Broadcasting in Wireless Networks

Ivan Stojmenovic[1] and Mahtab Seddigh[2]
[1]SITE, University of Ottawa, Ottawa, Ontario K1N 6N5, Canada ivan@site.uottawa.ca
[2]Nortel Networks, Ottawa, Ontario, Canada

Abstract

In a multihop wireless network, each node has a transmission radius and is able to send a message to one of its neighbors (one-to-one) or all of its neighbors (one-to-all) that are located within the radius. In a broadcasting task, a source node sends the same message to all the nodes in the network. Some existing solutions apply rebroadcasting from each clusterhead or border node in a clustered structure. In this paper, we propose to reduce the communication overhead of broadcasting algorithm by applying the concept of internal nodes. The maintenance of internal nodes requires much less communication overhead than the maintenance of cluster structure of nodes. In one-to-all broadcasting, only internal nodes forward the message, while in the one-to-one case messages are forwarded on the edges that connect two internal nodes, and on edges that connect each non-internal node with its closest internal node. Existing notions of internal nodes are improved by using node degrees instead of their IDs (which are used only as secondary keys), and by adding a shortest path criterion. Highest node degrees are also proposed for reducing the number of clusterheads and border nodes in a clustering algorithm. The reduction in communication overhead for broadcasting task, with respect to existing methods, is measured experimentally. Further savings are obtained if GPS and the concept of planar subgraphs are used for one-to-one networks. In case of one-to-all model, GPS may be used to eliminate neighbors that already received the message, and re-broadcast only if the list of neighbors that might need the message is nonempty. The important features of proposed algorithms are their reliability (reaching all nodes in the absence of message collisions), significant savings in the rebroadcasting, and their localized and parameterless behavior.

1. Introduction

Wireless networks consist of static or mobile hosts (or nodes) which can communicate with each other over the wireless links without any static network interaction. Each mobile host has the capability to communicate directly with another mobile hosts in its vicinity. They can also forward packets destined for other nodes. Examples of such networks are ad hoc local area networks, packet radio networks, and sensor networks. They are used in situations like disaster rescues, wireless conferences in the hall, battlefields, or monitoring objects in a possibly remote or dangerous environment.

In a broadcasting task, a source node sends the same message to all the nodes in the network. The broadcasting in literature is studied mainly for one-to-all model and mainly in the context of flooding, in which the same message is retransmitted by all nodes that receive it. Such flooding is used for route discovery, for example (see [BMJHJ]). Other applications in the context of wireless networks include paging a particular host or sending an alarm signal. Lauer [L] and Pagani and Rossi [PR] proposed to replace flooding by a method where each clusterhead and gateway (or border, as renamed in this paper) node in a clustered wireless network forwards the message exactly once. In broadcasting algorithm [HKB], full message is sent exactly once to each node, while short messages (which inform nodes about existence of full messages) are flooded. The concept of internal nodes is proposed in [WL]. We propose in this paper to restrict broadcasting to internal nodes only, since any other node is directly linked to an internal node. Savings in communication time are confirmed experimentally, in addition to savings coming from applying internal nodes rather than clustering that has demanding maintenance procedure. For the purpose of the comparison, we have also described a broadcast efficient clustering algorithm, where node degree was used as primary key and node ID as secondary in clusterhead decisions. Highest node degree and a shortest path criterion were also applied here to improve the concept of internal nodes.

One-to-one model of broadcasting emerges when narrow beam directional antennas may be used, or when each node receives messages on its own frequency, and each node is aware of the frequency of each of its neighbors. Broadcasting in one-to-one model is studied only in [HKB].

Ad hoc networks are best modeled by the graphs constructed in the following way. Each node A has its transmission range $t(A)$. Two nodes A and B in the network are neighbors (and thus joined by an edge) if the Euclidean distance between their coordinates in the network is less than the minimum between their transmission radii (i.e. $d(A,B) < \min \{t(A), t(B)\}$) [BCSW]. If all transmission ranges are equal, the corresponding graph is known as the *unit graph*. We have used random unit graphs in our experiments.

Global Position System (GPS) provides location information (latitude, longitude and possibly height) to hosts in a wireless network. GPS cards will be, in the near future, deployed in each car and possibly in every user terminal. For instance, Differential GPS offers accuracy of a few meters. If GPS is available to the nodes in the network, internal node maintenance is incorporated into location updates between neighboring nodes. The concept of Gabriel graph [BMSU] (a kind of planar subgraph) can be applied to further reduce the communication overhead of a broadcasting one-to-one task, as described in this paper.

The broadcasting algorithms in [NTCS] achieve high ratio of nodes receiving the message with reduced amount of rebroadcasting in one-to-all model. Each of the methods from [NTCS] has a parameter whose best value may depend on network conditions, which is a global information. The concept of *localized* algorithms was proposed in [EGHK], as distributed algorithms where simple local node behavior achieves a desired global objective. Our broadcasting algorithms are localized, since each node needs only the information from its neighbors. They are also degree independent in two senses: there is no parameter in the algorithms that is set according to the network average degree *d*, and the performance of proposed algorithms appears to be relatively stable with respect to *d*.

Relevant literature review is given in the next section. Sections 3 and 4 describe new broadcasting algorithms for one-to-all and one-to-one models. Performance evaluations for the two models are given in sections 5 and 6, followed by a conclusion.

