On the Cost of Participating in a Peer-to-Peer Network *

Nicolas Christin and John Chuang

School of Information Management and Systems, University of California, Berkeley

{christin,chuang}@sims.berkeley.edu

Abstract

In this paper, we model the cost incurred by each peer participating in a peer-to-peer network. Such a cost model allows to gauge potential disincentives for peers to collaborate, and provides a measure of the "total cost" of a network, which is a possible benchmark to distinguish between proposals. We characterize the cost imposed on a node as a function of the experienced load and the node connectivity, and express benefits in terms of cost reduction. We discuss the notion of social optimum with respect to the proposed cost model, and show how our model applies to a few proposed routing geometries for distributed hash tables (DHTs). We further outline a number of open questions this research has raised.

1 Introduction

A key factor in the efficiency of a peer-to-peer overlay network is the level of collaboration provided by each peer. In this paper, we take a first step towards quantifying the level of collaboration that can be expected from each node participating in an overlay, by proposing a model to evaluate the cost each peer incurs as a member of the overlay. We express the benefits of participating in the overlay in terms of a cost reduction.

Such a cost model has several useful applications, among which, (1) providing a benchmark that can be used to compare between different proposals, complementary to recent works comparing topological properties of various overlays [8, 12], (2) allowing for predicting disincentives, and designing mechanisms that ensure a protocol is *strategyproof* [15], and (3) facilitating the design of load balancing primitives.

This work is not the first attempt to characterize the cost of participating in a network. Jackson and Wolinsky [9] proposed cost models to analyze formation strategies in social and economic networks. More recent studies [5, 7] model network formation as a non-cooperative game, where nodes have an incentive to participate in the network, but want to minimize the price they pay for doing so. These studies assume that each node has the freedom to choose which links it maintains, whereas we assume that the overlay topology is constrained by a protocol. Moreover, our approach extends previously proposed cost models [5, 7, 9], by considering the load imposed on each node in addition to the distance to other nodes and degree of connectivity.

In the remainder of this paper, we first introduce our proposed cost model, before discussing the notion of "social optimum," that is, the geometry that minimizes the sum of all costs over the entire network. We then apply the cost model to several routing geometries used in recently proposed distributed hash table (DHT) algorithms [10, 12, 17, 18, 19], and compare the costs incurred by each geometry. We conclude by discussing some open problems this research has uncovered.

2 Proposed cost model

The model we propose applies to any peer-to-peer network where nodes request and serve items, or serve requests between other nodes. This includes peer-to-peer file-sharing systems [1], ad-hoc networks [6], distributed lookup services [17, 19], or application-layer multicast overlays [2, 4, 11], to name a few examples. Formally, we define an overlay network by a quadruplet (V, E, K, F), where V is the set of nodes in the network, E is the set of directed edges, K is the set of items in the network, and $F : K \rightarrow V$ is the function that assigns items to nodes. Each node $u \in V$ is assigned a unique identifier (integer or string of symbols), which, for the sake of simplicity, we will also denote by u. We define by $K_u = \{k \in K : F(k) = u\}$ the set of items stored at node $u \in V$. We have $K = \bigcup_u K_u$, and we assume, without loss of generality, that the sets K_u are disjoint.[1] We characterize each request with two independent random

*This work is supported in part by the National Science Foundation under grant number ANI-0085879.

[1] If an item is stored on several nodes (replication), the replicas can be viewed as different items with the exact same probability of being requested.

variables, $X \in V$ and $Y \in K$, which denote the node X making the request, and the item Y being requested, respectively.

Consider a given node $u \in V$. Every time an item $k \in K$ is requested in the entire network, node u is in one of four situations:

