

Technical Report CU-CS 1074-10
September 2010

GPS Assisted Broadcasting in Mobile Ad-Hoc Networks

by

Alireza Mahdian
Department of Computer Science
University of Colorado at Boulder
Boulder, Colorado 80309

Chris GauthierDickey
Department of Computer Science
University of Denver
Denver, Colorado 80208

Mario Lopez
Department of Computer Science
University of Denver
Denver, Colorado 80208

Shivakant Mishra
Department of Computer Science
University of Colorado at Boulder
Boulder, Colorado 80309

ABSTRACT

Broadcast protocols play a vital role in building services and applications for multihop, mobile, wireless ad hoc networks (MANET). This paper describes the design, implementation and evaluation of a new broadcast protocol for a multihop, mobile wireless ad hoc network that provides significantly better performance than the currently existing protocols. In particular, this protocol provides lower end-to-end delivery latency and higher coverage than the existing protocols. It requires relatively small number of message exchanges, minimizes collisions, and tolerates dramatic changes in the network due to node movement. The protocol uses the unit disk graph model and a clever combination of FDMA and TDMA.

1 Introduction

A mobile ad hoc network (MANET) is a self-configuring network of mobile devices (nodes) connected by wireless links. In such networks, each node has a limited transmission range, so that a message sent from a node can be received by all nodes within its transmission range. Two nodes that are not directly connected to each other can communicate by using one or more intermediate nodes as relay nodes (multihop). Each node in a MANET can move independently in any direction, and as a result, communication links can change frequently in these networks. Broadcast is a fundamental operation in a multihop, mobile, wireless ad hoc networks as it enables a source node to efficiently send a message to all other nodes in the network. A large number of network protocols, services and applications depend on an underlying broadcast operation to work correctly. Examples include updating routing tables, disseminating sensor network commands, and disseminating emergency messages.

Broadcasting in MANETs is complicated by three important issues. First, because wireless transmission is by nature broadcast based, there is the issue of interference in the wireless medium. When two or more nodes transmit a message to a common neighbor at the same time, the common node cannot receive any of these messages, because the wireless signals from the two sender nodes collide at the receiver node. Thus, intermediate nodes that forward a broadcast message must time their transmissions appropriately to minimize signal collisions. Second, due to the dynamic nature of MANETs where nodes may move at any time, there is no well-established infrastructure. As a result, maintaining routing tables is a major challenge. Finally, resources are relatively scarce in MANET. In particular, nodes have limited battery life and bandwidth is limited. Message transmissions consume both power and bandwidth. Hence, it is important to design broadcast algorithms that minimize overall power consumption and number of message exchanges.

There are four important criteria based on which a broadcast algorithm is evaluated: end-to-end latency, coverage, power consumption, and ability to cope with dramatic changes in the network due to node movement. End-to-end latency is the time interval between the time when the source sends out a message and the time when the last node in the network receives the message. Naturally, it is desirable to have low end-to-end latency. Coverage is the percentage of nodes in the network that successfully receive a broadcast message. While the goal is to get 100% coverage, it becomes difficult due to node movements, collisions, and a need to reduce end-to-end latency and minimize energy consumption. Power consumption is the overall energy consumed by a broadcast. Past research has shown that message transmissions result in maximum power consumption. Thus, the number of messages exchanged must be minimized to reduce overall power consumption. Finally, the ability to cope with dramatic changes in the network measures the variation in end-to-end latency, coverage and power consumption in the presence of different rates and frequency of node movement.

In this paper, we describe the design, implementation and an extensive evaluation of a new broadcast protocol for MANETs. The protocol uses the unit disk graph model and a clever combination of FDMA and TDMA to provide significantly better performance than the currently existing broadcast protocols. In particular, this protocol provides lower end-to-end delivery latency

and higher coverage than the existing protocols. It requires relatively small number of message exchanges, minimizes collisions, and tolerates quite well any dramatic changes in the network due to node movement.

The rest of this paper is organized as follows. Given the importance of broadcast in MANET, it is not surprising that a large number of broadcast algorithms have been proposed in recent years [1, 3, 5, 6, 8, 11]. Section 2 provides a summary of these algorithms. Section 3 provides a detailed description of our protocol and Section 4 provides an extensive evaluation. Finally, Section 5 concludes the paper.

