A study on the redundancy of flooding in unstructured p2p networks

Spiridoula V. Margariti
Department of Information and Telecommunication Technology
T.E.I. of Epirus, Arta, Greece, GR-47100
and
Department of Computer Science
University of Ioannina, Ioannina, Greece, GR-45110
smargari@cs.uoi.gr

Vassilios V. Dimakopoulos
Department of Computer Science
University of Ioannina, Ioannina, Greece, GR-45110
dimako@cs.uoi.gr

Abstract

In this work we consider flooding, a fundamental mechanism for network discovery and query routing, in unstructured peer-to-peer networks. Flooding has well known properties such as fast responses and quick network coverage but at the same time it suffers from high overheads due to unnecessarily generated traffic (duplicate messages). While there has been a significant amount of research on strategies that try to moderate this drawback, there has been no work that aims at quantifying it. This is the subject of the present paper; we analyze the behavior of flooding related to duplicate messages and provide simple bounds and approximate models to assess the associated overheads.

1 Introduction

Unstructured peer-to peer (p2p) networks consist of a large population of networked computers that offer resources and operate at fully decentralized manner. Such systems are usually quite large, highly dynamic and each node (or computer) has only information for a small subset of the other participating nodes. Unstructured p2p networks are usually modeled by graphs, which provide the tools to characterize their structure and predict their behavior [1]. Various networks like the Internet, social contact networks, biological networks, webpage networks can be described as random graphs. Random graph models were pioneered by Erdős and Rényi [2], through their uniform model, known as $G(N, m)$, where N nodes are connected by

m edges, which are uniformly at random chosen out of the $N(N-1)/2$ possible edges.

In the absence of any information, these systems use mainly flooding and its variants to provide search facilities. In flooding, the node (peer) that initiates the search sends a query message to all its neighbors. Any neighbor that does not know about the resource propagates the message to all its neighbors, and so on, until the resource is discovered or some termination conditions are reached.

For example, one of the fundamental issues is that of locating a desired resource. Because in unstructured p2p networks there is no relation between topology and resource placement, all search methods are mainly oblivious and flooding is one the most widely employed ones (the Gnutella protocol [3] is one prominent example). Another closely related operation is that of network discovery, where a node is interested to discover all reachable nodes (possibly within a particular distance). This operation can be considered as a special type of query where no peer knows about the requested resource and thus forwards it to all its neighbors. For our purposes here, network discovery and search will be considered equivalent and we will use them interchangeably.

It is well known that flooding, by nature, produces an enormous amount of messages that are transmitted over the network. Thus the search process is very fast but at the same time it is non scalable and quite expensive. Achieving high node coverage in the network yields better search performance but overloads the network with unnecessary messages. There are many works that acknowledge the excessive traffic produced by flooding [4], [5], [6], [7] and many others that propose strategies to reduce it. Some of the latter rely on probabilistic flooding where a node propagates a query message to its neighbors with a given probability, on random or generalized random graphs [8], [9], [10]. Gaeta et al [11], [12] derive analytical models to determine the impact of overlay topology and compare flooding-based schemes while also proposing efficient flooding strategies with tunable parameters. Gkantsidis [13] proposes a combination of flooding and random walks. Finally others try to restrict the search space by guiding search to certain edges or groups of nodes [14], [15], [16], [17], [18], or use other algorithms to eliminate excessive traffic [19], [20].

Despite this large amount of work that tries to reduce the overheads associated with flooding, to the best of our knowledge, there has been no study that tries to *quantify* the amount of this excessive traffic; this is exactly the subject of this paper. We are interested in knowing how significant the redundant traffic produced during flooding is. Our main contribution is thus a novel analysis for the overheads of plain flooding. More specifically, we derive simple but general bounds on the number of redundant messages produced, based on the number of total message transmissions and the degrees of the nodes in the network. Additionally, we present an intuitive model that approximates the flooding overheads in every step of the procedure.

The rest of the paper is organized as follows: in Section 2 we present the details of the problem we study here. In Section 3 we derive general bounds on the number of redundant messages produced by flooding. In Section 4 we present an approximate model for the number of duplicate messages and validate it through simulations. Some possible extensions of this model are discussed in Section 5 and finally the paper concludes with a summary in Section 6.

In what follows, along with the analysis we provide experimental results based on a detailed simulator we have constructed. A simulation session begins by constructing a network of given parameters (e.g. number of nodes, edges, average node degree or degree distribution). The topologies currently supported include uniformly random graphs (in the Erdős-Rényi sense), random regular graphs and power-law random graphs based on the Barabasi model [21]. In addition, the simulator can generate topologies with a desired clustering coefficient, based on the algorithms of Heath and Parikh [22]. A simulation run involves selecting a random root node which starts a query and uses flooding to propagate the query message to the other nodes of the simulated p2p network. User-configurable parameters include the options to limit the search to within a particular distance from the root node and to detect/eliminate unnecessary duplicate messages, based on the discussion that follows. Calculated metrics include the number of message transmissions, the number of visited nodes and the number of duplicate message receptions.

2 Unstructured P2P Networks & Flooding

In distributed systems, like p2p networks, nodes form an overlay network over the physical network and each connection between a pair of nodes is a logical link. The overlay network is modeled as a random undirected graph $G = (V, E)$, where $|V| = N$ are the vertices which correspond to system nodes (peers) and E is the set of edges (or links) connecting the peers. In typical networks, each node is adjacent to a subset of the others nodes in the system. We denote d_v the degree (number of neighbors) of node v. It is known that $\sum_v d_v = 2|E|$. Finally, let $\overline{d} = \frac{1}{N}\sum_v d_v$ be the average degree in the network.

Fig. 1 illustrates the problems of flooding; node v is about to propagate a message to both its neighbors (u and w). Assume, as an example, that each message transmission takes exactly 1 time unit (step). In the scenario of Fig. 1, an arrow denotes a message transmission and its label is the step of the procedure the transmission occurs at. Nodes x and y will receive the message *twice* in the second step through different paths. Since there is no coordination, in the third step nodes u and w will receive the same message another two times, leading to a large number of redundant receptions of the same message (*duplicate* messages). Duplicate messages are denoted by the gray arrows.