1. Node u does not hold or request k, and is not on the routing path of the request. Node u is not subject to any cost.

2. Node u requests item k. In our model, we express the benefits of participating in a peer-to-peer network in terms of latency reduction, similar to related proposals, e.g., [7]. In particular, we assume that the farther the node v holding k is from u (in a topological sense), the costlier the request is. If there is no path between nodes u and v, the request cannot be carried out, which yields an infinite cost. More precisely, we model the cost incurred by node u for requesting k as $l_{u,k}t_{u,v}$, where $t_{u,v}$ is the number of hops between nodes u and v, and $l_{u,k}$ is a (positive) proportional factor. We define the *latency cost* experienced by node u, L_u, as the sum of the individual costs $l_{u,k}t_{u,v}$ multiplied by the probability $k \in K_v$ is requested, that is

$$L_u = \sum_{v \in V} \sum_{k \in K_v} l_{u,k} t_{u,v} \Pr[Y = k] \ , \qquad (1)$$

with $t_{u,v} = \infty$ if there is no path from node u to node v, and $t_{u,u} = 0$ for any u. With this definition, to avoid infinite costs, each node has an incentive to create links such that all other nodes holding items of interest can be reached. An alternative is to store or cache locally all items of interest so that the cost of all requests reduces to $l_{u,k}t_{u,u} = 0$.

3. Node u holds item k, and pays a price $s_{u,k}$ for serving the request. For instance, in a filesharing system, the node uses some of its upload capacity to serve the file. We define the *service cost* S_u incurred by u, as the expected value of $s_{u,k}$ over all possible requests. That is,

$$S_u = \sum_{k \in K_u} s_{u,k} \Pr[Y = k] \ .$$

4. Node u does not hold or request k, but has to forward the request for k, thereby paying a price $r_{u,k}$. The overall *routing cost* R_u suffered by node u is the average over all possible items k, of the values of $r_{u,k}$ such that u is on the path of the request. That is, for $(u, v, w) \in V^3$, we consider the binary function

$$\chi_{v,w}(u) = \begin{cases} 1 & \text{if } u \text{ is on the path from } v \text{ to } w, \\ & \text{excluding } v \text{ and } w \\ 0 & \text{otherwise,} \end{cases}$$

and express R_u as

$$R_u = \sum_{v \in V} \sum_{w \in V} \sum_{k \in K_w} r_{u,k} \Pr[X = v] \Pr[Y = k] \chi_{v,w}(u) \ . \qquad (2)$$

In addition, each node keeps some state information so that the protocol governing the overlay operates correctly. In most overlay protocols, each node u has to maintain a neighborhood table and to exchange messages with all of its neighbors. The number of neighbors corresponds to the out-degree $\deg(u)$ of the node, resulting in a *maintenance cost* M_u that is characterized by

$$M_u = m_u \deg(u) \ ,$$

where $m_u \geq 0$ denotes the cost associated with maintaining a link with a given neighbor.

Last, we define the *total cost* C_u imposed on node u as

$$C_u = L_u + S_u + R_u + M_u \ .$$

We can use C_u to compute the total cost of the network, $C = \sum_{u \in V} C_u$. Note that the expression of C_u only makes sense if S_u, R_u, M_u , and L_u are all expressed using the same unit. Thus, the coefficients $s_{u,k}$, $r_{u,k}$, $m_{u,k}$ and l_u have to be selected appropriately. For instance, $l_{u,k}$ is given in monetary units per hop per item, while m_u is expressed in monetary units per neighbor entry.

3 Social optimum

The first question we attempt to address is whether we can find a social optimum for the cost model we just proposed, that is, a routing geometry that minimizes the total cost C. We define a routing geometry as in [8], that is, as a collection of edges, or topology, associated with a route selection mechanism. Unless otherwise noted, we assume shortest path routing, and distinguish between different topologies. We discuss a few simplifications useful to facilitate our analysis, before characterizing some possible social optima.

Assumptions For the remainder of this paper, we consider a network of $N > 0$ nodes, where, for all $u \in V$ and $k \in K$, $l_{u,k} = l$, $s_{u,k} = s$, $r_{u,k} = r$, and $m_u = m$.[2] We suppose that the network is in a steady-state regime, i.e., nodes do not join or leave the network, so that the values l, s, r and m are constants. We also suppose that

[2]While very crude in general, this simplification is relatively accurate in the case of a network of homogeneous nodes and homogeneous links containing fixed-sized keys such as used in DHTs.

requests are uniformly distributed over the set of nodes, that is, for any node u, $\Pr[X = u] = 1/N$. We make a further simplification by choosing the mapping function F such that all nodes have an equal probability of serving a request. In other words, $\sum_{k \in K_u} \Pr[Y = k] = 1/N$, which implies $S_u = s/N$ regardless of the geometry used. Moreover, if we use $E[x]$ to denote the *expected value* of a variable x, Eqs. (1) and (2) reduce to $L_u = lE[t_{u,v}]$ and $R_u = rE[\chi_{v,w}(u)]$, respectively. Last, we assume that no node is acting maliciously.

