A Small-world DHT Built on Generalized Network Coordinates

Liz Ribe-Baumann
Advisor: Kai-Uwe Sattler
Ilmenau University of Technology
Ilmenau, Germany
elizabeth.ribe-baumann@tu-ilmenau.de

ABSTRACT

Large-scale distributed hash tables (DHT) are typically implemented without respect to node location or characteristics, thus producing physically long routes and squandering network resources. Some systems have integrated round trip times through proximity-aware identifier selection (PIS), proximity-aware route selection (PRS), and proximity-aware neighbor selection (PNS). While PRS and PNS tend to optimize existing systems, PIS deterministically selects node identifiers based on physical node location, leading to a loss of scalability and robustness. The trade off between the scalability and robustness gained from a DHT's randomness and the better allocation of network resources that comes with a location-aware, deterministically structured DHT make it difficult to design a system that is both robust and scalable *and* resource conserving. We present initial ideas for the construction of a small-world DHT which mitigates this trade off by retaining scalability and robustness while effectively integrating round trip times and additional node quality with the help of Vivaldi network coordinates.

1. INTRODUCTION AND MOTIVATION

Increasingly, large amounts of data are being stored in a distributed manner over wide area networks. When such data is stored in a distributed hash table (DHT), the DHT's structure provides the user with guarantees on data availability while the system remains scalable and robust. DHT applications vary widely and are used to either store index information about the location of data or to store the data itself. Mobile games and file sharing systems based on BitTorrent are both examples of real-world applications where DHTs have been used to store data indexes. A DHT that stores application data directly can be considered a form of distributed data base, with real-world applications such as mobile ad hoc networks, sensor networks, or online gaming. In either case, it is important that the underlying DHT is robust and scalable to ensure that data is not lost. Unfortunately, DHTs often waste valuable resources by failing to incorporate location information - one highly visible loss of resources stems from inefficient routing protocols. A message traverses a typical DHT on links that are completely location unaware, so a message originating in Berlin and headed for Paris may hop from continent to continent before returning to its destination. Take for example a DHT in which peers have restricted power availability, such as in sensor networks. Then the problem of inefficient routing becomes more than a simple question of latency: roundabout routes squander power, thus increasing the risk of undeliverability.

However, the integration of location information often comes at the cost of robustness and scalability by reducing the randomness that was intentionally integrated into the design of DHTs: the random selection of identifiers ensures scalability with the load distributed uniformly throughout the network and increases robustness and peer autonomy while random links - such as those used in Symphony - reduce the expected routing length in the overlay network. The balance of structure and randomness, although not fully understood, appears to play an important role in a network's routing capabilities, as demonstrated by Kleinberg on a set of small-world graphs [8].

This work-in-progress aims to develop a robust, scalable DHT protocol that effectively rations network resources. To achieve this, node location and additional node information (such as node power availability or reliability) must be integrated while maintaining a balance of network structure and randomness. Initial ideas include a small-world DHT built within a Vivaldi-based coordinate system which holds additional node information in its "height" vector. Unlike similar constructs, the proposed DHT's node identifiers are chosen randomly and its links chosen within the *network coordinate space*, rather than the node identifier space. The resulting model could thus help to soften the trade off between the stability and scalability that comes with randomness and the information that can be incorporated into a structured system, providing better use of network resources with less loss of robustness.

Work related to proximity aware DHTs and network coordinates is outlined in Section 2. Vivaldi coordinates and their suitability for conveying additional node information are then discussed in Section 3, followed by an introduction of small-world graphs and the DHT Symphony in Section 4. In Section 5, the suggested network coordinate and DHT construction is delineated and then, in Section 6, future work for the analysis of these suggestions is discussed along with open questions.

EDBT 2010, March 22-26, 2010, Lausanne, Switzerland.

