Peer-to-Peer Netw. Appl. (2011) 4:92–105
DOI 10.1007/s12083-010-0067-1

Ad-hoc limited scale-free models for unstructured peer-to-peer networks

Durgesh Rani Kumari · Hasan Guclu · Murat Yuksel

Received: 29 November 2008 / Accepted: 9 March 2010 / Published online: 9 May 2010

Abstract Several protocol efficiency metrics (e.g., scalability, search success rate, routing reachability and stability) depend on the capability of preserving structure even over the churn caused by the ad-hoc nodes joining or leaving the network. Preserving the structure becomes more prohibitive due to the distributed and potentially uncooperative nature of such networks, as in the peer-to-peer (P2P) networks. Thus, most practical solutions involve unstructured approaches while attempting to maintain the structure at various levels of protocol stack. The primary focus of this paper is to investigate construction and maintenance of scale-free topologies in a distributed manner without requiring global topology information at the time when nodes join or leave. We consider the *uncooperative behavior* of peers by limiting the number of neighbors to a pre-defined hard cutoff value (i.e., no peer is a major hub), and the *ad-hoc behavior* of peers by rewiring the neighbors of nodes leaving the network. We also investigate the effect of these hard cutoffs and rewiring of ad-hoc nodes on the P2P search efficiency.

Keywords Scale-free networks · Hard cutoff · Preferential attachment · Power law

1 Introduction

Stability and scalability of highly dynamic networks mainly depends on the capability of preserving structure even over the churn caused by the ad-hoc nodes joining or leaving the network. In decentralized peer-to-peer (P2P) networks, the overlay topology (or connectivity graph) among peers is a crucial component in addition to the peer/data organization and search. Topological characteristics have profound impact on the efficiency of search on P2P networks as well as other networks. It has been well-known that search on small-world topologies can be as efficient as $O(\ln N)$ [27], and this phenomenon has recently been studied on P2P networks [25, 26, 32, 40]. The best search efficiency in realistic networks can be achieved when the topology is scale-free (power-law), which offers search efficiencies like $O(\ln \ln N)$. Key limitation of scale-free topologies is the high load (i.e., high degree) on very few number of hub nodes. In a typical unstructured P2P network, peers are not willing to maintain high degrees/loads as they may not want to store large number of entries for construction of the overlay topology.

An earlier version of this work appeared in IEEE International Conference on Peer-to-Peer Computing (P2P), 2008.

D. R. Kumari (✉) · M. Yuksel
Computer Science & Engineering Dept.,
University of Nevada—Reno, Reno, NV 89557, USA
e-mail: dkumari@cse.unr.edu, dkumari@gtiemail.com

M. Yuksel
e-mail: yuksem@cse.unr.edu

H. Guclu
School of Mathematical Sciences,
Rochester Institute of Technology,
Rochester, NY 14623, USA
e-mail: guclu@rit.edu

H. Guclu
Computer Science & Engineering Department,
Istanbul Sehir University, Istanbul, Turkey
e-mail: gucluh@gmail.com

So, to achieve fairness and practicality among all peers, *hard* cutoffs on the number of entries are imposed by the individual peers, which makes the overall network a "limited" one. Effect of such hard cutoffs on search efficiency can be significant [24].

Due to the uncooperative nature of peers in a P2P network, protocols cannot completely rely on methods working with full cooperation of peers. For example, peers may not want to store large number of entries for construction of the overlay topology, i.e., connectivity graph. Even though characteristics of the overlay topology is crucial in determining the efficiency of the network, peers typically do not want to take the burden of storing excessive amount of control information for others in the network, thereby imposing *hard cutoffs* on the amount of control information to be stored. Yet another key issue is the construction of scale-free overlay topologies without global information. There are several techniques to generate a scale-free topology [1, 5], by using *global* information about the current network when a node joins or leaves. Such global methods are not practical in P2P networks, and *local* heuristics in generating such scale-free overlay topologies must be employed. In other words, there must be local and simple operations when peers are joining or leaving the P2P overlay, and also causing a minimal inefficiency to the search mechanisms to be run on the network.

The primary focus of this paper is to investigate construction and maintenance of scale-free topologies in a distributed manner without requiring global topology information at the time when nodes join or leave. We consider the *uncooperative behavior* of peers by limiting the number of neighbors to a pre-defined hard cutoff value, and the *ad-hoc behavior* of peers by rewiring the neighbors of nodes leaving the network. We also investigate the effect of these hard cutoffs and the rewiring of ad-hoc nodes on the P2P search efficiency.

The rest of the paper is organized as follows: First, we provide motivation for this work, outline the key parameters to be considered, and briefly state the major contributions and findings of the work. Then, we survey the previous work on P2P networks in Section 2. In Section 3, we survey the previous work on scale-free topology generation and briefly cover the importance of *cutoff* in the scale-free network and *Preferential Attachment (PA) with Hard Cutoffs*. We introduce our topology generation techniques using local heuristics and briefly describe the algorithm showing join and leave process of a node in the growing network, in Section 4. In Section 5, we present our simulation results of degree distribution of the nodes. We also discuss the efficiency of three search algorithms i.e Flooding (FL), Normalized Flooding (NF) and Random Walk (RW) on the topology generated by our simulations. We conclude by summarizing our current work and outlining the directions for the future work in Section 6.

1.1 Contributions and major results

Our work uncovers the relationship between the ad-hoc behavior of peers (i.e., how frequent they join/leave) and the efficiency of search over an overlay topology where each peer can (or is willing to) store a maximum number of links to other peers. In our model, we parameterize (i) the ad-hoc behavior of nodes by the probability that a node leaves μ, (ii) the amount of local information to be used at the time of *join* by knowledge radius from the point the node attempts to join, τ_j (i.e., the node knows about the local topology covering τ_j hops away from the point the node attempts to join the network), and (iii) the amount of local information to be used at the time of *leave* by knowledge radius from the location of the leaving node, τ_l (i.e., each neighbor of the leaving node knows about the local topology covering τ_l hops away from itself). We also define the maximum number of links to be stored by peers as the *hard cutoff*, k_c, for the degree of a peer in the network as compared to *natural cutoff* which occurs due to finite-size effects. Our contributions include:

- *Guidelines for generating scale-free topologies over ad-hoc nodes:* We introduce a generic model that can assign availability of different amount of local topology information at the times when a node joins or leaves. Our model provides a way of tuning churn (i.e., ad-hocness) of the network and studying how to balance state information for nodes joining or leaving.
- *Search efficiency on ad-hoc limited scale-free topologies:* Through extensive simulations, we studied efficiency of Flooding (FL), Normalized Flooding (NF), and Random Walk (RW) on the topologies generated by our model with different μ, τ_j, τ_l, and k_c values.
- *Rewiring methodologies for designing peer leave algorithms for unstructured P2P networks:* Our study yielded several guidelines for peers leaving an unstructured P2P network, so that the search performance of the overall overlay topology remains high.

Our study revealed several interesting issues. We found that having more global information about the topology at the time of leave is significantly more helpful than having it at the time of join. We show that the degree distribution can be kept scale-free and the search efficiency can be kept very high by simply keeping τ_l at reasonably high values, e.g., 2–3.

2 Related work

Previous work on P2P network protocols can be classified into *centralized* and *decentralized* ones. As centralized P2P protocols (e.g. Napster (http://www.napster.com)) proved to be unscalable, the majority of the P2P research has focused on decentralized schemes. The decentralized P2P schemes can be further classified into sub-categories: *structured*, *unstructured*, and *hybrid*.

