

Multipoint Relaying for Flooding Broadcast Messages in Mobile Wireless Networks

Amir Qayyum, Laurent Viennot, Anis Laouiti

Project Hipercom, INRIA Rocquencourt, B.P. 105, 78153 Le Chesnay Cedex, France
Email: Amir.Qayyum@inria.fr, Laurent.Viennot@inria.fr, Anis.Laouiti@inria.fr

Abstract

In this paper we discuss the mechanism of multipoint relays (MPRs) to efficiently do the flooding of broadcast messages in the mobile wireless networks. Multipoint relaying is a technique to reduce the number of redundant re-transmissions while diffusing a broadcast message in the network. We discuss the principle and the functioning of MPRs, and propose a heuristic to select these MPRs in a mobile wireless environment. We also analyze the complexity of this heuristic and prove that the computation of a multipoint relay set with minimal size is NP-complete. Finally, we present some simulation results to show the efficiency of multipoint relays.

keywords: multipoint relays, mobile wireless networks, flooding of broadcast messages

networks. To obtain a satisfactory performance from these techniques or algorithms, they must be made adequate to this new and challenging environment.

Specifically, ad-hoc radio networks have an inherent capacity for broadcasting, i.e., with one emission, a node can reach all the nearby nodes. Using this capability for optimizing broadcast messages in such networks is a challenging task. A compromise has to be made between a small number of emissions and the reliability. Several techniques are described in the literature to limit or optimize the flooding of broadcast traffic [1], [2], [3], [4], [5], [6], [7]. Some results of comparison between these techniques can be found in [8]. Our paper discusses in detail the mechanism of “multipoint relaying” as one of the possible solution, first presented in [2]. Comparison with other techniques is reserved for future work.

1 Introduction

The research relating to mechanisms and protocols used in the wired networks is becoming mature. As a result, these mechanisms and protocols are now classified according to their relative domains of application, based on performance results obtained in those specific areas.

For mobile wireless networks, the research is still in its earlier stage. There is less consensus about the applicability of different existing techniques and algorithms in these new type of

1.1 Requirements of a mobile wireless environment

In “mobile” “wireless” networks, each of these two words put before us a list of requirements, and the daunting task is to fulfill them to their best. The mobility implies the limited lifetime of neighborhood or topology information received at any time, because of the movement of nodes. This implies that the information be updated regularly, otherwise it becomes invalid. More frequently

the information is updated, more the mobility of nodes can be handled correctly and efficiently.

The wireless nature of the medium implies the limited bandwidth capacity available in a frequency band. It is further reduced by the high bit error rate in radio transmissions. This makes it a scarce and hence a precious resource. Every effort is done to consume it very prudently. Hence, while designing a protocol using wireless links, the main task is to reduce the unnecessary use of this bandwidth.

Therefore, the requirements of these two environments are completely opposite to each other. Mobility requires more traffic to be sent in the network to keep the nodes informed of the changes, and at the same time, wireless medium does not allow to be used abundantly for unnecessary traffic. The compromise is to manage the mobility of nodes while using minimum of the bandwidth resources.

1.2 Flooding of broadcast messages in the network

The type of control traffic that is generated to manage the mobility of nodes in a network is mostly the information that a node declares about its relative movement, its new position, or its new neighborhood, etc. Some times, this information is useful only in the neighborhood of the node which is declaring the information. Therefore, the information is not required to be propagated in whole of the network to reach every node. But in many cases, not only the immediate neighbors of the declaring node, but the other far away nodes also need to know the topological changes occurring anywhere in the network. In these situations, lot of message passing is required in the network to keep the information consistent and valid at each node, by regularly announcing the changes due to mobility, or failure of links, etc.

The announcements about link changes are destined to each node of the network. But often all the nodes of network are not in the radio range of

each other to communicate directly. So there must be a mechanism to reach the far away nodes in order to keep them informed of the latest changes. The concept of intermediate nodes which serve as relays to pass the messages between the source and the destination is one of the solution.