2 Related Work

Williams and Camp classified broadcasting protocols by various properties such as node densities, mobility, and traffic rates, and broadly defined them into four categories: simple flooding, probability-based methods, area-based methods, and neighbor knowledge methods [14]. Our protocol is a combination of area-based and neighbor knowledge methods.

Minimum latency broadcast scheduling (MLBS) is known to be NP-hard [4]. As such, approximation algorithms have been developed to efficiently broadcast by reducing collisions. In simple flooding, the source sends a message to all neighbors and each neighbor that receives a new broadcast message further sends it to its neighbors until all nodes receive the message [10].

Chen et al. designed a constant approximation algorithm for interference aware broadcast that considers both the broadcast and interference range of transmissions in order to reduce the *makespan*, which is the shortest time for every node to receive a message [3]. Their centralized algorithm provides collision free scheduling with a worse-case performance no more than 26 times the optimum solution when the interference range is twice the transmission range. Our protocol provides collision free scheduling while being fully distributed.

Using unit-graphs as the network model, Huang et al. developed minimum-latency broadcast scheduling algorithms which improved on the best-known approximation algorithms, resulting in latencies as low as $R + O(\log R)$, where R is the radius of the network from the perspective of the broadcast source [6]. Their centralized algorithm is collision free, though their distributed algorithm is not.

Gandhi et al. further show that minimizing latency in broadcasting in ad hoc networks is NP-complete and then develop a distributed, collision free broadcast algorithm that guarantees the latency and number of transmissions [5]. Their distributed algorithm relies on two DFS traversals in the network to create the schedule. Our protocol solely relies on positional and neighbor information to schedule the broadcast.

The distributed construction of energy efficient broadcast trees was researched by Ahluwalia and Modiano and uses a two-stage approach to constructing the single-source broadcast trees which minimize energy usage [1]. Papadimitriou and Georgiadis further provide a polynomial approximation algorithm to construct similar energy-efficient broadcast trees [11]. While we do not focus on building energy efficient broadcasts, the small number of messages sent by our protocol will result in low energy consumption.

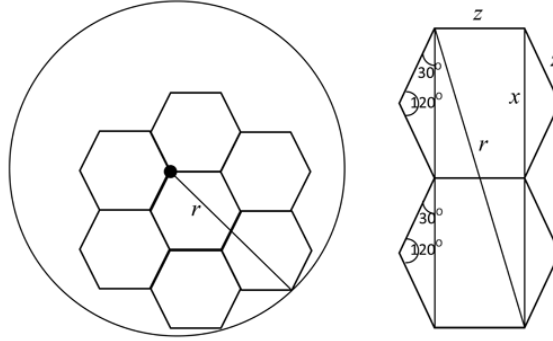


Figure 1: Relationship between the edge of a hexagon and the transmission range of a node.

An important issue that should be addressed in wireless ad hoc networks is the problem of interference and conflicts, e.g., when the same broadcast message from two different nodes collides due to simultaneous transmissions. Mahjourian et al. designed an approximation algorithm for conflict-aware broadcast scheduling to address this issue [8]. Our protocol also avoids collisions, but uses positional information to schedule broadcasts.

3 Description

In this section we describe our broadcasting algorithm which employs a subset of nodes to forward the message. The selected forwarding nodes would construct a dominating set on the networks graph and cover all nodes. In order to construct the dominating set, we use a hexagonal tessellation of the entire grid (terrain). The size of each hexagon is proportional to the transmission range of a node. For simplicity, we assume that the transmission range of all nodes is same. Also, we use the unit disk graph (UDG) model to derive the network graph. This means that two nodes are connected to each other if a disk with a radius equal to the transmission range around one of the nodes would cover the other as well.