In the structured P2P networks, data/file content of peers is organized based on a keying mechanism that can work in a distributed manner, e.g. Distributed Hash Tables (DHTs) [35]. The keying mechanism typically maps the peers (or their content) to a logical search space, which is then leveraged for performing efficient searches. In contrast to the structured schemes, unstructured P2P networks do not include a strict organization of peers or their content. Since there is no particular keying or organization of the content, the search techniques are typically based on flooding. Thus, the searches may take very long time for rare items, though popular items can be found very fast due to possible leveraging of locality of reference [34] and caching/replication [10].

The main focus of the research on unstructured P2P networks has been the tradeoff between state complexity of peers (i.e., number of records needed to be stored at each peer) and flooding-based search efficiency. The minimal state each peer has to maintain is the *list of neighbor peers*, which construct the overlay topology. Optionally, peers can maintain *forwarding table*s (also referred as routing tables in the literature) for data items in addition to the list of neighbor peers. Thus, we can classify unstructured P2P networks into two based on the type(s) of state peers maintain: (i) *per-data* unstructured P2P networks (i.e., peers maintain both the list of neighbor peers and the per-data forwarding table), and (ii) *non-per-data* unstructured P2P networks (i.e., peers maintain only the list of neighbor peers).

Non-per-data schemes are mainly Gnutella-like schemes (http://www.gnutella.wego.com), where search is performed by means of flooding query packets. Search performance over such P2P networks has been studied in various contexts, which includes pure random walks [19], probabilistic flooding techniques [20, 31], and systematic filtering techniques [38].

Per-data schemes (e.g. Freenet (http://freenetproject.org)) can achieve better search performances than non-per-data schemes, though they impose additional storage requirements to peers. By making the peers maintain a number of <key,pointer> entries peers direct the search queries to more appropriate neighbors, where “key” is an identifier for the data item being searched and the “pointer” is the next-best neighbor to reach that data item. This capability allows peers to leverage associativity characteristics of search queries [9]. Studies ranged from grouping peers of similar interests (i.e., peer associativity) [9, 26] to exploiting locality in search queries (i.e., query associativity) [8, 34]. Our work is applicable to both per-data and non-per-data unstructured P2P networks, since we focus on the interactions between search efficiency and topological characteristics.

Previous study on node isolation caused by churn in unstructured P2P networks introduced a general model of resilience [39]. In this study, joining and rewiring processes were based on age-biased neighbor selection, where a formal analysis included two age-biased techniques of neighbor selection. In maximal age-selection approach, the joining node selects uniformly randomly m alive nodes from the network and connects to the one with maximal age. It follows the same process when a dead link is detected. However, in age-biased random walk selection approach, the probability of a node to be selected by another peer is proportional to its current age. Another study introduced self-organizing super peer network architecture [18], where super peers maintain the cache with pointers to files that are recently requested and on the other hand client peers dynamically select super peers offering best search results.

3 Scale-free network topologies

Recent research shows that many natural and artificial systems such as the Internet [17], World Wide Web [2], scientific collaboration network [6], and e-mail network [16] have power-law degree (connectivity) distributions. These systems are commonly known as power-law or scale-free networks since their degree distributions are free of scale (i.e., not a function of the number of network nodes N) and follow power-law distributions over many orders of magnitude. This phenomenon has been represented by the probability of having nodes with k degrees as $P(k) \sim k^{-\gamma}$ where γ is usually between 2 and 3 [5]. Scale-free networks have many interesting properties such as high tolerance to random errors and attacks (yet low tolerance to attacks targeted to hubs) [3], high synchronizability [22, 23, 28], and resistance to congestion [36].

The origin of the scale-free behavior can be traced back to two mechanisms that are present in many systems, and have a strong impact on the final topology [5]. First, networks are developed by the addition of new nodes that are connected to those already present in the system. This mechanism signifies continuous expansion in real networks. Second, there is a higher probability that a new node is linked to a node that already has a large number of connections. These two features led to the formulation of a growing network model first proposed by Barabási and Albert that generates a scale-free network for which $P(k)$ follows a power law with $\gamma=3$. This model is known as *preferential attachment* (PA or rich-gets-richer mechanism) and the resulting network is called *Barabási-Albert* network [1, 5].

In this study, we use a simple version of the PA model [5]. The model evolves by one node at a time and this new node is connected to m (number of stubs) different existing nodes with probability proportional to their degrees, i.e., $P_i=k_i/\sum_j k_j$ where k_i is the degree of the node i. The average degree per node in the resulting network is $2m$ and the minimum degree is m.

Scale-free networks are very robust against random failures and attacks since the probability to hit the hub nodes (few nodes with very large degree) is very small and attacking the low-degree satellite nodes does not harm the network. On the other hand, deliberate attacks targeted to hubs through which most of the traffic go can easily shatter the network and severely damage the overall communication in the network. For the same reason the Internet is called “robust yet fragile” [15] or “Achilles’ heel” [3, 4].

Scale-free networks also have *small-world* [37] properties. In small-world networks the diameter, or the mean hop distance between the nodes scales with the system size (or the number of network nodes) N logarithmically, i.e., $d\sim\ln N$. The scale-free networks with $2<\gamma<3$ have a much smaller diameter and can be named *ultra-small* networks [11], behaving as $d\sim\ln\ln N$. When $\gamma=3$ and $m\geq 2$, d behaves as $d\sim\ln N/\ln\ln N$. However, when $m=1$ and $\gamma=3$ the Barabási–Albert model turns into a tree and $d\sim\ln N$ is obtained. Also when $\gamma>3$, the diameter behaves logarithmically as $d\sim\ln N$. Since the speed/efficiency of search algorithms strongly depend on the average shortest path, scale-free networks have much better performance in search than other random networks.

3.1 The cutoff

One of the important characteristics of scale-free networks is the natural cutoff on the degree (or the maximum degree) due to finite-size effects. Natural cutoff k_{nc} can be defined as [13] the value of the degree above which one expects to find at most one vertex, i.e.,

$$N\int_{k_{nc}}^{\infty} P(k)dk \sim 1 \ . \tag{1}$$

By using the degree distribution for the scale-free network and the exact form of probability distribution (i.e., $P(k)=(\gamma-1)m^{\gamma-1}/k^{\gamma}$), one obtains

$$k_{nc}(N) \sim mN^{1/(\gamma-1)}, \tag{2}$$

which is known as the *natural* cutoff of the network. The scaling of the natural cutoff can also be calculated by using the extreme-value theory [7]. For the scale-free networks generated by PA model ($\gamma=3$) the natural cutoff becomes

$$k_{nc}(N) \sim m\sqrt{N}. \tag{3}$$

3.2 Preferential attachment with hard cutoffs

The natural cutoff may not be always attainable for most of the scale-free networks due to technical reasons. One main reason is that the network might have limitations on the number of links the nodes can have. This is especially important for P2P networks in which nodes can not possibly connect many other nodes. This requires putting an artificial or *hard* cutoff k_c to the number of links one node might have.

In order to implement the hard cutoff in PA, we simply did not allow nodes to have links more than a fixed hard cutoff value during the attachment process. This modified method generates a scale-free network in which there are many nodes with degree fixed to hard cutoff instead of a few very high degree hubs and the degree distribution still decays in a power law fashion. The degree distribution of PA model with cutoff is slightly different than that of PA without a cutoff in terms of exponent and an accumulation of nodes with degree equal to hard cutoff. PA model, in its original form, has a degree distribution exponent $\gamma=3$ for very large networks. However, when a hard cutoff is imposed it is observed that the absolute value of degree distribution exponent decreases [24].