Full mesh Consider a full mesh, that is, a network where any pair of nodes is connected by a (bidirectional) edge, i.e., $t_{u,v} = 1$ for any $v \neq u$. Nodes never any route any traffic and $\deg(u) = N - 1$. Thus, for all u, $R_u = 0$, $L_u = l(N-1)/N$, and $M_u = m(N-1)$. With $S_u = 1/N$, we get $C_u = 1/N + l(N-1)/N + m(N-1)$, and, summing over u,

$$C = 1 + l(N-1) + mN(N-1) \ .$$

Let us remove a link from the full mesh, for instance the link $0 \rightarrow 1$. Because node 0 removes an entry from its neighborhood table, its maintenance cost M_0 decreases by m. However, to access the items held at node 1, node 0 now has to send traffic through another node (e.g., node 2): as a result, L_0 increases by l/N, and the routing cost at node 2, R_2, increases by r/N^2. So, removing the link $0 \rightarrow 1$ causes a change in the total cost $\Delta C = -m + l/N + r/N^2$. If $\Delta C \geq 0$, removing a link causes an increase of the total cost, and the full mesh is the social optimum. In particular, the full mesh is the social optimum if the maintenance cost is "small enough," that is, if

$$m \leq l/N + r/N^2 \ . \tag{3}$$

Note that, as $N \rightarrow \infty$, the condition (3) tends to $m = 0$. In fact, we can also express $\Delta C \geq 0$ as a condition on N that reduces to $N \leq \lfloor l/m + r/l \rfloor$ when $m \ll l^2/r$, using a first-order Taylor series expansion.

Star network Suppose now that Eq. (3) does not hold, and consider a star network. Let $u = 0$ denote the center of the star, which routes all traffic between peripheral nodes. That is, $\chi_{v,w}(0) = 1$ for any $v \neq w$ $(v, w > 0)$. One can show [3] that $R_0 = r(N-1)(N-2)/N^2$, $L_0 = l(N-1)/N$ and $M_0 = m(N-1)$, so that the cost C_0 incurred by the center of the star is

$$C_0 = m(N-1) + \frac{s}{N} + \frac{l(N-1)}{N} + \frac{r(N-1)(N-2)}{N^2} \ . \tag{4}$$

Peripheral nodes do not route any traffic, i.e., $R_u = 0$ for all $u > 0$, and are located at a distance of one from the center of the star, and at a distance of two from the $(N-2)$ other nodes, giving $L_u = l(2N-3)/N$. Further, $\deg(u) = 1$ for all peripheral nodes. Hence, $M_u = m$, and the total cost imposed on nodes $u > 0$ is

$$C_u = m + \frac{s + l(2N-3)}{N} \ . \tag{5}$$

A proof by identification [3] indicates that $C_0 = C_u$ can only hold when N is a constant, or when $l = r = m = 0$. The difference $C_0 - C_u$ quantifies the (dis)incentive to be in the center of the star.

Summing Eqs. (4) and (5), we obtain $C = 2m(N-1) + s + 2l(N-1)^2/N + r(N-1)(N-2)/N^2$. On the one hand, removing any (directed) link from the star either causes a node to be unreachable or prevents a node from contacting any of the other nodes. In either case, $C \rightarrow \infty$. On the other hand, adding a link to the star also causes the cost C to increase, when Eq. (3) does not hold. For instance, consider, without loss of generality, adding the link $1 \rightarrow 2$: M_1 increases by m, L_1 decreases by l/N (the items held at node 2 can now be reached in one hop), and R_0 decreases by r/N^2 (traffic from 1 to 2 is not routed through 0 anymore). All other costs are unchanged. Hence, the change in the cost C is $\Delta C = m - l/N - r/N^2$, which is positive if Eq. (3) does not hold. Therefore, adding or removing a link to a star when Eq. (3) is not satisfied cannot lead to a social optimum.