If a message is for a specific destination, the determination of intermediate nodes is simple: all the nodes which form the path (if it exists) from the source up to the destination are the intermediate nodes. These nodes agree upon a mechanism to re-transmit the message, on their turn, so that the message is successfully transferred to the destination. Different routing protocols designate, in different ways, these intermediate nodes for unicast packet forwarding.

The problem arises when a packet is not destined to a specific node, rather it is a broadcast message for all the nodes in the network. The task of determining the intermediate nodes who will forward the packet is not trivial in this case. The nodes should behave such that the message is reached to every node in the network. A simple solution is that each node re-transmits the message, when it receives the first copy of the message. Fig 1 shows an example where a packet originated by node *S* is diffused up to 3-hops with 24 retransmissions. The packet is retransmitted by all the intermediate nodes in order to diffuse it in the network. This technique is known as “pure flooding”. It is simple, easy to implement, and gives a high probability that each node, which is not isolated from the network, will receive the broadcast message. The inconvenience of this technique is that it consumes a large amount of bandwidth because of so many redundant retransmissions.

In certain conditions, and particularly in the “wireless” networks, the availability of limited resources in terms of bandwidth capacity requires to restrict the traffic as much as possible. If this constraint of wireless medium is not considered while designing an algorithm, the network may suffer from performance degradations due to high over-

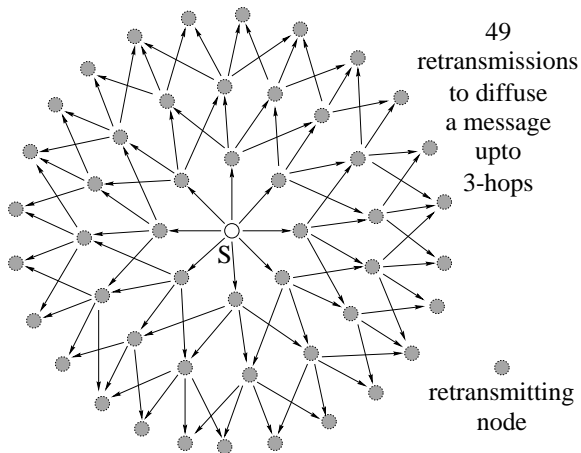


Figure 1: Diffusion of a broadcast message using pure flooding

loads or congestion, when the flooding of broadcast packets is launched in the network. On one hand, broadcast messages need some mechanism of flooding, specially in mobile environment to keep the mobile nodes remain in contact by regularly diffusing the updates. But on the other hand, it is not appreciable either to affect the actual working of the system due to this additional control traffic.

Every protocol uses some kind of flooding of control messages, for its functioning [9], [10]. It becomes very advantageous to optimize the resource consumption of the flooding process. Many techniques are described in the literature to limit the flooding of broadcast traffic and each technique has its own area of application and has its own advantages and disadvantages. Here, we will discuss the mechanism of “multipoint relaying” as one of the possible solution.

2 Multipoint relaying

The concept of “multipoint relaying” is to reduce the number of duplicate re-transmissions while forwarding a broadcast packet. This technique restricts the number of re-transmitters to a small set of neighbor nodes, instead of all neigh-

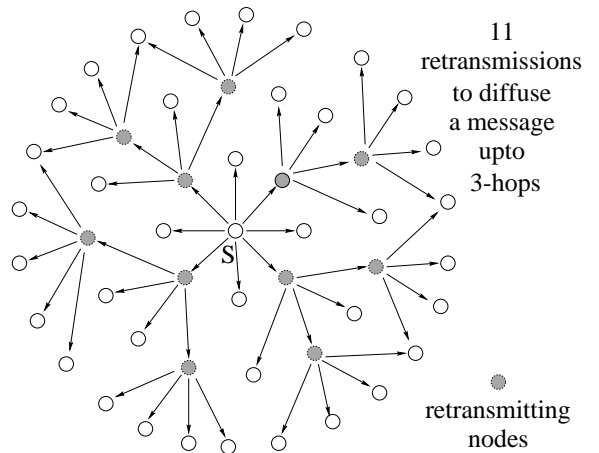


Figure 2: Diffusion of a broadcast message using multipoint relays

bor, like in pure flooding. This set is kept small as much as possible by efficiently selecting the neighbors which covers (in terms of one-hop radio range) the same network region as the complete set of neighbors does. This small subset of neighbors is called multipoint relays of a given network node. The technique of multipoint relays (or MPRs) provides an adequate solution to reduce flooding of broadcast messages in the network, while attaining the same goal of transferring the message to every node in the network with a high probability. Fig 2 shows an example where a broadcast message of node S is diffused in the network using the multipoint relays. In this case, it took only 11 retransmissions for a message to reach up to 3-hops.