2. RELATED WORK

Well established DHTs such as the Content Addressable Network (CAN) [17], Chord [22], and Kademlia [13] were designed to route efficiently without location information. Proximity-awareness has begun to permeate areas related to DHTs, including caching and replication protocols and hybrid overlays [4],[11], as well as the DHTs' original designs. Suggestions for the integration of locality have generally been directed at reducing overall traffic (or average round trip times) by modifying a combination of a DHT's characteristics: proximity-aware identifier selection (PIS) integrates location information into the selection of identifier keys; proximity-aware neighbor selection (PNS) uses the round trip times (RTTs) between nodes to establish "favorable" links; and proximity-aware route selection (PRS) chooses individual routing steps based on location information [7].

Each of these approaches is dependent on some form of network distance, and distance is defined differently from application to application. Approaches to measuring distance range from super-peers that measure and supply distance information to other peers (IDMaps [6]), to triangulation-measurements taken from selected peers (Global Network Positioning [15]), to landmark binning [18], to network coordinate systems generated from measurements piggybacked on all communication (Vivaldi [2]). In a network coordinates system, each node computes its own (dynamic) coordinates which can be used to calculate the approximate network distance between any two nodes without them ever directly communicating. In this sense, each node acts autonomously and establishes a position within the network.

The effectiveness of each proximity-aware modification depends on both the accuracy of the distance measurement protocol as well as its integration into the network protocol. PIS protocols such as Mithos [23] and SAT-Match [19] must cope with the inherent trade offs associated with selecting identifiers based on location: randomly selected identifiers are essential for the scalability and security of the system and help to balance load while location-aware identifiers prevent messages from taking extensive scenic routes (on the physical level). On the other hand, PNS and PRS protocols such as DHash++ [3] alter the overlay network's links instead of its nodes, substituting existing links and routing steps with ones that are considered more local. However, substitutions are primarily drawn from a node's direct neighborhood and not from the entire network.

3. COORDINATE SYSTEMS - VIVALDI

Network coordinate systems provide convenient and inexpensive *estimates* of network distances by embedding the network into a given space, for example a 2-dimensional plane [16], the d-dimensional Euclidean space [1], the d-dimensional Euclidean space with an additional "height" dimension [2], or a hyperbolic space [20]. Network coordinate systems were constructed with the primary goals of reducing message latency or network traffic and therefore restrict their measurements and predictions to RTTs between nodes. However, networks with specific limitations could profit from the integration of additional information such as battery power (e.g., for mobile or sensor networks) or node reliability (e.g., file sharing networks). Consider for a moment how these characteristics could be integrated into a network coordinate system which predicts some "distance" between nodes: both power limitations and reliability of a single node are felt by all other nodes in the network and, consequently, might be expressed by amplifying a node's distance to each node in the network.

SimplestVivaldi(y, l_{xy}):
  1. $e = l_{xy} - d_V(x^c, y^c)$
  //computes the error of the RTT measurement l_{xy}
  2. $x^c = x^c + \delta * e * u(x^c - y^c)$
  //adds an error in the direction of the force vector

Figure 1: Pseudo code for decentralized coordinate adjustment. Node x has measured the RTT l_{xy} to y and updates its coordinates. $u(x^c - y^c)$ is the unit vector in the direction of the force between x^c and y^c and the parameter δ is used for adjustments.

Considering this behavior, Vivaldi network coordinates [2] appear to be an apposite starting point. $d+1$-dimensional Vivaldi coordinates are composed of a d-dimensional Euclidean vector with an additional "height" dimension. For two nodes x and y with coordinates $x^c = (x_1, x_2, \ldots, x_{d+1})$ and $y^c = (y_1, y_2, \ldots, y_{d+1})$, the Vivaldi-distance $d_V(x^c, y^c)$ between x and y's network coordinates is:

$$d_V(x^c, y^c) = \sqrt{(x_1 - y_1)^2 + \ldots + (x_d - y_d)^2} + x_{d+1} + y_{d+1}. \qquad (1)$$

As Vivaldi was constructed to approximate round trip times between nodes, the additional hight dimension represents the expensive last hop in routing incurred by the access-link within an ISP's network that separates a node from the highly connected internet core. However, this height dimension could prove instrumental for the integration of additional node information in node selection.