2. Literature review

2.1. Broadcasting

Broadcasting task was sometimes studied in the context of address serving. In a mobile ad hoc network in particular, due to host mobility, such operations are expected to be executed more frequently (such as finding a route to a particular host, paging a particular host, and sending up an alarm signal). In a ‘flat’ packet radio network, all of the nodes are visible to each other with respect to routing and, therefore, no location tracking is necessary. However, a flat network organization is not scalable, and hence some form of hierarchical clustering should be employed to hide information. The SURAN packet radio network is a prime example of a hierarchically clustered packet radio network, and we shall describe location tracking techniques in that context. Each node has a hierarchical address used in routing packet to that node. This hierarchical address is the sequence of enclosing clusters, starting with the highest level and ending with the lowest level, in which the node resides. Address servers located in each bottom-level cluster (thus associated with each clusterhead) keep track of a node’s hierarchical address. When a node wants to send traffic to another node, it queries the address server present in its cluster for the address of the destination node. To answer the query, the address server in turn may query other address servers. Each address server maintains a cache of responses from other address servers, which can be used to respond to future requests. For a detailed discussion on location tracking in hierarchical packet radio networks and the SURAN address server design, refer to [L].

Algorithms for address service require that an address server either be able to search for an address (by sending messages to all the other address servers asking if particular node is in their cluster), or update the other address servers (send messages indicating that particular node has joined or left its cluster). These searches and updates can use any of a variety of algorithms, including flooding, multicast along a spanning tree, and sending a packet directly to each address server. Flooding is reliable, but uses substantial network bandwidth; multicast on a spanning tree is very efficient, but is vulnerable to clusterhead failures; searching by sending a request to each address server is reliable.

Pagani and Rossi [PR] described a broadcasting protocol for ad hoc networks. It is based on clustered organization of nodes. Nodes are divided into clusters, with one of them serving as clusterhead in each cluster. Each clusterhead has direct link to any of the nodes in its cluster. Thus two nodes in the same cluster have hop distance at most two. They used algorithms for dividing nodes into such clusters as described in [EWB, GT]. In the broadcasting protocol [PR], the source node forwards the message to its clusterhead (CH), which then initiates the construction of virtual spanning tree of all CHs (two neighboring CHs are at hop distance two) by forwarding the message to all of them. More precisely, the message is sent to all neighboring CHs which in turn forward it to their neighboring CHs. Any node (ordinary node or a CH) rejects duplicate copies of the same message (that is, does not forward it upon repeated receptions). All CHs broadcast the message (with one transmission) to the nodes in its own cluster. The protocol [PR] is therefore similar to address searching algorithm [L] that sends a request to each address server (that is, CH). One packet is assumed for each message in [PR].

Since the performance of the broadcasting algorithm [PR, L] depends on the structure of clustering algorithm, we shall briefly describe them. The distributed clustering algorithm by Lin and Gerla [LG] is initiated by all nodes whose ID is lowest among all their neighbors (local lowest ID nodes). They broadcast their decision to create clusters (with them as CHs) to all their neighbors. Each node may hear the broadcasts by its neighbors and select the lowest ID among neighboring CHs, if any. If all neighbors which have lower ID sent their decisions and none declared itself a CH, the node decides to create its own CH and broadcasts its ID as cluster ID. Otherwise, it chooses neighboring CH with lowest ID, and broadcasts such decision. Thus each node broadcasts its clustering decisions after all its neighbors with lower IDs have already done so. Every node can determine its cluster and only one cluster, and transmits exactly one message during the algorithm. A sophisticated maintenance procedure for cluster formation when nodes move are also described in [LG].

Gerla and Tsai [GT] described a modified version of algorithm from [P], in which the highest degree node in a neighborhood becomes the clusterhead. More precisely, such nodes are elected as CHs, and their neighbors are then covered. The process then continues for the remaining uncovered nodes. An uncovered node is elected as a clusterhead if it is has the highest degree among all its uncovered neighbors. Although the algorithm is expected to perform well on many randomly defined graphs (as reported in [GT]), it may not produce any CH for graphs which do not have any node with the highest number of neighbors (interval and triangular graphs, where almost all nodes have degrees two and three, respectively). Thus the algorithm must be modified, as proposed in the next section.

The paper [HKB] describes a broadcasting algorithm called *SPIN* for sending a message from a node in sensor network to all other nodes. Each node that receives the datum that is being broadcast will forward corresponding *meta-datum* which has considerably shorter bit length (e.g. 16 bytes instead of 500). Sensor's *ID*, or a combination of variable names (e.g. (x,y)-coordinates of sensor, possibly rounded), are two examples. The actual datum could be all data that particular sensor collects. Neighboring nodes that did not yet receive the meta-datum will reply with request to get actual datum. Node will respond by sending the actual datum to all nodes that requested it. Suppose that node A has d neighbors, and that it forwards meta-datum (or actual datum) to k of these neighbors. The energy spent by node A to transmit is k* *bitsize* * 600mW (*bitsize* is 16 or 500). The energy needed by any node to receive such message is *bitsize** 200mW. The experiments carried on particular static sensor network and particularly selected parameters show 60% energy savings for *SPIN,* in one-to-one networks, and 80% for one-to-all case, with performance very close to theoretically optimal one (which is the case when actual data are sent without previous meta-data).

Ni, Tseng, Chen and Sheu [NTCS] studied the broadcast storm problem. Because radio signals are likely to overlap with others in a geographic area, a straightforward broadcasting by flooding is usually very costly and will result in serious redundancy, contention, and collision. They identified this broadcast storm problem by showing how serious it is through analyses and simulations. Several schemes (probabilistic, counter-based, distance-based, location-based, and cluster-based) to reduce redundant rebroadcasts and differentiate timing of rebroadcasts to alleviate this problem are proposed in [NTCS]. These schemes achieve high percentage of delivery rate with low number of retransmissions. However, they do not guaranty the delivery, which may be critical in some instances (e.g. alarm signal). In this paper, we are primarily interested in reliable broadcasts, whose goal is to ensure all hosts receive the message.