Multipoint relaying technique works in a distributed manner, designed in view of the mobile and disperse nature of the network nodes. Each node calculates its own set of multipoint relays, which is completely independent of other nodes’ selection of their MPRs. Each node reacts when its neighborhood nodes change and accordingly modifies its MPR set to continue covering its two-hop neighbors.

An important aspect of the multipoint relays is the manner in which these multipoint relays are

selected by each node. The goal is to achieve the maximum performance by selecting an optimal set of MPRs by each node. But this task is not a trivial one. If the mechanism of selecting the MPRs is too simple, it may not select efficiently the MPRs in a dynamic and complex situation, and the expected performance gain would not be achieved. If the algorithm of MPR selection is very complex and sophisticated to provide a near to optimal MPR set, it may become difficult to implement it. A highly sophisticated algorithm may generate its own control traffic, to gather information for its functioning, which becomes comparable to the saving in flooding of messages. Thus, there must be a compromise in designing such an algorithm for the selection of multipoint relays: it should be easy to implement, and it should give near to optimal MPR set in “majority” of cases.

The information required to calculate the multipoint relays is the set of one-hop neighbors and the two-hop neighbors, i.e. the neighbors of the one-hop neighbors. To obtain the information about one-hop neighbors, most protocols use some form of HELLO messages, that are sent locally by each node to declare its presence. In a mobile environment, these messages are sent periodically as a keep alive signals to refresh the information. To obtain the information of two-hop neighbors, one solution may be that each node attaches the list of its own neighbors, while sending its HELLO messages. With this information, each node can independently calculate its one-hop and two-hop neighbor set. Once a node has its one- and two-hop neighbor sets, it can select a minimum number of one-hop neighbors which covers all its two-hop neighbors.

2.1 Heuristic for the selection of multipoint relays

We propose here one heuristic for the selection of multipoint relays. To select the multipoint relays for the node x , let us call the the set of one-hop neighbors of node x as $N(x)$, and the set of

its two-hop neighbors as $N^2(x)$. Let the selected multipoint relay set of node x be $M\mathcal{P}\mathcal{R}(x)$. The heuristic can be stated as:

1. Start with an empty multipoint relay set $M\mathcal{P}\mathcal{R}(x)$
2. First select those one-hop neighbor nodes in $N(x)$ as multipoint relays which are the only neighbor of some node in $N^2(x)$, and add these one-hop neighbor nodes to the multipoint relay set $M\mathcal{P}\mathcal{R}(x)$
3. While there still exist some node in $N^2(x)$ which is not covered by the multipoint relay set $M\mathcal{P}\mathcal{R}(x)$:
 - (a) For each node in $N(x)$ which is not in $M\mathcal{P}\mathcal{R}(x)$, compute the number of nodes that it covers among the uncovered nodes in the set $N^2(x)$
 - (b) Add that node of $N(x)$ in $M\mathcal{P}\mathcal{R}(x)$ for which this number is maximum.

To analyze the above heuristic, first notice that the second step permits to select some one-hop neighbor nodes as MPRs which must be in the $M\mathcal{P}\mathcal{R}(x)$ set. Otherwise the $M\mathcal{P}\mathcal{R}(x)$ will not cover all the two-hop neighbors. These nodes will be selected as MPRs in the process, sooner or later. Therefore, if the second step is omitted, the multipoint relay set can still be calculated with success, i.e. it will cover all the two-hop neighbors. The presence of step 2 is for optimizing the MPR set. Those nodes which are necessary to cover the two-hop set $N^2(x)$ are all selected in the beginning, which helps to reduce the number of uncovered nodes of $N^2(x)$ at the start of the normal recursive procedure of step 3.

