983.

[34] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proc. ACM SIGCOMM 2001*, pages 149–160, 2001.

[35] D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, pages 440–442, 1998.

[36] H. Zhang, A. Goel, and R. Govindan. Incrementally improving lookup latency in distributed hash table systems. In *ACM SIGMETRICS 2003*, pages 114–125, June 2003.

[37] B. Y. Zhao, L. Huang, J. Stribling, S. C. Rhea, A. D. Joseph, and J. D. Kubiatowicz. Tapestry: A resilient global-scale overlay for service deployment. *IEEE Journal on Selected Areas in Communications*, 22(1), Jan. 2004.