RNG-based searching and broadcasting over Internet graphs and peer-to-peer computing systems

Oscar Escalante[1], Tania Pérez[1], Julio Solano[1], Ivan Stojmenovic[2]
[1]DISCA, IIMAS, UNAM, Circuito Escolar s/n, Ciudad Universitaria, Coyoacán, México D.F. 04510, México
[2]SITE, University of Ottawa, Ottawa, Ontario K1N 6N5, Canada ivan@site.uottawa.ca

Abstract

***Abstract* - In a broadcasting problem, a message is sent from a source to all the other nodes in the network. It can be used for topology updates or to search for a file or information in a peer-to-peer computing system. Blind flooding is a classical mechanism for broadcasting, where each node retransmits received message to all its neighbors. Despite its important advantages, an increase in the number of requests or the size of the routing area produces communication overheads that limit the scalability of blind flooding, especially in networks with dynamic topologies. In this work we propose a new broadcasting and searching scheme over the Internet based on the Relative Neighbourhood Graph (RNG). We extended the definition of RNG in order to use alternative metrics other than geometric. Delay, congestion, random numbers or geographic distances are examples of such metrics that can be used efficiently in RNG based broadcasting. RNG is a sparse connected overlay network which is defined based on local information at each node. This new scheme is compared to the existing Flooding and Rumor Mongering (or Gossip) schemes to evaluate its performance. Our parameterless RNG based scheme guarantees delivery to each node with considerably reduced number of messages with respect to flooding, and has comparable amount of message to Rumor Mongering/Gossip scheme that does not guarantee delivery to each node and also uses parameters whose best value depend on underlying network density**.

Keywords: Simulations, graph theory, peer-to-peer computing, broadcasting

1. Introduction

New challenges emerge from the increasing popularity of dynamic decentralized networks. Technologies like *Bluetooth*, new file sharing systems over the Internet such as *Gnutella* or *KaZaA* and new *ad hoc* networks generated by the strong need of interconnection of spontaneous communicating devices are currently subjects of ongoing research.

In order to adapt to changes in these conditions, a scheme running on a dynamic decentralized network usually needs to know about topology changes, link and node status, changes in position, changes in the set of neighbors for each node, etc. An example of this can be seen in the OSPF (Open Shortest Path First) protocol for propagating topology updates on Internet's IPv6 [17]. This task is usually accomplished by broadcasting techniques sending a message to every node. Broadcasting also has other applications, for example the election of a leader for processes initialization or search for particular information among the nodes.

Blind flooding is a classical approach to broadcast messages across a network. Blind flooding starts with the source node disseminating a message among its neighbors. Whenever a node receives the message for the first time, it sends one copy to all of its neighbors, except the node it received the message from. The scheme stops after each node has received at least one copy of the message. Despite of its broad use, blind flooding has important disadvantages making its use inappropriate in many dynamic networks, due to the amount of retransmitted messages to cover the entire network (among other reasons).

Alternatives proposed in the literature to alleviate the problems presented by blind flooding include different types of probabilistic node selection algorithms such as the Rumor Mongering or Gossip protocol [17]. In this work we propose a new scheme based on the *Relative Neighborhood Graphs* (RNG). We show how broadcasting over the RNG using a suitable distance metric can outperform both blind flooding and Rumor

Mongering schemes. Examples of a distance to define neighborhoods may include delay, congestion, random numbers or the classic geographic distances. The idea behind this is to avoid congesting the whole network by selecting a specific subset of the topology to spread the message. Our main work has been done over synthetic Internet graphs which model the real Internet network to a certain degree according to current studies on the Internet topology. The advantage of RNG based scheme over flooding and Rumor Mongering/Gossip schemes is that it guarantees delivery while minimizing the total number of messages, and uses no parameters in its functioning. Peer-to-peer networks are created in an *ad hoc* fashion and new nodes can connect to any node on the network they want. Therefore the exact global topology is not known and the individual nodes have only a very local view of the network. The main advantage of RNG is that it generates sparse connected overlay network based only on such local information available to each node. The overlay networks uses only edges already available in the underlay network, thus avoiding the overhead needed for establishing routes for virtual edges in, for example, distributed hash table approaches [2], [6], [21] for searching in peer to peer networks.

This paper is organized as follows: section 2 provides a literature review of the concepts involved on this work. Section 3 explains the new scheme, that is, the application of the RNG concept for broadcasting over Internet Graphs. Performance evaluation is given in Section 4. Section 5 contains some concluding remarks.

2. Literature review

A. Internet Graphs

There have been several attempts to model the Internet. Anyone with an Internet connection can explore links between *routers* from the local computer to any server in the world by means of a `traceroute` utility. Some relevant properties of the Internet such as diameter or degree distribution can be determined using similar exploring tools. While the early simulations of Internet topology were conducted on random generated graphs recent results have shown that they do not accurately reflect the properties of Internet topology [5, 1, 14].

Interest in modeling new algorithms and protocols has originated the study of Internet from different approaches according to its interconnection level. At the *router level* the nodes are the routers and the edges are the physical connections between them. At the *inter-domain* (or Autonomous System) *level* each domain, composed of hundreds of routers and computers, is represented by a single node and an edge is established between two domains if there is at least one route that connects them [1]. These concepts can be also extended at the *application level* [20] where nodes are computer applications and edges are the communication links between them.