3 Complexity analysis on the computation of multipoint relays

This section is devoted to the analysis of the computation of the multipoint relays. We will show that unfortunately, finding a multipoint relay set with minimal size is NP-hard. Nevertheless we will see that the above heuristic is within a $\log n$ factor from optimality. Let us first give a formal definition of the problem.

3.1 Formal definitions

If x is a node of the network, we denote by $N(x)$ the set of its one-hop neighbors. $N(x)$ is called the neighborhood of x . (Here we consider that $x \notin N(x)$.) Let $N^2(x)$ denote the two-hop neighbors of x .

If y is a one-hop neighbor of x , we also say that x covers y . Or we will simply say that y is a neighbor of x . Moreover, if S and T are sets of nodes, we say that S covers T iff every node in T is covered by some node in S . A set $S \subseteq N(x)$ is a multipoint relay set for x if S covers $N^2(x)$, or equivalently $\cup_{y \in N(x)} N(y) - N(x) \subseteq \cup_{y \in S} N(y)$. A multipoint relay set for a node x is optimal if its number of elements is minimal among all the multipoint relay set for x . We call this number the optimal multipoint relay number for x .

3.2 NP-completeness

We prove that the following problem is NP-complete:

Multipoint Relay: Given a network (i.e. the set of one-hop neighbors for each node), a node x of the network and an integer k , is there a multipoint relay set for x of size less than k ?

First of all, notice that this problem is easier than the problem of finding an optimal multipoint relay set. If an optimal set is known, simply computing its size and comparing it to k allows to an-

swer the question. Let us now show that the Multipoint Relay Problem is NP-complete.

It is obviously in NP since taking a random set in $N(x)$, one can easily check in polynomial time if it is a multipoint relay set and if its size is less than k . To prove that it is NP-complete, we prove that the following Dominating Set Problem which is known to be NP-complete [11] can be reduced to the Multipoint Relay Problem in polynomial time:

Dominating Set Problem: Given a graph (i.e. a set of nodes and a set of neighbors for each node) and a number k , is there a dominating set of cardinality less than k ? Where a dominating set is a set S of nodes such that any node of the graph is either in S or in the neighborhood of some node in S .

Let G be a graph with node set V and let $M(x)$ denote the neighborhood of any $x \in V$. We construct a reduction as follows. Let us make a copy of V and denote with a prime the copies: x' denotes the copy of x for any $x \in V$ and S' denotes the set of copies of the elements of any set $S \subseteq V$ (V' denotes the set of all the copies). Let s be an element neither in V nor in V' . Consider a network where the nodes are $\{s\} \cup V \cup V'$ and where the neighborhoods are the following (see Figure 3.2 for an example):

$$\begin{aligned} N(s) &= V, \\ N(x) &= \{x'\} \cup M(x)' \text{ for } x \in V, \\ N(x') &= \{x\} \cup M(x) \text{ for } x \in V \end{aligned}$$

Such a data structure can easily be computed in polynomial time. We claim that the answer to the Multipoint Relay Problem for the node s of the computed network with the integer k is valid for the Dominating Set Problem for the considered graph with the same integer k . It is sufficient to prove that any multipoint relay set S for the network is associated with a dominating set of the graph with same cardinality. S is a subset of $N(s) = V$. We show that S itself is a

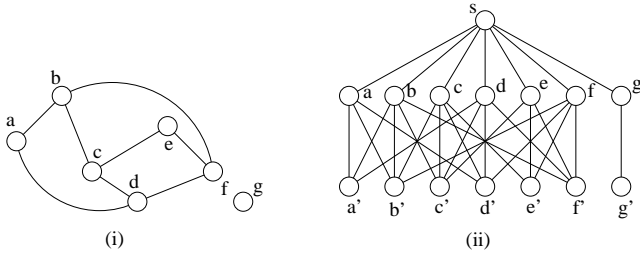


Figure 3: (i) A graph. (ii) The network obtained by the reduction. $\{b, c, g\}$ is a dominating set in (i) and a multipoint relay set for s in (ii).

dominating set of the graph. Consider a node $x \in V$ and its copy x' . As S is a multipoint relay set, x' is the neighbor of some node $y \in S$. As $N(y) = \{y'\} \cup M(y)'$ by definition, we have either $x' = y'$ or $x' \in M(y)'$, or equivalently, $x = y$ or $x \in M(y)$. This means that x is in S or is the neighbor of some node in S . S is thus a dominating set and the proof is achieved.