From the above study, when Eq. (3) holds (i.e., N or m is small), the social optimum is the full mesh. When Eq. (3) does not hold, repeatedly removing links from the full mesh decreases the cost C until a star topology is reached. Thus, a centralized topology seems to be desirable when N and/or m are significant, while the objective is to minimize the total amount of resources used in the whole network to maintain the overlay. However, we stress that we do not consider robustness against attack, fault-tolerance, or potential performance bottlenecks, all being factors that pose practical challenges in a centralized approach, nor do we offer a mechanism creating an incentive to be in the center of the star. Furthermore, determining under which conditions on l, s, r, m and N the star is the social optimum is an open problem.

4 Case studies

We next apply the proposed cost model to a few selected routing geometries and compare the results with those obtained in our study of the social optimum. We present the various costs experienced by a node in each geometry, before illustrating the results with numerical examples.

4.1 Analysis

Due to space limitations, we omit here most of the details in the derivations, and instead refer the reader to a companion technical report [3] for complete details.

De Bruijn graphs De Bruijn graphs are used in algorithms such as Koorde [10] and ODRI [12], and present very desirable properties, such as short average routing distance and high resiliency to node failures [12]. In a de Bruijn graph, any node u is represented by an identifier string $(u_1, \ldots, u_D)$ of D symbols taken from an alphabet of size Δ. The node represented by $(u_1, \ldots, u_D)$ links to each node represented by $(u_2, \ldots, u_D, x)$ for all possible values of x in the alphabet. The resulting directed graph has a fixed out-degree Δ, and a diameter D.

The maintenance, routing, and latency costs experienced by each node in a De Bruijn graph all depend on the position of the node in the graph [3]. Denote by V' the set of nodes such that the identifier of each node in V' is of the form $(h, h, \ldots, h)$. Nodes in V' link to themselves, and the maintenance cost is $M_u = m(\Delta - 1)$ for $u \in V'$. For nodes $u \notin V'$, we have $M_u = m\Delta$.

For any node $u \in V$, the routing cost R_u is such that $0 \leq R_u \leq r\rho_{\max}/N^2$, where $\rho_{\max}$ denotes the maximum number of routes passing through a given node, or maximum *node loading*, with (see [3]):

$$\rho_{\max} = \frac{(D-1)(\Delta^{D+2} - (\Delta-1)^2) - D\Delta^{D+1} + \Delta^2}{(\Delta-1)^2} .$$

One can show by contradiction that with shortest-path routing, nodes $u \in V'$ do not route any traffic, so that the lower bound $R_u = 0$ is reached for $u \in V'$. One can also show that $R_u = r\rho_{\max}/N^2$ when $\Delta \geq D$ for the node $(0, 1, 2, \ldots, D-1)$.

We further prove in [3] that the latency cost is bounded by $L_{\min} \leq L_u \leq L_{\max}$ where

$$L_{\min} = \frac{l}{N}\left(D\Delta^D + \frac{D}{\Delta - 1} - \frac{\Delta(\Delta^D - 1)}{(\Delta-1)^2}\right) ,$$

and

$$L_{\max} = l\frac{D\Delta^{D+1} - (D+1)\Delta^D + 1}{N(\Delta - 1)} .$$

We have $L_u = L_{\max}$ for nodes $u \in V'$, and $L_u = L_{\min}$ for the node $(0, 1, \ldots, D-1)$ when $\Delta \geq D$. Note that we can simplify the expressions for both $L_{\min}$ and $L_{\max}$ when $N = \Delta^D$, that is, when the identifier space is fully populated.

D-dimensional tori We next consider D-dimensional tori, as in CAN [17], where each node is represented by D Cartesian coordinates, and has $2D$ neighbors, for a maintenance cost of $M_u = 2mD$ for any u.

Routing at each node is implemented by greedy forwarding to the neighbor with the shortest Euclidean distance to the destination. We assume here that each node is in charge of an equal portion of the D-dimensional space. From [17], we know that the average length of a routing path is $(D/4)N^{1/D}$ hops.[3] Because we assume that the D-dimensional torus is equally partitioned, by symmetry, we conclude that for all u,

$$L_u = l\frac{DN^{1/D}}{4} .$$

To determine the routing cost R_u, we compute the node loading as a function $\rho_{u,D}$ of the dimension D. With our assumption that the D-torus is equally partitioned, $\rho_{u,D}$ is the same for all u by symmetry. Using the observation that the coordinates of two consecutive nodes in a path cannot differ in more than one dimension, we can compute $\rho_{u,D}$ by induction on the dimension D [3]:

$$\rho_{u,D} = 1 + N^{\frac{D-1}{D}}\Bigg(-N^{\frac{1}{D}} + D\Bigg(N^{\frac{1}{D}} - 1 + \left(\left\lfloor \frac{N^{\frac{1}{D}}}{2}\right\rfloor - 1\right)\left(\left\lceil \frac{N^{\frac{1}{D}}}{2}\right\rceil - 1\right)\Bigg)\Bigg) .$$