Faloutsos *et al.* [5] have studied the Internet topology at both levels and concluded that in each case the degree distribution follows a power-law pattern. They proposed four power-laws and suggested the use of these laws to measure the realism of synthetically generated Internet graphs. The challenge in the Internet topology modeling task comes from a huge and constantly evolving topology [7]. A generation model that captures and represents its properties is proposed in [5, 3, 19]. Magoni and Pansiot [15] introduced a method performing a sample on a real Internet map. Their method complies with the four power laws presented by Faloutsos *et al*. and with the distance properties such as the number of shortest paths between node pairs and tree size distributions observed on Internet maps. An Internet modeler based on map sampling is described and analyzed in [14], [15].

Several simultaneous studies have been conducted which have improved our understanding of the Internet topology. We shall now present a general overview of the models used during our simulations and the generators used to create such topologies.

Barabási and Albert [3] proposed a model to construct scale free networks. This model is applicable not only to Internet topologies but to a variety of fields such as biology, social sciences and chemistry. *Scale free* networks are those in which a node has degree k with probability k^{γ} for some constant γ. The Barabási model is based on two main ideas. First, most real network systems grow in time. Second, most real networks exhibit preferential connectivity: a new node is most likely to be attached to an important existing node with large connectivity.

Waxman's method [25] introduces a link between a pair of nodes with a probability which depends on their distance. Although the Waxman model fails to reproduce the power law properties [15], [19], it offers a good reference for generation models, additionally it was one of the first attempts to model the Internet topology as a non random graph. Palmer and Steffan [19] introduced a method which partitions recursively the set of nodes in equal parts and favor links for nodes inside each part based on some parameters.

In our experiments we used the Waxman, Palmer and Steffan, and Barabási models to generate the topologies to which the RNG algorithm was applied. We used these three models to evaluate the performance of our new scheme when using different generation models. Once the models to generate the graphs have been chosen, there is a need to use a generator to create them. Currently, there are many topology generators, each of them implements a particular set of generation models. Its selection is based on the characteristics of the topology to be generated e.g. the number of nodes and edges or its structural properties. For this work, we used two topology generators: BRITE [16] and one provided by Palmer and Steffan [19].

BRITE (Boston University Representative Internet Topology Generator) is a universal topology generator created by Medina, Lakhina, Matta and Byers. It supports multiple generation models including models for flat interdomain, flat Router and hierarchical topologies. Models can be enhanced by assigning links attributes such as bandwidth and delay. Additionally, it provides the user with a Graphical User Interface and a configuration file to easily specify diverse topology generation parameters. We used BRITE to generate the topologies for the Barabási and Waxman models. The main reason to choose it as generator was its flexibility and feasibility to modify the parameters and its implementation in Java which allowed developing a platform independent application.

Jin et all [26] proposed DIP (Distance Information Protocol), based on IDMaps (Internet Distance Map Service) by Francis et all [27] which provides distance estimates between any two hosts on the Internet. The distance measure can be latency, bandwidth or other metrics.

B. Flooding

In the broadcasting process, a node wishing to disseminate a message across the network starts by sending a copy of this message to all of its neighbors. The amount of time it takes a node to receive a message and then forward it on to its selected neighbors is called a *round*. Several nodes may have rounds running in parallel. The algorithm finishes, or *converges*, when all the nodes in the network have received a copy of the message. Broadcasting converges in *O(diam)* rounds, where *diam* is the diameter of the network, because it takes at most *diam* rounds for a piece of data to travel from one end of the network to the other.

In the blind flooding scheme, whenever a node receives the message for the first time it forwards that message to its neighbors, except to the node from which it just received it. The method disseminates information quickly in a network with enough bandwidth and no loss prone links [8]. Nevertheless, it has disadvantages prohibiting its use in networks with a dynamic topology behavior or where requests for information are frequent and must be known by all the participants. The main disadvantage of blind flooding is its extensive use of the bandwidth, if dense underlay networks are used. At node level, each message retransmission consumes processing time and the available bandwidth on both incoming and outgoing links. An illustrative analysis of the blind flooding performance was presented in [20].

Although the strict definition of blind flooding rules that messages are retransmitted to all neighbors (except the one it came from), we will now show that the protocol remains somewhat incomplete, with two variants that can be used to complete it.

The type of blind flooding defined in the Gnutella file sharing system is a flooding that follows a model where edges may be used for sending the same message in both directions (i.e. node *A* to node *B* and vice versa). It will be referred here as the **Double Flooding** model. The total number of messages in double flooding model of blind flooding protocol is $1+(d-1)n$ [20] where d is average network density or average node degree (that is, the average number of neighbors per node), while n is the number of nodes in the network. That is, each of n nodes retransmits to its $d-1$ neighbors on average, plus one more message from the source node since it has no predecessor in the process. In the implementation [20], each node may receive several copies of the same message before deciding about retransmission. When such decision is made, the node eliminates only the last neighbor from which the same message was received, and then transmits simultaneously to all the others (including those the message was received from previously). Therefore it is possible that the same message travels simultaneously on the same link in both directions.