In the probabilistic scheme [NTCS], each node rebroadcasts the first copy of a received message with a given probability p. In the counter-based scheme, each node rebroadcasts the message if and only if it received the message from less than C neighbors. In the distance-based scheme, message is retransmitted if and only if the distance to each neighbor that already retransmitted the message is $>D$ (the distance may be measured by signal power or GPS). In the location-based scheme (which requires GPS for its application), message is retransmitted if and only if the additional area that can be covered if the node rebroadcasts the message (divided by the area of circle with transmission radius) is greater than the threshold A. A simplified version of the method is to rebroadcast the message if the node is not located inside the convex hull of neighboring nodes that already retransmitted the message. In the cluster-based scheme, lowest ID clustering algorithm [LG] is applied, and one of above four methods is then applied on clusterhead and border nodes (that is, on the dominating set created by clustering the nodes).

Experiments in [NTCS] have measured reachability *RE* (that is, delivery rate, the ratio of nodes receiving the message), the saved rebroadcasts *SRB* (the ratio of nodes that do not rebroadcast the message), and average latency until the last host receives the message. One hundred nodes are placed in an area of varying size and fixed transmission radius R, resulting in graphs with varying average degrees. RE and SRB in probabilistic, counter and distance-based method was highly dependent on the average node degree. Thus, in order to have a good overall performance, each node needs to know the average node degree, an information which is of global nature and not locally available to nodes. Cluster-based broadcasting is discussed elsewhere in this paper. Location-based method is, according to authors, the best performing method, taking advantage of GPS. However, [NTCS] observed that the reachability at sparser areas is unacceptable, even after clustering is applied first. *SRB* for low degree graphs seems to be low (compared to data obtained in this paper), for *RE* in the range 80-90%. We shall obtain guaranteed delivery (if the impact of collision is ignored) with higher *SRB* in this paper. Figure 1 illustrates a possible problem with location-based and all other methods mentioned here. Node *A* received broadcast message from its only neighbor *B*. According to each of probabilistic, counter, distance or location-based methods, *A* will rebroadcast the message although there is no other neighbor that needs to receive the message. Even lowest ID clustering may treat *A* as clusterhead (with 50% chance), thus imposing rebroadcast at *A*. This paper proposes to prevent such nodes from rebroadcasting.

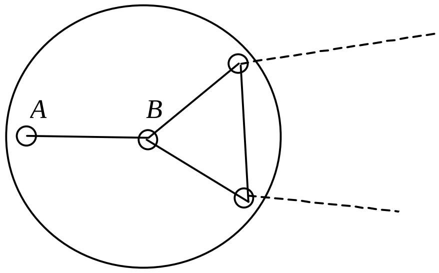

Figure 1. Node *A* receiving message from its only neighbor *B*

Let *disk(u,v)* be the disk with diameter *(u,v)*. Then, the Gabriel graph *GG(S)* is a graph in which edge *(u,v)* is present if and only if *disk(u,v)* contains no other points of *S*. *GG(S)* is a planar graph (that is, no two edges cross each other). The intersection of connected unit graph and Gabriel graph (defined over the same set of points) is a connected planar graph [BMSU].

Bose, Morin, Stojmenovic and Urrutia [BMSU] proposed a broadcasting algorithm for unit graph *G*, which is based on finding a planar subgraph *G"* of *G* (by intersecting *G* with Gabriel graph of the same nodes) and applying the algorithm of De Berg, van Kreveld, van Oostrum and Overmars [BKOO] which enumerates all the faces, edges, and vertices of a connected embedded planar graph *G*. It requires no memory at the nodes of the graph and uses only O(1) additional memory in the packet that is traveling around the network. The algorithm works by defining a spanning tree on the faces of *G* and performing depth first search on this spanning tree in $O(n^2)$ time, where *n* is the number of vertices of *G*. Although no memory requirement at nodes is a desirable property for a fully distributed algorithm, the message complexity high and impractical.

A number of centralized (where each node is assumed to know the full graph topology) broadcasting algorithms were proposed in literature. We are interested her only in distributed algorithms. Deterministic broadcasting in networks where nodes know only their *id* (as an integer between 1 and $Z=O(n)$) are studied in [CGGP]. An $O(n)$ worst case algorithm was proposed in [CGGP] for model with collision detection. In this paper, we are interested in expected time complexity, or average behavior. A randomized protocol that runs in expected time $O(D \log n + \log^2 n)$ was given in [BGI], where *D* is network diameter and *n* is number of nodes. [BGI] also pointed to exponential gap between deterministic and randomized algorithms. Our research and review are concentrated on the constant of proportionality, that is, the exact latency or number of rounds needed for broadcasting a message.

2.2. Dominating sets and internal nodes

Let *G* be the graph that corresponds to given wireless network. A set is dominating if all the nodes in *G* are either in the set or neighbors of nodes in the set. Nodes that belong to a dominating set will be called *internal* nodes for *G*. Routing based on a connected dominating set is frequently used approach [WL], where the searching space for a route is reduced to corresponding internal nodes. Such routing is suggested for shortest path and for dynamic source routing [WL], which do not use location information in routing decisions. If dominating set concept is applied, routing tables need to contain only information about internal nodes, and their size is therefore reduced. It is desirable, in this context, to create dominating set with minimal possible ratio of internal nodes. However, the savings in the communication overhead in maintaining routing tables shall not be overshadowed by the overhead imposed in maintaining dominating set structure.

We observe that clustering process is an example of creating a dominating set. The dominating set consist of all clusterhead nodes and border nodes. Border nodes have neighbors from at least two clusters (that is, they connect two CHs). The main drawback of using cluster structure as dominating set is its significant communication overhead for maintaining the structure in a moving environment. [WL] gave a literature review of several existing dominating set definitions which have significant overhead in maintaining the structure and do not produce better ratios of internal nodes than the simple definitions given in [WL].