In a hexagonal tessellation, the plain is covered with non-overlapping hexagons of equal size. We say that a node j belongs to a hexagon i if it is either inside that hexagon or on its boundaries, and we show this relationship with $j \in i$. In this manner, each hexagon would have six other hexagons as its neighbors. We denote this set of neighboring hexagons for hexagon i with $H(i)$. We compute the size of each hexagon in such a way that a node placed inside or on the boundary of hexagon i would cover all nodes inside or on the boundaries of $H(i)$. Based on this definition, the worst case scenario, i.e. the maximum distance between two nodes in neighboring hexagons, occurs when a node is right at a corner of a hexagon (see Figure 1). Applying our coverage constraint for neighboring hexagons, the size of each edge of a hexagon is computed as follows:

$$x = 2z \cos(\pi/6), \text{ also } r^2 = (2x)^2 + z^2. \text{ This means that } r^2 = (16 \cos(\pi/6)^2 + 1)z^2 \Rightarrow z = \frac{r}{2\sqrt{4 \cos(\pi/6)^2 + 1/4}} = \frac{r}{2\sqrt{5.25}}$$

Based on the size of each hexagon, a node belonging to hexagon i would cover all nodes in

$H(i)$ in addition to all nodes in i itself. Assuming that the network graph is connected, it is evident that selecting a single node from every non-empty hexagon would derive a dominating set. This dominating set is of course not independent since all nodes in two adjacent hexagons are connected to each other. Therefore they would form a subgraph that can be used to propagate the packets.

We use this idea to develop an efficient broadcasting algorithm. Our goal is to minimize the following metrics: end to end delay, number of transmissions, and number of collisions. The first and the second metrics are functions of how we build the dominating set and the last one depends on how we schedule message transmissions. First we explain our approach, and then provide theoretical bounds and experimental values for our three metrics.

Based on Figure 2, transmission from two nodes can collide if the two nodes transmit at same time and distance between them is less than or equal to twice the transmission range. In other words, a collision occurs when two nodes are two hop neighbors (i.e. they are at the two hop neighborhood of each other) and they are transmitting at the same time. If we color the hexagons so that all the hexagons covered by a disk of radius $2 \times \text{transmission range}$ are assigned different colors¹, we are guaranteed to have a collision free broadcasting if the message is transmitted by a single node from every non-empty hexagon at a timeslot which corresponds to the color assigned to that hexagon. This way we obtain our goal of minimizing the number of transmission collisions. Assuming that the transmission time for each direct link between two nodes is α^2 , the end to end delay at node j cannot be less than $BFS(j) \times \alpha$ where $BFS(j)$ denotes the height of j in the breadth first search tree with the source of the broadcast being the root of the BFS tree. Using the above coloring approach, the end to end time at node j is at most $85 \times BFS(j) \times \alpha$.

If we color with only *seven* colors we can get a hexagonal tessellation in which for each hexagon x , the set of hexagons $x \cup H(x)$ have different colors. This implicitly means that every pair of adjacent hexagons have different colors as well. The problem with this coloring is shown in Figure 3. The dots show the transmitting nodes and the circles around each dot shows its transmission range. Each node belongs to a gray shaded hexagon. Suppose that the center node is transmitting at timeslot 0 (based on its color) then the other 6 nodes in the gray shaded hexagons can also transmit at the same time causing collisions in all hexagons that are covered by at least two circles. This problem is caused because for a node k belonging to hexagon i , there are exactly 6 other hexagons with the same color (timeslot) as i inside the two hop neighborhood of k ; the two-hop neighborhood is represented by a disk of radius $2 \times \text{transmission range}$ centered at node k . A way to avoid this problem is to do another layer of coloring for each hexagon. Let $C_1(i)$ and $C_2(i)$ denote the first and second layer colors of hexagon i . This coloring is done in such a way that every pair of hexagons that are covered by a disk of radius $2 \times \text{transmission range}$ and have the same first layer color are assigned different second layer colors. This second layer of coloring can be mapped to

- different frequencies, which can be achieved by employing Frequency Division Multiple Access (FDMA), or
- smaller timeslots using Time Division Multiple Access (TDMA).

¹We can do this coloring with 85 colors.

²This is a function of both bandwidth and packet size.