In order to control the extend of flooding, a *time-to-live* (TTL) parameter is usually attached to each message which gets decremented at each hop. The further transmission of the message stops whenever its TTL count is reduced to zero. The aim is to limit the search space within a particular distance. In Fig. 1 a TTL value of 2 would eliminate the duplicate messages produced in the third step.

To make the problem more concrete, assume a random d-regular network and a totally blind flooding setting, where a node forwards any received message to all its neighbors, except the node it received the message from. Since the degree of any node v is d, any message reception will result in $d - 1$ message transmissions by v. If there were $r(j)$ message receptions at step j, then there will be

$$r(j + 1) = (d - 1)r(j)$$

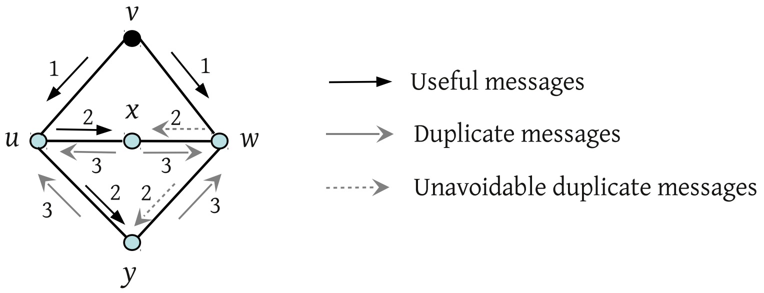


Figure 1: Example of flooding on a simple network and produced messages.

message transmissions at step $j+1$. The recursion easily gives

$$r(j) = d(d-1)^{j-1}, \quad j \geq 1,$$

since $r(1) = d$ for the root node. The total number of messages transmitted up to time t is then equal to:

$$M(t) = \sum_{j=1}^{t} r(j) = d\frac{(d-1)^t - 1}{d-2} > (d-1)^t, \tag{1}$$

which is exponential on the time step.

In Fig. 2(a) we show the number of messages predicted by (1), along with a simulation run on a random 5-regular network of 100,000 nodes, that shows the close match. On the other hand, in Fig. 2(b) we have plotted the total number of nodes that have received the query message after each simulation step, for the same network (curve with white circles). It is should be clear that after a few steps, the number of such nodes is disproportionally smaller than the number of transmitted messages. Based on Fig. 1, this means that a large number of messages are retransmitted to already visited nodes, i.e. they are unnecessary duplicate messages.

It is clear that the situation is undesirable and this is why *duplicate detection mechanisms* (DDMs) need to be employed in order to limit unnecessary message transmissions. Such a mechanism is used for example in Gnutella where each query message is assigned a globally unique identifier (GUID) field [23]. When a peer receives a message, it stores its GUID in a local query cache (qcache) and keeps it there for some time. If the peer receives the same query message again (i.e. the same GUID), it simply discards it, avoiding unnecessary transmissions.

This method of detecting duplicates may require a large amount of memory, depending on the traffic that passes through each peer. High-traffic nodes will receive large volumes of query messages per time unit and should thus either use large qcaches or keep their qcache entries for smaller periods. It should also be noted that the mechanism can never be perfect in the sense that a duplicate message may always appear long after its GUID was removed from a peer's qcache, for example due to delays in the underlying physical network. Nevertheless, this simple mechanism is quite powerful and eliminates the majority of redundant traffic, as is shown in the lower curve in Fig. 2(a). For this plot we have used the same simulation parameters as in upper curve, but this time we have each peer maintain a perfect qcache. It can be easily seen that the behavior of flooding has now moved quite far from the exponential curve.

However, even by using a perfect DDM, there are certain duplicate messages that cannot be prevented. We term them as *unavoidable* duplicate messages and this is the kind of duplicate messages we are going to consider in the next sections. Assume that a perfect local DDM is employed at each peer. By perfect, as noted above, we mean that it never forgets an incoming message so as to avoid propagating it twice. By local we mean that it has no knowledge of what messages the other peers handle.

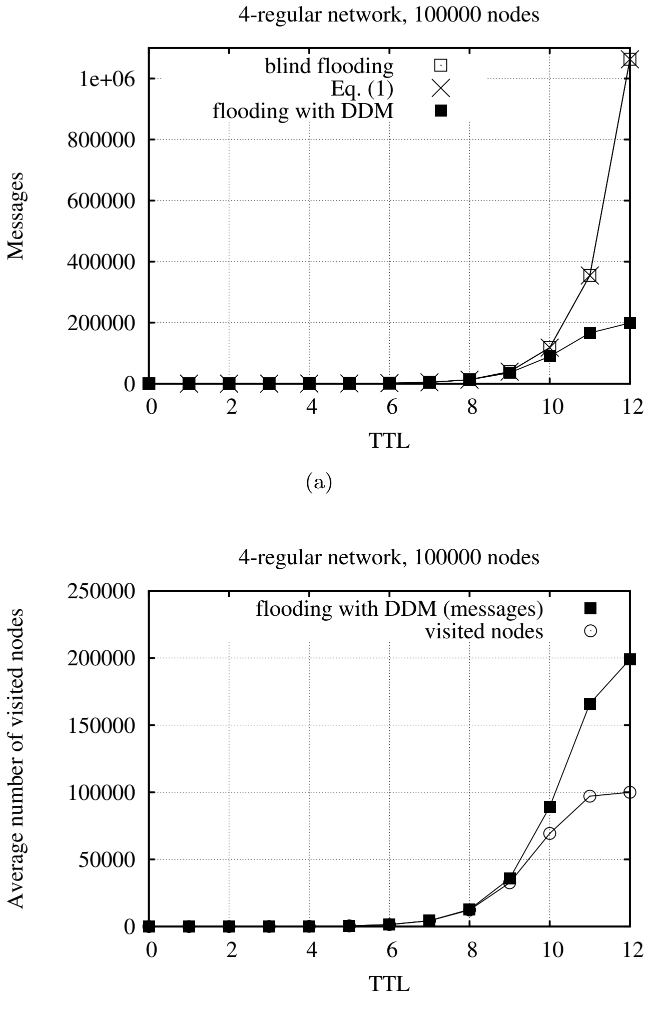
Definition 1 *All duplicate messages that occur when every node utilizes a perfect local duplicate detection mechanism are termed unavoidable duplicate messages.*

