Analyzing Network Coverage in Unstructured Peer-to-Peer Networks: A Complex Network Approach

Joydeep Chandra, Santosh Shaw, and Niloy Ganguly

Department of Computer Science & Engineering,
Indian Institute of Technology, Kharagpur-721302, India
{joydeep,santosh,niloy}@cse.iitkgp.ernet.in

Abstract. In this paper, we apply the theory predicting neighbor distribution of arbitrary random graphs to analyze the network coverage of the peers in unstructured peer-to-peer(p2p) networks that use TTL-based flooding mechanism for search and query. However, we find that for many cases, the theory cannot be directly applied to obtain correct estimate of network coverage due to the presence of certain types of edges that we refer as cross and back edges. It is also observed that the presence of cross and back edges in the p2p networks reduce the coverage of the peers and also generates large number of redundant messages, thus wasting precious bandwidth. We refine the theory and develop a model to estimate the network coverage of the peers in the presence of cross and back edges. We simulate our model for different networks with various degree distribution properties. The results indicate that our models provide good estimates of second neighbor and network coverage distribution. We perform a case study of the Gnutella networks to analyze the effects cross and back edges on network coverage and message complexity in these networks. Based on our study, we propose a new bootstrapping algorithm for Gnutella networks named HPC5 that substantially improves the network coverage and message complexity. The results have been validated using simulations.

Keywords: Peer-to-Peer Networks, Network Coverage Models, Overlay Networks, Gnutella.

1 Introduction

The unstructured peer-to-peer (p2p) networks like Gnutella [1][2], Kazaa[3] and FreeHaven[4] use broadcasting as their query and search mechanism. Thus the query and search performance of these p2p networks are directly proportional to the network coverage of the peers achieved through broadcasting. A high network coverage of the peers implies that queries reach a large subset of peers in the network, and thus yields better search performance. However, as of now, in most unstructured p2p networks like Gnutella and Kazaa, the fundamental strategy to improve coverage is to introduce more overlay links, thereby,

L. Fratta et al. (Eds.): NETWORKING 2009, LNCS 5550, pp. 690–702, 2009.

leading to huge Internet traffic. With the unbridled growth of the p2p networks in the past few years, the ISP's are facing a huge problem of network congestion and bandwidth consumption [5][6]. These problems are expected to be a big research challenge in the forthcoming days, and several recent works have started addressing these problems [5][7][8]. The scale of the problem increases as broadcast leads to redundant message generation and consequently wastage of precious bandwidth. However, properly analyzing the topological behavior of the networks and the impact of topology on coverage and redundancy can provide new insights in alleviating traffic and redundancy problems.

In this paper, we initially build up a basic analytical model to assess the network coverage of p2p networks that uses $TTL(2)$ based search and query mechanisms. We limit our study to $TTL(2)$ based networks, as search and query in popular unstructured networks like Gnutella uses $TTL(2)$ for most search cases. $TTL(3)$ is used only for rare searches; however, our models and results can easily be generalized for $TTL(3)$ searches as well. The basic model has been developed using the theory applied to derive the distribution of first and second neighbors of randomly selected node in large networks [9][10]. To the best of our knowledge, this work is a pioneering work that applies the theories used to estimate neighbor distribution in analyzing network coverage of p2p networks. Further we propose a refinement of our basic model to perfect the estimation of the neighbor distributions for networks that contain certain type of edges, which we refer as cross and back edges. The effect of these edges is to reduce the coverage of the peers and increase message redundancy. Thus for finite-sized networks with high cross and back edges, the results of the basic models tend to deviate from the simulation results. We study the impact of these edges on the network coverage of the peers and derive suitable models for the same. We compare the results of the refined model with the simulation results; the comparison reveals that the refined model produces accurate results of network coverage. Finally, we apply our derivations on Gnutella networks and estimate its coverage based on certain key statistics. We found that the existing Gnutella protocol generates a lot of redundant traffic and has low network coverage. Hence, we propose a bootstrapping mechanism named HPC5 to improve the network coverage of the Gnutella protocol. The superior performance of the proposed mechanism is validated using simulations.

The rest of the paper is organized as follows: The theoretical concepts of complex networks[1] related to our model are discussed next. In section 3 we discuss our derived model for network coverage in finite sized networks. In section 4, we analyze the Gnutella protocol and propose a new bootstrapping mechanism for Gnutella. The simulation results are stated and discussed in section 6. Finally we present our conclusion in section 7.