One can use the master-equation [29] approach to analyze the effects of the hard cutoff on the topological characteristics. We grow the network by introducing new nodes one by one for simplicity. Each new node links to m earlier nodes in the network. The probability that the new node attaches to a previous node of degree k is defined to be A_k/A, where A_k is the rate of attachment to a previous node and this rate depends only on the degree of the target node, while $A=\sum_{k=m}^{k_c-1} A_k N_k$ is the total rate for all events, and N_k is the number of

Algorithm 1 Network growth using paramaterized join and leave processes

//Global Variables and Functions
m—minimum degree
μ—probability of a node to leave the network
N—the maximum node ID of the existing network (the minimum node ID is 0)
G—graph of the existing network of M links and N nodes
PreferentialAttachment(G_1, G_2)—*a function that performs Preferential Attachment to G_1 by using the nodes in G_2, returns the number of successful new links*

// Join process of node i
void **Join**(i, τ_j)
N++
$numoflinks \leftarrow 0$
while $numoflinks < m$ **do**
  $N_{rand} \leftarrow$ Randomize(1,N) {Pick a random node from the existing network}
  $myG \leftarrow get\_subgraph(N_{rand}, \tau_j)$ {Get the subgraph including neighbor nodes of N_{rand} up to τ_j hops away}
  $numoflinks$ $+=$ PreferentialAttachment(G, myG)
end while

//Leave process of node i
void **Leave**(i, τ_l)
$myG \leftarrow get\_subgraph(N_{rand}, \tau_l)$ {Get the subgraph including neighbor nodes of N_{rand} up to τ_l hops away}
$remove(N_{rand})$ {Delete N_{rand} from the existing network}
$N = N - 1$
PreferentialAttachment(G,myG)

// Growth process of a network with N_{target} nodes, parameterized with τ_j and τ_l
void **Grow**(N_{target}, τ_j, τ_l)
for i=m+1; i<N_{target}; i++ **do**
  Join(N,τ_j)
  $num \leftarrow$ Random(0,1)
  if $N == N_{target}$ **then**
    break;
  end if
  if $num < \mu$ **then**
    $N_{del} \leftarrow$ Randomize(1,N)
    Leave(N_{del}, τ_l)
  end if
end for

nodes of degree k in the network. Thus A_k/A equals to the probability for the newly-introduced node to attach to a node of degree k. The new feature that we study is the effect of a hard cutoff on the degree of each node. Once the degree of a node reaches k_c, it is defined to become inert so that no further attachment to this node can occur. Thus only nodes with degrees $k = m, m+1, ..., k_c - 1$ are active. This restriction is the source of the cutoff in the definition of the total attachment rate. We now study the degree distribution, $N_k(N)$, as a function of the cutoff k_c and the total number of nodes in the network N.

The master equations for the degree distribution can be written by using the fact that N_k is proportional to N, and thus $N_k \to Nn_k$ as well as $A \to \nu N$ as

$$n_k = \begin{cases} -\dfrac{mn_m}{\nu} + 1 & k = m \\ \dfrac{(k-1)n_{k-1} - kn_k}{\nu} & k = m+1, ..., k_c - 1 \\ \dfrac{(k_c - 1)n_{k_c-1}}{\nu} & k = k_c \end{cases} . \quad (4)$$

By the nature of these equations, it is evident that n_{k_c} is of a different order than n_k with $k < k_c$. Starting with the solution $n_m = \nu/(m+\nu)$, we can find n_k by subsequent substitutions. This recursive approach gives us a chance to write n_k values as products [29] and by converting these products into Euler gamma functions we show that n_{k_c} scales as $k^{-\nu}$, while for $k < k_c$, n_k scales as $k^{-(\nu+1)}$. We can obtain the coefficient ν in $A = \nu N$ self consistently from $A = \sum_{k=m}^{k_c-1} A_k n_k \equiv \nu N$, or equivalently, $\nu = \sum_{k=m}^{k_c-1} A_k n_k$. By rewriting the sum above as a difference between two sums with limits from the minimum degree to ∞ and from cutoff to ∞ and by taking asymptotic limits [21] of large N and k_c we get

$$\nu \to 2 - \frac{2m}{k_c} . \quad (5)$$

This result shows that $n_k \sim k^{-(3-2m/k_c)}$ for $k < k_c$ and $n_{k_c} \sim k_c^{-(2-2m/k_c)}$ confirming the change in the degree distribution exponent [24]. This implies that any finite hard cutoff value decreases the degree distribution exponent, i.e., it makes the degree distribution flatter. A better search efficiency observed for a smaller cutoff can be explained by the increase in the degree distribution exponent [24].

3.3 Network dynamics

Ad-hoc scale-free networks have recently attracted considerable attention in the literature mainly because of its most-desired property of robustness to random attacks or failures. For example it was shown that [3] the diameter of the Internet at the autonomous system level, which is the most famous example of scale-free networks, would not be changed considerably if up to 2.5% of the routers were removed randomly. This is an order of magnitude larger than the failure rate. It was also shown in [3] that for a scale-free network of size 10,000 and a failure rate of 18%, the biggest connected component holds 8,000 nodes, whereas under the same conditions a random network can survive this failures by the biggest connected component of size 100.

Many models for ad-hoc scale-free networks in which the edges can appear and disappear [12, 14, 30] or some nodes are removed [33] have been studied. In the first set of studies as the nodes are joining to the network some links among the pre-existing nodes are rewired or moved randomly by some probability parameter. Depending on the parameters such models exhibit either exponential or power-law degree distributions. In [33], as the nodes are joining by preferential attachment some randomly selected nodes are deleted (i.e., left the network) along with their links from the network. If the nodes whose neighbors are deleted do not reconnect themselves to other nodes, it is observed that the degree distribution is a power law with an exponent ranging from 3 to infinity depending on the deletion probability, i.e., the probability that a node will leave the network. The authors proposed a remedy for the deletion that the neighbors of the deleted nodes select some other nodes in the network and connect by again using preferential attachment rules. In this case the degree distribution is still a power law but the exponent changes from 3 to 2 as the deletion probability goes from 0 to 1.

The main disadvantage of the ad-hoc scale-free models in the literature is that they lack localized algorithmic solutions. All requires global information to be available to nodes so that they can reconnect to randomly selected nodes in the network. For this reason we grow scale-free networks with local heuristics only to simulate the real-life situation in unstructured peer-to-peer systems. To parameterize our model we use two different time-to-live variables: τ_j and τ_l to describe the number of nodes available to a new node and to a neighbor of a deleted node, respectively. In the next section we explain our model and its parameters in detail.

4 Growing scale-free topologies with local heuristics

In the PA model and its ad-hoc variants as outlined in the previous section, the new node or the neighbor of a deleted node has to make random attempts to connect to the existing nodes with a probability depending on the degree of the existing nodes. To implement this in a P2P (or any distributed) environment, the new node has to have information about the global topology (e.g. the current number of degrees each node has for the PA model), which might be very hard to maintain in reality. Thus, in order for a topology construction mechanism to be practical in P2P networks, it must allow joining or rewiring of the nodes by just using locally available information. Of course, the cost of using only local information is expected to be loss of scale-freeness (or any other desired characteristics) of the whole overlay topology, which will result in loss of search efficiency in return. In this section, we present a practical method using local heuristics and no global information about the topology. This model imitates the method for finding peers in Gnutella-like unstructured P2P networks.