3.3 Analysis of the Proposed Heuristic

We prove that the heuristic proposed in section 2.1 computes a multipoint relay set of cardinality at most $\log n$ times the optimal multipoint relay number where n is the number of nodes in the network.

We give a proof directly inspired from [12] which is itself inspired from a general proof by Chvátal [13]. The first proof about an analogous heuristic was given in [14].

Let S_1 be the nodes selected in stage 2 of the above algorithm and let x_1, \dots, x_k be the nodes selected in stage 3 (x_i is the i th added node). Let S^* be a solution with minimal cardinality. First notice that $S_1 \subseteq S^*$ since any node in S_1 is the only neighbor of some node in $N^2(s)$. We will show that $|S - S_1| \leq \log n |S^* - S_1|$ which implies that the computed solution is within a factor $\log n$ from the optimal.

Let N^{2_1} be the set of nodes in $N^2(s)$ that are neighbors of some node in S_1 . We set $N^{2'} =$

$(s)N^2 - N^{2_1}$, $S' = S - S_1$, $S^{*'} = S^* - S_1$ and $N'(x) = N(x) \cap N^{2'}$ for each node $x \in N$. We associate a cost c_y with each node $y \in N^{2'}$. For each x_i chosen by the algorithm, a unit cost is equally divided among the nodes newly covered in N^2 . More formally: if x_i is the first neighbor of y added to S by the algorithm, then we set:

$$c_y = \frac{1}{|N'(x_i) - \cup_{j=1}^{i-1} N'(x_j)|}$$

The costs are linked with the cardinality of the computed solution in the following way:

$$|S'| = \sum_{y \in N^{2'}} c_y$$

We are going to show that for any node z in $S^{*'}$, we have:

$$\sum_{y \in N'(z)} c_y \leq \log |N'(z)| \quad (1)$$

Notice first that this implies immediately the result. Any node $y \in N^{2'}$ is the neighbor of some $x \in S^{*'}$ (remember that no node in S_1 is a neighbor of y by definition). We can thus deduce:

$$\begin{aligned} |S'| &= \sum_{y \in N^{2'}} c_y \leq \sum_{z \in S^{*'}} \sum_{y \in N'(z)} c_y \\ &\leq \sum_{z \in S^{*'}} \log |N'(z)| \leq |S^{*'}| \log n \end{aligned}$$

We still have to prove Inequation 1 to conclude. Let z be a node in $S^{*'}$ and let

$$u_i = |N'(z) - \cup_{j=1}^i N'(x_j)|,$$

for each $0 \leq i \leq k$ ($u_0 = |N'(z)|$)

be the number of neighbors of z in $N^{2'}$ which are still not covered after the choice of x_1, \dots, x_i . Let l be the first index such that $u_l = 0$. When x_i is chosen, $u_{i-1} - u_i$ neighbors of z are then covered. We can thus deduce:

$$\sum_{y \in N'(z)} c_y = \sum_{i=1}^l (u_{i-1} - u_i) \frac{1}{|N'(x_i) - \cup_{j=1}^{i-1} N'(x_j)|}$$

We then notice that the choice of x_i by the algorithm implies:

$$\begin{aligned} |N'(x_i) - \cup_{j=1}^{i-1} N'(x_j)| &\geq |N'(z) - \cup_{j=1}^{i-1} N'(x_j)| \\ &= u_{i-1} \end{aligned}$$

This implies:

$$\begin{aligned} \sum_{y \in N'(z)} c_y &\leq \sum_{i=1}^l (u_{i-1} - u_i) \frac{1}{u_{i-1}} \leq \int_{u_l}^{u_0} \frac{dt}{t} \\ &\leq \log u_0 \leq \log |N'(z)| \leq \log n \end{aligned}$$

The upper bound on the approximation factor follows. Notice that we can get a sharper bound on the approximation factor: it is bounded by $\log \Delta$ where Δ is the maximum number of two-hop nodes a one-hop node may cover. When a vertex covers at most 40 nodes, the approximation factor of the heuristic is below 3.7. When a vertex covers at most 100 nodes, the approximation factor of the heuristic is below 4.7.