[1] The theories developed to explain behaviors of large dynamic networks are loosely termed as Complex Network Theory [11].

2 Basic Model for First and Second Neighbor Distributions

Most unstructured p2p networks use flooding as a means for search and querying. Since flooding causes huge number of query packets to flow through the network, thus consuming huge bandwidth, most p2p networks use a TTL-based flooding scheme. The commonly used TTL value for ordinary searches in most networks is 2. Thus when a peer broadcasts a message with $TTL(2)$, the message reaches its immediate neighboring peers as well as their neighbors. Thus the coverage of a peer for a $TTL(2)$ broadcast is the sum of its first and second neighbors. Newman [10] derived models for the distribution of the number of first and second neighbors of a node in a large graph. Suppose, in a large network with N nodes (N is large), if p_k denotes the probability of any random node in a network having k first neighbors, then the first neighbor distribution — also referred as degree distribution — of the nodes can be represented using a generating function as,

$$G_0(x) = p_0 + p_1(x) + p_2(x^2) + p_3(x^3) \ldots . \tag{1}$$

Thus the coefficient of x^i in $G_0(x)$ gives the probability that any random node in the network will have degree i. The average number of neighbors of a node is given by,

$$\langle z \rangle = 1 \cdot p_1 + 2 \cdot p_2 + 3 \cdot p_3 + \ldots = G_0'(1). \tag{2}$$

Another important quantity is the distribution of the outgoing edges of a node reached by following a randomly chosen edge. If N_k denotes the number of nodes with degree k, then $p_k = \frac{N_k}{N}$, and the number of edges that leads to a node with degree k equals kN_k. Thus, the probability, $p_k^{(o)}$, of reaching a node with degree k by following a randomly chosen edge is,

$$p_k^{(o)} = \frac{kN_k/N}{\left(1 \cdot N_1 + 2 \cdot N_2 + 3 \cdot N_3 + \ldots + (N-1)N_{N-1}\right)/N} = \frac{kp_k}{\langle z \rangle}. \tag{3}$$

The generating function for the distribution of the outgoing edges of a node reached by following a random edge can be represented as,

$$G_1(x) = \frac{1}{\langle z \rangle} \cdot \left(\sum kp_k x^k\right) = \frac{G_0'(x)}{G_0'(1)}. \tag{4}$$

The coefficient of x^i in $G_1(x)$ gives the probability that any randomly chosen edge leads to a node with degree i. Suppose, we want to find the number of second neighbors of a node, P. Let $\hat{p}$ denote the connection probability between any two random nodes in the network. When N is large and $\hat{p} \to 0$, then the probability that an outgoing edge from a neighbor of P connects to another immediate neighbor of P, or to P itself is negligible. Moreover, under these conditions, the probability that two neighbors of P will have another common

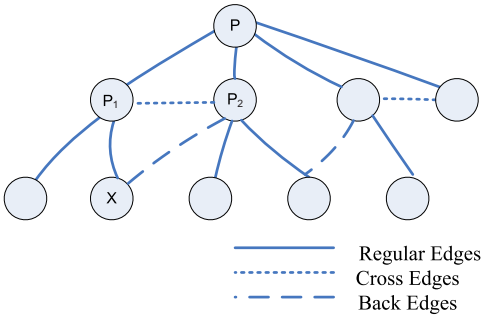


Fig. 1. A portion of a p2p topology. The solid lines indicate the regular edges that connect two peers. A fine broken line indicates a cross edge between two peers. A cross edge is an edge that connects directly two immediate neighbors of a peer. A heavy broken line indicates a back edge between two peers. Back edges are formed when a neighbor P_2 of peer P, connects to another peer, X, that is already connected to another neighbor P_1 of P.