Our model considers a substrate network of $m+1$ fully connected nodes to start with. The possible values for m can be from 1 to 3. Algorithm 1 in the paper presents the pseudo code in order to explain the growth of the overlay network. At each time step, a new node i is being added to the existing network by calling function Join(). The node i first connects to a randomly selected node from the existing network and then it targets to connect to $m-1$ more nodes. Each of the $m-1$ nodes is randomly selected from the nodes in the radius of τ_j of its latest neighbor. Slowly the network grows from $m+1$ nodes to the targeted number of nodes (N_{target}). At every time step a new node with m possible links is added to the network and one randomly chosen node is deleted with probability μ. Since the nodes are short-sighted, i.e., they do not have global information about the network, they can only choose from a subset of the network (*horizon*) they construct

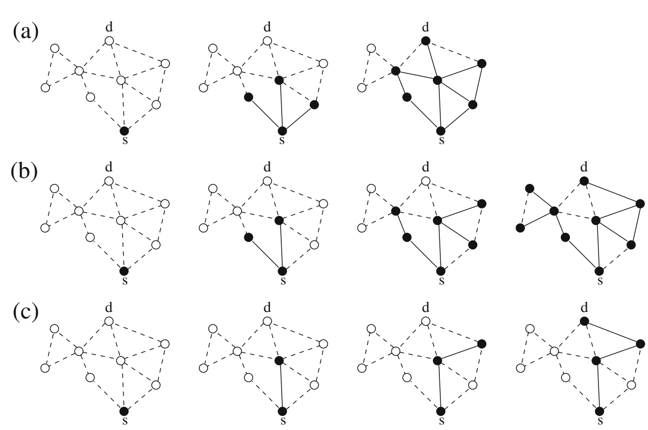


Fig. 1 Search strategies: **a** Flooding **b** Normalized flooding **c** Random walk

Table 1 Parameters of our topology construction framework

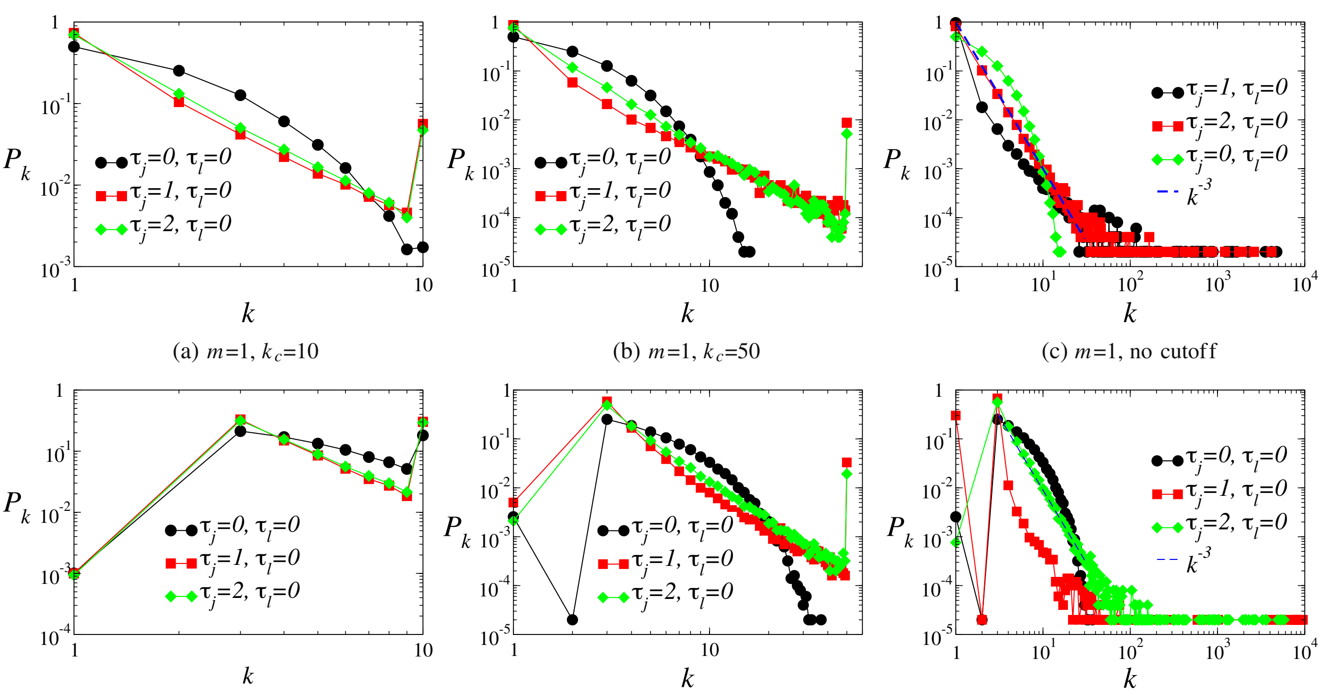
Symbol	Parameter description	Range
μ	Churn of the nodes	[0,1)
τ_j	Available information at join	≥ 0
τ_l	Available information at leave	≥ 0
k_c	Hard cutoff	≥ 1
m	Minimum degree (# of stubs)	≥ 1

instead of the whole network. The parameter τ_j and τ_l are the TTL values used by the nodes and denote the measure of locality in joining and leaving, respectively. A newly added node, first, select a random existing node and construct a set of nodes reachable in τ_j hops or less from that node. Then, this new node randomly selects a node from this set and connects itself with probability proportional to its degree. This probability is normalized by the total degree of the nodes in the set. The new node randomly selects other nodes in the set until its degree reaches m. If no node is left in the set to connect but the degree of the new node is less than m, it selects another random node from the network and continue this process. In the deletion case, the neighbors of the deleted node selects a node randomly from a set of nodes reachable in τ_l or less steps from the deleted node and connect by using the preferential attachment rule. Here, in both cases nodes cannot connect to other nodes with degrees equal to the hard cutoff.

There are special cases in this model: i) when $\tau_j = 0$, the horizon of the new nodes contain only the randomly selected nodes and the preferential attachment rule is invalid. In this case the new node connect to this single node in the horizon if its degree is less than the hard cutoff. ii) when $\tau_l = 0$ the neighbors of the deleted node do not have any node in their horizons so no rewiring occurs. These nodes just loses one of their links and they do nothing to compensate it. The model typically becomes the preferential attachment with global information when τ_j value is large and τ_l is zero and a BA network with $\gamma = 3$ is obtained.

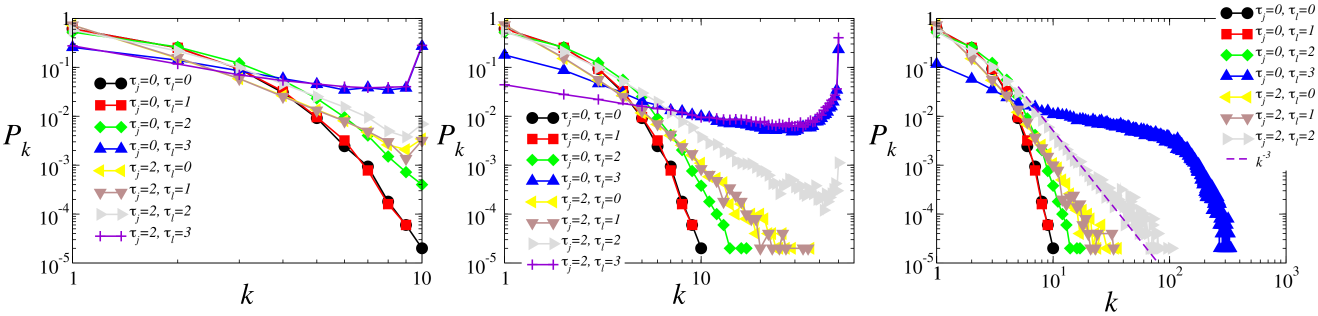
5 Simulations

In the previous sections, we introduced a framework to investigate the effects of join and leave processes in terms of scale-freeness of the topology being constructed within the context of ad-hoc unstructured P2P networks. Here, we study a number of message-passing algorithms that can be efficiently used to search items in P2P networks utilizing the scale-free degree distribution in sample networks generated by our topology construction algorithms. These search algorithms are completely decentralized and do not use any kind of global knowledge about the network. We consider three different search algorithms: *flooding* (FL), *normalized flooding* (NF), and *random walk* (RW).