For all u, R_u immediately follows from $\rho_{u,D}$ with

$$R_u = r\frac{\rho_{u,D}}{N^2} .$$

Plaxton trees We next consider the variant of Plaxton trees [16] used in Pastry [18] or Tapestry [20]. Nodes are represented by a string $(u_1, \ldots, u_D)$ of D digits in base Δ. Each node is connected to $D(\Delta - 1)$ distinct neighbors of the form $(u_1, \ldots, u_{p-1}, x, y_{p+1}, \ldots, y_D)$, for $p = 1 \ldots D$, and $x \neq u_p \in \{0, \ldots, \Delta - 1\}$. The resulting maintenance cost is $M_u = mD(\Delta - 1)$.

Among the different possibilities for the remaining coordinates $y_{p+1}, \ldots, y_D$, the protocols generally select a node that is nearby according to a proximity metric. We here assume that the spatial distribution of the nodes is uniform, and that the identifier space is fully populated, which enables us to pick $y_{p+1} = u_{p+1}, \ldots, y_D = u_D$. Thus, two nodes u and v at a distance of k hops differ in k digits, which, as described in [3], leads to

$$\Pr[t_{u,v} = k] = \frac{\binom{D}{k}(\Delta - 1)^k}{N} . \quad (6)$$

[3] Loguinov et al. [12] refined that result by distinguishing between odd and even values of N.

(Δ, D)	$L_{\min}$	$L_{\max}$	$\frac{L_{\max}}{L_{\min}}$	$R'_{\min}$	$R_{\max}$	$\frac{R_{\max}}{R'_{\min}}$
(2, 9)	7.18	8.00	1.11	3.89	17.53	4.51
(3, 6)	5.26	5.50	1.04	2.05	9.05	4.41
(4, 4)	3.56	3.67	1.03	5.11	13.87	2.71
(5, 4)	3.69	3.75	1.02	1.98	5.50	2.78
(6, 3)	2.76	2.80	1.01	5.38	9.99	1.86

Table 1: Asymmetry in costs in a de Bruijn graph ($l = 1, r = 1000$)

Using Eq. (6) in conjunction with the total probability theorem, leads, after simplification, to

$$R_u = r\frac{\Delta^{D-1}(D(\Delta-1)-\Delta)+1}{N^2} \ . \tag{7}$$

Furthermore, from $L_u = lE[t_{u,v}]$, Eq. (6) gives

$$L_u = l\frac{D\Delta^{D-1}(\Delta-1)}{N} \ . \tag{8}$$

Chord rings In a Chord ring [19], nodes are represented using a binary string (i.e., $\Delta = 2$). When the ring is fully populated, each node u is connected to a set of D neighbors, with identifiers $((u + 2^p) \mod 2^D)$ for $p = 0 \ldots D-1$. An analysis similar to that carried out for Plaxton trees yields R_u and L_u as in Eqs. (7) and (8) for $\Delta = 2$. Simulations confirm this result [19].

4.2 Numerical results

We illustrate our analysis with a few numerical results. In Table 1, we consider five de Bruijn graphs with different values for Δ and D, and X and Y i.i.d. uniform random variables. Table 1 shows that while the latency costs of all nodes are comparable, the ratio between $R_{\max}$ and the second best case routing cost,[4] $R'_{\min}$, is in general significant. Thus, if $r \gg l$, there can be an incentive for the nodes with $R_u = R_{\max}$ to defect. For instance, these nodes may leave the network and immediately come back, hoping to be assigned a different identifier $u' \neq u$ with a lower cost. Additional mechanisms, such as enforcing a cost of entry to the network, may be required to prevent such defections.