Wu and Li [WL] proposed a simple and efficient distributed algorithm for calculating connected dominating set in ad hoc wireless networks. They introduced three concepts of internal nodes. A node *A* is an *intermediate* node if there exist two neighbors *B* and *C* of *A* that are not direct neighbors themselves. They introduced also two rules that considerably reduce the number of internal nodes in the network. Let $N(u)$ be the (open) set of all neighbors of node *u*, and let $N[u]=N(u) \cup \{u\}$ be the corresponding closed neighbor set, that is the set of all neighbors and *u* itself. Suppose that each node has a unique *id* number (for example, a random number in [0,1], or its *x*-coordinate). Let us define *inter-gateway* nodes as intermediate nodes that are not eliminated by Rule 1, and let the *gateway* nodes be those intermediate nodes that are not eliminated by both rules.

Rule 1 [WL] is as follows. Consider two intermediate nodes *v* and *u*. If $N[v] \subseteq N[u]$ in *G* and $id(v) < id(u)$, then node *v* is not an inter-gateway node. In other words, if any neighbor of *v* is also a neighbor of *u*, and *v* is connected to *u* and has lower *id*, then any path via *v* can be replaced by a path via *u*, thus node *v* is not needed as internal node. We may also say that node *v* is 'covered' by node *u*. The number of internal nodes can be further reduced by applying Rule 2 [WL], as follows. Assume that, after applying Rule 1, *u* and *w* are two inter-gateway neighbors of a inter-gateway node *v*. If $N(v) \subseteq N(u) \cup N(w)$ in *G* and $id(v) = \min \{id(v), id(u), id(w)\}$, then node *v* is declared a non-gateway node. In other words, if each neighbor of *v* is a neighbor of *u* or *w*, where *u* and *w* are two connected neighbors of *v*, then *v* can be eliminated from the list of gateway nodes (when, in addition, *v* has lowest *id* among the three). The hop count between a source and destination node may increase by one in this process, since a segment *pvq* of the path between them is replaced by a segment *puwq*.

If GPS is available, enabling nodes to know the location of all their neighbors, each node can determine whether or not it is an intermediate, inter-gateway or gateway node in $O(k^2)$ computation time (where *k* is the number of its neighbors), and without any message exchanged with its neighbors for that purpose. If GPS is not available then the maintenance of internal node status requires each node to know the list of neighbors for each of its neighbors.

3. Broadcasting in one-to-all networks

3.1. Reducing ratios of internal nodes

We propose to replace node *id*s with a record (*degree, x, y*), where *degree* is the number of neighbors of a node, and *x* and *y* are its two coordinates in the plane. In both rules from [WL], nodes compare first their degrees, and node with higher degree has greater chances of remaining an internal node. In case of ties, *x*-coordinate is used to resolve (or an *ID*, if GPS is not available). If *x*-coordinates also happen to be the same, use *y*-coordinate for final decision. Such comparison rule will result in fewer remaining nodes in the graph. The example in Fig. 2 has used so defined record instead of *ID*s. The information about the degree of neighboring nodes may be gathered together with information about their location.

In addition to definitions of internal nodes given in the previous section, we introduce notions *of SP-intermediate, SP-inter-gateways,* and *SP-gateways* by restricting each dominating set to those nodes that are on the shortest path (*SP*) between two its neighboring nodes. More precisely, *SP-internal* (where *internal* means either *intermediate, inter-gateway,* or *gateway*) node is an internal node *X* such that there exist two neighbors *A* and *B* of *X* from *G* such that $|AX|+|XB| \leq |AY|+|YB|$ for any other internal node *Y*, if *Y* is another common neighbor of *A* and *B*. The information from two hop away neighbors (and GPS) is required to apply the concept. Each node may periodically broadcast the list of its neighbors, with their coordinates, to enable the application of *SP-internal* nodes concept. The shortest path may be replaced by another measure (such as the power or cost of nodes [SL2]) in the definition.

3.2. Broadcasting via internal nodes

We shall propose new broadcasting algorithms separately for each of one-to-one and one-to-all network models. In both cases, the communication overhead of broadcasting algorithm is reduced by applying the concept of internal nodes. Since any node is the network has a neighbor that is intermediate node, broadcasting task can be performed by retransmitting the message by each internal node, to all (in one-to-all model) or some (in one-to-one model) neighbors.

In one-to-all broadcasting, each internal node forwards the message to all its neighbors, which is counted as one message. Although the concept of short and long messages from [HKB] can be applied, we believe that the overhead introduced by acknowledging short messages by several neighbors, and collision effect involved, is higher than if each internal node simply forwards the long message to all its neighbors at once. The comparison with the full flooding method (where each node forwards the message exactly once) is also simplified, since it suffices to find the ratio of internal nodes in a network. Comparison with algorithm [PR] is also simplified, since the number of messages in algorithm [PR] is equal to the number of clusterheads and border nodes in the clustered graph.

For example, Fig. 2 shows an unit graph with 12 nodes. Eight of these nodes (those that re shaded and named) are intermediate, while five of them are inter-gateway nodes (they are also gateway nodes). Thus broadcasting based on the internal nodes concept will require 8, 5 and 5 message transmissions for the case of intermediate, inter-gateway and gateway nodes (respectively), compared to 12 in case of full flooding.