These concern messages that arrive at the same node through different paths in the network. In such cases, all the duplicate detection mechanism can do is stop propagating them *after* they arrive. Unavoidable duplicates are shown in Fig. 1 by dotted arrows. Even if a DDM eliminates most duplicates, unavoidable duplicates represent a sizable redundancy as exemplified in Fig. 2(b). The upper curve (black boxes) is a repetition of the message transmissions shown in Fig. 2(a), under the assumption of a perfect DDM mechanism while the lower one is the number of visited nodes as described above. At each step, the difference between the two curves gives the number of unavoidable duplicates.

3 General Bounds On Unavoidable Duplicates

In this section, we provide simple general bounds for flooding-based search in arbitrary networks where we assume that a perfect duplicate detection mechanism is in effect. In addition to the DDM, we assume that neighboring nodes won't simultaneously exchange the same message. That is, if nodes v and u are neighbors through edge e (see Fig. 3) and received the query message simultaneously from other nodes, then only one of them will utilize e to transmit the message so as to avoid a duplicate message transmission (this can be guaranteed for instance by a simple handshake protocol for each transmission). We require this so that our results are independent of the relative speeds of the nodes. If for example node u is faster than node v or receives the query message slightly earlier, then there will only be one duplicate message (which is unavoidable) transmitted towards v; v won't transmit the same message back to u. If, however, both nodes send the message to each other blindly at the same time, then there will be two duplicate messages, as shown in Fig. 3. Thus, depending on the relative node speeds, edge e may be crossed by the query message either once or twice. With the above assumption it is guaranteed that any edge in the network will be used *at most once* to carry the query message. Based on the above, we are interested in the number of unavoidable duplicate messages.

Consider a 'discovery' or 'ping' message that is flooded in order to reach as many nodes as possible and assume that a total of m messages where transmitted



(a)

(b)

Figure 2: (a) Blind flooding vs flooding with a duplicate detection mechanism, (b) visited nodes and messages under a DDM.

over the network. Let the number of different nodes discovered be n (including the discovery initiator). We have the following lemma.

Lemma 1 *If a discovery request produces m messages, the number of different discovered nodes is $\sqrt{2m} \leq n \leq m+1$.*

Proof: Suppose that the discovery request visited n different nodes, the initiator and $n-1$ other nodes. Clearly, if there are no duplicates, each message reaches a new node and thus $n-1 \leq m$, or $n \leq m+1$.

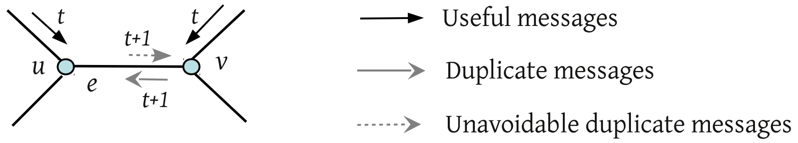


Figure 3: Duplicate message avoidance in neighboring nodes.

In addition, the subgraph induced by the n discovered nodes has at most $n(n-1)/2$ edges. Because of the duplicate detection mechanism and the above assumption, no edge is going to be used more than once in order to propagate the discovery request. Consequently,

$$m \leq \frac{n(n-1)}{2} \Rightarrow n \geq \sqrt{2m}.$$

Q.E.D.

Lemma 2 *If the maximum node degree is Δ and a discovery request produces m messages, the number of different discovered nodes is $2m/\Delta \leq n \leq m+1$.*

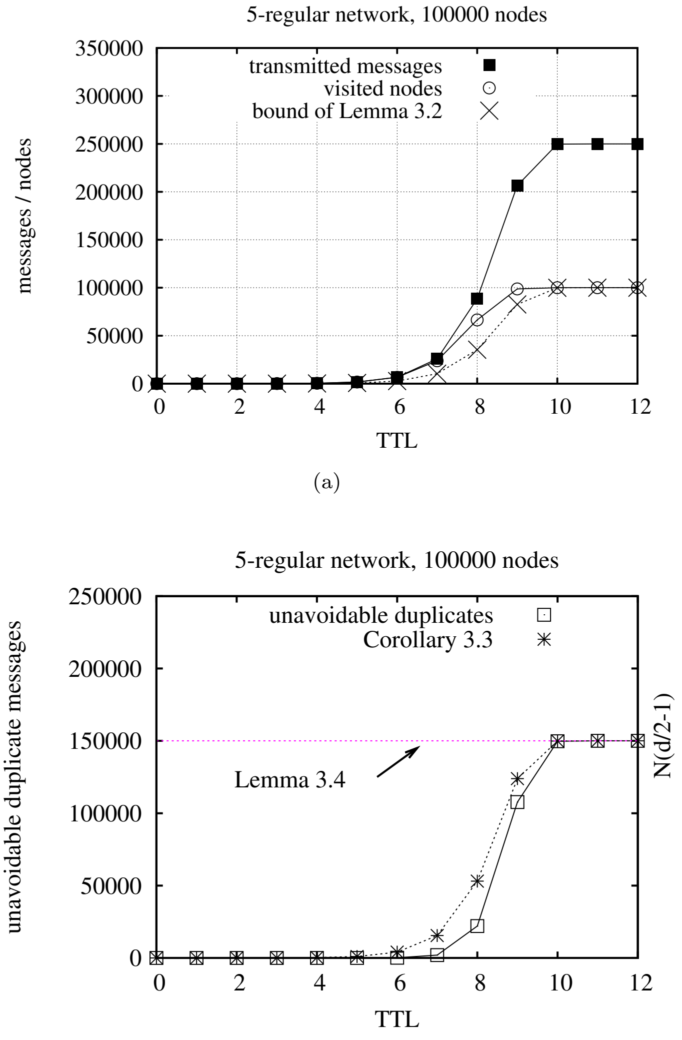
Proof: The proof is identical to that of Lemma 1, except that in the subgraph induced by the n discovered nodes, the edges are at most $n\Delta/2$ in number since the degree of each node is at most Δ.
Q.E.D.