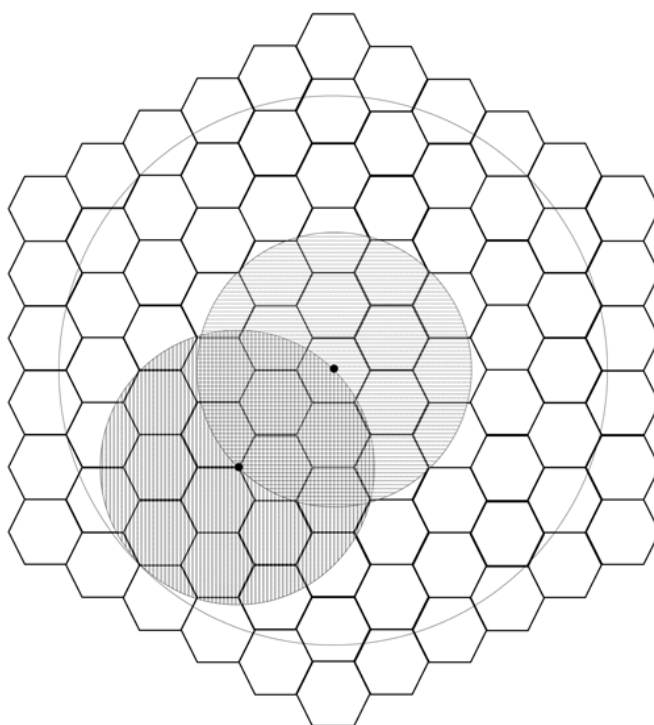


Figure 2: A simple collision scenario where the two nodes (represented by dots) transmit at the same time causing collision at the overlap of the shaded circles.

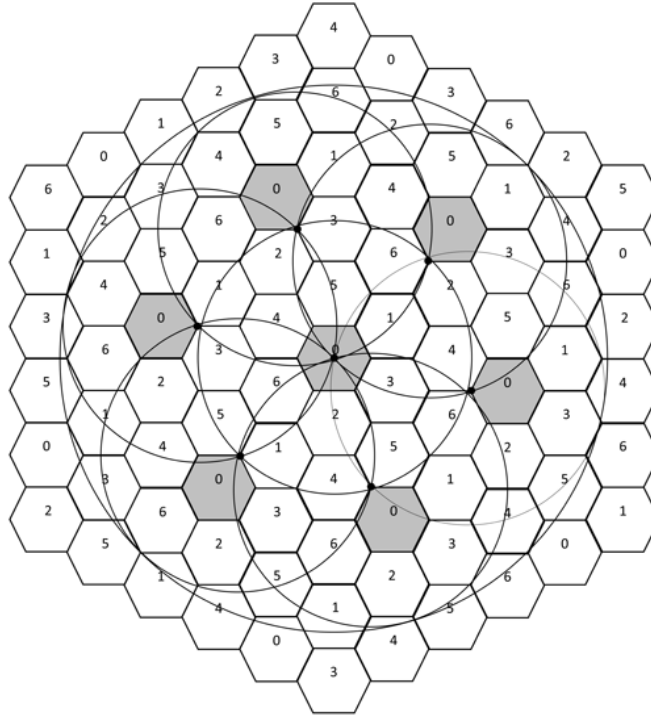


Figure 3: A scenario where the nodes at the same colored (gray) hexagons can still cause collision.

3.1 FDMA

Using different frequencies means that each node belonging to hexagon i would transmit at timeslot corresponding to $C_1(i)$ using frequency $C_2(i)$. Nowadays, different wireless standards allow for different wireless channels to be used, and this approach can be employed if the standard allows for seven different non-overlapping frequencies. Of course in this scenario each receiving node needs to tune into a particular frequency at each timeslot in order to receive the packet. This means for a very small period of time the receiving node needs to scan different frequencies (4 to be exact) in order to find the strongest signal which must be added when computing the end to end delay. This approach would result in a maximum end to end delay of $7 \times (BFS(j) + \beta)$ at node j , where β is the frequency tuning time³.

3.2 TDMA

Using smaller timeslots (*micro timeslots*) means that each timeslot (*macro timeslots*) that corresponds to a single first layer color is divided into seven smaller timeslots each corresponding to a second layer color. There are two constraints that need to be considered when determining the length of the micro and macro timeslots: the length of each micro timeslot should be large enough

³The value of β is hardware dependent and is usually in the range of 30 to 200 μs .