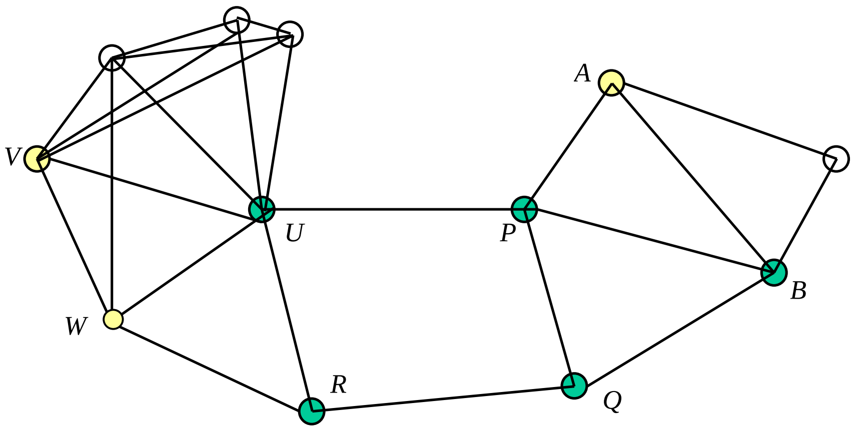

Figure 2. Intermediate nodes *AVWBUPQR,* inter-gateway and gateway nodes *BUPQR*

3.3. Improved clustering for broadcasting

In order to provide a fair comparison with cluster based broadcasting algorithm, we propose here a combined higher connectivity and lower *ID* clustering algorithm. Each node is assigned a pair *did=(d, ID),* where *d* is node degree and *ID* is its *ID*, which will be also called clusterhead priority. Let *did'= (d', ID')* and *did"= (d", ID").* Then *did'>did"* if *d'> d"* or (*d'=d"* and *ID' < ID"*). That is, a node has clusterhead priority over the other if it has higher connectivity or, in case of equal connectivity, has lower *ID.* The algorithms then follows the algorithm [LG], where lower *ID* clusterhead priority is replaced with higher *did* clusterhead priority described here. This algorithm will be referred to as the *ConID* algorithm.

This new algorithm produces reduced number of CHs and border nodes, as will be shown experimentally. Border nodes are nodes that have neighbors that belong to another cluster. In a cluster based broadcasting algorithm, each CH and border node will transmit the message exactly once. Therefore their total number needs to be optimized. For example, in Fig. 2 there are two CH nodes (*U* and *B*) and three border nodes (*P*, *Q*, and *R*) when this new algorithm (with *x*-coordinate serving as node *ID*) is applied. The lowest-ID algorithm [LT] produces four CHs (*V, R, P* and the rightmost node) and five border nodes (all except top three). Note that the count applies only if broadcasting is initiated by a node that is supposed to retransmit by the method. Otherwise (that is, if non-internal node or a node that is not a CH or border node is the source for a broadcasting task), the count should be increased by one, for transmitting the message from the source to one of its neighbors that is selected for broadcasting by the method. However, to simplify the comparison, we shall ignore that possible count difference in our experiments.

3.4. Improved broadcasting by neighbor elimination

We shall now propose an improvement for each of discussed broadcasting algorithms. It is based on the observation given in Figure 1, and availability of GPS. A node will re-broadcast the message only if it has a neighbor that might need the message. Thus some of neighbors are eliminated for re-broadcasting (they will, however, receive the message if it is sent because of other neighbors, as imposed by one-to-all model).

First, each node that is not supposed by the method to rebroadcast will assign itself to one of its possibly re-transmitting neighbors. In a clustered structure, this neighbor is the corresponding clusterhead. In the internal node structure we propose to assign the neighbor as follows: each non-internal node *A* will assign itself to neighboring internal node *B* which has the largest degree. In case of ties, use lowest ID among candidate neighbors. This rule attaches more neighbors to higher degree nodes thus possibly 'emptying' assigned list of low degree internal nodes.

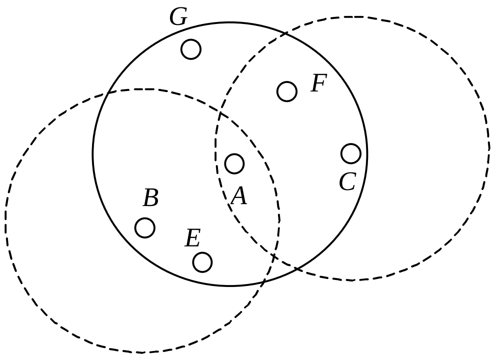

Figure 3. Node *A* eliminates neighbors *E* and *F* from its broadcast list

Second, neighbors that received one of message copies that arrived at a node *A* are also eliminated from the list of neighbors that might need the message. Consider, for example, node *A* in Fig. 3, which received twice the message which is being broadcast, from neighbors *B* and *C*. Neighbors *E* and *F* are eliminated from the broadcast list, since they received the same broadcast message from neighbors *B* and *C*, respectively. However, node *A* will, in this example, still re-broadcast the message because of neighbor *G* which is not 'covered' by *B* or *C*. Three circles around *A*, *B* and *C*, respectively, indicate their transmission ranges for these three internal nodes. Nodes *E*, *F* and *G* are either internal nodes (that is, belonging to dominating set), or non-internal nodes which are assigned to *A*. This scheme will further reduce the number of re-transmissions in a broadcasting task.

4. Broadcasting in one-to-one networks

In case of one-to-one network model, we shall adopt short and long messages concept from [HKB]. Each node will receive long message exactly once, which is optimal. Thus we will only try to optimize the number of short messages. The number is equal to the total number of edges in the algorithm [HKB]. We propose to reduce that number. Messages are only sent on edges connecting two internal nodes (one message per edge). The number of short messages is then equal to the number of edges in the subgraph of internal nodes. Each non-internal node, knowing all its internal node neighbors, will choose one of them and inform that one to send all broadcast messages to it. This algorithm will be refereed to as the *I-broadcast* algorithm (that is, broadcasting restricted to Internal nodes). The additional overhead (in this and in three more algorithms given below) when the source of broadcasting is not an internal node is ignored in order to simplify the discussion and experiments (it has no significant impact on the performance data).