Corollary 1 *If a discovery request produces m messages, the number of unavoidable duplicate messages is at most $m+1-\max\{\sqrt{2m}, 2m/\Delta\}$.*

Given the duplicate detection mechanism, and the assumption in the beginning of this section, it should be clear that no edge in the graph is going to be used more than once in the process of flooding a certain message. This means that the number of message transmissions can never exceed $|E|$, or $\overline{d}N/2$, where $\overline{d}$ is the average node degree. However, if all edges of the graph have been traversed, then all nodes in the network will have been contacted. The result is that out of the $\overline{d}N/2$ messages, exactly N nodes will have been discovered. Considering that $N-1$ messages are necessary to discover N different nodes, at most $\overline{d}N/2-(N-1)$ messages are duplicates, leading to the next Lemma.

Lemma 3 *For any discovery request, the number of unavoidable duplicate messages is at most $N(\overline{d}/2-1)+1$.*

In Fig. 4 we illustrate the bounds given above in the setting of a random 5-regular network of 100,000 nodes. The solid plots represent simulation data while the dashed ones represent derived bounds from the above analysis. In particular, in Fig. 4(a), the upper curve is the count of message transmissions (m). The lower curve is the lower bound on the number of discovered nodes, as obtained from Lemma 2, which, in this case, is applied with $\Delta = 5$. The middle curve is the actual number of discovered nodes during simulation. Fig. 4(b) depicts the unavoidable duplicate messages, where in addition, the upper limit on their number is given by the horizontal line as derived from Lemma 3.



(a)

(b)

Figure 4: (a) Message transmissions and visited nodes and (b) unavoidable duplicate messages.

4 A Simple Model

In this section we develop an approximate model for the behavior of flooding-based search when a duplicate message detection mechanism is in effect. We consider random networks of N nodes where the average node degree is given by $\overline{d}$, and where message transmissions are of the same speed, equal to 1 time unit (or step). Assume that some node v has initiated the flooding procedure and let S_t be the set of nodes that have been contacted through flooding, including node v, up to step t.

We denote by m_t be the expected number of messages transmitted exactly at step t and by n_t the expected number of nodes that lie in distance t from v. Finally, let

$$M_t = \sum_{i=1}^{t} m_i$$

and

$$N_t = \sum_{i=1}^{t} n_i$$

be the total number of messages and the total number of visited nodes, correspondingly, up to and including step t.

Consider now step $t+1$. It should be clear that the new nodes contacted during the $(t+1)$-th step are given by $S_{t+1} \setminus S_t$ and are all nodes in distance exactly $t+1$ from the root node, i.e. $n_{t+1} = |S_{t+1} \setminus S_t|$. Due to the duplicate detection mechanism, the only nodes to transmit new messages at step $t+1$ will be the n_t nodes in distance t. Since a node does not transmit the message back to the neighbor that delivered it, the nodes in question will only transmit to new nodes in distance $(t+1)$ or to already visited nodes in distance t. Given that the average node degree is $\overline{d}$, then there will be $(\overline{d}-1)n_t$ message transmissions at step $t+1$.

Because edges are uniformly random, message destinations are also uniformly randomly distributed. As a result, the probability that a message is destined for a new node, not already visited by time t, is given by $(N - N_t)/N$. Consequently the number of new nodes discovered exactly at step $t+1$ is given by

$$n_{t+1} = (\overline{d}-1)n_t \left(1 - \frac{N_t}{N}\right). \tag{2}$$

Using a similar argument we can derive the number of transmitted messages. In particular, out of the $(\overline{d}-1)n_t$ possible edges to transmit the message from the n_t nodes, only a portion of $(|E| - |E_t|)/|E|$ of them will be used,

$$m_{t+1} = (\overline{d}-1)n_t \left(1 - \frac{|E_t|}{|E|}\right), \tag{3}$$

where $|E_t|$ is the number of edges used up to step t and $|E|$ is the total number of edges in the network. Clearly, $|E| = \overline{d}N/2$. The number of edges used up to step t is approximated by

$$|E_t| = \frac{\overline{d}N_{t-1}}{2} + \frac{n_t}{2}. \tag{4}$$

This is because by time t the nodes in S_{t-1} will have utilized (for message transmissions) all their incident edges, which in number are equal to $\overline{d}N_{t-1}$, divided by 2 to account for counting each edge twice. Observe, however, that at the tth step, the edges leading to new nodes are only counted once, since the new nodes do not belong to S_{t-1}. These new nodes are n_t in number and are reached through approximately n_t edges, leading thus to the corrective $+n_t/2$ term. Notice that this is an approximation since there may exist two or more nodes in distance $t-1$