Vivaldi is a decentralized, lightweight, and adaptive simulation of a network of physical springs, with each node piggybacking its coordinates on network messages and measuring round trip times to neighbors. Imagine that every two nodes are connected by a spring with a length equal to the RTT between the nodes: The springs push and pull the nodes in the coordinate space until the system settles, at which point the nodes have been *spring embedded*. A very simple algorithm for Vivaldi coordinates can be found in Figure 1. Of course, RTT between nodes are dynamic and a single measurement rarely provides an adequate estimate of the average distance, so the network coordinates are constantly changing. The accuracy of Vivaldi network coordinates is disputed (see for example [9],[21]), but with modifications that stabilize the coordinates and dampen the effects of nodes with unreliable coordinates, there are strong indicators that they cope well with a fair amount of network dynamics [9]. Due to findings of Ledlie et al. [9] on the descriptive accuracy of Vivaldi coordinates using various dimension configurations, 3-dimensional Vivaldi coordinates (i.e., 2-dimensional Euclidean coordinates plus one height dimension) are used in the following considerations.

4. SMALL WORLD NETWORKS - SYMPHONY

Academic interest in small-world networks dates back to a series of studies by Stanly Milgram in the 1960's [14] in which letters were forwarded between randomly selected, geographically separated people with an average number of five to six steps. Milgram's findings let to the idea of "six degrees of separation" - and with it the small-world network. Following ideas from Watts and Strogatz [24], small-world networks have come to be characterized by high clustering, low average path lengths, and a combination of randomness and structure. (Note that the term small-world network has various meanings; in this paper, it refers to a network that emulates all of these characteristics of social networks.) Clustering reflects cliquishness, or the natural tendency for one person's acquaintances to be acquainted themselves. Watts and Strogatz used a clustering coefficient - the average over all nodes of the portion of existing edges to possible edges between each node's neighbors - and a characteristic path length - the average over all pairs of nodes of the length of the shortest path joining each pair of nodes - to analyze different classes of networks.

A completely random network, as introduced by Erdős and Rényi in 1959 [5], is a network in which each pair of nodes is joined by and edge with independent probability p. Such graphs have low clustering coefficients and low average path lengths, meaning that they are highly connected but lack the wanted structure of small-world networks. On the other hand, regular ring lattice networks, in which nodes are arranged in a ring and then connected to their k nearest neighbors, demonstrate high clustering coefficients and high average path lengths, so they have ample structure but lack the connectedness of small-world networks [24]. Watts and Strogatz suggested a network model based on an undirected regular ring lattice to which randomness has been "injected" - for each node, each of its edges is rewired to some other random node with probability p, forbidding duplicate edges and multiple rewiring of edges. This gives us the regular ring lattice for $p = 0$, a network similar to the Erdős-Rényi model for $p = 1$, and networks which display high clustering and low average paths for a broad spectrum of p values - i.e., the Watts and Strogatz's small-world model.

Following Watts and Strogatz's findings, Kleinberg posed two engaging questions: "Why should there *exist* short chains of acquaintances linking together arbitrary pairs of strangers?" and "Why should arbitrary pairs of strangers be able to *find* short chains of acquaintances that link them together?" [8] Kleinberg introduces a similar small-world model based on a two-dimensional lattice of nodes. With parameters k, q, and r, each node x is joined to every node y within lattice distance k ($d(x, y) \leq k$) and to another q randomly chosen *long distance* contacts. Every potential long distance node y's probability of becoming one of x's long distance contacts is proportional to $d(x, y)^{-r}$. In short, Kleinberg found that the network could route most efficiently for $r = 2$ with an expected delivery time of $O(\log^2(n))$ and conjectured that the success of distributed routing algorithms depends on nodes' ability to interpret structural cues.