When GPS is not available, intermediate, inter-gateway and gateway node concept may be applied. If GPS is available to nodes then the concept of SP-internal nodes may be applied as well. In addition, it is possible to apply the planar graph construction using Gabriel graphs. In case of unit graphs, the test for a given edge (whether or not it belongs to the Gabriel graph) may be applied to neighboring nodes only. Since location update between neighboring nodes is performed regularly for other reasons (i.e. routing), the selection of edges for the planar graph is local and requires no additional communication overhead. Edge *(u,v)* belongs to the constructed planar subgraph if and only if none of the common neighbors of *u* and *v* is located inside the disk with diameter *(u,v).*

We specify three location based broadcasting algorithms as follows. *P-broadcast* algorithm constructs planar subgraph (Gabriel graph) for all the nodes. The number of short messages in the algorithm is therefore equal to the total number of edges in the planar subgraph. *PI-broadcast* algorithm applies the planar subgraph construction first, and then applies the internal nodes concept on the subgraph. The result is different from the internal nodes applied on the whole graph. *IP-broadcast* algorithm changes the order of concept application compared to the previous algorithm. Internal nodes are first identified in the whole graph, and then the obtained subgraph (containing only internal nodes) is further reduced to planar one by the Gabriel graph construction.

5. Performance evaluation for one-to-all networks

The experiments were carried in several phases. In the first phase (section 5.1), an ideal MAC protocol is assumed, which provides for collision free broadcasting. The number of nodes that re-broadcast the message is counted, and compared to total number of nodes to evaluate the savings. The second phase (section 5.2) involves a real simulation using a MAC protocol, as described below. The third phase includes the proposed neighbor elimination scheme, while the fourth phase adds negative acknowledgement to the scheme for guaranteed delivery of message to all nodes (the experiments for last phase are in progress).

5.1. Internal nodes vs cluster based broadcasting

We have first compared the overhead of two clustering algorithms, *LowestID* and *ConID*. Table 1 gives the ratios of CH nodes and border (B) nodes, and the combined CH+B ratios, for random connected graphs with *n*= 100 nodes, and degrees from 4 to 12. The results indicate that *ConID* has consistently between 12% and 30% lower ratios compared to *LowestID*. We have also compared them for *n*=50, 200, 500 and 1000 nodes, and obtained similar results (more precisely, the CH+B ratios are quite close to those given for *n*=100). The percentage of border nodes increases with degree, while the percentage of CH nodes decreases for both algorithms. The CH+B ratio appears to be relatively independent from degree *d*, for each of algorithms, around 65% for *LowestID* and 52% for *ConID*. Thus the broadcasting algorithm based on *ConID* will have also less communication overhead. The comparison was made for static networks only. The cluster structure of nodes changes due to node movements, and existing maintenance algorithm do not provide for the preservation of highest degree CH preference. Thus the presented ratios for *ConID* appear to be the optimal one.

		LowestID		*ConID/LowestID %*		*ConID*	
d	CH	B	CH+B	%	CH+B	CH	B
4	0.31	0.36	0.67	75	0.5	0.25	0.25
5	0.27	0.39	0.66	70	0.46	0.21	0.25
6	0.24	0.37	0.61	84	0.51	0.2	0.31
7	0.21	0.42	0.63	87	0.55	0.19	0.36
8	0.2	0.47	0.67	76	0.51	0.16	0.35
9	0.18	0.41	0.59	88	0.52	0.15	0.37
10	0.18	0.46	0.64	84	0.54	0.15	0.39
11	0.16	0.48	0.64	81	0.52	0.13	0.39
12	0.15	0.52	0.67	78	0.52	0.12	0.4

Table 1. Ratios of CHs and border nodes in *LowestID* and *ConID* algorithms for 100 nodes

Table 2 compares clustering and internal node methods in terms of percentage of nodes that broadcast the message. Clearly, *intermediate* and *SP-intermediate* nodes concept does not achieve satisfactory gains. Gains for *inter-gateway* nodes are comparable to *LowestID* method, while *SP-inter-gateway* nodes are better than *LowestID* but somewhat worse than *ConID*. *Gateway* nodes method is comparable to *ConID*, while *SP-gateway* nodes are best overall, achieving broadcasting with guaranteed delivery with only 39-49% of nodes retransmitting the message. Interestingly, the ratios appear to be relatively stable with respect to degree *d*, for each of cluster of internal node based method. Thus the performance of the broadcasting algorithm, in terms of communication overhead, does not depend significantly on *d*. More precisely, ratios are quite stable for clustering and *(SP-)inter-gateway* methods, somewhat increasing (with increasing *d*) for *intermediate* and *SP-intermediate* methods, and somewhat decreasing for *(SP-)gateway* methods.

method/degree	4	5	6	7	8	9	10
LowestID	67	66	61	63	67	59	64
ConID	50	46	51	55	51	52	54
intermediate	80	84	88	91	92	94	95
inter-gateway	65	65	67	68	69	69	70
gateway	60	57	54	51	50	47	45
SP-intermediate	70	73	75	78	80	82	83
SP-inter-gateway	52	54	56	57	60	60	60
SP-gateway	49	48	45	44	43	40	39

Table 2. Percentage of broadcast nodes for each method for *n*=100 nodes and *d*=4-10

Recall that this comparison involves only retransmissions involved in direct broadcasting tasks, and for initial clustering structure. Communication overhead involved in maintaining clustered or internal node structures is not included. As noted before, maintenance of clustered structure (in the presence of moving nodes) is a non-trivial operation and may involve significant amount of message traffic. On the other hand, internal node structure requires only communication with neighbors when topology changes, and only when GPS is not available. If GPS is available, no additional overhead (except the one for location updates, also needed for the clustering task) occurs for non-SP methods, with minimal overhead for SP-based methods. Moving nodes pose additional problem to clustered structure, by loosening the highest degree CH property, which effectively moves *ConID* toward *LowestID* algorithm unless a global re-clustering (which also means additional overhead) occurs. Therefore, internal nodes concept does seem to perform broadcasting task with significantly lower communication overhead compared to existing method based on clustering, even if clustering process is optimized (as proposed in this paper). Clustering based broadcasting methods are therefore not evaluated further in this paper.