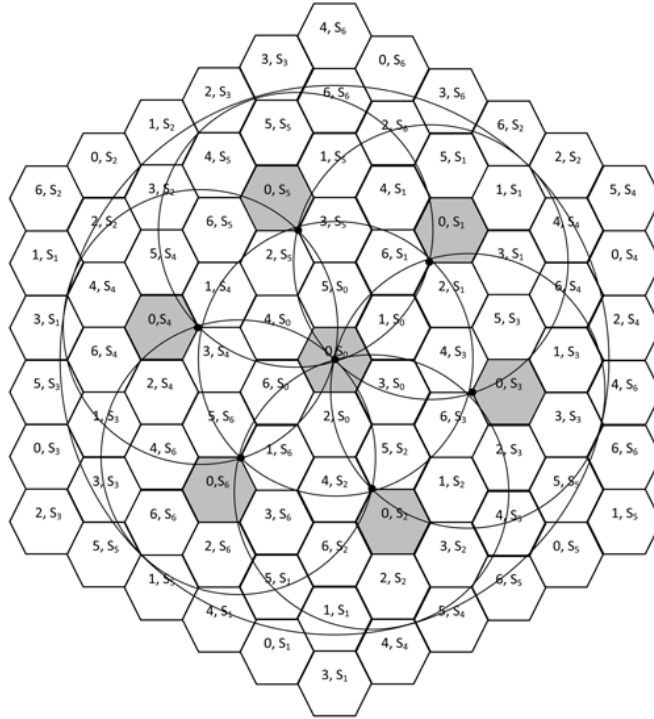


Figure 4: A collision free two layer coloring of the hexagonal tessellation of the plain.

to carry a single packet while the length of the macro timeslot should be small enough to accommodate the rate of packet generation at the source. This approach would result in a maximum end to end delay of $7 \times 7 \times BFS(j)$ at node j .

Assuming that the interference range of each node is twice its transmission range, both FDMA and TDMA approaches avoid interference as well, since the two nodes with the same first and second layer colors are more than $3 \times transmission\ range$ apart. Figure 4 shows a valid coloring for TDMA and FDMA approaches.

3.3 TDMA over FDMA

While the IEEE 802.11a standard at 5 Ghz provides 13 orthogonal (non-overlapping) channels [2], the widely used IEEE 802.11b and g standards would only provide 3 non-overlapping channels⁴. This means that using the standalone FDMA approach would be practically infeasible in many WiFi networks. The problem with the TDMA approach is that the end to end delay is high compared to the FDMA approach. In this section, we describe another approach that overcomes these problems. This hybrid approach uses the TDMA solution over FDMA solution to achieve smaller delay while using smaller number of frequencies. The goal here is to use at most three different

⁴The Zigbee standard and WiMax would also provide more than 7 non-overlapping channels.

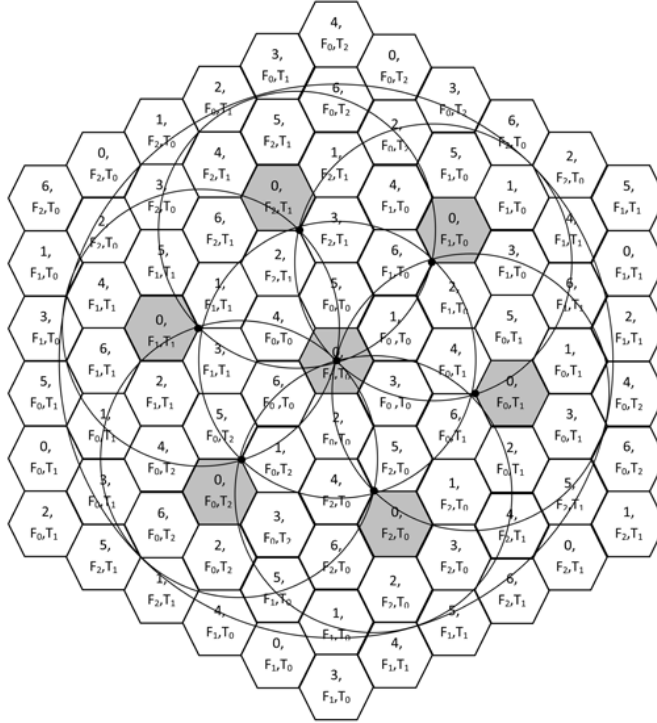


Figure 5: A valid three layer coloring for TDMA over FDMA, F denotes the frequency while T denotes the micro timeslot.

frequencies while minimizing the end to end delay.