In this article we also define and consider the **Single Flooding** model as the process in which each link in the network can be used only once to send the same message and consequently only in one direction. That is, while node *A* sends to node *B*, *B* cannot and does not simultaneously sends to *A*. Instead, node *B* processes every message received from *A*, and if the same message is found in the queue for transmission to the sender node *A*, it is eliminated from the queue and never transmitted to *A*. In this single flooding model, *B* can receive the same message from several neighbors before being able to retransmit it to the remaining neighbors. The total number of messages in single flooding model is therefore equal to the total number of edges in the network used for broadcasting, since every edge is used exactly once in the process. Single flooding is justified since each node in double flooding needs to memorize each incoming message, to avoid forwarding the same message several times to all (except sender) neighbors. We simply propose to extend the memorization to include all senders of the same message, to be stored as long as the message is waiting in the queue for transmitting to any of neighbors. In fact, this memorization can be restricted to each link independently, so that the same message arriving from one endpoint is eliminated from the queue of the other link endpoint. Obviously single flooding reduces the message overhead. A detailed comparison of the number of messages involved in each variant will be given in Section 4.

C. The Gnutella network

Gnutella is the common example of a pure *peer-to-peer* network, used for free sharing of files among its participants, representing a proper scenario for the overlay network building method we proposed in this paper. Gnutella provides its services of searching for files and peer discovering via its overlay network by broadcasting messages to all peers in the network. The two types of messages of our interest are `PING` and `QUERY`. A `PING` message is essentially a request for a host to announce itself in the network while a `QUERY` is the primary mechanism for searching the network. These messages are propagated through the network by means of a double flooding-based blind flooding. Various studies have been conducted around the Gnutella network, on modeling its topology [10] and on proposing solutions to the flooding limitations [20].

The flooding process in Gnutella is implemented by using two mechanisms designed to restrict flooding scope and limit the number of redundant messages [10]:

Mechanism 1: Time-to-Live Bounds. Time-to-Live (TTL) is a governing mechanism that prevents messages from traveling farther than a specified number of hops, defined by an initial TTL value. TTL bounds are implemented by including in each message header a TTL value field. When a node receives a message it first checks to see if its TTL value is greater than zero. If not, the node continues the flood with a decremented TTL. Otherwise the message is dropped.

Mechanism 2. Unique Message Identification. Unique Message Identification (UID) is a mechanism that prevents unique messages from being transmitted more than once from each node. This mechanism is implemented by including in each message header a UID (a unique ID label, or unique sequence number). When a node receives a message it checks to see if it has previously seen that message. If it has, the message is dropped and not forwarded. Otherwise, the node stores the new

UID in a local table, and then continues the flood.

In accordance with those mechanisms the flooding cost in Gnutella is calculated with the following formula: $c = 1 + n(d-1)$ [20], where d is the average degree and n is the number of nodes. Basically, the UID mechanism prevents a message from being forwarded more than one time by the same node, but does not prevent a link from being used in both directions. Thus Gnutella uses double flooding based blind flooding.

D. Gossip

A proposed alternative to flooding is *Rumor Mongering* or *Gossip* [20]. This scheme defines a class of probabilistic schemes where the message is sent to a set of neighbors of the current node selected randomly. In [20], Portmann and Seneviratne evaluated a specific type of Gossip scheme called *Blind Counter Rumor Mongering*. This scheme determines how messages are routed at each node and it has the following structure:

```
A node n initiates a broadcast
by sending the message m to B of
its neighbors, chosen at random.
When (node p receives the
message from node q),
  IF (p has received the message
no more than F times)
      p sends m to at most B
      randomly chosen neighbors
      that p knows have not yet
      seen the message.
```

Pseudocode for the Gossip protocol

Here, B specifies the maximum number of neighbors a message is forwarded to and F determines the number of times a node forwards the same message to B of its neighbors. The parameters F and B can be chosen to control the properties of the scheme.

A node p knows if its neighbor q has already seen the message m only if p has sent it to q previously, or if p received before the message from q. That is, p has a register S of nodes to which it has sent a copy of m and also a register R of nodes from which have received m, the union of sets R and S allows p identifying if q has already seen m [13]. In the case where the number of valid neighbors for a node is less than B, the message is only forwarded to this smaller number of neighbors.

The Gossip protocol may have several variants depending on the assumptions, such as memorization capabilities and simultaneous use of the same link in both directions. We define **Single Gossip** as the single flooding based Gossip process in which a link can be used in only one direction to send the same message. Thus, no simultaneous transmission in both directions is possible in this variant. Our understanding is that such version was meant in [13] and such version was used in our experiments. **Double Gossip** is defined as the double flooding based Gossip process in which a link may be used in both directions to send the same message, node memorizes received message but not the previous senders except the very last one. This means that the list of previous senders is actually not implemented. Our experiments with this double Gossip version produced data that coincided with the data for Gossip protocol presented in [20]. Therefore we suspect that the list of previous senders was actually not implemented in [20], despite the description (given above) that leads to a different conclusion. There exists also a **single/double Gossip** variant. In such variant, the list of previous senders is maintained but simultaneous use of a link in both directions is allowed. As a result, two nodes may transmit the same message to each other simultaneously before including each other in the respective sender lists. In fact, this can be further classified into variant where the message is sent to all selected neighbors simultaneously, and variant where message is sent to transmission queues of selected neighbors for independent processing, still allowing concurrent transmission in both direction of the same edge. None of the variants of single/double Gossip was implemented by us and consequently does not appear in experimental data.