(a) m=1, k_c=10 (b) m=1, k_c=50 (c) m=1, no cutoff
(d) m=3, k_c=10 (e) m=3, k_c=50 (f) m=3, no cutoff

Fig. 2 Degree distributions when there is no churn (i.e., μ= 0): $P(k)$ for various networks generated by our framework for varying τ_j



(a) m=1, k_c=10 (b) m=1, k_c=50 (c) m=1, no cutoff

Fig. 3 Degree distributions over ad-hoc nodes (i.e., μ=0.3): $P(k)$ for various networks generated by our framework for varying τ_j and τ_l

Goals of our simulation experiments include:

- *Effect of churn on the search efficiency in an uncooperative environment with hard cutoffs:* Churn of nodes joining or leaving the network affects the search efficiency, i.e., *number of hits per unit time*. Further, applying hard cutoffs on such ad-hoc scale-free topologies reduces the degree distribution exponent. We are interested in observing the effect of this churn and hard cutoffs on the search efficiency for three search algorithms, i.e., FL, NF, and RW. This extends our previous work in [24], which focussed on the effect of hard cutoffs *only*.
- *Ad-hoc scale-free topology construction with global vs. local information:* Though we showed in the previous section that using local information when a peer is joining yields a less scale-free topology, the effect of this on search efficiency still needs to be shed light on. Our simulations aim to investigate this too.

5.1 Search algorithms

We use three search techniques to evaluate our ad-hoc scale-free topologies:

Flooding (FL) FL is the most common search algorithm in unstructured P2P networks. In search by FL, the source node s, sends a message to all its nearest neighbors. If the neighbors do not have the requested item, they send on to their nearest neighbors excluding the source node [see Fig. 1a]. This process is repeated a certain number of times, which is usually called *time-to-live (TTL)*.

Normalized Flooding (NF) In NF, the minimum degree m in the network is an important factor. NF search algorithm proceeds as follows: When a node of degree m receives a message, the node forwards the message to all of its neighbors excluding the node forwarded the message in the previous step. When a node with larger degree receives the message, it forwards the message only to randomly chosen m of its neighbors except the one which forwarded the message. The NF mechanism is illustrated in Fig. 1b. In this simple network with $m = 2$, the source node sends a message to its randomly chosen two neighbors and these neighbors forward the message to their randomly chosen two neighbors. In the third step, the message reaches its destination.

Random Walk (RW) RW or multiple RWs have been used as an alternative search algorithm to achieve even better granularity than NF. In RW, the message from the source node is sent to a randomly chosen neighbor. Then, this random neighbor takes the message and sends it to randomly selected one of its random neighbors excluding the node from which it got the message. This continues until the destination node is reached or the total number of hops is equal to TTL. A schematic of RW can be seen in Fig. 1c. RW can also

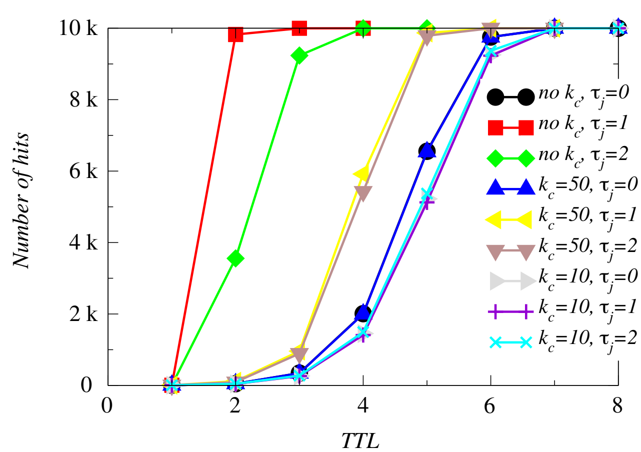


Fig. 4 Flooding (FL) performance over topologies with m=3 and no churn

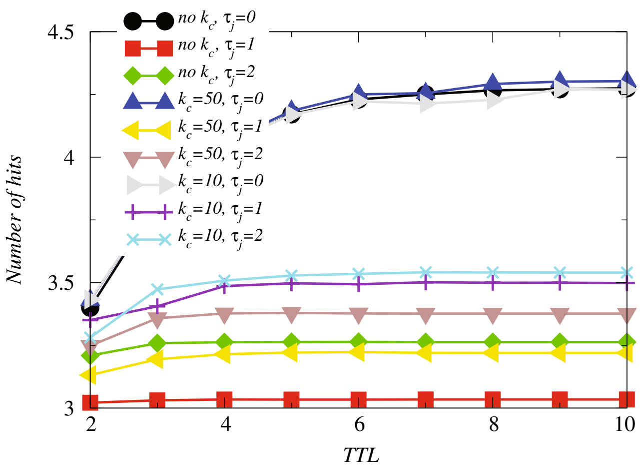


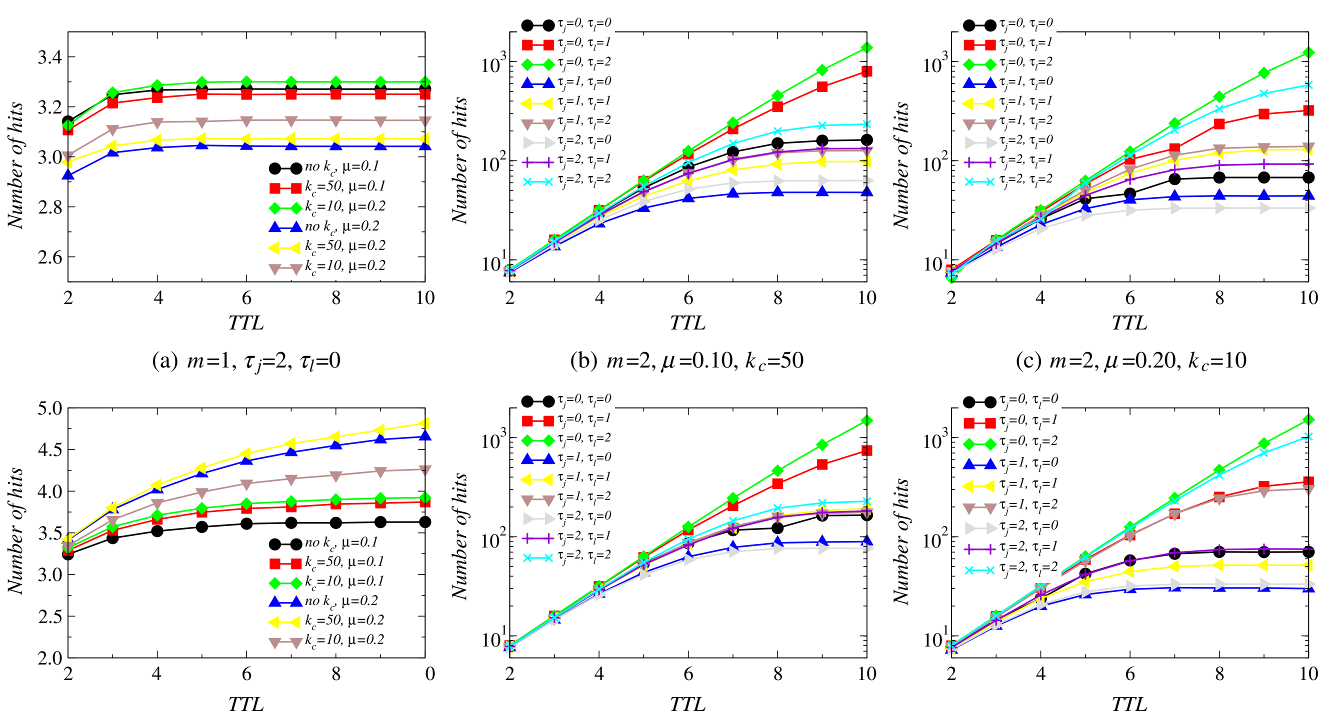
Fig. 5 Flooding (FL) performance over topologies generated with k_c=10 and m=3

be seen as a special case of FL where only one neighbor is forwarded the search query, providing the other extreme situation of the tradeoff between delivery time and messaging complexity.