node as neighbor is also negligible. According to power property of generating functions, $[G_1(x)]^k$ gives the distribution of the number of outgoing edges of k independent nodes. Thus, the distribution of the number of second neighbors of a node P, is given by,

$$S(x) = \sum_k p_k \left[G_1(x)\right]^k = G_0(G_1(x)). \tag{5}$$

The total network coverage of a peer in p2p networks that use $TTL(2)$ flooding scheme is the sum of its number of first and second neighbors. Using the above stated results, we can derive the generating function for the probability distribution of the total network coverage of any peer in the network. Thus, the distribution of the total node coverage of a peer P that deploys a $TTL(2)$ flooding mechanism is represented by the generating function $C(x)$ as,

$$C(x) = G_0(x) \cdot S(x) \tag{6}$$

Using these expressions, we can obtain the expected $TTL(2)$ coverage, $\langle c \rangle$ of a peer which is given as,

$$\langle c \rangle = C'(1). \tag{7}$$

Limitations: The above stated derivations can be used to model the expected first neighbor, second neighbor and total network coverage of any random node in a network. In unstructured p2p networks that use TTL-based flooding for search and query, the query messages reach the adjacent neighbors upto a number of hops, specified by the TTL value. Thus these derivations can be used to model the reachability of the queries in these networks. However, these expressions provide correct reachability distributions only when the peers reached from a

source node through a TTL-based message does not form any cycles among themselves. But, for many real cases, this condition fails to hold. Since p2p systems behave like social networks, the peers inherently form many short length cycles. The cycles that affect the coverage of the peers are referred to as *cross* and *back* edges as shown in fig. 1. A cross edge is formed when two adjacent nodes of a peer, say P gets connected by an edge, whereas a back edge is formed when a neighbor (say P_2) of P connects to another peer, say X, that is already connected to some other neighbor (say P_1) of P. The presence of back and cross edges reduces the coverage of a given source peer that use $TTL(2)$ flood. This can be directly interpreted from the figures 3(a) and 3(b), where the mean number of second neighbors of a peer is actually much less than predicted by our basic model. Thus in these cases, the basic model does not produce correct results for coverage of peer in a network that uses broadcast mechanism. In order to understand the actual coverage of the peers with a given degree distribution and given probability of back and cross edges, we need to develop a model that captures the effect of back and cross edges in the networks.

3 Network Coverage in Finite-sized Networks: Refined Model

We derive models for coverage of a peer that uses $TTL(2)$ flooding mechanism, when the degree distribution of the network is known. We also assume that the probability of a random edge being a back edge with respect to any source peer is fixed and given as b. We derive the second neighbor and the coverage distribution of any random peer in the network, while analytically deriving the cross-edge probability and eliminating its effects. We assume the probability that a random peer is of degree k be given as p_k for all possible values of k. We derive the peer coverage for these graphs, that deploys a $TTL(2)$ broadcast mechanism for query and search.

Let us assume that a network has N peers and a random peer P has k first neighbors. Initially, we intend to find the distribution of the number of outgoing edges, which are not cross edges, of a first neighbor of P. If a peer has j outgoing edges, then i unique neighbors has to be chosen from $N-(k+1)$ peers — since there are k first neighbors of P and P itself, so the total unique peers present in the network from which P will have to choose is $N-(k+1)$. This can be done in $\binom{N-k-1}{i}$ ways. The rest of the $j-i$ peers has to be chosen from k peers, and this can be done in $\binom{k}{j-i}$ ways. Thus, for any first neighbor of P having a total of j outgoing edges, the probability of having i edges that are not cross edges, is given as

$$R_{k,i,j} = \frac{\binom{N-k-1}{i}\binom{k}{j-i}}{\binom{N-1}{j}}.$$

Hence, for any random neighbor of P (having k first neighbors), the distribution for having i non-cross edges, is given by

$$R_{k,i} = \sum_{j=i}^{k+i} q_j \left[\frac{\binom{N-k-1}{i}\binom{k}{j-i}}{\binom{N-1}{j}} \right], \tag{8}$$

where q_j is the probability of having j outgoing edges of a peer, reached by selecting a random edge — as obtained from the coefficent of x^j of $G_1(x)$ (Eq. 4). One must note that the values of j range from i to $k+i$. For any value of $j < i$, the probability $R_{k,i,j}$ becomes equal to zero. Similarly, when $j > k+i$, the number of non-cross outgoing edges from j must be greater than i and hence $R_{k,i,j}$ is again equal to zero.