We conclude this section by presenting the experimental data from [20] that coincided with experimental data we obtained for Double Gossip protocol. The results in Table 1 are obtained using topologies generated with the Barabási model with n nodes and average node degree d, with values n = 1000 and d=4, 6 and 8 respectively. Due to the parametric nature of the Gossip protocol the *percentage of reached nodes* (the percentage of nodes that receive a copy of the sent message) varies according to the values of *B* and *F*. The average number of messages in the Double Gossip protocol is compared to the Gnutella flooding cost (representing 100%) to obtain the *percentage of cost* of Gossip with respect to double flooding based blind flooding. A deeper analysis of broadcasting cost in the Gossip protocol is presented in Section 4.

		d=4		d=6		d=8	
B	**F**	**Cost**	**Reach**	**Cost**	**Reach**	**Cost**	**Reach**
2	1	30%	55%	27%	68%	19%	67%
2	2	53%	82%	51%	92%	42%	93%
3	1	49%	76%	46%	87%	38%	88%
3	2	69%	93%	67%	98%	61%	99%

Table 1: Results for the Double Gossip protocol when using different average node degrees

E. Internet searching

In this article we also consider Internet search problem in addition to Internet broadcasting. Broadcasting can be used in the context of searching as one possible approach. However, different approaches are already developed. A popular Internet search technique which received enormous attention recently is distributed hash table (DHT) method [2], [6], [21]. In this approach, a hash function maps any file name to an integer value (key). Each node is responsible for all keys in an interval between two integers. A node joining the network shall send information about each of its files to a particular node in the network, decided by the same hash function. The file name is hashed, and information about the file (and possibly the whole file in some implementations) is sent to the node responsible for the obtained key. When a node searches for the file with given name, it applies the same hash function and then contacts the node responsible for the obtained key. In this way, supply and demand meet at a neutral node, the same one since the same hash function was applied. Two important problems remain to make this process work. One is to decide about the topology for all nodes so that adding new nodes and new information, and searching for information, is fast. Many different topologies were already considered, such as mesh, torus, chord, deBruijn, hypercube, Delaunay triangulations etc. [21]. The topology needs to be easily adjustable when new nodes are added or some nodes deleted from the network. The topology shall also offer simple and efficient routing between any two nodes, preferably greedy routing. The other problem, perhaps more important, is how to actually construct desired topology in the space of integer keys. Nodes that are supposed to be connected according to desired topology are normally not direct neighbors in the underlay network (e.g. Internet). They therefore need to search for each other, which looks like the problem we started with in the first place, and which we would like to perform efficiently. Once two nodes discover a route between then, the route is memorized and followed whenever communication between them in overlay network is required. The routing tables for the purpose are normally small with respect to the size of network, basically of the order of the density of the overlay network. To avoid straight broadcasting as means of discovering route between desired pair of nodes (and repeatedly doing this for every edge in the overlay network), a solution based on attempting initial 'reasonable' ID assignment, and then applying a greedy routing in the ID space, is performed. For ID calculation, [21] proposes to establish a virtual spring between a candidate member and its fixed neighbors. The tension of each spring is set to inversely proportional to the distance measured. Then this virtual equivalent of a physical system is allowed to settle, achieving the minimum energy state. This minimum energy location of the candidate node in the (multidimensional) space is directly used for its ID [21]. As mentioned, to establish a route between two nodes, for each hop, node holding the 'discovery' message chooses the best neighbor in greedy fashion among fixed neighbors, to advance toward the destination that will complete the establishment of a virtual link between the two nodes in overlay network.

Although DHT method for Internet search gained a lot of attention recently, and was considered as even an ultimate solution, it was recently observed that it has an inherent problem that cannot be resolved, which prohibits its use over the Internet. According to DHT, a given host needs to store the content assigned to it by the algorithm. However, the host may not like to store certain type of content (e.g. pornographic images, copyright protected files). Therefore we did not compare our approach with DHT, since our approach does not have the mentioned problem.

E. The Relative Neighborhood Graphs

The *Relative Neighborhood Graph* (RNG) is a geometric and graph theoretic concept proposed by Toussaint [24]. Let $d(x,y)$ be the Euclidean distance between nodes x and y. In a RNG, a pair of nodes a and b are linked together if $d(a,c) \geq d(a,b)$ or $d(b,c) \geq d(a,b)$ for any other node c. In other words, there is no node inside the intersection region of circles centered at a and b and radius $d(a, b)$. Also, one can say that $d(a,b)$ is in RNG if edge ab is not the strictly largest edge in any triangle.

Toussaint proved some properties of the RNG that make it attractive for broadcasting. The resulting RNG of a random set of points in the plane is a *planar graph* which in particular, has its number of edges bounded by $3n - 6$. Experiments show that the average density of a RNG graph is about 2.4.

It is not difficult to observe that a flooding algorithm performs better on a network with a small number of edges or without redundant links, in fact the theoretical minimum cost for broadcasting a message to the entire network, would be the transmission of one message per node to be reached, which in a network with N nodes results in N-1 messages. As stated in [20] this can be achieved with a spanning tree graph. An interesting property of RNG is its local behavior. A classic spanning tree would join the set of nodes with the minimum possible number of edges for a connected graph, but algorithms building spanning trees require global knowledge of the network. On the other hand, RNG only requires local information of distance to the nodes' neighbors to determine if a link should be eliminated.