5.2 Results

We simulated the three search algorithms FL, NF, and RW on the topologies generated by our framework with three different parameters: (i) churn, $0 < \mu < 1$, (ii) available information during join, $\tau_j \geq 0$, (iii) available information during leave, $\tau_l \geq 0$, (iv) hard cutoff, $k_c > 1$, and (v) minimum degree (number of stubs), m. These parameters are listed in Table 1 as well. By assigning different values to each of these parameters, we generated topologies with 10,000 nodes. We used different k_c values from 10 to 100 (or just a few in this range), in addition to the natural cutoff, i.e., no hard cutoff. We varied τ_j and τ_l from 0 to 3. Minimum degree values (or m) in our topologies were 1, 2, or 3. We studied smaller values of μ from 0 to 0.3, reflecting no churn to 30% churn, respectively. We performed 5 realizations of our results.

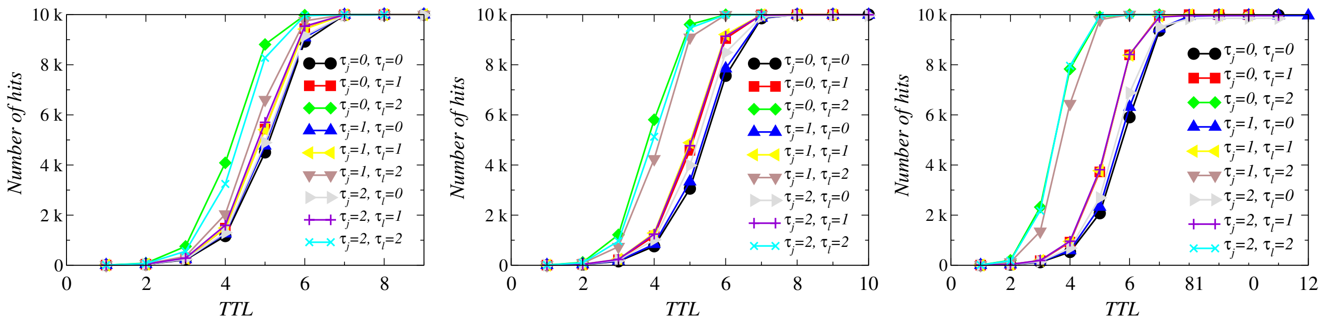
We varied the TTL (i.e., time-to-live) values of search queries in FL and NF to the point we reach the system size. To compare search efficiencies of RW and NF fairly, we equated TTL of RW searches to the number of messages incurred by the NF searches in the same scenario. Thus, for the search efficiency graphs of RW when TTL is equal to a particular value such as 4, this means that the number of hits corresponding to that TTL=4 value is obtained by simulating a RW search with TTL equal to the number of messages that were caused by an NF search using a TTL value of 4. A similar normalization was done in [20].



(a) m=1, τ_j=2, τ_l=0 (b) m=2, μ=0.10, k_c=50 (c) m=2, μ=0.20, k_c=10

(d) m=1, τ_j=2, τ_l=2 (e) m=2, μ=0.10, k_c=10 (f) m=2, μ=0.20, k_c=50

Fig. 6 Normalized Flooding (NF) performance over topologies generated with various m, τ_j, and τ_l values



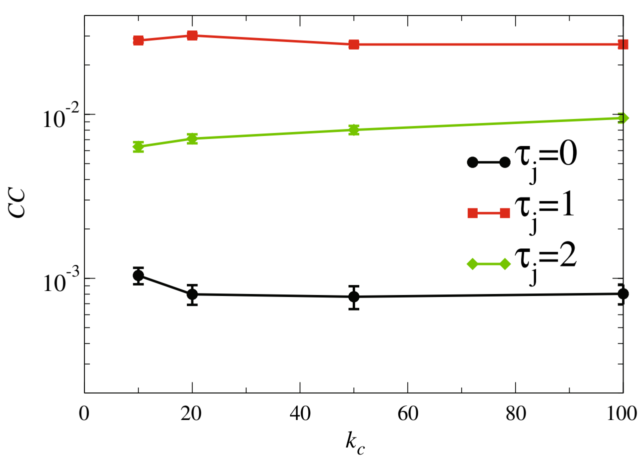
(a) μ=0.1 (b) μ=0.2 (c) μ=0.3

Fig. 7 Performance of random walk over topologies with m =1 and no churn

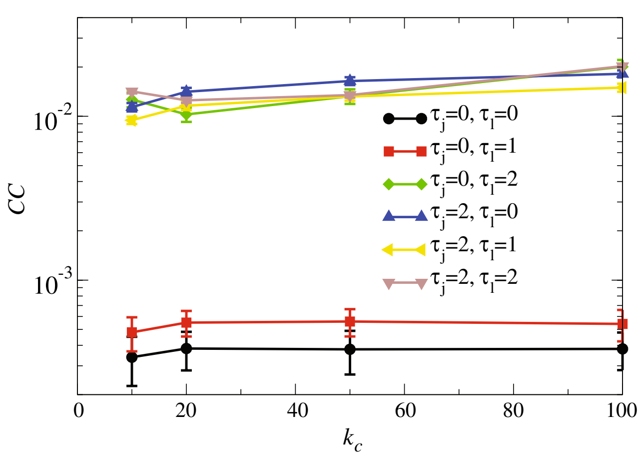
5.2.1 Effects on degree distribution

Our simulation results show the effect of churn and hard cutoff on the degree distribution of the topologies. Figure 2 shows the degree distribution of the topologies when there is no churn, i.e., μ=0. Similarly, Fig. 3 shows the degree distributions when the nodes are ad-hoc with μ=0.3. It is known that using more global information (i.e., knowing more of the network topology) helps to generate better scale-free topologies with lower power-law exponent. This phenomenon is clearly shown in Fig. 2, i.e., the degree distribution shifts from an Exponential one to a power-law one as τ_j increases from 0 to 2 (i.e., the joining node uses the topology information at a larger horizon). This is true for both m=3 and m=1, though larger m makes the shift a little less apparent. Further, the hard cutoff, k_c, only affects this process by simply bounding the very large hubs to the cutoff without affecting the transition from Exponential to power-law.

An interesting result being revealed in Fig. 3 is that τ_l has much more significant effect than τ_j in shifting the degree distribution from Exponential to power-law. This is even more apparent for smaller values of the cutoff.



(a) μ = 0 and m = 3



(b) μ = 0.3 and m = 3

Fig. 8 Random Walk (RW) performance over topologies generated with various m, τ_j, τ_l, and k_c values

5.2.2 Effects on search efficiency

In flooding by far the most important parameter when there is no deletion in the network is the cutoff which determines the number of distinct nodes one can reach from a node, see Fig. 4. In this case, τ_j is also an important parameter which changes the network from an exponential to a scale-free one and give better efficiency in flooding.