Kleinberg's work inspired the design of Manku et al.'s DHT Symphony [12]. In a Symphony network with n nodes, each node randomly chooses a node identifier from the unit interval $[0, 1)$ and positions itself on a ring. Each node establishes links with its two immediate neighbors on the ring as well as q *long distance links* chosen with the help of a probability density function (pdf) $p_n(x)$ for $x \in [1/n, 1]$ the distances to nodes:

$$p_n(x) = \frac{1}{x \ln(x)}.$$

In order to establish a long link, a node draws a distance d from $p_n(x)$ and then searches for the manager of the point at distance d away from itself on the ring. Although Kleinberg attributed the success of his routing algorithm for $r = 2$ in part to the fact that the distances of the long distance contacts were approximately uniformly distributed, Symphony's $p_n(x)$ implies that the probability density of a long distance link being established at a distance d is *inversely proportional* to d. Nonetheless, Manku et al. showed that the expected path length of unidirectional routing on Symphony with $q = O(1)$ long links is $O(\frac{1}{q} \log^2(n))$.

5. RESEARCH IDEAS - DHT CONSTRUCTION

The following ideas for DHT construction stem from the apparent disparities between the works of Watts and Strogatz, Kleinberg, and Manku et al. and actual DHT implementations. If we consider a network as a set of nodes with geographical locations, then a small-world network which aims at conserving network resources would be built on top of this geography. But current DHTs built on this geography either neglect scalability and robustness by deterministically choosing node IDs based on location or fail to achieve a small-world network by strictly favoring geographically close nodes. Although integrating network coordinates into a small-world DHT is far from a novel idea, using network coordinates to express more than mere latency and forming small-world overlay links within the network coordinate space as opposed to the key identifier space seem to be unexplored areas. By circumventing deterministically chosen node identifiers and building a small-world network in the network coordinate space, such a model could better retain desirable small-world graph properties such as robustness and low average path length.

5.1 Initialization

Assume that we have a network in which each node x possesses a universal measure of node quality μ_x in some interval $[-a, b]$, where large values of μ_x signify poor node quality and low values high node quality. Since links cannot be established within the network coordinate space until nodes have found suitable network coordinates, we initiate the network with the small-world DHT Symphony [12] with q long distance links. Node identifiers are chosen in this phase and later, similar to other ring shaped DHTs, uniformly at random from the real numbers on the interval $[0, 1)$ that wraps around. Nodes position themselves on the ring of keys where key values increase in a clockwise direction and thereafter manage the subinterval of keys that corresponds to this identifier as defined in Symphony. Each node is linked to its k nearest neighbors on the ring of node identifiers - let us call these links a node's *short links* and all other links its *long links*.

With an initial DHT in place, a 3-dimensional Vivaldi coordinate system is initialized and nodes' coordinates are piggybacked on network messages. Once the network coordinates are stable enough, each node x first integrates its node quality parameter μ_x into its network coordinates x^c and then begins to replace its original long links with long links found within the network coordinate space. Two sets of network coordinates are used for each node x: one set x^c which is used to establish x's "raw" coordinates, and another "adjusted" set x^a which incorporates node quality and is used to estimate the Vivaldi distance between x and other nodes when choosing and maintaining neighbors. The adjusted coordinates are obtained by replacing the node's height component x_h with an amplified height component $x_h^a := max\{0, x_h + \mu_x\}$, resulting in the new coordinates $x^a := (x_1, x_2, x_h^a)$. With this adjustment, x will either increase or decrease its Vivaldi distance to the entire network, so that every node will chose it with either a lower or higher probability, respectively (see the search algorithm below). Nodes decide whether to use two sets of 3-dimensional coordinates x^c and x^a or one 4-dimensional set (x_1, x_2, x_h, μ_x) from which every network node can determine x^c and x^a. Vivaldi distance is measured using the equation for $d_V(x^c, y^c)$ in (1).

5.2 New long links

In order to generate a small-world-like network, long links are established based on a probability density function over the distances between nodes in the network coordinate space. We choose Manku et al.' approach: The smaller the distance between two nodes' network coordinates, the higher the probability that they are joined by a link. Each node finds new long links by drawing *distances* from a pdf and then searching for a node at approximately this distance in the network coordinate space: A node x that draws a distance d from its pdf sends a search message to a select number of neighbors with its adjusted coordinates x^a and the distance d. These neighbors check their routing tables for any node(s) y with

$$d_V(x^a, y^a) \in (d - \varepsilon, d + \varepsilon)$$

for some tolerance level $\varepsilon > 0$, returning any hits. Each node then forwards the search message to a neighbor v from its set of neighbors N with

$$d_V(x^a, v^a) = \min_{\{z \in N)\}} \{d_V(x^a, z^a) : d_V(x^a, z^a) \notin (d - \varepsilon, d + \varepsilon)\}.$$

After a set number of iterations, x chooses between any returned hits. If no hits are returned, then a new distance is drawn and the process begins again. Once a new long link has been found and established, x randomly selects a long link from the initial Symphony construction to be replaced.