3. RNG based broadcasting over Internet graphs

In this article we propose the use of the RNG based broadcasting over Internet topologies. Since Internet nodes are difficult to pinpoint around the globe, we also propose the use of different metrics other than the geometric (Euclidean) distance. This metrics include the delay taken by messages across a link, a random number possibly under some distribution, etc. RNGs have been calculated with the L_p [11] and L_1, L_∞ metrics [18] (see also a review at [9]), but as far as we know it hasn't been defined with arbitrary metric and/or applied to Internet graphs.

We can observe that Relative Neighborhood Graphs are connected for any distance metric, which is proven in the following theorem.

Theorem 1. *The Relative Neighborhood Graph is connected on any metric space.*

> **Proof**. Let $\delta(a,b)$ represent the "distance" (as specified by selected metric) between two nodes a and b. Follow Theorem 3.1 from [24] using δ instead of d. We obtain that the Minimal Spanning Tree (MST) calculated with this metric is a subset of the RNG. The MST is connected therefore RNG is also connected ■

The Relative Neighborhood Graph is therefore applied on the internet topology to define an overlay network, according to the concepts described by Toussaint [24]. Here, it is important to note some key facts in the application of the RNG concept. The Toussaint's notion of neighborhood refers to the physical proximity among nodes, in fact in [24] the RNG is built from a given set of nodes initially disconnected whose interconnection is defined in terms of their physical proximity. In the case of Internet graphs (IGs) the RNG approach is slightly different because the initial graph is already connected with the edges distribution following particular patterns (power laws). Here the concept of neighborhood refers to the interconnection proximity rather than to the physical, in this sense two nodes are said to be neighbors if they are connected if communication link between them exists, regardless of their physical locations. Although in general overlay networks may use virtual link between two nodes which are not directly connected (but the route between them exists), we investigate here only overlay networks that are subgraphs of underlying IGs.

The core function of the RNG based broadcasting protocol is RNG graph derivation. The pseudo code to generate the RNG of a graph is described below. It starts with edges of the original graph and eliminates some edges in the process.

```
FOR_EACH node i
  FOR_EACH i.neighbor
    neighbors_list←i.neighbor
  FOR_EACH node j IN
                  neighbors_list
    FOR_EACH j.neighbor
      IF j.neighbor IS IN
           neighbors_list
         k←j.neighbor
         edgeij←distance(i,j)
         edgeik←distance(i, k)
         edgejk←distance(j, k)
         dmax←
           dmax(edgeik, edgejk)
         if dmax<edgeij
           DELETE longer edgeij
```

Pseudocode for obtaining the RNG

RNG based broadcasting can be used with both single flooding and double flooding models. In case of single flooding model, the number of messages sent is equal to the number of edges in the graph. In case of double flooding model, the number of messages sent is $1+(d-1)n$, where n is the number of nodes in the network, while d is the average density (number of neighbors) in the RNG graph with the metric applied. This average density d depends obviously on the metric applied. These measures are used in the cost definitions in the simulation section that follows.

4. Performance evaluation

A. Simulation

The broadcasting simulation over Internet Graphs (IG) is a two-step approach: first a topology is generated with a given number of nodes and edges giving a topology with a specific node average degree. Second, the RNG concept is applied to reduce the number of redundant edges in the graph. To generate the Internet like topologies, we use the three models described in section 2.A. The number of nodes n remains constant (n=1000) while the number of edges is modified by changing the average node degree d of the graph.

The performance of a given protocol is evaluated using two basic metrics: *reachability* and *cost*. The reachability is the percentage of nodes reached by a particular broadcast protocol. Note that blind flooding and RNG based protocol have reachability 100%, while Gossip protocol does not guarantee delivery to all nodes and therefore reachability data for it are included in tables. It is necessary to define the cost metric(s) in order to evaluate the savings obtained with the broadcasting protocols presented in this paper. A suitable metric is counting the number of retransmitted messages needed to reach the nodes in the network, since this number directly affects the available network resources in the nodes and links. Such an absolute measure is not easy to interpret, and a better measure is the relative cost with respect to a standard broadcasting scheme. Standard scheme being blind flooding, the (relative) cost is normally measured as the ratio of message count in a given protocol and message count in double or single flooding based blind flooding. The cost is directly related to the method used to diffuse the information, so we present how this cost is calculated in the scenarios used in our simulations.

The costs for the blind flooding and RNG based methods were already discussed in this article. The cost of broadcasting by Gossip protocol, as stated in [20] and [13], it is very difficult to obtain by analytical expressions, even for relatively simple topologies. As in [20] and [13], we also resort to simulation to compare the performance of Gossip to blind flooding and to RNG based broadcasting methods.

The cost is measured by performing 1000 broadcasting operations, each on separately generated IG and with randomly selected source. Each generated IG, in all three generators used, had n=1000 nodes. Each broadcasting operation starts by randomly selecting a starting node that sends a copy of the message to all its neighbors initiating the process. A round finishes when all the reached nodes have forwarded the message according to the scheme. Then, the total number of forwarded messages to complete the round gives the cost of that round. Finally the average cost is the average number of forwarded messages when performing all 1000 simulations.