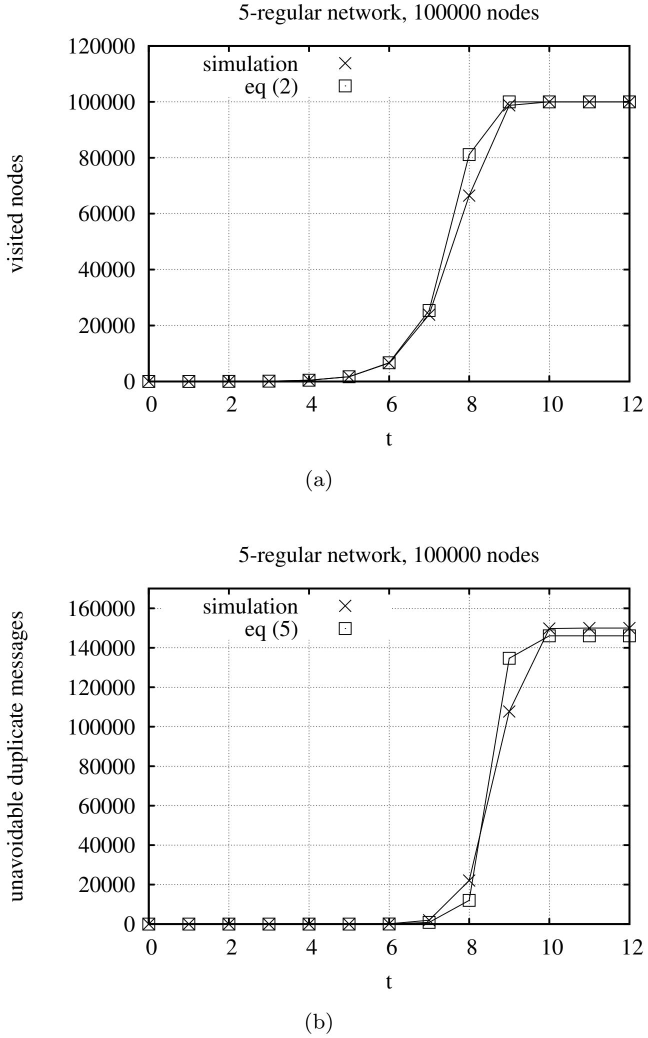
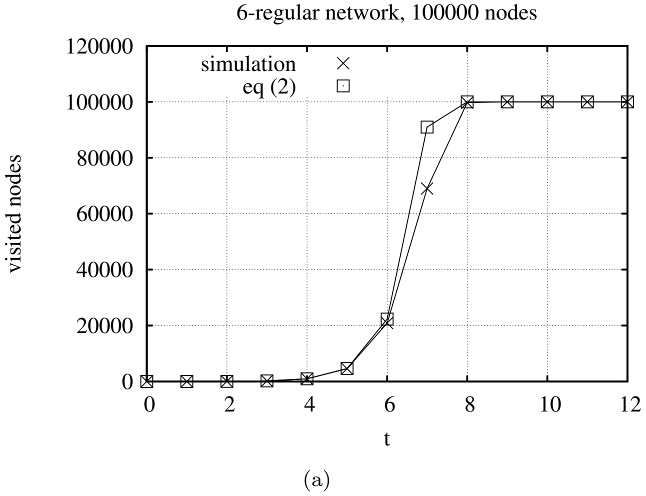


Figure 5: Visited nodes (a) and duplicate messages (b) in a random 5-regular of 100,000 nodes.

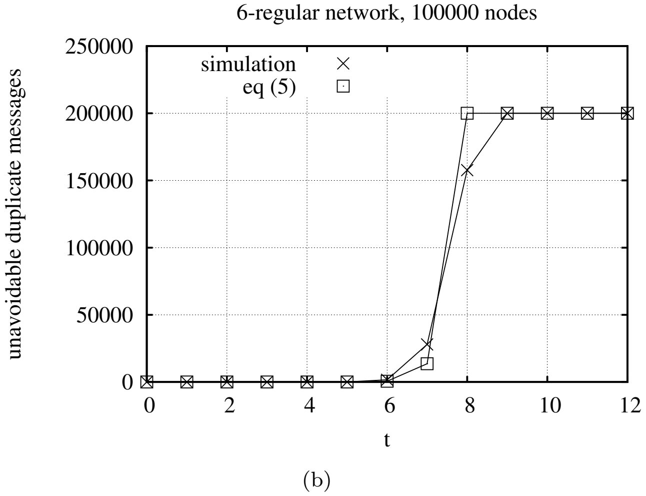
adjacent to a node in distance t. Based on the above, we can calculate the expected number of duplicate messages as:

$$\mathrm{dups}_{t+1} = m_{t+1} - n_{t+1}. \tag{5}$$

We note that (2) is reminiscent of the discrete logistic equation, which has been used widely in the study of population growth patterns. Korhonen and Kurhinen [24] use the logistic function to model information diffusion in a mobile encounter network under several assumptions, while Oyama et al. [25] suggest the logistic equation as a theoretical tool to approximate simulation results for data dissemi-



(a)



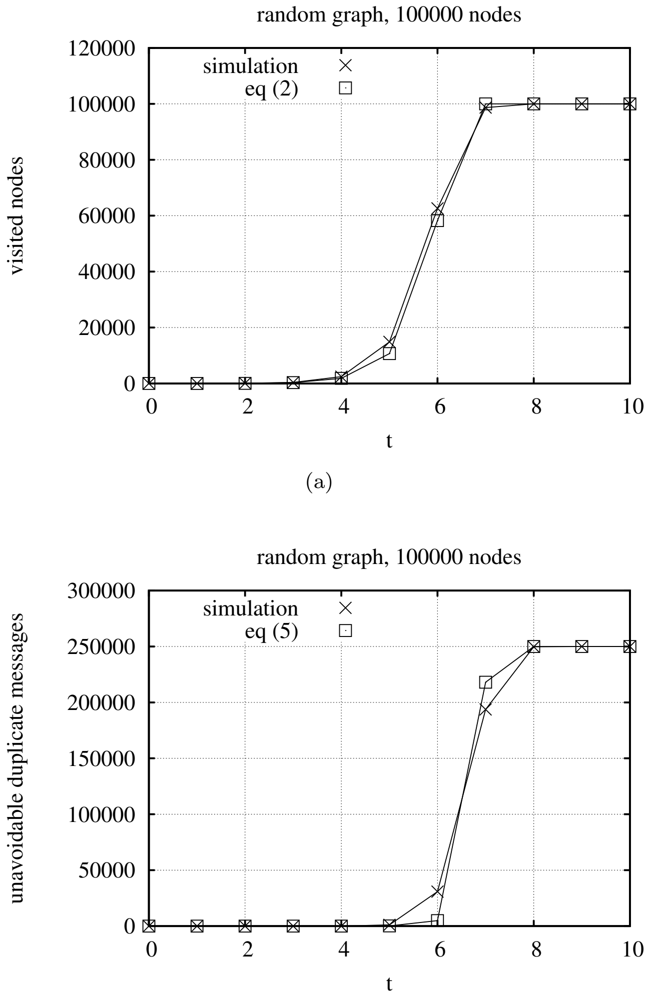
(b)

Figure 6: Visited nodes (a) and duplicate messages (b) in a random 6-regular of 100,000 nodes.

nation over mobile p2p nets.