Thus, the distribution of the total number of non-cross outgoing edges from any random neighbor (out of k first neighbors) of P, can be represented using generating function as,

$$\hat{R}_k(x) = \sum_{i'} R_{k,i'} x^{i'}. \tag{9}$$

Thus, the distribution of the total number of non-cross outgoing edges from all the k first neighbors of P can be found from the power property of generating functions and is given as,

$$\Gamma_k(x) = \left[\hat{R}_k(x)\right]^k = \Gamma_{k,0} + \Gamma_{k,1}x^1 + \Gamma_{k,2}x^2 + \ldots (say), \tag{10}$$

$$= \sum_m \Gamma_{k,m} x^m. \tag{11}$$

Now, suppose the probability that any random edge is a back edge with respect to the source node P is known and is denoted as b, then for a neighbor X with t non-cross edges, the distribution of the number of non-back edges with respect to source node P can be represented by the generating function as,

$$Q_t(x) = \sum_{\gamma \le t} \binom{t}{\gamma} (1-b)^\gamma (b)^{t-\gamma} x^\gamma, \tag{12}$$

$$= 0 \text{ for } \gamma > t. \tag{13}$$

Thus, $Q_t(x)$ gives the distribution of the number of edges, out of a total of t edges, from neighbor X of P that connects to distinct nodes. The distribution of the actual number of unique peers to which k first neighbors of P connect is given by,

$$A_k(x) = \sum_{t'} \Gamma_{k,t'} Q_{t'}(x), \tag{14}$$

and the distribution of the number of unique second neighbors for any random peer in a network is,

$$\hat{S}(x) = \sum_{k'} p_{k'} A_{k'}(x). \tag{15}$$

The total coverage of the network will be given by

$$\hat{C}(x) = G_0(x) \cdot \hat{S}(x) \tag{16}$$

The model is experimentally validated for networks with Poisson and power-law degree distributions. The validation results are presented cohesively in section 6 after we explain the Gnutella model and the improvement algorithm in 4. We also verified the model for Gnutella networks based on its certain key statistics.

4 Analysis of the Gnutella Protocol

One of the aims of the paper is to verify the correctness of the refined model in Gnutella networks. Moreover, we propose techniques to modify the existing Gnutella protocol to eliminate back and cross edges thus significantly enhancing its coverage. We have build up a Gnutella prototype in order to carry out the experiments. The prototype is built by studying the basic model of Gnutella, its bootstrapping protocol, and its basic search technique. Certain key statistics of the Gnutella network, based on studies conducted by several researchers[1][12][13][14] is also used to develop a prototype that we discuss next. Our prototype reveals that Gnutella forms large number of back and cross edges. Since Gnutella uses $TTL(2)$ flooding for most ordinary searches, these back and cross edges generate large redundant queries at the peers. Based on these observations, we propose certain changes in the bootstrapping protocol of Gnutella that reduces query redundancy and improves network coverage (results presented in section 6).

Basic Model: Gnutella 0.6 is a two-tier overlay network, consisting of two types of nodes : *ultra-peer* and *leaf-peer* (the term peer represents both ultra and leaf peer). An ultra-peer is connected with a limited number of other ultra-peers and leaf-peers. A leaf-peer is connected with some ultra-peers. However, there is no direct connection between any two leaf-peers in the overlay network.

Bootstrapping and Handshaking Protocol: Many software clients are used to access the Gnutella network (like *Limewire, Bearshare, Gtk-gnutella*). The most popular client software, Limewire's handshake protocol is used in our prototype. Through handshaking, a peer establishes connection with any other ultra-peer. To start handshake protocol a peer first collects the address of an online ultra-peer from a pool of online ultra-peers. A peer can collect the list of online peers from *hardcoded* address/es and/or from *GwebCache* systems [15] and/or through *pong-caching* and/or from its own hard-disk which has obtained a list of online ultra-peers in the previous run [12]. A handshake protocol is used to make new connections [1,2].

Basic Search Technique: The network follows limited flood based query search. A query of an ultra-peer is forwarded to its leaf-peers with $TTL(0)$ and to all its ultra-neighbors with one less TTL only when $(TTL > 0)$. A leaf-peer

does not forward query received from an ultra-peer. On the other hand ultra-peers perform query searching on behalf of their leaf peers. The query of a leaf-peer is initially sent to its connected ultra-peers. All the connected ultra-peers simultaneously forward the query to their neighbor ultra-peers up to a limited number of hops. While $TTL(2)$ is used for most searches, $TTL(3)$ is used for rare searches.