Our simulations also show that this effect can be relieved by increasing the minimum degree in the network as it can be seen in Fig. 5. More interestingly, churn plays an important role in the efficiency of search algorithms. Negative effect of the high churn (high μ) can be eliminated by increasing the available information in rewiring, i.e., by increasing τ_l in both flooding and normalized flooding, see Fig. 6. In some cases in normalized flooding higher churn yields better efficiency for enough high values of τ_l. Figures 7 and 8, describe the behavior of Random Walk search algorithm on different network topologies. We observed that qualitatively random walk shows a similar behavior as normalized flooding except that random walk is more vulnerable to isolated clusters in the network. Both random walk and normalized flooding results show that the best performance is obtained when the available information in leaving is high ($\tau_l = 2$) but low in joining ($\tau_j = 0$). This demonstrates an interesting finding that using more information for rejoining of the nodes of a deleted node is more important than that in joining new nodes. We also observe that the for a fixed τ_l increasing τ_j does not increase search efficiency but decreases. We believe the mechanism of rewiring after node deletion works in a peculiar way. It makes the network better connected in terms of search performance if the new nodes are not using too much global information in joining. We also believe this should manifest itself when put in a theoretical framework combining the structure and search performance.

5.2.3 Effects on clustering coefficient

In this part, we discuss about the effects of cutoff and churn on clustering coefficient. The clustering, also known as *transitivity* is the measure of how tightly connected the neighbors of a node are. In other words transitivity means the presence of a high number of triangles in the network. An alternative definition of clustering coefficient which we also use in this paper is defined as follows [37]. Suppose c_i is the local clustering coefficient of the node i. Its value is obtained by counting the number of edges (denoted by e_i) among the neighbors of the node i and dividing this number to the maximum possible number of edges among the neighbors, i.e., $k_i(k_i - 1)/2$, where k_i is the degree of the node i ($c_i = 0$ for $k_i = 0$ or 1):

$$c_i = \frac{2e_i}{k_i(k_i - 1)} \tag{6}$$

The clustering coefficient of the network is calculated by taking the average of the local clustering coefficients of all the nodes in the network:

$$C = \langle c \rangle = \frac{1}{N} \sum_i c_i \tag{7}$$

By definition, $0 \leqslant c_i \leqslant 1$ and $0 \leqslant C \leqslant 1$.

Cutoffs have a noticeable effect on clustering coefficient. Figure 9 illustrates this behavior very well. We have considered two cases to describe this characteristic (a) m=3, μ=0 and (b) m=3, μ=0.30. For the first case there is no effect of τ_l as μ=0 therefore we observed the behavior of clustering coefficient on

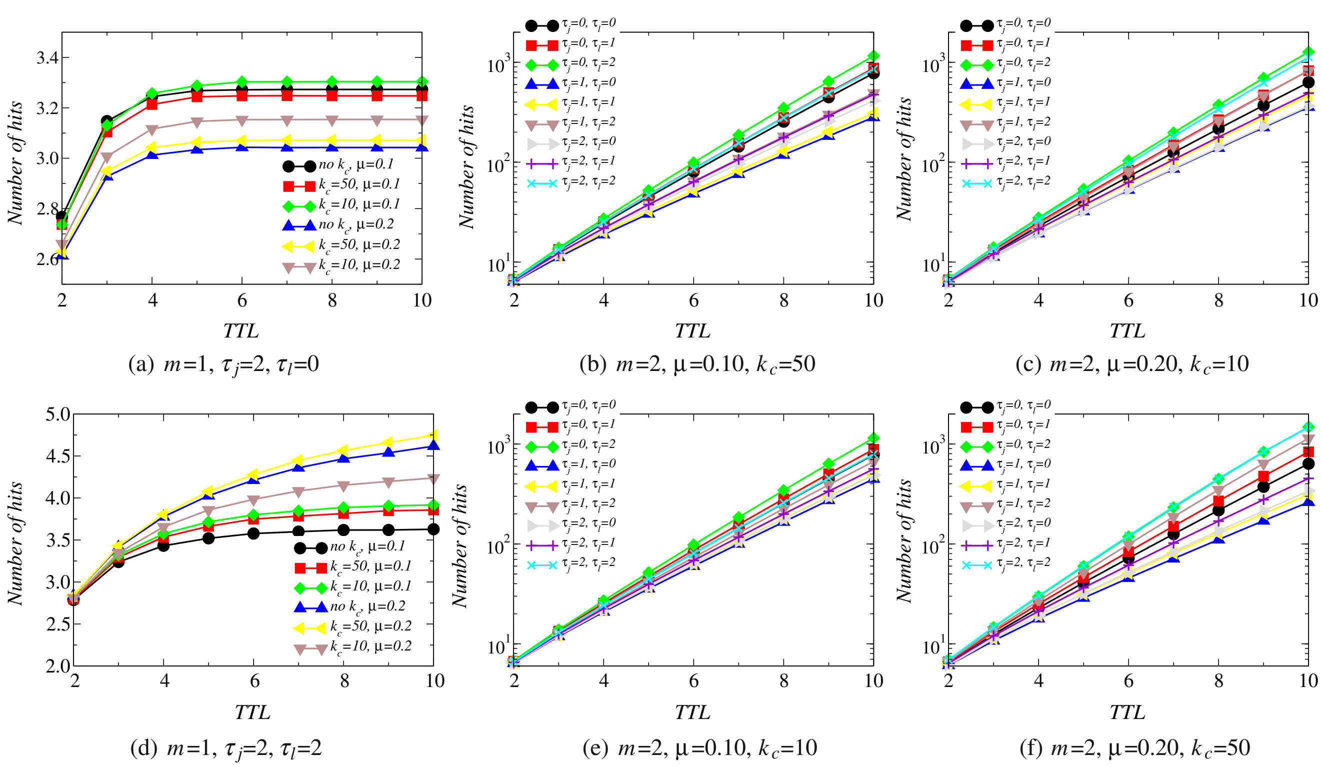


Fig. 9 Clustering coefficient over cutoffs

increasing value of cutoffs on different values of τ_j. However, for second case we observed this trend on different values of τ_j and τ_l. It's interesting to find out that increasing cutoffs, increased the values of clustering coefficients in most of the cases. After analyzing both the graphs it can be stated that the clustering coefficient of the network increases with the increase in the cutoff. This relationship between hard cutoffs and the clustering coefficient can be attributed to the fact that rewiring process gets limited in being able to connect to close by (i.e., within τ_l hops) peers. We also observe that the clustering coefficient is very low, if not zero, for $\tau_j = 0$ and $\tau_l = 0, 1$, and increasing τ_l to 2 increases it considerably.

6 Summary and discussions

In summary, we worked on an ad-hoc limited scale-free network model for unstructured peer-to-peer networks. We first developed localized joining and leaving schemes for the peers and measure the efficiency of search algorithms such as flooding and normalized flooding. By considering the fact that the peers do not want to store too many links information we also imposed a hard cutoff on the degree a node can have and analyzed its effect on the search efficiency. We parameterized the locality of the joining and leaving schemes by two parameters: τ_j (for joining) and τ_l (for leaving) which are the number of hops nodes will use to construct sets of nodes from which they will randomly choose other nodes and attempt to connect by using the preferential attachment rules and by observing the hard cutoff. Typically, high values of these parameters will make the network a preferential attachment network with degree distribution exponent 3. We also modeled the random deletion (i.e., churn) of the nodes by a probability parameter.

Our search simulations show that the negative effects of the low cutoff and high probability of deletion can be eased by increasing the minimum degree in the network. This also helps one to avoid the pathological case of minimum degree being 1, for which the network will likely to have isolated clusters hindering the efficiency of the search algorithms. To remedy the negative influence of high leave rate which destroys the scale-freeness in the network we enlarged the locality of the leaving scheme, i.e., increasing τ_l for a fixed τ_j and cutoff will increase the efficiency of normalized flooding. Our findings are directly applicable to current unstructured P2P networks in which the peers leave the network unexpectedly and they have an upper limit for degree.

Acknowledgements This work was supported in part by National Science Foundation under awards 0627039 and 0721542. Dr. Guclu was supported by the Rochester Institute of Technology seed grant. Authors would like to thank Sid Redner for fruitful discussions.