A node x's probability density function over the distances depends on the radius of the node space as measured from x's position (i.e., with a space W: $\max_{y \in W} d_W(x, y)$) as well as the total number of nodes. This is fairly straightforward in the node identifier space where the diameter of the space (the number of possible IDs) is known and approximately twice the radius as measured from any node, but the radius of the network coordinate space can only be estimated and will be drastically different for different nodes. Nonetheless, each node x can maintain an approximation r_x of the distance to the furthest node in the network coordinate space as well as an estimation n_x of the number of nodes in the network, which are then used to generate a node-specific probability density function. Analogous to the harmonic distribution used in Symphony [12], we want a pdf for the distance d drawn by node x with $x^a = (x_1, x_2, x_h^a)$ which is defined on the interval $[\frac{1}{n_x} + x_h^a, r_x]$ and is inversely proportional to d:

$$\begin{aligned} p_x(d) &= \frac{1/d}{\int_{(1/n_x + x_h^a)}^{r_x} \frac{1}{d} \mathbf{d}d} \\ &= \frac{1}{d * \ln\left(\frac{n_x r_x}{1 + n_x x_h^a}\right)}. \end{aligned}$$

Note that the integral over the interval $[\frac{1}{n_x} + x_h^a, r_x]$ is one.

5.3 Estimating network radius and size

The size of the network N_x as observed by node x is estimated with a standard estimation protocol for nodes placed uniformly on a ring. By adding the size of the sections of the ring managed by s different nodes and then dividing by s, we obtain an unbiased estimation of n [10],[12].

Unfortunately, the estimation of r_x is more difficult. A preliminary idea is to piggyback the network coordinates of the source node on each network message until that message reaches its destination. These network coordinates are read by each node on the message's path. Each node routinely calculates its distance to the source nodes of messages passing though and stores the longest s distances with a timestamp in a radius table. Once the timestamp of a distance in the radius table is determined aged, that distance is deleted. The largest value in the radius table is used to approximate r_x.

5.4 Routing protocol

Routing for lookups is performed identical to unidirectional routing in Symphony [12]. A node which needs to lookup a hash key h in $[0, 1)$ forwards the lookup to the node x in its routing table that minimizes the clockwise distance between h and x in the node identifier space.

5.5 System dynamics

Node joins and leaves. Node failures are treated just as node failures in other ring-shaped DHTs. To join the network, a node x must know one current member of the network. x choses a node ID uniformly at random from $[0, 1)$ and then looks up its current manager y and establishes links with y and its nearest k neighbors on the ring. Since x must first find suitable Vivaldi coordinates, it initializes its long distance links as in Symphony and then replaces them one by one as described in the network initialization process.

Network coordinate changes. Vivaldi network coordinates continue to adjust to network changes, for example when nodes' positions change or nodes join or leave the network. Since links are chosen based on their Vivaldi distances, it makes sense to rewire a link once its distance has changed significantly. By periodically comparing a link's original distance to its current distance, a node can determine when individual links should be replaced.

5.6 Network coordinate vs. node ID space

In the node ID space, each node is joined to its k nearest neighbors, creating a "backbone" for the network. A node x's long links $y_1, y_2, \ldots, y_q$, which are chosen in the network coordinate space, have a strong bias towards nodes with smaller Vivaldi distances. However, every y_i has a node identifier chosen independently and uniformly at random from the interval $[0, 1)$, and thus, the long links are distributed uniformly at random in the node identifier space. Viewed in the network coordinate space, the network consists of a random Hamiltonian cycle and long links connecting nodes at a distance d with probability density inversely proportional to d, thus resembling the small-world network as proposed by Manku et. al. But viewed in the node ID space, nodes are connected to their closest k neighbors as well as an additional q long links chosen uniformly at random, thus resembling the small-world networks of Watts and Strogatz [24] and Kleinberg [8].