To analyze the broadcasting processes, we deployed a series of classes and methods in the simulator. All the methods were programmed in Java, making use of the JDSL (Java Data Structures Library) [23] to implement some of the routines related to graph theory. The modularity design of the simulator allows switching between different broadcasting algorithms which as will is useful when comparing the performance of the RNG and Gossip protocols.

B. Performance of RNG over Internet graphs

The following table summarizes the results of applying RNG over the synthetically generated Internet graphs. To calculate the distances among nodes we simply assigned to each edge a random value between the interval (0, 1) which may represent the congestion or delay in the communication link. On real applications this information could be represented by the round-trip-time measures on the links.

Data in Table 2 is organized as follows: Average Degree is the average network density in the original graph. The Average Degree (RNG) is the average network density in the overlay network, after RNG algorithm was applied over the original graph. The *%Cost* is the percentage ratio of the number of forwarded messages in RNG based broadcasting versus blind flooding, both with single flooding model. That is, the cost is the percentage of edges from the original graph that remain in the RNG with random number as metric on edges. Three different Internet topology generators were used.

Barabási			Waxman			Palmer and Steffan		
Average Degree	Av. Degree (RNG)	%Cost	Average Degree	Av. Degree (RNG)	%Cost	Average Degree	Av. Degree (RNG)	%Cost
4.8	3.84	87	4	3.94	98	3.91	3.78	96
5.49	4.56	83	6	5.72	95	5.86	5.55	94
11.2	7.54	67	8	7.33	91	7.81	7.11	91
16.64	9.14	54	10	8.74	87	9.77	8.24	84
21.64	10.68	49	20	13.35	66	19.53	11.67	76
26.77	11.68	43	30	14.65	48	29.3	12.07	41

Table 2: Relative cost of single flooding RNG based broadcasting when using the Barabási, Waxman and Palmer and Stefan models.

It follows from Table 2 that the results obtained with the Barabási model have the highest savings among the three. In the case of the Barabási model, we can observe that an average node degree of approximately 20 achieves a reduction of almost 50%. Sparser original networks will have fewer savings: the savings in the cost is about 17% for network with an average node degree of approximately 6. This observation is important for Internet applications having a dynamic topology behavior such as the Gnutella network analyzed in [10]. In that work, the author demonstrated that peers in a Gnutella network have a high degree of interconnection with other peers in the network.

C Single versus double flooding based RNG and blind flooding broadcasting

The performance of single and double flooding based schemes, when all edges are used, can be compared theoreticaly as follows. In case of single flooding based schemes, the number of messages sent is equal to $nd/2$, where n is the number of nodes in the network, and d is its average density. If double flooding based scheme is used instead, the number of messages raises to $1+(d-1)n$. Therefore the relative cost of single flooding over double flooding is $nd/(2+2n(d-1)) = 1/(2/(nd) + 2(1-1/d))$. When n is large number, this approximates to $d/(2d-2) = 0.5/(1-1/d)$. For

d=4 this is 2/3, for d=6 this becomes 3/5, for d=8 it becomes 4/7, for d=10 it is 5/9, and for large d this approaches 50% of savings, as expected, since one instead of two directions on each edge is used.

D. Comparison of RNG and Gossip schemes

In [20] the Gossip protocol is presented as an alternative to blind flooding. We show in this section a comparison between RNG and Gossip. This is done by measuring the reachability and cost, as defined in this article. Table 3 shows the comparing costs with respect to the Gnutella (double) flooding representing 100%. Given that the RNG scheme guarantees reaching the all nodes but Gossip does not, we show multiple combinations of Gossip parameters B and F for making a more appropriate comparison.

Results in Table 3 show the performance data when using the Double flooding model over the RNG (DF-RNG) and the Double Gossip algorithms for Barabási topologies with n nodes, where n=1000, and d average degree, where d = 4, 6, 8, 10, 16, 20 and 24.

As shown before, the flooding Gnutella cost c can be theoretically calculated with the formula $c = 1 + n(d-1)$ that represents the total cost (100%). The results presented here were obtained counting directly the number of forwarded messages. The difference from those calculated with the formula comes from intrinsic properties in the implementation of the Barabási model. As a result of implementation, the final average degree d has a value approximately 1% smaller than the desired value.

Average degree	Cost of Gnutella Flooding	Double Gossip				RNG %Cost	Ratio RNG/Gossip
		B	F	%Reach	%Cost		
4	2998	2	2	82.3	53.4	96.10	1.80
4	2998	3	1	76.3	48.7	96.10	1.97
4	2998	3	2	93.4	69.2	96.10	1.39
6	4992	2	2	91.8	51.2	91.37	1.78
6	4992	3	1	87.1	46.3	91.37	1.97
6	4992	3	2	97.7	66.9	91.37	1.37
8	6989	2	2	92.8	41.6	87.58	2.11
8	6989	3	1	88.3	38.0	87.58	2.30
8	6989	3	2	98.7	61.1	87.58	1.43
10	8971	2	2	92.3	36.4	83.84	2.30
10	8971	3	1	88.0	29.5	83.84	2.84
10	8971	3	2	98.9	56.9	83.84	1.47
16	14929	2	2	91.3	21.7	72.95	3.36
16	14929	3	1	86.4	17.4	72.95	4.19
16	14929	3	2	98.8	38.4	72.95	1.90
20	18891	2	2	91.0	17.1	66.28	3.88
20	18891	3	1	86.8	13.8	66.28	4.80
20	18891	3	2	98.5	30.3	66.28	2.19
24	22845	2	2	90.9	14.1	59.80	4.24
24	22845	3	1	86.2	11.3	59.80	5.29
24	22845	3	2	98.4	25.0	59.80	2.39

Table 3: Comparison of costs and reachability when using the Double Gossip and DF-RNG schemes both compared to the Gnutella flooding.

