On the Connectedness of Peer-to-Peer Overlay Networks*

Weisheng Si MingChu Li
Software School, Dalian University of Technology, Dalian, China
Email: {siweish, mingchul}@dlut.edu.cn

Abstract

Peer-to-Peer (P2P) applications typically use overlay networks to forward the content queries. However, due to the distributed fashion of the overlay establishment, the overlay networks in P2P applications are not guaranteed to be connected. In this paper, we introduce a new concept called 'buddy-assignment graph' for the P2P overlay networks and prove a theorem that the connectedness of a P2P overlay network, being irrelevant to the kind of its overlay graph, is equivalent to the connectedness of its buddy-assignment graph. Then, with the aid of this theorem, we show that (1) when all hosts are active in the P2P overlay networks, two random buddies for each host suffice to make the connectedness probability close to 1 *as well as approximating to* 1 *with the increase of the overlay scale and (2) when hosts are only active in certain probability, though the connectedness probability decreases with the increase of the overlay scale, the connectedness probability can still be very quickly raised to almost* 1 *by increasing the number of buddies for each host.*

1. Introduction

Since the emergence of Napster in 1999, Peer-to-Peer (P2P) applications have gained great popularity. In practical use, Gnutella [3] and KaZaA [7] represent two famous examples of the P2P applications today. In research area, considerable activity is also done on various aspects of the P2P applications. For example, many content lookup services such as Chord [11], CAN [8], Tapestry [12], and Pastry [9] are proposed; [10] investigates the host characteristics; and [1] proposes a new scalable P2P system called 'Gia' that achieves three to five orders of magnitude improvement in the total system capacity.

In this paper, we focus on the connectedness of the overlay networks in the P2P applications. Briefly, an overlay network is an application-layer network topology established by the P2P hosts. Figure 1 shows an example of the overlay networks. P2P applications such as Gnutella or KaZaA typically build their overlay networks in the following way: each host maintains a list of some other hosts (usually called 'buddies') in the P2P application, and the overlay network is built by letting each host contact its buddies. Due to the distributed fashion of such overlay establishment, the P2P overlay networks are not guaranteed to be connected. For example, in KaZaA, if all buddies of hosts in America only exist in America and all buddies of hosts in Asia only exist in Asia, then the whole KaZaA network will be at least divided into two separated overlay networks — one in America and the other in Asia.

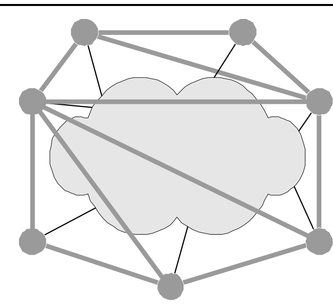

Figure 1. An example of the P2P overlay networks. The shaded circles and lines depict hosts and connections in the overlay network respectively.

Since the main use of overlay networks is to

* This research is supported by the National Natural Science Foundation of China under Grant No. 90412007.

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05)
0-7695-2281-5/05 $20.00 © 2005 **IEEE**

forward the content queries in the P2P applications, the connectedness of overlay networks will significantly affect the scope that the queries can reach. To deepen the understanding of the overlay connectedness in P2P applications, in this paper we study the connectedness probability for the P2P overlay networks under the assumption that the buddies of each host are randomly assigned. Here by 'randomly' we mean that for a host, the probability of any other host to be assigned as its buddy is equal. We note that in practice (e.g., in Gnutella or KaZaA), some well-known buddies are generally exploited to improve the overlay connectedness, but what we are interested in this paper is what will occur if we simply take the random assignment approach.

Under the above random assignment assumption, we study the connectedness probability of the P2P overlay networks in two cases. In Case 1, we let all P2P hosts be active and in Case 2, we let all P2P hosts be active only in certain probability, considering that in an actual P2P environment, hosts are not always participating in the overlay network. It is worth noting that **for both cases, we obtain very encouraging results:** for Case 1, two random buddies for each host suffice to make the connectedness probability close to 1 as well as approximating to 1 with the increase of the overlay scale and for Case 2, though the connectedness probability decreases with the increase of the overlay scale, the connectedness probability can be very quickly raised to almost 1 by increasing the number of buddies for each host.