Key Statistics: Certain key statistics of the current Gnutella network are as follows [1][14]: Currently the number of peers in Gnutella Network is around 2000k, out of which 100k are live at any point of time[12]; the number of ultra-peers is around 15–16% of the total number of live peers. An ultra-peer connects to a maximum of 32 other ultra-peers, and to a maximum of 30 leaf peers, where as a leaf-peer connects to a maximum of 3 ultrapeers. The average number of neighboring ultra-peer of an ultra-peer is 25, whereas the average number of neighboring leaves is 22.

The key statistics points to some interesting analysis. If 15% of the live peers are ultra, then with 100k live peers, there are around N=15k ultra-peers. Since each ultra-peer connects to an average of 25 ultra-peers, so the average connection probability of an ultra-peer with another ultra is approximately $p^{uu} = .0017$, thus $p^{uu} \approx N^{-.67}$. As discussed in [11], when the connection probability between two random peers in a network with N peers increases beyond $N^{-\frac{2}{3}}$, a large number of short cycles of length 3 and 4 are created. Hence, in a Gnutella network, a high number of cross and back edges will exist, leading to huge traffic redundancy. To handle the problem of redundancy, we propose a mechanism named HPC5 for topology generation in Gnutella networks that eliminates cycles of length shorter than five in the network. We define a topology containing cycles not less than length r as a *Cycle-r* topology. The underlying rationale behind this proposal is that with a TTL(2) flooding, a cycle-5 topology will not generate any redundant messages at any node. We state our proposed HPC5 mechanism next. We simulated HPC5 and compared the network coverage and message complexity of peers with the Gnutella 0.6; the results are presented in section 6.

5 HPC5: Handshake Protocol for Cycle-5 Networks

As stated earlier, the major objective of HPC5 protocol is to eliminate the possibility of forming short length cycles (cycles of length 3 or 4). Figure 2 illustrates the proposed HPC5 graphically. It shows the various possibilities when peer-1 requests other online ultra-peers to be its neighbor, given that, peer-2 is already a neighbor of peer-1. In figures 2(a) and 2(b), the possibility of the formation of triangle and quadrilateral arises if a 1^{st} or 2^{nd} neighbor of peer-2 is selected. However, this possibility is discarded in fig. 2(c) and a cycle of length 5 is formed. HPC5 exactly ensures that.

Each peer maintains a list of its 1^{st} and 2^{nd} neighbors, which contains only ultra-peers (because a peer only sends request to an ultra-peer to make neighbor). The 2^{nd} ultra-neighbors of a leaf-peer represent the collection of 1^{st}

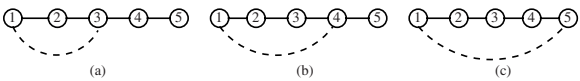


Fig. 2. Selection of neighbor by peer-1 after making peer-2 as a neighbor

ultra-neighbors of the connected ultra-peers. To keep updated knowledge, each ultra-peer exchanges its list of 1^{st} neighbors periodically with its neighbor ultra-peers and sends the list of 1^{st} neighbors to its leaf-peers. To do this with minimal overhead, piggyback technique can be used in which an ultra-peer can append its neighbor list to the messages passing through it.

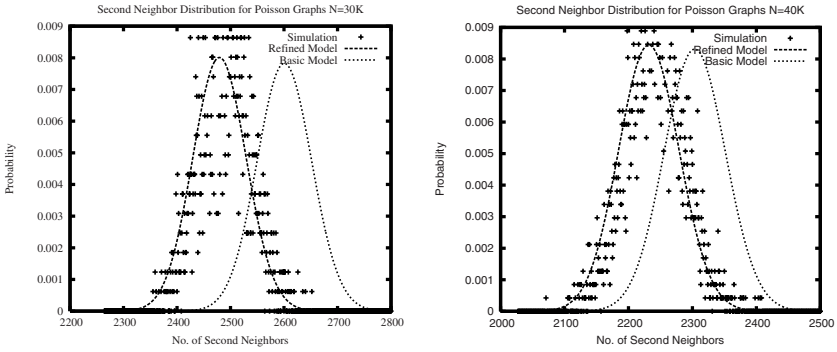
The three steps of modified handshake protocol (HPC5) is described below.