It can be observed that the RNG scheme provides better performance than the Gossip scheme and that the savings increase as the density of the network increases for average degree values lower than 10. But for average degree values greater than 10 we can see that the savings obtained with RNG grow at a slower rate than that presented with the Gossip.

In Table 4 we present a comparison between the Single Gossip and the single flooding based RNG schemes. The data are given with respect to the Gnutella double flooding based scheme. The cost of single and double based Gossip protocols were very similar, since the performance is mainly dictated by parameters *B* and *F*. This Gossip implementation (Single Gossip) is the one that we believe was intended in [20] because it takes into account all the concepts presented in [13] allowing nodes to keep track of the neighbors that have sent messages to them, because in that case those nodes do not need to receive a new copy of the message *m*. The results obtained with our implementation of the Single Gossip protocol are shown below:

Average degree	Cost of Gnutella Flooding	Single Gossip				RNG %Cost	Ratio RNG/Gossip
		B	F	%Reach	%Cost		
4	2998	2	2	84.4	51.7	64.7	1.25
4	2998	3	1	81.3	49.4	64.7	1.31
4	2998	3	2	95.5	62.0	64.7	1.04
6	4992	2	2	93.4	48.3	55.7	1.15
6	4992	3	1	90.1	45.6	55.7	1.22
6	4992	3	2	98.8	56.5	55.7	0.99
8	6989	2	2	93.2	39.4	50.9	1.29
8	6989	3	1	90.3	37.8	50.9	1.35
8	6989	3	2	99.1	52.1	50.9	0.98
10	8971	2	2	94.5	35.1	47.5	1.35
10	8971	3	1	89.6	30.0	47.5	1.58
10	8971	3	2	99.4	48.1	47.5	0.99
16	14929	2	2	93.3	22.2	39.8	1.79
16	14929	3	1	87.9	17.6	39.8	2.26
16	14929	3	2	99.4	38.2	39.8	1.04
20	18891	2	2	92.8	17.4	35.8	2.06
20	18891	3	1	87.4	13.8	35.8	2.59
20	18891	3	2	99.2	30.6	35.8	1.17
24	22845	2	2	92.2	14.3	32.1	2.24
24	22845	3	1	86.8	11.4	32.1	2.81
24	22845	3	2	98.9	25.1	32.1	1.28

Table 4: Comparison of costs and reachability when using the Single Flooding Gossip and SF-RNG schemes both compared to Gnutella Flooding.

From data in Table 4 we can see that for low *d* values there is a reduction in the cost of Single Gossip which is predictable because some redundant messages were eliminated, therefore there is a slight growth in the percentage of reached nodes. These variations are almost imperceptible for *d* greater than 10, which can be explained because nodes have more neighbors to send the messages, reducing the likelihood of reselecting them. Again it can be seen that the Single Gossip protocol provides lower costs than the RNG.

Conclusions

We have introduced a RNG based broadcasting scheme for broadcasting over Internet topologies and presented it as an alternative to the blind flooding and Gossip approaches. Savings in message retransmissions are very significant for topologies with average degrees greater than 10 with average savings up to 25%. In general, savings increase with density when applying RNG based broadcasting.

We also showed that broadcasting over the RNG may also represent an alternative to the Gossip protocol since our algorithm guarantees a 100% of nodes reachability, while due to its probabilistic nature, Gossip does not. That is, RNG reduces the flooding cost without loosing the reachability of nodes in the network. Furthermore, only local nodes' information is needed to build up the Relative Neighborhood Graph. Although it can be argued that the Gossip protocol shows more savings in message retransmission we believe that these savings are apparent and not necessarily conduct to better performance in general because it does not consider nodes' local information, additional an advantage of the RNG based broadcasting is that messages are routed through minimal paths depending on the selected metric, per example delay or geographic distance.

There are several possible extensions of this work, which are planned by this group. First, an alternative to RNG will be considered. A recently published ingenious notion of localized minimal spanning tree [12] might be a suitable candidate, since it appears that geometric distance used there can again be replaced by an arbitrary metric. If geometric distance used, average density of about 2.4 in RNG can be reduced to about 2.04 in LMST (Local Minimum Spanning Tree) [12].

We would like to compare the overall cost of searching using our RNG or LMST based overlay network which use direct links and the cost of searching using the overlay networks as constructed in [21]. Each hop in a search following [21] in fact corresponds to several hops in underlying network, and all of them are to be counted. The goal is to estimate under what scenarios our RNG/LMST method may require less overall hops than DHT based schemes (for particular kind of overlay networks). Additional measure will be the update costs in both cases, when nodes are added or deleted in both cases. The hop measure may be replaced by some other measure such as overall delays in both cases.