The rest of the paper is structured as follows. Section 2 gives the definitions and theorems introduced by us to analyze the connectedness of the P2P overlay networks. Section 3 presents our experimental and theoretical results on the connectedness probability in detail. Finally, section 4 concludes this paper.

2. Definitions and Theorems

In this paper, we consider an overlay network as a graph G with node set V and edge set E, where V consists of nodes abstracting hosts in P2P applications and E consists of edges abstracting overlay connections among the hosts. We call such a graph G an 'overlay graph'. Note that we consider the overlay graph as an *undirected* graph, hence its connectedness in the *undirected* sense. With the above abstraction, when we say an overlay network is connected or disconnected, we exactly mean that its corresponding overlay graph is connected or disconnected.

Overlay networks in P2P applications adopt different kinds of overlay graphs. For example, Gnutella [3] and KaZaA [7] use *unstructured* graphs; Chord [11] and CAN [8] use *k-node-connected* graphs; and [6] proposes the use of the *de Bruijn* graphs to obtain short routing distances and high resilience to host failures in the overlay network. Every P2P application has its own overlay protocol to establish its overlay graph, and how these overlay protocols operate is beyond the scope of this paper. Here we only acknowledge that one main function of these P2P overlay protocols is to establish the connected overlay graphs with the given buddy assignment.

To make our connectedness study independent of the kinds of overlay graphs adopted, we introduce the concept of the 'buddy-assignment graph' for the P2P overlay networks. We first give its definition below. Then, we will show that the connectedness of a P2P overlay network, being irrelevant to the kind of its overlay graph, is actually determined by its 'buddy-assignment graph'.

Buddy-Assignment Graph For a P2P overlay network, consider the same node set V in its overlay graph. Define R, a relation on V, as the set of the ordered pairs (u, v) satisfying that $u, v \in V$ and u has buddy v. Denote the set of active nodes, a subset of V, V_{act}. Given R and V_{act} of an overlay, its buddy-assignment graph B is obtained as follows:

1. B has the node set of V_{act};
2. For each $(u, v) \in R$, add edge uv into the edge set of B, if both $u, v \in V_{act}$.

Note that (1) we consider the buddy-assignment graph also as an *undirected* graph, since in overlay networks, once a host contacts its buddy, its buddy learns the host's address and can contact the host back, despite the fact that the buddy relationship is directional; (2) the buddy-assignment graph is different from the overlay graph — while the former is simply an abstraction of the buddy relationship used by the overlay protocols to establish the latter, the latter has the actual usage of forwarding the content queries and providing resilience to host failures; and (3) in the following, we also consider the overlay graph has node set V_{act}, discarding those inactive nodes.

To help understanding the concept of the buddy-assignment graph, Figure 2 shows an example of the buddy-assignment graph and the established overlay graph for a P2P overlay network that uses *complete graph* as the overlay graph. Below we give the theorem

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05)
0-7695-2281-5/05 $20.00 © 2005 IEEE

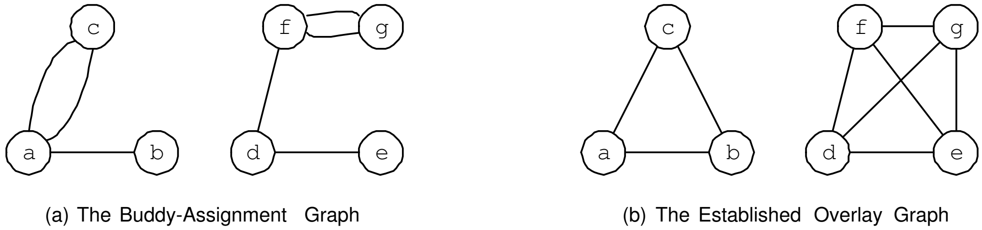

Figure 2. An example of the buddy-assignment graph and the established overlay graph for an overlay network that uses *complete graph* as the overlay graph. Here $V = V_{act} = \{a, b, c, d, e, f, g\}$ and $R = \{(a, c), (b, a), (c, a), (d, e), (d, f), (f, g), (g, f)\}$, which results in a two-component overlay graph.