1. The initiator peer first sends a request to a remote ultra-peer which is not in its 1^{st} or 2^{nd} neighbor set. The request header contains the type of the initiator peer. The presence of remote peer in 2^{nd} neighbor set implies the possibility of 3-length cycle. In fig. 2, peer-1 cannot send request to peer 2 or 3, on the other hand peer 4 & 5 are eligible remote ultra-peers.
2. The recipient replies back with its list of 1^{st} neighbors and the neighbor-hood acceptance/rejection message. If the remote peer discards the connection in this step, the initiator closes the connection and keeps the record of neighbors of the remote peer for future handshaking process; on acceptance of the invitation by the remote-peer, the initiator peer checks the presence of at least one common peer between its 2^{nd} neighbor set (say A) and the 1^{st} neighbor set of the remote peer (say, B). A common ultra-peer between sets A and B indicates the possibility of 4-length cycle. In fig. 2, peer 3 is in the second neighbor set of 1, and in the first neighbor set of 4. Thus 1 and 4 cannot form neighbors.
   If no common peer is present between sets A and B then the initiator sends *accept connection* to remote peer.
   Otherwise the initiator sends *reject connection* to remote peer.

Thus HPC5 prevents the possibility of forming a cycle of length 3 or 4 and generates a cycle-5 network. We simulated the network coverage of the peers as shown in fig. 5; the results indicate that our proposed protocol has much improved network coverage as compared to Gnutella 0.6 that allows formation of a Cycle-3 topology.

6 Simulation Results

In this section, we present simulation results generated to validate theoretical correctnes for second neighbor distribution of various networks with different degree distributions, including an arbitrary distribution generated by the Gnutella prototype, and compared the results with our derived models. Moreover, we present results of the impact of HPC5 on the network coverage and message complexity of the Gnutella network.



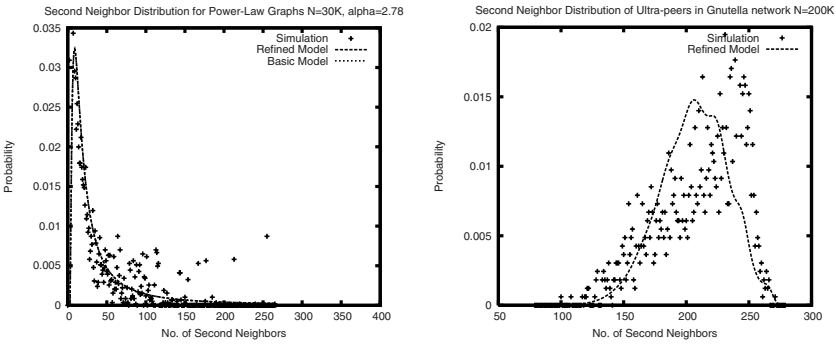
(a) N=30K (b) N=40K

Fig. 3. Second neighbor distribution of a random peer with $k = 51$ and 48 first neighbors for Erdos-Renyi networks (poisson degree distribution) with total peers $N = 30$K and 40K respectively. The value of $\hat{p}$ is .0017 for $N = 30$K and .0012 for $N = 40$K, and the back edge probability b is .04 and .03 respectively. The points show the simulation results, the heavy broken lines show the results of refined model, compared to the results of basic model (fine broken lines).

6.1 Second Neighbor Distribution

We considered networks like Erdos-Renyi networks[16][17], power law networks[18], and also arbitrary networks generated by the Gnutella prototype, for our simulations. For each of these cases, we simulated the second neighbor distribution of the peers for a given first degree k, and estimated back edge probability, b. We discuss the simulation details and results for each of these networks.

Case: Erdos-Renyi Graphs: Erdos-Renyi graphs[16][17] are random graphs, in which any two peers in the network are connected with a fixed probability $\hat{p}$. We simulated the second neighbor distribution in Erdos-Renyi networks with N =30K and 40K, for a back edge probability of .04 and .03 respectively, and for the first neighbor value k =51 and 48 respectively. The connection probability $\hat{p}$ was taken as .0017 and .0012 respectively. As seen in fig. 3, the results of the simulation for $N = 30K$ and $40K$ matches well with our refined model, where as the basic model considerably deviates from the simulation results. However, it can be seen that for the basic model, the closeness of fit is more for $N = 40$K as compared to $N = 30$K. When the size of the network, N, is increased considerably from 30K (fig. 3(a)) to 40K (fig. 3(b)), or if the connection probability between two random nodes, $\hat{p}$ is further reduced, then the simulation results matches well with the basic model as well as our refined model. This is because, when N increases or the connection probability $\hat{p}$ is considerably reduced, then the chances of forming cross or back edges by the peers become almost negligible, and thus in these limiting conditions, the basic model matches well with the simulation results. However, as $\hat{p}$ increases, the number of neighbors of the peers increases; thus the number of unique peers that has not been selected by