4 Simulations

Some simulations have also been done to study the performance of the proposed heuristic in the computation of multipoint relays. The objective of the simulations was to compare two types of algorithms for the diffusion of packets in the radio networks: one is pure flooding technique, and the second is diffusion of packets using multipoint relays. The simulations aim at evaluating the behavior of these algorithms in the conditions of high error rates, either due to radio transmission problems or because of dynamic environment with rapidly changing topologies. We were interested in seeing the impact of these errors on the network with these two techniques. Moreover, we studied the limits of error rate up to which the algorithm of multipoint relays is able to ensure the diffusion and can guarantee satisfactory results.

4.1 Simulation model

Our study relates to large networks, in terms of number of nodes. We considered dense networks, so the nodes had a significant number of links with their neighbors. In order to assure the existence of a path from a node to all other nodes in the network, we considered the connected networks only, *i.e.* without any partitions or isolated nodes. The graph of the network was composed of a grid of nodes and their links. All the nodes were placed on the grid, to form a square network region. A radio range radius was defined, and all the nodes which were inside this radius were considered as the direct, one-hop neighbors. For all the simulations, we considered a graph of 1024 nodes placed on a 32x32 grid.

The simulations consisted of varying the probability of error of reception from 0 to 100%, and diffusing a message of a node in the whole network. This procedure was repeated for each node of the network to calculate an average of these values, for each value of the error probability.

In our simulations, we adopted certain assumptions to appropriately define the area of our study, which is the impact of error of reception on the diffusion of packets. These assumptions are as follows:

- The messages are broadcast messages which do not require an explicit acknowledgement to confirm the reception. Hence there was no retransmission when error of reception occurred;
- There are no uni-directional links. Each link between a pair of nodes is a perfect bi-directional link;
- The only traffic exists in the network is that of the diffusion of broadcast packet;
- Each node retransmits a packet (if it has to retransmit according to the protocol) only once;

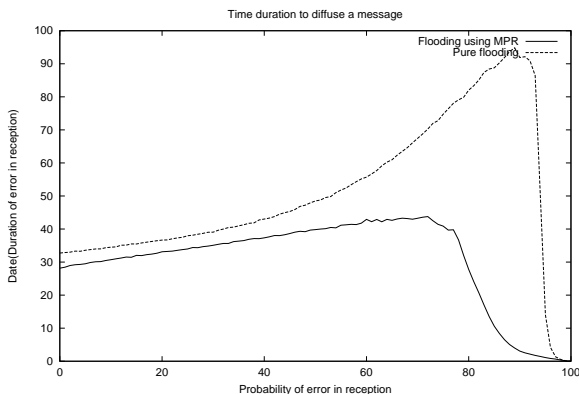


Figure 4: Completion of diffusion in network

- There is a synchronization among the transmissions. Channel is time-slotted and each transmission takes one slot;
- Each time a node transmits a packet, its one-hop neighbors receive this packet with probability P , where P is a percentage which lies between 0 and 100.

For a node to transmit, it was necessary that none of its neighbors up to 2-hops are transmitting. We call this as blocking of transmissions up to 2-hops. It was used to eliminate the problem of interference when a node receives two radio transmissions at the same time by two of its neighbors, which are not neighbors themselves.

4.2 Simulation results

Here we discuss some of the simulation results that we have obtained.

The Figure 4 shows that when a message was diffused in the network, how much time it took (in terms of clock ticks) so that all the nodes of the network get that message. It can be seen that pure flooding took more time as compared to multipoint relay technique, to diffuse the message in the network. In the Figure 5, we compare the time at which the transmission activity ended in the network which was started to diffuse a message.