References

1. Albert R, Barabási A-L (2000) Topology of evolving networks: local events and universality. Phys Rev Lett 85:5234
2. Albert R, Jeong H, Barabási A-L (1999) Diameter of the World Wide Web. Nature 401:130
3. Albert R, Jeong H, Barabási A-L (2000) Error and attack tolerance of complex networks. Nature 406:378
4. Albert R, Jeong H, Barabási A-L (2000) Error and attack tolerance of complex networks (correction). Nature 409:542
5. Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286:509
6. Barabási A-L, Jeong H, Neda Z, Ravasz E, Schubert A, Vicsek T (2002) Evolution of the social network of scientific collaborations. Physica A 311:590
7. Boguná M, Pastor-Satorras R, Vespignani A (2004) Cut-offs and finite size effects in scale free networks. Eur Phys J B 38:205
8. Chawathe Y, Ratnasamy S, Breslau L, Lanham N, Shenker S (2003) Making Gnutella-like systems scalable. In: Proceedings of ACM SIGCOMM
9. Cohen E, Fiat A, Kaplan H (2003) Associative search in peer-to-peer networks: harnessing latent semantics. In: Proceedings of conference on computer communications (INFOCOM)
10. Cohen E, Shenker S (2002) Replication strategies in unstructured peer-to-peer networks. In: Proceedings of ACM SIGCOMM
11. Cohen R, Havlin S (2003) Scale-free networks are ultrasmall. Phys Rev Lett 90:058701
12. Dorogovtsev S, Mendes J (2000) Scaling behavior of developing and decaying networks. Europhys Lett 52:33–39
13. Dorogovtsev S, Mendes J (2002) Evolution of networks. Adv Phys 51:1079
14. Dorogovtsev S, Mendes J, Samukhin A (2001) Anomalous percolation properties of growing networks. Phys Rev E 64:066110
15. Doyle JC, Alderson DL, Li L, Low S, Roughan M, Shalunov S, Tanaka R, Willinger W (2005) The 'robust yet fragile' nature of the Internet. Proc Natl Acad Sci 102:14497
16. Ebel H, Mielsch M-I, Bornholdt S (2002) Scale-free topology of e-mail networks. Phys Rev E 66:035103(R)
17. Faloutsos M, Faloutsos P, Faloutsos C (1999) On power-law relationships of the Internet topology. ACM Comput Commun Rev 29:251
18. Garbacki P, Epema DHJ, van Steen M (2007) Optimizing peer relationships in a super-peer network. In: Proceedings of IEEE international conference on distributed computing systems (ICDCS)
19. Gkantsidis C, Mihail M, Saberi A (2003) Random walks in peer-to-peer networks. In: Proceedings of conference on computer communications (INFOCOM)

20. Gkantsidis C, Mihail M, Saberi A (2005) Hybrid search schemes for unstructured peer-to-peer networks. In: Proceedings of conference on computer communications (INFOCOM)
21. Graham RL, Knuth DE, Patashnik O (1994) Concrete mathematics: a foundation for computer science, 2nd edn. Addison-Wesley, Reading
22. Guclu H (2005) Synchronization landscapes in small-world-connected computer networks. Ph.D. thesis, Rensselaer Polytechnic Institute, arXiv:cond-mat/0601278
23. Guclu H, Korniss G, Toroczkai Z (2007) Extreme fluctuations in noisy task-completion landscapes on scale-free networks. Chaos 17:026104
24. Guclu H, Yuksel M (2007) Scale-free overlay topologies with hard cutoffs for unstructured peer-to-peer networks. In: Proceedings of IEEE ICDCS
25. Hui K, Lui J, Yau D (2006) Small-world overlay p2p networks: Constructing and handling of dynamic flash crowd. Comput Networks 50:2727
26. Iamnitchi A, Ripeanu M, Foster I (2004) Small-world file-sharing communities. In: Proceedings of conference on computer communications (INFOCOM)
27. Kleinberg J (2000) Navigation in a small world. Nature 406:845
28. Korniss G (2007) Synchronization in weighted uncorrelated complex networks in a noisy environment: optimization and connections with transport efficiency. Phys Rev E 75:051121
29. Krapivsky P, Redner S (2001) Organization of growing random networks. Phys Rev E 63:066123
30. Krapivsky P, Redner S (2002) A statistical physics perspective on web growth. Comput Networks 39:261
31. Kumar A, Xu J, Zegura EW (2005) Efficient and scalable query routing for unstructured peer-to-peer networks. In: Proceedings of conference on computer communications (INFOCOM)
32. Merugu S, Srinivasan S, Zegura E (2005) Adding structure to unstructured peer-to-peer networks: the use of small-world graphs. J Parallel Distrib Comput 65:142
33. Sarshar N, Roychowdhury V (2004) Scale-free and stable structures in complex ad hoc networks. Phys Rev E 69: 026101
34. Sripanidkulchai K, Maggs B, Zhang H (2003) Efficient content location using interest-based locality in peer-to-peer systems. In: Proceedings of conference on computer communications (INFOCOM)
35. Stoica I, Morris R, Karger D, Kaashoek H, Balakrishnan H (2001) Chord: a scalable peer-to-peer lookup service for internet applications. In: Proceedings of ACM SIGCOMM
36. Toroczkai Z, Bassler K (2004) Jamming is limited in scale-free systems. Nature 428:716
37. Watts D, Strogatz S (1998) Collective dynamics of 'small-world' networks. Nature 393:440
38. Xu J (2003) On the fundamental tradeoffs between routing table size and network diameter in peer-to-peer networks. In: Proceedings of conference on computer communications (INFOCOM)
39. Yao Z, Wang X, Leonard D, Loguinov D (2007) On node isolation under churn in unstructured p2p networks with heavy-tailed lifetimes. In: Proceedings of IEEE international conference on computer communication
40. Zhang H, Goel A, Govindan R (2002) Using the small-world model improve Freenet performance. In: Proceedings of conference on computer communications (INFOCOM)

Durgesh Rani Kumari is a Software Developer and Database Administrator at GameTech International—Reno. She has received her M.S. degree in Computer Science from University of Nevada, Reno in 2009. She received her Bachelors degree from University of Pune, India in Computer Engineering in 2005. She has more than 3 years of experience in the field of Computer Networks and Software Development. Her research interest lies in various networking topics such as peer-to-peer networking and multicasting.

Hasan Guclu received B.S. and M.S. degrees in physics from Middle East Technical University, Ankara, Turkey, in 1998 and 2001, respectively, and the Ph.D. degree in physics from Rensselaer Polytechnic Institute, Troy, NY, in 2005. He is an assistant professor at the School of Mathematical Sciences, Rochester Institute of Technology, Rochester, NY, and concurrently an associate professor on leave from the Department of Computer Science and Engineering, Istanbul Sehir University, Istanbul, Turkey. Formerly, he was a postdoctoral fellow at the Complex Systems Group of Los Alamos National Laboratory, Los Alamos, NM. His research is on complex networks and their applications in peer-to-peer/sensor networks, computational epidemiology, and parallel and distributed computation systems. He is a member of IEEE, APS, AAAS, Sigma Xi and Sigma Pi Sigma.

Murat Yuksel is an Assistant Professor at the University of Nevada—Reno. He received M.S. and Ph.D. degrees in Computer Science from Rensselaer Polytechnic Institute in 1999 and 2002 respectively. He holds a B.S. degree in Computer Engineering from Ege University, Izmir, Turkey. His research is on various networking issues such as wireless routing, free-space-optical mobile ad-hoc networks (FSO-MANETs), network modeling and economics, peer-to-peer, protocol design, and performance analysis. He is a member of IEEE, ACM, and Sigma Xi.