Next, we provide an illustration by simulation of the costs in the different geometries. We choose $\Delta = 2$, for which the results for Plaxton trees and Chord rings are identical. We choose $D = \{2, 6\}$ for the D-dimensional tori, and $D = \log_\Delta N$ for the other geometries. We point out that selecting a value for D and Δ common to all geometries may inadvertently bias one geometry against another. We emphasize that we only illustrate a specific example here, without making any general comparison between different DHT geometries.

We vary the number of nodes between $N = 10$ and $N = 1000$, and, for each value of N run ten differently seeded simulations, consisting of 100,000 requests each, with X and Y i.i.d. uniform random variables. We plot the latency and routing costs averaged over all nodes and all requests in Fig. 1. The graphs show that our analysis is validated by simulation, and that the star provides a lower average cost than all the other geometries. In other words, whenever practical, a centralized architecture appears more desirable to the community as a whole than a distributed solution. This relatively counter-intuitive result needs to be taken with a grain of salt, however, given the scalability and resiliency concerns linked to a centralized architecture, and the need for incentive mechanisms to compensate for the asymmetry of a star network.

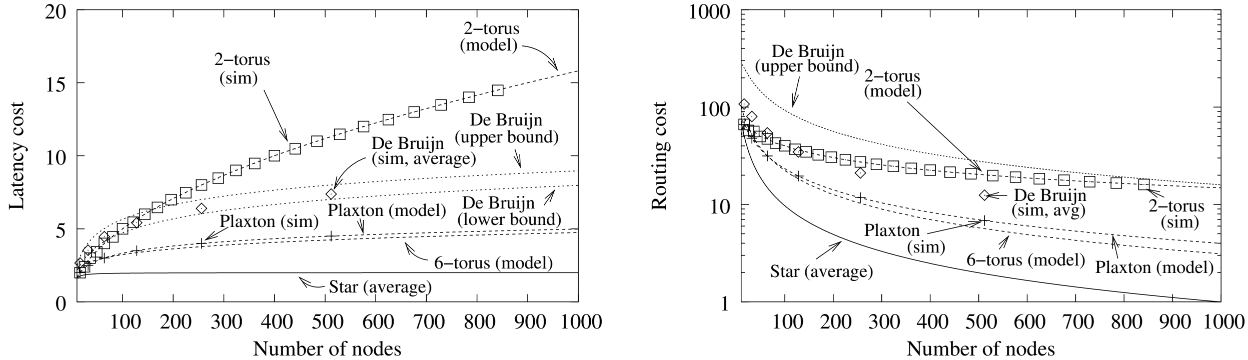
[4] That is, the minimum value for R_u over all nodes but the Δ nodes in V' for which $R_u = 0$.

5 Discussion

We proposed a model, based on experienced load and node connectivity, for the cost incurred by each peer to participate in a peer-to-peer network. We argue such a cost model is a useful complement to topological performance metrics [8, 12], in that it allows to predict disincentives to collaborate (peers refusing to serve requests to reduce their cost), discover possible network instabilities (peers leaving and re-joining in hopes of lowering their cost), identify hot spots (peers with high routing load), and characterize the efficiency of a network as a whole.

We showed that, when the number of nodes is small, fully connected networks are generally the most cost-efficient solution. When the number of nodes is large, star networks may be desirable from the point of view of overall resource usage. This result leads us to conjecture that, when feasible, centralized networks, where the “center” consists of a few fully connected nodes can be an interesting alternative to completely distributed solutions, provided that incentive mechanisms to handle network asymmetries are in place.

We believe however that this paper raises more questions than it provides answers. First, we only analyzed a handful of DHT routing geometries, and even omitted interesting geometries such as the butterfly [13] or geometries based on the XOR metric [14]. Second, applying the proposed cost model to deployed peer-to-peer systems such as KaZaA/FastTrack, which is based on in-



(a) Latency cost ($l = 1$) (b) Routing cost ($r = 1000$)

Figure 1: Latency and routing costs. Curves marked "sim" present simulation results.

terconnected star networks, could yield some insight regarding user behavior. Third, for the mathematical analysis, we used strong assumptions such as identical popularity of all items and uniform spatial distribution of all participants. Relaxing these assumptions is necessary to evaluate the performance of a geometry in a realistic setting. Also, obtaining a meaningful set of values for the parameters (l, s, r, m) for a given class of applications (e.g., file sharing between PCs, ad-hoc routing between energy-constrained sensor motes) also remains an open problem. Finally, identifying the minimal amount of knowledge each node should possess to devise a rational strategy, or studying network formation with the proposed cost model are other promising avenues for further research.