RNG and LMST can also be considered as possible topologies for DHT based Internet search. To se used, the problem associated with addressing, ID selection and routing need to be resolved first.

Acknowledgements

This research is supported by CONACYT project 37017-A and NSERC grants.

References

[1] R. Albert and A. Barabási, *Statistical mechanics of complex networks*, Reviews of modern physics, 74 (2002), pp. 47–97.

[2] H. Balakrishnan, F. Kaashoek, D. Karger, R. Morris, and I. Stoica, *Looking up data in P2P systems*, Communications of the ACM, February 2003, pp 43-48

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

[4] O. Escalante-Mendieta, *Intelligent flooding over the Internet. BSc. Thesis*, Universidad Nacional Autónoma de México, IIMAS, October 2002.

[5] M. Faloutsos, P. Faloutsos, and C. Faloutsos, *On power-law relationships of the internet topology*, in SIGCOMM, 1999, pp. 251–262.

[6] P. Ganesan, Q. Sun; H. Garcia-Molina, *YAPPERS: A Peer-to-Peer Lookup Service Over Arbitrary Topology*, IEEE INFOCOM 2003.

[7] R. Govindan and H. Tangmunarunkit, *Heuristics for internet map discovery*, in IEEE INFOCOM 2000, Tel Aviv, Israel, March 2000, IEEE, pp. 1371–1380.

[8] W. Heinzelman, J. Kulik, and H. Balakrishnan, *Adaptive protocols for information dissemination in wireless sensor networks*, in MOBICOM, Seattle, 1999, pp. 174–185.

[9] J. W. JAROMCZYK AND G. T. TOUSSAINT, *Relative neighborhood graphs and their relatives*, Proc. IEEE, 80 (1992), pp. 1502–1517.

[10] M. A. JOVANOVIC, *Modeling Large-scale Peer-to-Peer Networks and a Case Study of Gnutella. Master Thesis.* DECE, University of Cincinnati. June 2000.

[11] J. KATAJAIEN, *The region approach for computing relative neighborhood graphs in the lp metric*, Computing, 40 (1988), pp. 147–161.

[12] N. LI, J.C. HOU AND L. SHA, *Design and analysis of an MST based topology control algorithm*, Proc. IEEE INFOCOM, 2003.

[13] M.-J. LIN, K. MARZULLO AND S. MASINI, *Gossip versus deterministic flooding: Low message overhead and high reliability for broadcasting on small networks*. Technical Report CS1999-0637, University of California, San Diego, Computer Science and Engineering, 1999.

[14] D. MAGONI AND J.-J. PANSIOT, *Internet topology analysis and modeling*, in CCW'01 - 16th IEEE Annual Computer Communications Workshop, Charlottesville, Virginia, October 2001.

[15] D. MAGONI AND J.-J. PANSIOT, *Internet topology modeler based on map sampling*, in ISCC'02 - 7th IEEE Symposium on Computers and Communications, Giardini Naxos, Italy, July 2002, pp. 1021–1027.

[16] A. MEDINA, A. LAKHINA, I. MATTA, AND J. BYERS, *BRITE: Universal topology generation from a user's perspective*, Tech. Rep. 2001-003, NEC, 1 2001.

[17] J. MOY, A. LINDEM, AND R. DUBE, *OSPF protocol charter*, IETF, http://www.ietf.org/html.charters/ospf-charter.html, 2002.

[18] J. O'ROURKE, *Computing the relative neighborhood graph in L_1 and L_∞ metrics*, Pattern Recognition, 14, pp. 45–55.

[19] C. PALMER AND J. STEFFAN, *Generating network topologies that obey power laws*, in IEEE GLOBECOM 2000, San Francisco, CA, November 2000, IEEE.

[20] M. PORTMANN AND A. SENEVIRATNE, *The cost of application-level broad-cast in a fully decentralized peer-to-peer network*, in ISCC 2002, Italy, July 2002.

[21] R. RINALDI AND M. WALDVOGEL, *Routing and data location in overlay peer-to-peer networks.* Research Report RZ-3433, IBM, July 2002.

[22] SEDDIGH, J. SOLANO AND I. STOJMENOVIC, *RNG and internal node based broadcasting algorithms in wireless one-to-one network*, ACM Mobile com-putting and communications review, 5 (2001), pp. 37–44.

[23] R. TAMASSIA, M. GOODRICH, ET AL, *JDSL - The Java Data Structures Library in Java,* Department of Computer Science, Brown University, http://www.jdsl.org/, 2002.

[24] G. TOUSSAINT, *The relative neighborhood graph of a finite planar set, Pattern recognition*, 12 (1980), pp. 261–268.

[25] B. WAXMAN, *Routing of multipoint connections*, IEEE Journal on Selected Areas in Communications, 6 (1988), pp. 1617–1622.

[26] Y. Jin, B. Zhang, V. Pappas, L. Zhang, S. Jamin, DIP: Distance information protocol for IDMaps, IEEE Int. Symp. on Computers and Communications ISCC, Turkey, July 2003.

[27] P. Francis, S. Jamin, C. Yin, Y. Yin, D. Raz, Y. Shavitt, and L. Zhang, IDMaps: A global Internet host distance estimation service, ACM/IEEE Transactions on Networking, Oct. 2001.