6. ANALYSIS AND RESEARCH PROBLEMS

The proposed DHT protocol avoids deterministically choosing node identifiers and should therefore experience better robustness and scalability than other DHTs which are embedded in a network coordinate space, however, it is unclear how the two "layers" of small-world networks will cojoin and effect resource consumption (in particular routing efficiency). In order to compare the efficiency of this DHT protocol with other proximity-aware and proximity-unaware DHTs, measures for scalability, robustness, round trip times, and resource conservation must first be established. Any such measure must invariably depend on the underlying distribution of the nodes in the network coordinate space, since each distribution directly effects the DHT's construction and characteristics. Perhaps the simplest suitable network model is a 3-dimensional grid: Let the nodes be arranged on a 3-dimensional grid i nodes wide, j nodes deep, and k nodes high with $n := i * j * k$ nodes. The node in row x, column y, and at height z is called $u_{x,y,z}$ and the distance between two nodes (i.e. the Vivaldi distance) is $d_V(u_{x,y,x}, u_{\tilde{x},\tilde{y},\tilde{z}}) := \sqrt{(x-\tilde{x})^2 + (y-\tilde{y})^2} + z + \tilde{z}$.

But what exactly are we interested in for each of the given criteria and how do we quantify them? These are open questions and will vary from application to application, but some feasible measures follow:

Scalability. Load balancing is a key issue in scalability, and one possible measure is the variance of the nodes' loads shortly after a fraction n/m of the current nodes are added uniformly at random to a randomly chosen smaller portion of the grid (which must provide finer granularity).

Robustness. The ability of a network to cope with node failures is considered a characteristic of robustness. Since failures can be characterizes as random failures or attack failures, both should be examined. The first could be quantified by the expected number of routing hops between any two nodes after m independent uniformly distributed node failures, the second by the expected number of routing hops between any two nodes after a subset of the grid with dimensions i'xj'xk' chosen uniformly at random fails.

Round trip times. The expected Vivaldi distance traversed when routing between two nodes in the network is a logical measure.

Network resources. With an extended model where each node x chooses a node reliability parameter μ_x from $[-a, b]$ uniformly at random, this measure should quantify the use of nodes with respect to their resources. The expected sum of the resource levels used in a single route could be a useful measure.

Future work will involve modeling and simulating the network in order to collect information from these measures. Theoretical comparisons of various overlays with respect to these measures could prove difficult, but preliminary work to obtain an analytical framework on which such comparisons could be built must be a concurrent part of this work. Although we are directly interested in the above criteria, their behavior seems to be strongly linked to the amount and kind of randomness and structure within a given network. Examining the criteria together with a suitable measure for network randomness or structure could therefore prove insightful.

Beyond these measures, there are many open questions regarding the suggested DHT. The above protocol contains several parameters for which we have no current suggestions or comparisons: the number of short links in the DHT, the tolerance ε in the search algorithm, the number of nodes to initially forward a search message to, the number of iterations to perform a search, the number of distances in a radius table and their lifespan, etc. Furthermore, the suggested pdf is a logical extension of the pdf from Symphony, but other probability density functions proportional to $\frac{1}{d^2}, \frac{1}{d^3}, \ldots$ could be better suited. Although we proposed a single node reliability parameter μ, multiple parameters might be more practical for systems that value characteristics differently. By storing each of the parameters in its own dimension, nodes could then choose which characteristics to integrate into their individual distance calculations.

These initial ideas are based on small-world networks, although much about small-world networks themselves is not yet understood. Small world networks heuristically reflect our understanding of social networks, thus combining short average paths with high clustering, but the goal in computer networks is to establish fast, efficient, and reliable networks and routing. We are unaware of any work that vets disparate classes of networks in order to learn more about what makes a network capable (or incapable) of good routing. Such work could eventually find an optimal class of graphs with a routing algorithm that minimizes the average routing path length.