on the connectedness relationship between the buddy-assignment graph and the overlay graph:

Theorem 1 *For a P2P overlay network, its overlay graph G is connected if and only if its buddy-assignment graph B is connected.*

Proof For a P2P overlay network, if B is connected, it implies that the address of any host in this overlay can actually reach any other host in the same overlay. Therefore, the overlay protocol in this overlay network has enough information to establish a connected overlay graph, no matter what kind of overlay graph is to be used. So G is connected. On the other hand, if B is not connected, say B_1 and B_2 are two connected components of B with no connections to each other, then nodes in B_1 must have no buddies in B_2 and nodes in B_2 must have no buddies in B_1, so nodes in B_1 and B_2 have no way to learn each other, which must result in two connected components G_1 and G_2 in G, with G_1 and G_2 having the same sets of nodes as B_1 and B_2 respectively. Therefore, G is not connected. ∎

With the above theorem, it is easy to prove the following corollary:

Corollary 1 *For a P2P overlay network, if its buddy-assignment graph B has connected components B_1, B_2, ..., B_k ($k \geq 1$), then the established overlay graph G has connected components G_1, G_2, ..., G_k, with G_i having the same node set as B_i respectively, $i \in 1 \ldots k$; and vice versa.*

Having the above results, we can study the connectedness of the P2P overlay networks only by looking at their buddy-assignment graphs, the connectedness results of which can be easily interpreted to the P2P overlay networks.

3. Experimental and Theoretical Results

In this section, we present the experimental and theoretical results on the connectedness probability of the P2P overlay networks in two cases, with both cases under the assumption that the buddies for each host are randomly assigned. All of our probability results are obtained from the corresponding buddy-assignment graphs of the P2P overlay networks of these two cases, and according to Theorem 1 and Corollary 1, these results also hold for the P2P overlay networks.

Our experimental study presented in the following two subsections is done with a simulator called "buddygraph" developed by us. Given the parameters of a kind of buddy-assignment graph (e.g., number of nodes in the graph, number of buddies for each node, and active probability of nodes, etc.), this simulator can produce the connectedness probability for this kind of graphs by conducting a large number (up to 10^8) of experiments on it with buddies assigned randomly each time.

Note that we obtain the connectedness probabilities for the kinds of buddy-assignment graphs studied by doing experiments is because up to now, no general formulae for calculating the connectedness probabilities for these kinds of graphs are found yet.

3.1. Case 1 — All Hosts Active

In Case 1, we assume the following on the P2P overlay networks:

1. All overlay hosts are active.

2. Each overlay host has an equal number of distinct random buddies.

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05)
0-7695-2281-5/05 $20.00 © 2005 IEEE

We denote the buddy-assignment graph for this case $B_m^{(n)}$, where n is the number of nodes in the graph and m is the number of buddies for each node.

Using our "buddygraph" simulator, we conduct experiments on $B_m^{(n)}$'s with various n, m values. For each n, m value pair, we compute the connectedness of 10^8 graphs. Our experiment results are summarized in Table 1. From this table, we can see that (1) when $m = 2$, the probability for $B_m^{(n)}$ to be connected is very close to 1 (larger than 0.9999) for all experimented n, while approximating to 1 with the growth of n and (2) when $m = 3$, for all experimented n, no disconnected $B_m^{(n)}$'s are spotted during the 10^8 experiments.

By mathematical calculation, it is easy to obtain $Prob\,(B_2^{(6)}\ \text{is connected}) = 1 - 10^{-5}$ and $Prob\,(B_2^{(7)}\ \text{is connected}) = 1 - 7 \times 75^{-3} \simeq 1 - 1.659 \times 10^{-5}$, which assures us the experimental results are correct. Also, it is apparent that $Prob\,(B_2^{(n)}\ \text{is connected}) = 1$, when $n = 3, 4$, or 5.