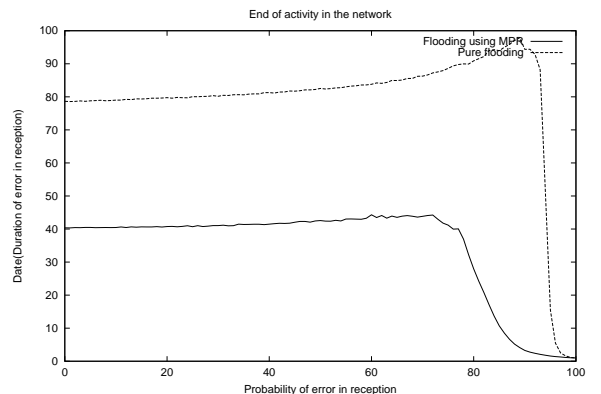


Figure 5: End of activity in the network

As expected, the pure flooding took almost double the time as compared to multipoint relay technique. This behavior can be explained as a result of packet retransmission by each and every node of the network, even when it is not needed. This can be proved by comparing the two graphs, and we can observe that when the packet is successfully diffused in the network, multipoint relaying technique took quite less time to stop further retransmissions, but pure flooding continued to retransmit, as each node must retransmit the packet, once, on its turn.

The Figure 6 shows how many of the nodes have retransmitted the message, on the average. For multipoint relaying, this figure was quite low because only selected nodes had retransmitted the packet, still achieving the comparable performance (as shown in rest of the graphs). In case of pure flooding, obviously it was all the 1024 nodes which retransmitted. As a consequence, the Figure 7 shows that in pure flooding, on average, the nodes have received too many duplicate copies of the same message as compared to the case of multipoint relaying.

In optimizing the flooding mechanism and reducing the traffic by multipoint relaying, there is a small price to pay, and that is the robustness of the protocol in varying conditions of error rates. In pure flooding, each node retransmits without

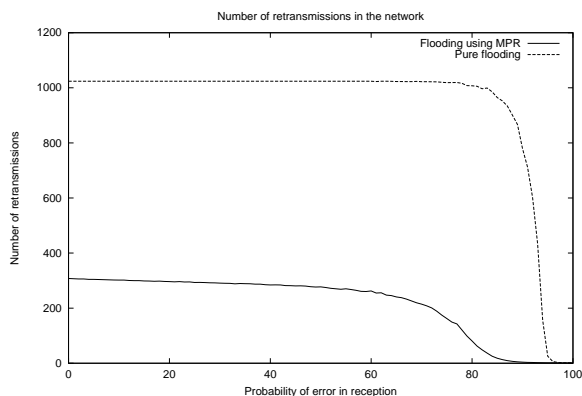


Figure 6: Number of retransmitting nodes

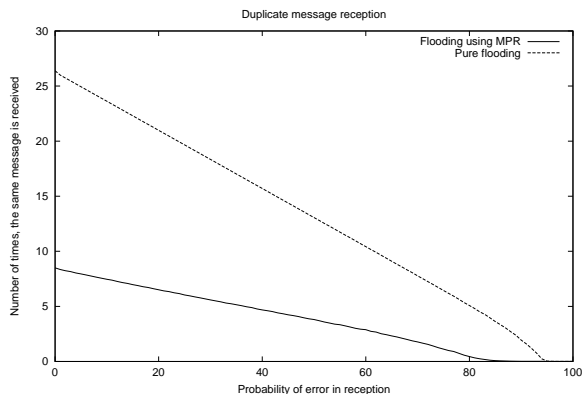


Figure 7: Number of duplicate receptions

exception, so there are more chances that the message reaches all the nodes, as compared to multipoint relaying, where only a selected number of nodes propagate the message. Figure 8 shows this fact, by comparing two protocols. We observe that when the error probability is higher than 20 or 25%, the multipoint relaying technique starts losing the packets, and some nodes do not receive the message because of these errors.