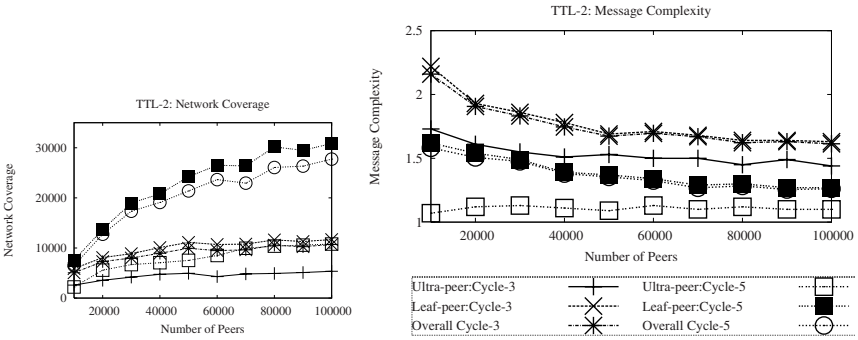
(a) Power-Law (b) Gnutella Network

Fig. 4. Second neighbor distribution of a peer with $k = 4$ first neighbors for power law network with $N = 30$K peers, $\alpha = 2.78$ and back probability $b = 0$, and a Gnutella network with $N \approx 27$K ultra peers and $b = .05$. The points show the simulation results, the heavy broken lines indicate the results of our refined model. For the case of power-law, the results of basic model matches exactly with the refined model.

other peer reduces. Hence cross and back edges are formed and the basic model fails to model the second neighbor of the peers precisely.

Case: Power-Law Graphs: We simulated the power-law graphs using the degree distribution given as $p_k \sim k^{-\alpha}$, where α is a constant that varies between 2 and 3 for all real networks that follows power law distribution[11][18]. The power-law network topology was generated using the well known configuration model[10]. We simulated the second neighbor distribution for varying number of peers from $N = 10K$ to $30K$, and for α varying from 2.31(high) to 3.0(low). Figure 4(a) shows the second neighbor distribution for peers with $k = 4$ first neighbors in a network with $N = 30$K and for $\alpha = 2.78$. Here, interestingly the results match well for both refined model, as well as the basic model. This is because, the power-law networks hold an important property; a majority of the peers in these networks have very low degree and only a very few peers have very high degree. As, most of the peers have very low connectivity, the chances of forming cross and back edges are inherently very less in power law networks, hence the simulation results match well with the basic as well as the refined model.

Case: Gnutella Network: Here we used the degree distribution of ultra-peers (considering only ultra-peer to ultra-peer connectivity) generated by the Gnutella prototype that we have implemented. In our Gnutella implementation we considered $N = 200$K total peers that have around 26842 ultra-peers; the ultra-peers connects to a maximum of 32 other ultra-peers, the average connectivity being 25. Figure 4(b) plots the second neighbor distribution of the peers that have $k = 9$ first degree neighbors with a back edge probability $b = .045$. The estimate of the back edge probability was obtained from the simulation of the Gnutella



(a) Network Coverage (b) Message Complexity

Fig. 5. Network Coverage and message complexity with TTL 2 for cycle-3 and cycle-5 networks

prototype. We observe that the simulated results fit well with that of our model. The minor difference is due to the method in which the back edge estimation is made. Unlike our consideration in the refined model, the back edge probability in this case is not same for all the peers and hence minor differences in the simulated results can be observed.

6.2 Impact of HPC5 on Gnutella Networks

We simulated the effect of existing Gnutella topology and our proposed HPC5 mechanism on the *network coverage* and *message complexity* of the network. We define message complexity as the average number of messages required to discover a peer in the overlay network whereas network coverage implies the number of unique peers explored during query propagation in limited flooding. The simulation results are shown in fig. 5. The network coverage of the peers improves by a maximum amount of 10%, whereas, the message complexity of the overall networks almost reaches 1, when HPC5 is used. Thus using HPC5 leads to significant improvement in network coverage and message complexity as compared to the cycle-3 networks in traditional Gnutella.