The model of (2)–(5), while simple, serves for a very effective approximation as we have confirmed in our simulation experiments. In Fig. 5–7 we present simulation results along with what the model gives. In Fig. 5 and 6 we consider a random regular network of 100,000 nodes with node degree 5 and 6 respectively, while in Fig. 7, the network is random (non-regular) with $\overline{d} = 7$.

The plots exemplify the accuracy of our model. Its limitations stem mainly from the simplifying approximation we used in (4) and the fact that the globally average network degree ($\overline{d}$) we use in our formulas, is different for the average degree of



(a)

(b)

Figure 7: Visited nodes (a) and duplicate messages (b) in a random network of 100,000 nodes, with $\overline{d} = 7$.

the visited nodes at each step. Nevertheless, the model succeeds to capture the essential dynamic behavior of the flooding procedure.

5 Extending the model

In this section we discuss how the applicability of the simple model of Section 4 can be extended to networks other than the standard random ones.

5.1 Networks with Given Clustering Coefficient

Clustering is a common characteristic found in many real networks such as social, biological and technological networks [26, 27] and it expresses the property that two neighbors of a node v may also be neighbors themselves. The *clustering coefficient* (cc) of the network is measure of the clustering property and it defined as [26]:

$$cc = \frac{3 \times N_\Delta}{\sum_{v \in G} \binom{d_v}{2}}$$

where d_v is the degree of a node v and N_Δ is the number of triangles (i.e. triads of nodes which are neighbors to each other) present in a network. We consider here how the simple model of Section 4 can be extended to random graphs with tunable clustering coefficient which have been studied recently [26, 22].

Since cc represents the probability that two nodes with a common neighbor are also neighbors themselves, only a fraction $1 - cc$ of the edges a node uses at step $t + 1$ will lead to new (unvisited) nodes. Consequently (2) is modified as follows in order to account for the given clustering coefficient:

$$n_{t+1} = (\overline{d} - 1) n_t (1 - cc) \left(1 - \frac{N_t}{N} \right). \tag{6}$$

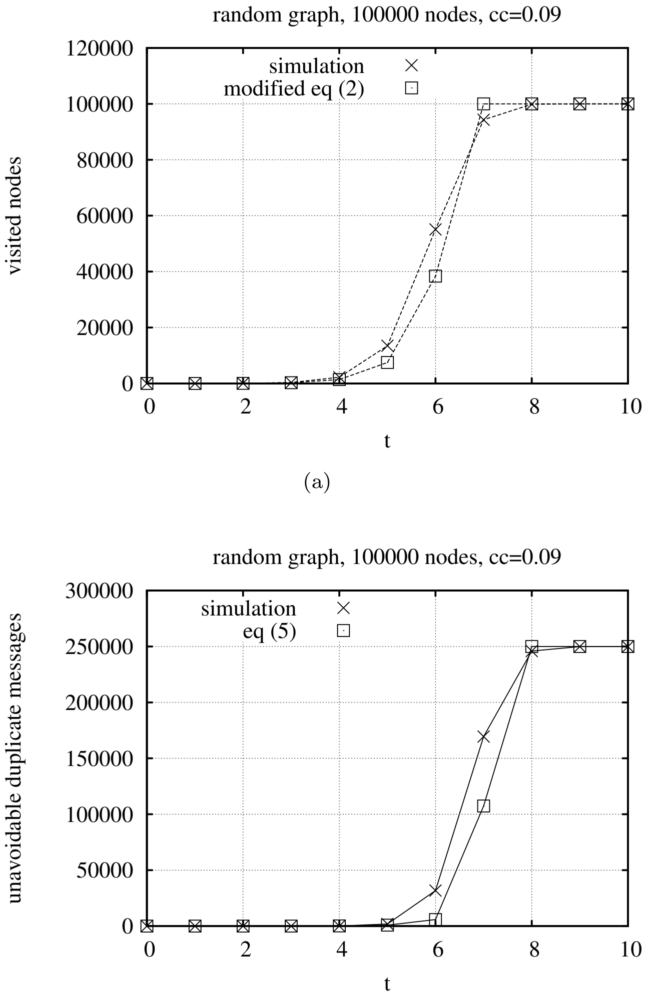
Our simulator has been extended to generate random graphs with a given degree distribution and clustering coefficient according to the algorithms of Heath and Parikh [22]. We have conducted expirements using the updated model, using (6) instead of (2), for various random graphs with Poisson-distributed degrees. In Figs. 8 and 9 we present results for a random graph with 100000 nodes, average node degree of 7 and cc value of 0.09, and 0.14, respectively. The plots demonstrate that the modified model succeeds in following closely the behavior of flooding in this class of topologies.

5.2 Power-Law Networks

The topological organization of many real networks obeys power law degree distributions (the probability that a node has degree k is $\sim k^{-a}$) [28]. These networks grow according to two basic principles: a) a new node is connected to m nodes already present in the network and b) the new node chooses existing nodes with probability proportional to their degree (preferential attachment property).

Our simple model is not directly applicable to this topology class. Because the vast majority of nodes have very small degree and very few nodes are highly connected, the average network degree ($\overline{d}$) is very different from the average degree of visited nodes at each step of the procedure. If the average node degree, $\overline{d}_t$, in distance t from the originating node was known then our model would also be applicable—the only difference being the use of $\overline{d}_t$ instead of $\overline{d}$ in Eqs. (2)–(5).