5 Conclusions

We have seen the performance of the two techniques, and according to the simulation results, the multipoint relaying has shown superiority

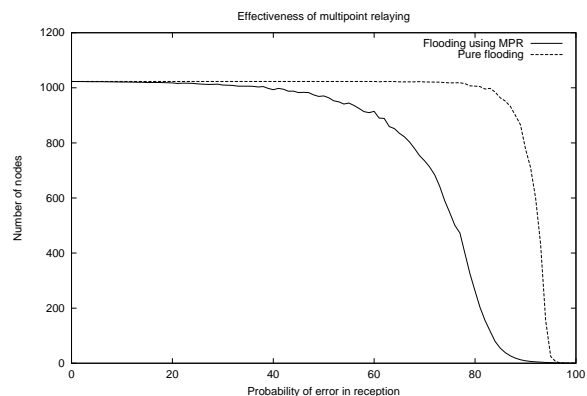


Figure 8: Effectiveness of diffusion in different conditions of error rate

over pure flooding scheme. The results of the simulation show that although the classic technique of pure flooding to diffuse a message in the network is more reliable and robust, it consumes a large amount of bandwidth as its cost. On the other hand, multipoint relaying gives equally good results, with much less control traffic, when the errors of reception remains less than 20%. In general, it's a quite realistic assumption to consider these errors as less than 10% in a network. So we can conclude that in the range of error rate which is most common, the multipoint relaying gives us quite satisfactory results, with a tremendous gain in performance due to quite a less traffic.

References

- [1] Philippe Jacquet, Pascale Minet, Paul Muhlethaler, and Nicolas Rivierre. Increasing reliability in cable-free radio lans: Low level forwarding in hiperlan. *Wireless Personal Communications*, 4(1):65–80, January 1997.
- [2] ETSI STC-RES10 Committee. Radio equipment and systems: High performance radio local area network (hiperlan) type 1, func-

- tional specifications, June 1996. ETS 300-652.
- [3] S. Datta, I. Stojmenovic, and J. Wu. Internal node and shortcut based routing with guaranteed delivery in wireless networks. In Proc. IEEE Int. Conf. on Distributed Computing and Systems, April 16-19 2001. Phoenix, AR.
- [4] G. Lauer. Address servers in hierarchical networks. In Proc. ICC, pages 443–451, 1988.
- [5] Wei Peng and Xi-Cheng Lu. On the reduction of broadcast redundancy in mobile ad hoc networks. In Proc. First Annual Workshop on Mobile and Ad Hoc Networking and Computing, pages 129–130, August 11 2000. Boston, USA.
- [6] S.Y. Ni, Y.C. Tseng, Y.S. Chen, and J.P. Sheu. The broadcast storm problem in a mobile ad hoc network. In Proc. MobiCom, pages 151–162, August 1999. Seattle.
- [7] J. Wu and H. Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In Proc. DIAL M, pages 7–14, August 1999. Seattle, USA.
- [8] I. Stojmenovic, M. Seddigh, and J. Zunic. Internal node based broadcasting algorithms in wireless networks. In Proc. IEEE Hawaii Int. Conf. on System Sciences, January 2001.
- [9] M. S. Corson, S. Papademetriou, P. Papadopoulos, V. Park, and A. Qayyum. An internet manet encapsulation protocol (imep) specification. Internet Draft, draft-ietf-manet-imep-spec-01.txt, August 21 1998, Work in progress.
- [10] P. Jacquet, P. Muhlethaler, and A. Qayyum. Optimized link state routing (olsr) protocol. Internet Draft, draft-ietf-manet-olsr-01.txt, February 7 2000, Work in progress.
- [11] M. R. Garey and D. S. Johnson. Computers and intractability, a guide to the theory of NP-completeness. W.H. Freeman, 1979.
- [12] T. Cormen, C. Leiserson, and R. Rivest. Introduction to Algorithms, chapter 37.3. MIT Press, 1990.
- [13] V. Chvátal. A greedy heuristic for the set-covering problem. Mathematics of Operation Research, 4(3):233–235, 1979.
- [14] S. Sahni and T. Gonzales. P-complete approximation problems. Journal of the ACM, 23:555–565, 1976.