References

[1] The annotated Gnutella protocol specification v0.4, June 2001. `http://rfc-gnutella.sourceforge.net/developer/stable/index.html`.

[2] S. Banerjee, B. Bhattacharjee, and C. Kommareddy. Scalable application layer multicast. *Proc. ACM SIGCOMM'02*, pp. 205–217, Pittsburgh, PA, Aug. 2002.

[3] N. Christin and J. Chuang. On the cost of participating in a peer-to-peer network. Tech. rep., UC Berkeley, Dec. 2003. `http://p2pecon.berkeley.edu/pub/TR-2003-12-CC.pdf`. See also: arXiv:cs.NI/0401010.

[4] Y.-H. Chu, S. Rao, and H. Zhang. A case for endsystem multicast. *Proc. ACM SIGMETRICS'00*, pp. 1–12, Santa Clara, CA, June 2000.

[5] B.-G. Chun, R. Fonseca, I. Stoica, and J. Kubiatowicz. Characterizing selfishly constructed overlay networks. To appear in *Proc. IEEE INFOCOM'04*, Hong Kong, March 2004.

[6] C. Perkins (ed). *Ad hoc networking*. Addison-Wesley, Boston, MA, 2000.

[7] A. Fabrikant, A. Luthra, E. Maneva, C. Papadimitriou, and S. Shenker. On a network creation game. *Proc. ACM PODC'03*, pp. 347–351, Boston, MA, July 2003.

[8] K. Gummadi, R. Gummadi, S. Gribble, S. Ratnasamy, S. Shenker, and I. Stoica. The impact of DHT routing geometry on resilience and proximity. *Proc. ACM SIGCOMM'03*, pp. 381–394, Karlsruhe, Germany, Aug. 2003.

[9] M. Jackson and A. Wolinsky. A strategic model for social and economic networks. *J. Econ. Theory*, 71(1):44–74, Oct. 1996.

[10] M. F. Kaashoek and D. Karger. Koorde: A simple degree-optimal distributed hash table. *Proc. IPTPS'03*, pp. 323–336, Berkeley, CA, Feb. 2003.

[11] J. Liebeherr, M. Nahas, and W. Si. Application-layer multicast with Delaunay triangulations. *IEEE J. Select. Areas Commun.*, 20(8):1472–1488, Oct. 2002.

[12] D. Loguinov, A. Kumar, V. Rai, and S. Ganesh. Graph-theoretic analysis of structured peer-to-peer systems: routing distances and fault resilience. *Proc. ACM SIGCOMM'03*, pp. 395–406, Karlsruhe, Germany, Aug. 2003.

[13] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: a scalable and dynamic emulation of the butterfly. *Proc. ACM PODC'02*, pp. 183–192, Monterey, CA, July 2002.

[14] P. Maymounkov and D. Mazières. Kademlia: A peer-to-peer information system based on the XOR metric. *Proc. IPTPS'02*, pp. 53–65, Cambridge, MA, Feb. 2002.

[15] C. Ng, D. Parkes, and M. Seltzer. Strategyproof computing: Systems infrastructures for self-interested parties. *Proc. 1st Workshop on Econ. of Peer-to-Peer Syst.*, Berkeley, CA, June 2003.

[16] C. G. Plaxton, R. Rajamaran, and A. Richa. Accessing nearby copies of replicated objects in a distributed environment. *Theory of Comp. Syst.*, 32(3):241–280, June 1999.

[17] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network. *Proc. ACM SIGCOMM'01*, pp. 161–172, San Diego, CA, Aug. 2001.

[18] A. Rowston and P. Druschel. Pastry: Scalable, decentralized object location and routing for large scale peer-to-peer systems. *Proc. ACM Middleware'01*, pp. 329–350, Heidelberg, Germany, Nov. 2001.

[19] I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup protocol for Internet applications. *IEEE/ACM Trans. Net.*, 11(1):17–32, Feb. 2003.

[20] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J. Kubiatowicz. Tapestry: A resilient global-scale overlay for service deployment. *IEEE J. Select. Areas Commun.*, 22(1):41–53, Jan. 2004.