The theoretical explanation of the above almost certain connectedness can be found in [2], which discusses the connectivity of one kind of random graph [4] called the 'random m-orientable graph', in which each node randomly generates m edges. Concretely, [2] shows that for a random m-orientable graph $G_m^{(n)}$, where n is the number of nodes in the graph and m is the number of random edges generated at each node, the following properties in terms of graph *vertex connectivity* and *edge connectivity*[1] hold:

$$\text{For } m \geq 2,\ \lim_{n\to\infty} Prob\,(C_v(G_m^{(n)}) = m) = 1. \quad (1)$$

$$\text{For } m \geq 2,\ \lim_{n\to\infty} Prob\,(C_e(G_m^{(n)}) = m) = 1. \quad (2)$$

In the above equations, $C_v(G_m^{(n)})$ denotes the *vertex connectivity* and $C_e(G_m^{(n)})$ denotes the *edge connectivity* of $G_m^{(n)}$ respectively. As an aside, in case of $m = 1$, it can be easily derived from the results in [5] that when $n \to \infty$, $Prob\,(G_1^{(n)}\ \text{is connected}) \to 0$.

According to our assumptions on the P2P overlay networks in this case, $B_m^{(n)}$ is an instance of the $G_m^{(n)}$, so Equation (1) and (2) of course hold for $B_m^{(n)}$. Since $Prob\,(B_m^{(n)}\ \text{is connected})$ is apparently larger than $Prob\,(C_v(B_m^{(n)}) = m)$ and $Prob\,(C_e(B_m^{(n)}) = m)$, we can obtain:

$$\text{For } m \geq 2,\ \lim_{n\to\infty} Prob\,(B_m^{(n)}\ \text{is connected}) = 1. \quad (3)$$

This equation explains the almost certain connectedness of $B_m^{(n)}$ shown in Table 1.

Note that the overlay graphs are not the same graphs as $B_m^{(n)}$, so we cannot infer their *vertex connectivity* or *edge connectivity* using Equation (1) and (2). However, with Theorem 1, we can definitely interpret the connectedness results of $B_m^{(n)}$ to the overlay graphs. In summary, for P2P overlay networks in this case we obtain that:

- The probability of overlay connectedness is close to 1 (larger than 0.9999) for any overlay scale, as long as each host has at least two random buddies.
- When each host has at least two random buddies, the probability of overlay connectedness approximates to 1 with the growth of the overlay scale.

3.2. Case 2 — Hosts Active in Certain Probability

In a typical P2P environment, hosts are not always active. So in this subsection we study the overlay connectedness under the following assumptions:

1. Each overlay host is active in an equal probability (< 1).
2. Each overlay host has an equal number of distinct random buddies.

We denote the buddy-assignment graph for this case $B_m^{(n,p)}$, where n is the total number of nodes in the graph, p is the probability for a node to be active, and m is the number of buddies for each node.

We use our "buddygraph" simulator to conduct experiments on the $B_m^{(n,p)}$'s with various n, m, and p values. In our experiments, we find many disconnected $B_m^{(n,p)}$'s have the monolithic form that one connected component consists of most of the nodes and all the other connected components consist of only one node. For example, an experiment on $B_9^{(1000,1/3)}$ typically produces the result like (303, 1, 1, 1), meaning that total 306 nodes are active and four connected components (with the node number 303, 1, 1, and 1 respectively) are generated. Our experiment results are summarized in Table 2, which gives the probability of the disconnected $B_m^{(n,p)}$'s (denoted by P) and the probability of the disconnected $B_m^{(n,p)}$'s with the monolithic

1 The *vertex connectivity* of a graph is defined as the minimum number of vertices (nodes) the deletion of which disconnects the graph. The *edge connectivity* is defined the same way in terms of edges.