5.2. Internal nodes vs location based broadcasting

We have developed a simulator using C. A simplified version of the MAC specification in IEEE 802.11 standard is referenced to simulate carrier sense multiple access with collision avoidance (CSMA/CA) behavior among hosts. Since the speed of broadcasting a message is significantly larger than the node mobility, we assume that nodes are static while broadcasting is in progress. The experiments were therefore carried on random unit graphs, defined as follows. Each of *n* nodes is chosen by selecting its *x* and *y* coordinates at random in the interval [*0,m*). We experimented with *n*=100 nodes, as in [NTCS]. The average node degree *d,* the transmission radius *R*, and the map size *m* are related to each other. In order to compare our results with those from [NTCS], and to comply with IEEE 802.11 standard, we decided to fix radius *R* to 500 meters. The map sizes from [NTCS] are equal to *s*R*, for *s*=1, 3, 5, 7, 9, 10 and 11. The corresponding average node degrees *d* are 96.5, 25.4, 10.4, 5.6, 3.5, 2.9, and 2.4, respectively.

It is assumed that one broadcasting task at a time is in the network, and no other message traffic while broadcasting is in progress (this is fair assumption for comparing various broadcasting methods). We used the same parameters as in [NTCS]: the bit rate is 1M per second, the slot time is 20us (microseconds), the packet size is 280 bits which, with required overhead, took 2536us in their simulation. Global synchronization can be achieved by adding some dummy bits so that the transmission takes integer *p*=2540/20=127 number of slots. Acknowledgements are not sent. In this protocol, when a node *A* receives a packet to be transmitted, it first waits for an interframe spacing *DIFS* period (*DIFS=2* in our experiments). Node *A* then chooses a random integer *BC* (backoff counter) in interval [0..31]. The backoff counter determines the number of transmission free slots as sensed by *A*. During periods in which channel is clear, *A* decrements *BC*. When *BC* reaches 0, *A* transmits the packet. Once a node starts a transmission, it transmits continuously for *p* slots until packet is fully transmitted. Thus a neighboring node receives the packet if it receives collision free transmissions for the duration of *p* consecutive slots.

The performance metrics for comparison are [NTCS]:

- Reachability (*RE*): the number of nodes receiving the broadcast message divided by total number of nodes that are reachable from source (the graph may not be connected);
- Saved ReBroadcast (*SRB*): *(r-t)/r,* where *r* is the number of nodes receiving the broadcast message, and *t* is the number of nodes that actually transmitted the message.

	1 x 1	3 x 3	5 x 5	7 x 7	9 x 9	11 x 11
Intermediate	100	100	100	97	96	98
Inter-gateway	100	100	99	96	96	98
Gateway	100	99	97	94	95	98
SP-intermediate	100	100	99	97	96	98
SP-inter-gateway	97	100	98	87	82	83
SP-gateway	98	97	94	81	73	70
Location 0.1871	100	100	97	90	72	72
Location 0.0913	100	100	98	96	87	79
Location 0.0469	99	99	99	100	88	81
Location 0.0251	98	99	99	98	90	83
Location 0.0134	97	99	99	99	94	83

Table 3. Reachability for internal nodes and location based methods

Location based method is the best method among those presented in [NTCS], according to their experimental data. We will therefore compare proposed internal node based broadcasting methods only with the location based method (see tables 3 and 4). The average number of reachable nodes from source was 100, 100, 99.8, 85.7, 36.1, 17.2 and 11.3, respectively.

	1 x 1	3 x 3	5 x 5	7 x 7	9 x 9	11 x 11
Intermediate	1	1	5	14	22	32
Inter-gateway	97	32	29	34	37	43
Gateway	98	76	54	45	41	45
SP-intermediate	67	14	16	24	33	39
SP-inter-gateway	99	41	37	40	45	52
SP-gateway	99	81	60	49	47	53
Location 0.1871	90	78	52	37	30	22
Location 0.0913	78	55	36	26	17	12
Location 0.0469	67	42	26	17	12	8
Location 0.0251	61	36	21	12	8	5
Location 0.0134	56	32	28	10	8	6

Table 4. Saved Rebroadcasts for internal nodes and location based broadcasting

Among internal node based methods, the gateway nodes seem to provide best combined values for *RE* and *SRB*. *RE* values for location based method are comparable to gateway nodes method for *s*=1, 3, 5, and 7, but are lower for *s*= 9 and 11. *SRB* data are comparable to gateway nodes method only for *A*=0.1871 and only for *s*= 3 and 5, and are lower otherwise. Therefore, since a method is to be selected for arbitrary network density, the gateway nodes based broadcasting method is apparently better than any of methods proposed in [NTCS], and has no parameter associated with it. This is especially valid for networks with average degrees of nodes under 10, which is the range that includes throughput efficient networks.

The latencies of various methods were also compared. Gateway based method, for instance, has lower latencies that any of location based methods (for various parameter values). Differences were small for *A*=0.1871 but were increasing with increased value of threshold *A*. We then added the neighbor elimination scheme to the proposed internal node based broadcasting schemes. The differences in reachability data were negligible. Intermediate and *SP*-intermediate schemes benefited significantly from the elimination scheme in terms of *SRB* and latencies, while other methods had improvements in 1-6% range.