To verify our argument we simulated power-law networks using the Barabasi-Albert model [21]. Initially we performed flooding in order to estimate $\overline{d}_t$ for every step t of the procedure. We then utilized the estimated values as described above. Indicative results are shown in Fig. 10 for a network with 100000 nodes and $m = 2$ (numbers of link for each new node) which confirm our theory.



(a)

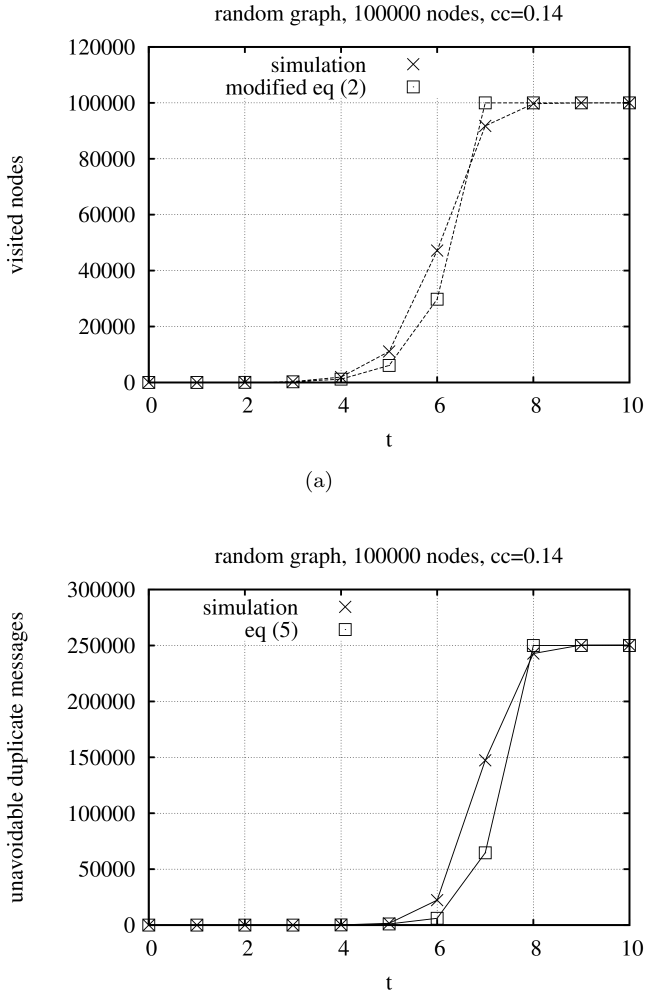
(b)

Figure 8: Visited nodes (a) and duplicate messages (b) in a random network of 100,000 nodes, with $\overline{d} = 7$ and clustering coefficient cc=0.09.

6 Conclusion

In this work we have considered flooding in unstructured p2p networks. We have analyzed the excessive overheads related to blind flooding, which justify the deployment of duplicate detection mechanisms. Such mechanisms are able to reduce the unnecessary overheads substantially but cannot however eliminate the 'unavoidable' duplicate messages that emanate from the overlay network topology itself and the multiplicity of different paths among nodes.

We have derived basic bounds for the number of such unavoidable duplicate

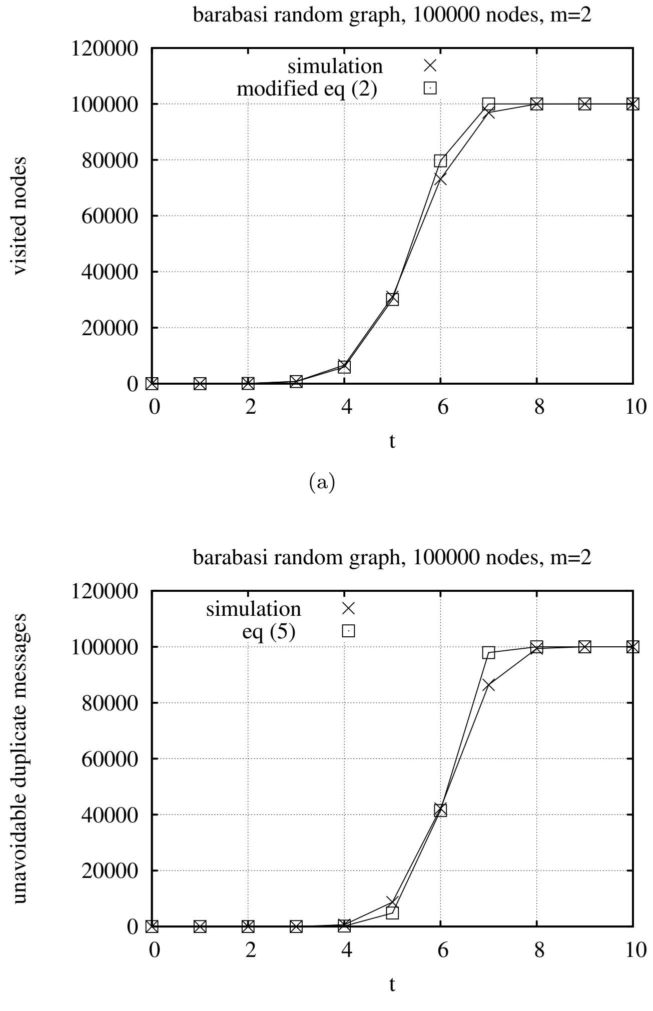


(a)

(b)

Figure 9: Visited nodes (a) and duplicate messages (b) in a random network of 100,000 nodes, with $\overline{d} = 7$ and clustering coefficient cc=0.14.

messages in any p2p network, which we also demonstrated through simulation. In addition, we developed a simple but effective model for the behavior of flooding at every step of the procedure, applicable to unstructured p2p overlays modeled as random graphs of various classes. We are currently working on the refinement of our model and its extensions.