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05)
0-7695-2281-5/05 $20.00 © 2005 IEEE

n	m	Number of Experiments	Times of Being disconnected	Probability of Being Connected
6	2	10^8	991	$1 - 9.91 \times 10^{-6}$
7	2	10^8	1642	$1 - 1.642 \times 10^{-5}$
20	2	10^8	83	$1 - 8.3 \times 10^{-7}$
200	2	10^8	0	1
1000	2	10^8	0	1
20	3	10^8	0	1
200	3	10^8	0	1
1000	3	10^8	0	1

Table 1. Buddy-assignment graph connectedness probability when all nodes are active.

n	P	$P_{monolith}$
50	0.034024	0.933569
200	0.090376	0.988548
1000	0.352576	0.991749
5000	0.882018	0.985340
10000	0.985964	0.974174

(a) $m = 9, p = 1/3$

n	P	$P_{monolith}$
50	0.000951	0.950578
200	0.001404	0.998575
1000	0.005474	0.999817
5000	0.025763	0.999764
10000	0.050232	0.999930

(b) $m = 15, p = 1/3$

n	P	$P_{monolith}$
50	0.000000	1.000000
200	0.000000	1.000000
1000	0.000000	1.000000
5000	0.000000	1.000000
10000	0.000000	1.000000

(c) $m = 30, p = 1/3$

n	P	$P_{monolith}$
50	0.032175	0.897627
200	0.086751	0.982133
1000	0.326212	0.998229
5000	0.857042	0.984567
10000	0.979371	0.973909

(d) $m = 12, p = 1/4$

n	P	$P_{monolith}$
50	0.000000	1.000000
200	0.000219	1.000000
1000	0.000729	1.000000
5000	0.003188	0.999686
10000	0.006288	0.999841

(e) $m = 24, p = 1/4$

n	P	$P_{monolith}$
50	0.000000	1.000000
200	0.000000	1.000000
1000	0.000000	1.000000
5000	0.000000	1.000000
10000	0.000002	1.000000

(f) $m = 40, p = 1/4$

Table 2. Probabilities on disconnected $B_m^{(n,p)}$'s during 10^6 experiments. Here P denotes the probability of disconnected $B_m^{(n,p)}$'s among all experimented $B_m^{(n,p)}$'s and $P_{monolith}$ denotes the probability of the disconnected $B_m^{(n,p)}$'s with the monolithic form among all disconnected $B_m^{(n,p)}$'s.

form among all disconnected $B_m^{(n,p)}$'s (denoted by $P_{monolith}$). We see from this table that:

- With m, p not changing, the connectedness probability of the $B_m^{(n,p)}$'s (i.e., $1 - P$), decreases with the growth of n. This is contrary to Case 1, where the connectedness probability approximates to 1 with the growth of n.
- Increasing the value of m can sharply increase the connectedness probability of the $B_m^{(n,p)}$'s, and a moderate m can almost ensure the connectedness of the $B_m^{(n,p)}$'s (e.g., in Table 2 (c), $m = 30$ gives us $1 - P \simeq 1$).
- Most of the disconnected $B_m^{(n,p)}$'s have the aforementioned monolithic form, implying that it is very unlikely for an overlay to be divided into several multiple-node components under the assumptions in this case.

Table 3 gives the probability density on the number of connected components in the $B_m^{(1000,p)}$'s with various m and p values in 10^6 experiments. Note in this table, P_i denotes the probability of the i-component $B_m^{(1000,p)}$'s, where i means the number of connected components in

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05)
0-7695-2281-5/05 $20.00 © 2005 IEEE

m	p	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
9	1/3	0.647292	0.275264	0.064575	0.010863	0.001593	0.000177	0.000033	0.000003
12	1/3	0.952162	0.046452	0.001347	0.000038	0.000001	0	0	0
15	1/3	0.994630	0.005346	0.000024	0	0	0	0	0
30	1/3	1.000000	0	0	0	0	0	0	0
12	1/4	0.671448	0.260163	0.057698	0.009273	0.001262	0.000143	0.000011	0.000002
16	1/4	0.952653	0.045935	0.001385	0.000025	0.000002	0	0	0
24	1/4	0.999271	0.000726	0.000003	0	0	0	0	0
40	1/4	1.000000	0	0	0	0	0	0	0

Table 3. Probability density on the number of connected components for $B_m^{(1000,p)}$ in 10^6 experiments.

the graph; P_1 means the probability of the connected $B_m^{(1000,p)}$'s; and when $i > 8$, the i-component $B_m^{(1000,p)}$'s are not spotted in 10^6 experiments, so we omit their probability values from this table. This table shows that:

- Increasing the value of m can not only quickly increase the probability of the connected $B_m^{(n,p)}$'s, but also significantly reduce the number of connected components in the disconnected $B_m^{(n,p)}$'s.
- The number of connected components in the $B_m^{(n,p)}$'s are very small in most cases and the probability density for the number of connected components drops quickly with the increase of the number of the connected components.