6. Performance evaluation for one-to-one networks

The performance evaluation of broadcasting algorithms in one-to-one network model was done using random connected unit graphs with *n*=100 nodes. The ratio of all edges in graph that are used for message (re)broadcasting over all edges in the graph is measured. This corresponds to savings in message retransmissions. Since our algorithms use portion of original messages, without adding any new message, rebroadcast savings can be claimed without using MAC layer in experiments. The amount of collision is further reduced, and even better comparison (if MAC layer were used) would have been obtained. Table 5 shows the percentages of retransmissions, compared to retransmissions using all edges in the graph. Because of similarity of results and space constraints, we show only the results with applying gateway node concept of internal nodes. The superiority of proposed methods is apparent. For example, for higher degree graphs, planar graph concept alone leads to significant reduction in number of message retransmissions, e.g. only 23% of retransmissions remained for degree 15 graphs. Internal node concept is 2-6% more efficient than planar graph concept. *PI-broadcast* gave minor improvements over *P-broadcast*, while *IP-broadcast* lead to notable improvement over *I-broadcast* and was best overall method.

degree	4	5	6	7	8	9	10	15	20	25	30	35	40
P-broadcast	67	57	50	45	41	37	34	23	18	14	12	10	9
PI-gateway	61	52	46	42	38	35	32	22	17	14	11	10	8
IP-gateway	57	47	41	36	32	29	26	17	12	9	7	6	5
I-gateway	61	52	47	41	38	35	32	21	15	12	9	7	6

Table 5. Percentages of retransmissions for one-to-one model

Conclusions

Our experiments have demonstrated the efficiency of proposed broadcasting algorithms for both one-to-all and one-to-one models. The broadcasting reliability is achieved with significant reduction in the number of re-broadcasting messages, resulting in reduced contention and collision problems in the network. In addition, internal nodes concept has reduced the maintenance communication cost compared to clustered structure.

We believe that further savings may be achieved by improving proposed algorithms in various ways. We plan to design broadcasting algorithm with guaranteed delivery (that is, 100% RE) by sending negative acknowledgements from nodes that experienced collisions. The elimination scheme was discussed here only for one-to-all model, and may be similarly applied to one-to-one case. Next, schemes that will use the location of the source may be applied, if the source location is part of broadcast message. Generally, each receiving node will forward the message only to neighbors that are further than itself from the source. However, if a broadcast message does not arrive within a reasonable time from a neighbor closer to the source, that neighbor will receive the message as well. Next, neighbors that are further will receive no message if there is evidence that another neighbor will forward the same message to such neighbors. We will investigate such methods in our future work.

References

[BKOO] M. de Berg, M. van Kreveld, R. van Oostrum, and M. Overmars, Simple traversal of a subdivision without extra storage, Int. J. of Geographic Information Systems, 11, 1997, 359-373.

[BGI] R. Bar-Yehuda, O. Goldreich, A. Itai, On the time complexity of broadcast in radio networks: an exponential gap between determinism and randomization, Proc. ACM Symp. Principles of Distributed Computing, 1987, 98-108.

[BMJHJ] J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, J. Jetcheva, A performance comparison of multi-hop wireless ad hoc network routing protocols, Proc. MOBICOM, 1998, 85-97.

[BMSU] P. Bose, P. Morin, I. Stojmenovic and J. Urrutia, Routing with guaranteed delivery in ad hoc wireless networks, 3rd int. Workshop on Discrete Algorithms and methods for mobile computing and communications, Seattle, August 20, 1999, 48-55.

[CGGP] B.S. Chlebus, L. Gasienec, A. Gibbons, A. Pelc, Deterministic broadcasting in unknown radio networks, Proc. SODA, 2000, to appear.

[EGHK] D. Estrin, R. Govindan, J. Heidemann, S. Kumar, Next century challenges: Scalable coordination in sensor networks, Proc. MOBICOM, 1999, Seattle, 263-270.

[EWB] A. Ephremides, J.E. Wieselthier and D.J. Baker, A design concept for reliable mobile radio networks with frequency hoping signaling, Proc. IEEE 75, 1987, 56-73.

[GT] M. Gerla and J.T.C. Tsai, Multicluster, mobile, multimedia radio network, Wireless networks, 1, 1995, 255-265.

[HKB] W.R. Heinzelman, J. Kulik, H. Balakrishnan, Adaptive protocols for information dissemination in wireless sensor networks, Proc. MOBICOM, Seattle, 1999, 174-185.

[L] G. Lauer, Address servers in hierarchical networks, Proc. ICC, 1988, 443-451.

[LG] C.R. Lin and M. Gerla, Adaptive clustering for mobile wireless networks, IEEE J. Selected Areas in Communications, 15, 7, 1997, 1265-1275.

[NTCS] S.Y. Ni, Y.C. Tseng, Y.S. Chen, J.P. Sheu, The broadcast storm problem in a mobile ad hoc network, Proc. MOBICOM, Seattle, Aug. 1999, 151-162.

[P] A.K. Parekh, Selecting routers in ad hoc wireless networks, ITS, 1994.

[PR] E. Pagani and G.P. Rossi, Providing reliable and fault tolerant broadcast delivery in mobile ad-hoc networks, Mobile Networks and Applications, 4, 1999, 175-192.

[SL1] I. Stojmenovic and X. Lin, GEDIR: Loop-free location based routing in wireless networks, IASTED Int. Conf. on Parallel and Distributed Computing and Systems, Nov. 3-6, 1999, Boston, MA, USA, 1025-1028.

[SL2] I. Stojmenovic and Xu Lin, Power-aware localized routing in wireless networks, IEEE Int. Parallel and Distributed Processing Symp., Cancun, Mexico, May 1-5, 2000, 371-376.

[WL] J. Wu and H. Li, On calculating connected dominating set for efficient routing in ad hoc wireless networks, Proc. DIAL M, Seattle, Aug. 1999, 7-14.

This research is supported by NSERC research grant.