(a)

(b)

Figure 10: Visited nodes (a) and duplicate messages (b) in a barabasi network of 100,000 nodes, with $m = 2$.

References

[1] M.E.J. Newman, *The structure and function of complex networks*, SIAM REVIEW 45 (2003), p. 167.

[2] B. Bollobas, *Random Graphs*, Cambridge University Press, 2001.

[3] M. Ripeanu and I.T. Foster, *Mapping the Gnutella Network: Macroscopic Properties of Large-Scale Peer-to-Peer Systems*, in *IPTPS*, 2002, pp. 85–93.

[4] S. Jin and H. Jiang, *Novel approaches to efficient flooding search in peer-to-peer networks*, Comput. Netw. 51 (2007), pp. 2818–2832.

[5] Z. Zhu, P. Kalnis, and S. Bakiras, *Dcmp: A distributed cycle minimization protocol for peer-to-peer networks*, IEEE Trans. Parallel Distrib. Syst. 19 (2008), pp. 363–377.

[6] V.V. Dimakopoulos and E. Pitoura, *On the performance of flooding-based resource discovery*, IEEE Trans. Parallel Distrib. Syst. 17 (2006), pp. 1242–1252.

[7] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, *Search and replication in unstructured peer-to-peer networks*, in *ICS*, 2002, pp. 84–95.

[8] K. Oikonomou, D. Kogias, and I. Stavrakakis, *Probabilistic flooding for efficient information dissemination in random graph topologies*, Comput. Netw. 54 (2010), pp. 1615–1629.

[9] R. Gaeta and M. Sereno, *Generalized probabilistic flooding in unstructured peer-to-peer networks*, IEEE Transactions on Parallel and Distributed Systems 22 (2011), pp. 2055–2062.

[10] S. Crisostomo, U. Schilcher, C. Bettstetter, and J. Barros, *Probabilistic flooding in stochastic networks: Analysis of global information outreach*, Computer Networks 56 (2012), pp. 142 – 156.

[11] R. Gaeta and M. Sereno, *Random graphs as models of hierarchical peer-to-peer networks*, Perform. Eval. 64 (2007), pp. 838–855.

[12] R. Gaeta and M. Sereno, *On the evaluation of flooding-based search strategies in peer-to-peer networks*, Concurr. Comput. : Pract. Exper. 20 (2008), pp. 713–734.

[13] C. Gkantsidis, M. Mihail, and A. Saberi, *Hybrid search schemes for unstructured peer-to-peer networks*, in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, Vol. 3, march, 2005, pp. 1526 – 1537 vol. 3.

[14] Z. Kun, N. Zhendong, Z. Yumin, and Y. Jun, *Group-based search in unstructured peer-to-peer networks*, in *Proceedings of the 28th IEEE conference on Global telecommunications*, GLOBECOM'09, Honolulu, Hawaii, USA, IEEE Press, Piscataway, NJ, USA, 2009, pp. 1493–1498.

[15] R. Zhang and Y.C. Hu, *Assisted peer-to-peer search with partial indexing*, IEEE Trans. Parallel Distrib. Syst. 18 (2007), pp. 1146–1158.

[16] S. Vuong and J. Li, *Efa: an efficient content routing algorithm in large peer-to-peer overlay networks*, in *Peer-to-Peer Computing, 2003. (P2P 2003). Proceedings. Third International Conference on*, sept., 2003, pp. 216 – 217.

[17] S. Jiang, L. Guo, and X. Zhang, *Lightflood: an Efficient Flooding Scheme for File Search in Unstructured peer-to-peer systems*, in *ICPP*, 2003, pp. 627–635.

[18] C. Papadakis, P. Fragopoulou, E. Athanasopoulos, E. Markatos, M. Dikaiakos, and A. Labrinidis, *A Feedback-based Approach to Reduce Duplicate Messages*, in *Unstructured Peer-to-Peer Networks, Integrated Workshop on Grid Research*, 2005.

[19] J. Wang, Y. Li, F. Gong, and W. Chen, *A New Strategy of Resource Searching in Unstructured P2P Network*, in *Communication Software and Networks, 2010. ICCSN '10. Second International Conference on*, feb., 2010, pp. 32 –36.

[20] Y. Liu, N. Xiong, L. Zhu, J.H. Park, and J. Gao, *An effective simulation method for search strategy in unstructured p2p network*, Simulation Modelling Practice and Theory 18 (2010), pp. 456 – 469.

[21] A. Reka and Barabási, *Statistical mechanics of complex networks*, Rev. Mod. Phys. 74 (2002), pp. 47–97.

[22] L.S. Heath and N. Parikh, *Generating random graphs with tunable clustering coefficients*, Physica A: Statistical Mechanics and its Applications 390 (2011), pp. 4577 – 4587.

[23] A. Klemm, C. Lindemann, M.K. Vernon, and O.P. Waldhorst, *Characterizing the query behavior in peer-to-peer file sharing systems*, in *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, Taormina, Sicily, Italy, ACM, New York, NY, USA, 2004, pp. 55–67.

[24] V. Korhonen and J. Kurhinen, *Logistic model for modeling mobile encounter network*, in *Information and Communications Technology, 2007. ICICT 2007. ITI 5th International Conference on*, dec., 2007, pp. 151 –155.

[25] Y. Oyama, H. Sasaki, H. Iwasaki, and Y. Nishiura, *Data Dissemination Dynamics in Mobile P2P Network*, in *Modelling, Simulation, and Optimization - 2005*, 2005, pp. 471–802.

[26] M.E.J. Newman, *Random Graphs with Clustering*, Physical Review Letters 103 (2009), p. 058701.

[27] M. Ángeles Serrano and M. Boguñá, *Tuning clustering in random networks with arbitrary degree distributions*, Phys. Rev. E 72 (2005), p. 036133.

[28] A.L. Barabási and R. Albert, *Emergence of scaling in random networks*, Science 286 (1999), pp. 509–512.