7 Conclusion

In this paper, we developed suitable models that quantify the coverage of the peers in networks that perform $TTL(2)$ searches. The models based on generating function formalism provides a strong theoretical foundation needed to understand the relation between the topology of a network and the achievable performance through TTL-based searches. Using the derived model, we provided an insight of the topological impact on network coverage and message complexity of the peers in Gnutella. The model revealed low network coverage and high message complexity in existing Gnutella, and helped us to propose a modified

bootstrap mechanism named HPC5 that showed improvement of almost 10% in network coverage and 30% in message complexity. The models can be extended further for higher values of TTL, and also for obtaining coverages in networks with high clustering coefficients. However, a more elegant methodology to calculate back edges needs to be developed in future.

References

1. Gnutella and Limewire, `http://www.limewire.org`
2. Gnutella Protocol Specification 0.6, `http://rfc-gnutella.sourceforge.net`
3. Liang, J., Kumar, R., Ross, K.: The KaZaA Overlay: A Measurement Study. In: Proceedings of the 19th IEEE Annual Computer Communications Workshop, Bonita Springs, Florida (2004)
4. The FreeHaven Project, `http://www.freehaven.net`
5. Aggarwal, V., Feldmann, A., Scheideler, C.: Can ISP'S and P2P Users Cooperate for Improved Performance?. SIGCOMM Comput. Commun. Rev. 37, 29–40 (2007)
6. Sen, S., Wang, J.: Analyzing Peer-to-Peer Traffic Across Large Networks. In: IMW 2002: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment, pp. 137–150. ACM, New York (2002)
7. Bindal, R., Cao, P., Chan, W., Medved, J., Suwala, G., Bates, T., Zhang, A.: Improving Traffic Locality in BitTorrent via Biased Neighbor Selection. In: 26th IEEE International Conference on Distributed Computing Systems, 2006. ICDCS 2006, pp. 66–77 (2006)
8. Choffnes, D.R., Bustamante, F.: Taming the Torrent: A Practical Approach to Reducing Cross-ISP Traffic in Peer-to-Peer Systems. In: SIGCOMM 2008, Seattle, Washington, USA, pp. 363–374. ACM, New York (2008)
9. Dorogovtsev, S.N., Goltsev, A.V., Mendes, J.F.F.: Critical Phenomena in Complex Networks. Reviews of Modern Physics 80 (2008)
10. Newman, M.E., Strogatz, S.H., Watts, D.J.: Random Graphs with Abitrary Degree Distributions and Their Applications. Phys. Rev. E Stat. Nonlin. Soft. Matter. Phys. 64 (2001)
11. Albert, R., Barabasi, A.-L.: Statistical Mechanics of Complex Networks. Reviews of Modern Physics 74, 47 (2002)
12. Karbhari, P., Ammar, M.H., Dhamdhere, A., Raj, H., Riley, G.F., Zegura, E.W.: Bootstrapping in Gnutella: A Measurement Study. In: Barakat, C., Pratt, I. (eds.) PAM 2004. LNCS, vol. 3015, pp. 22–32. Springer, Heidelberg (2004)
13. Stutzbach, D., Rejaie, R.: Capturing Accurate Snapshots of the Gnutella Networks. In: IEEE INFOCOM, pp. 2825–2830 (2005)
14. Stutzbach, D., Rejaie, R., Sen, S.: Characterizing Unstructured Overlay Topologies in Modern P2P File-Sharing Systems. In: Internet Measurment Conference, USENIX Association, pp. 49–62 (2005)
15. GwebCache System, `http://www.gnucleus.com`
16. Erdos, P., Renyi, A.: On Random Graphs I. Publ. Math. Debrecen 6, 290–297 (1959)
17. Erdos, P., Renyi, A.: On the Evolution of Random Graphs. Publ. Math. Inst. Hungar. Acad. Sci. 5, 17–61 (1960)
18. Barabasi, A.L., Albert, R.: Emergence of Scaling in Random Networks. Science 286, 509–512 (1999)