7. REFERENCES

[1] M. Costa, M. Castro, A. Rowstron, and P. Key. Pic: Practical internet coordinates for distance estimation. pages 178–187, 2003.

[2] F. Dabek, R. Cox, F. Kaashoek, and R. Morris. Vivaldi: a decentralized network coordinate system. pages 15–26, 2004.

[3] F. Dabek, J. Li, E. Sit, J. Robertson, M. F. Kaashoek, and R. Morris. Designing a dht for low latency and high throughput. In *IN PROCEEDINGS OF THE 1ST NSDI*, pages 85–98, 2004.

[4] M. El Dick, E. Pacitti, and B. Kemme. Flower-cdn: a hybrid p2p overlay for efficient query processing in cdn. In *EDBT '09: Proceedings of the 12th International Conference on Extending Database Technology*, pages 427–438, New York, NY, USA, 2009. ACM.

[5] P. Erdős and A. Rényi. On random graphs. I. *Publ. Math. Debrecen*, 6:290–297, 1959.

[6] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang. Idmaps: a global internet host distance estimation service. *IEEE/ACM Trans. Netw.*, 9(5):525–540, 2001.

[7] K. Gummadi, R. Gummadi, S. Gribble, S. Ratnasamy, S. Shenker, and I. Stoica. The impact of dht routing geometry on resilience and proximity. In *SIGCOMM*

'03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, pages 381–394, New York, NY, USA, 2003. ACM.

[8] J. Kleinberg. The small-world phenomenon: an algorithm perspective. In *STOC '00: Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 163–170, New York, NY, USA, 2000. ACM.

[9] J. Ledlie, P. Gardner, and M. Seltzer. Network coordinates in the wild. In *In Proceeding of USENIX NSDIŠ07*, 2007.

[10] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: A scalable and dynamic emulation of the butterfly. In *Proceedings of the 21st annual ACM symposium on Principles of distributed computing.* ACM Press, 2002.

[11] B. Maniymaran, M. Bertier, and A.-M. Kermarrec. Build one, get one free: Leveraging the coexistence of multiple p2p overlay networks. In *Distributed Computing Systems, 2007. ICDCS '07. 27th International Conference on*, pages 33–33, June 2007.

[12] G. S. Manku, M. Bawa, P. Raghavan, and V. Inc. Symphony: Distributed hashing in a small world. In *In Proceedings of the 4th USENIX Symposium on Internet Technologies and Systems*, pages 127–140, 2003.

[13] P. Maymounkov and D. Mazieres. Kademlia: A Peer-to-peer Information System Based on the XOR Metric. In *IPTPS*, 2002.

[14] S. Milgram. The small world problem. *Psychology Today*, 2:60–67, 1967.

[15] T. S. E. Ng and H. Zhang. Predicting internet network distance with coordinates-based approaches. In *In INFOCOM*, pages 170–179, 2001.

[16] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica. Geographic routing without location information. In *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 96–108, New York, NY, USA, 2003. ACM.

[17] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content addressable network. In *SIGCOMM'01*, 2001.

[18] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. Topologically-aware overlay construction and server selection. 2002.

[19] S. Ren, L. Guo, S. Jiang, and X. Zhang. Sat-match: a self-adaptive topology matching method to achieve low lookup latency in structured p2p overlay networks. In *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*, pages 83–, April 2004.

[20] Y. Shavitt and T. Tankel. On the curvature of the internet and its usage for overlay construction and distance estimation. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, pages –384, March 2004.

[21] M. Steiner and E. W. Biersack. Where is my peer? evaluation of the vivaldi network coordinate system in azureus. In *NETWORKING '09: Proceedings of the 8th International IFIP-TC 6 Networking Conference*, pages 145–156, Berlin, Heidelberg, 2009. Springer-Verlag.

[22] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *SIGCOMM'01*, pages 149–160, 2001.

[23] M. Waldvogel and R. Rinaldi. Efficient topology-aware overlay network. *SIGCOMM Comput. Commun. Rev.*, 33(1):101–106, 2003.

[24] D. J. Watts and S. H. Strogatz. Collective dynamics of'small-world'networks. *Nature*, 393(6684):409–10, 1998.