Applying the results from Table 2 and Table 3 to the overlay networks in this case, we can obtain that:

1. Increasing the number of buddies for each host can significantly improve the overlay connectedness, and a moderate number of buddies for each host can almost guarantee the overlay connectedness.
2. In case of disconnected overlays, the overlays will still have good connectivity in that most of the overlay hosts form a large connected component and the number of total connected components is very small in general.

4. Conclusions

The connectedness of the overlay networks is essential for the forwarding of content queries in the P2P applications. To deepen the understanding of the connectedness of the P2P overlays, in this paper we presented both experimental and theoretical results on the connectedness probability for the P2P overlays under the assumption that the buddies for each host are randomly assigned. We first introduced a new concept called 'buddy-assignment graph' and proved a theorem that the connectedness of a P2P overlay network, being irrelevant to the kind of its overlay graph, is actually determined by its buddy-assignment graph. Then, with the aid of this theorem, we mainly obtained **two very encouraging results:**

- When all hosts in the P2P overlay networks are active, a minimum of two random buddies for each host can guarantee that the connectedness probability of the overlays is almost 1 as well as approximates to 1 with the increase of the overlay scale.
- When hosts in the P2P overlay networks are only active in certain probability, the connectedness probability of the overlays drops with the increase of the overlay scale. However, it can be very quickly raised to almost 1 by enlarging the number of buddies for each host.

References

[1] Y. CHAWATHE, S. RATNASAMY, L. BRESLAU, N. LANHAM, and S. SHENKER. Making Gnutella-like P2P Systems Scalable. In *Proceedings of ACM SIGCOMM 2003*, pages 407–418, Aug. 2003.

[2] T. I. Fenner and A. M. Frieze. On the connectivity of random m-orientable graphs and digraphs. *Combinatorica*, 2:347–359, 1982.

[3] Gnutella. Gnutella project. http://www.gnutella.com.

[4] S. Janson, T. Luczak, and A. Rucinski. *Random Graphs*. John Wiley and Sons, INC., 2000.

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05)
0-7695-2281-5/05 $20.00 © 2005 IEEE

[5] L. Katz. Probability of indecomposability of a random mapping function. *Annals Of Mathematical Statistics*, 25:512–517, 1955.

[6] D. Loguinov, A. Kumar, V. Rai, and S. Ganesh. Graph-Theoretic Analysis of Structured Peer-to-Peer Systems: Routing Distances and Fault Resilience. In *Proceedings of ACM SIGCOMM 2003*, pages 395–406, Aug. 2003.

[7] S. Networks. Kazaa. http://www.kazaa.com/.

[8] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A Scalable Content-Addressable Network. In *Proceedings of ACM SIGCOMM 2001, San Diego*, pages 149–160, Aug. 2001.

[9] A. Rowstron and P. Druschel. Pastry: scalable, distributed object location and routing for large-scale peer-to-peer systems. In *Proceedings of IFIP/ACM Middleware 2001*, Heidelberg, Germany, Nov. 2001.

[10] S. Saroiu, P. Gummadi, and S. Gribble. A Measurement Study of Peer-to-Peer File Sharing Systems. In *Proceedings of Multimedia Computing and Networking, 2002.*, Jan. 2002.

[11] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications. In *Proceedings of ACM SIGCOMM 2001, San Diego*, pages 160–172, Aug. 2001.

[12] B. Y. Zhao, J. Kubiatowicz, and A. Joseph. Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing. Technical Report UCB/CSD-01-1141, University of California, Berkeley, April 2001.

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS'05)
0-7695-2281-5/05 $20.00 © 2005 IEEE