The hexagonal coloring is more complex in this approach comparing to the simple symmetrical coloring used in the previous two. In this approach, we have three layers of coloring as opposed to the two layer coloring introduced before. The first layer corresponds to the macro timeslots, while the second and third layers correspond to frequency and micro timeslots respectively. The main constraint is on the second layer coloring where we are bound to use at most three colors, which correspond to the three non-overlapping frequencies. The second layer implements FDMA and the third layer implements TDMA. In this approach, we use the same first layer coloring, while the seven colors in the second layer are mapped to two layers, i.e. a second layer with three colors that correspond to the three non-overlapping frequencies and a third layer with another three colors that correspond to micro timeslots (Figure 5). With this approach, the maximum end to end delay at node j is $7 \times 3 \times (BFS(j) + \beta)$

3.4 Detailed description

The protocol starts by each node sending out and receiving periodic hello messages to obtain its one hop neighborhood information. When a node initiates a broadcast, it would send its GPS location as part of the message along with the timestamp of the message. The source of the broadcast is

always considered to be at the center of a hexagon with first and second and possibly third layer colors of 0. We assume that the source node broadcasts at timeslot 0. Upon receiving a message at a node k , it computes the boundaries of the hexagon that it belongs to based on the location of the source node. It then computes the first and second layer colors of its hexagon. In the end, it decides to forward the message if it is has the smallest id amongst all of its one hop neighbors that belong to the same hexagon⁵. Suppose node k that belongs to hexagon i becomes a forwarding node. Depending on the approach being used, it will forward the packet in one of the following manners:

- If the FDMA approach is used: Assuming that the current timeslot is TS using this approach node k transmits the packet with frequency $C_2(i)$ at time that corresponds to the start of timeslot x where x is computed as follows:
 - in case $(TS \bmod 7) < C_1(i)$: $x = TS + (C_1(i) - (TS \bmod 7))$.
 - in case $(TS \bmod 7) \leq C_1(i)$: $x = TS + (C_1(i) + 7 - (TS \bmod 7))$.
- If the TDMA approach is used: Assuming that the current macro timeslot is MTS and the current micro timeslot is mTS node k transmits the packet at the start of macro timeslot x and micro timeslot y where they are computed as follows:
 - in case $(MTS \bmod 7) < C_1(i)$ or $(MTS \bmod 7) = C_1(i) \wedge (mTS \bmod 7) \leq C_2(i)$: $x = MTS + (C_1(i) - (MTS \bmod 7))$ and $y = C_2(i)$.
 - otherwise: $MTS + (C_1(i) + 7 - (MTS \bmod 7))$ and $y = C_2(i)$.
- If TDMA over FDMA solution is used: This is very much like the TDMA approach except for the fact that node k would transmit the message with frequency corresponding to $C_3(i)$.

At each node a message would be forwarded at most once. This means that only the first copy of the message is forwarded. Computing the boundaries of the hexagon in which a node belongs to can be done in constant time using three hashing function one for every pair of parallel edges. The three layer colors can be computed with a linear time algorithm.

4 Experimental Results

In this section we present the results of three studies conducted in order to compare our solutions with several existing techniques. Existing techniques can be grouped into four different categories [14]:

- Simple Flooding: it requires each node to rebroadcast all packets.
- Probability Based Methods: these methods use some basic understanding of the network topology to assign a probability to a node to rebroadcast.

⁵The assumption here is that every node has a unique ID in the network.

- Area Based Methods: they assume nodes have common transmission distances; a node will rebroadcast only if the rebroadcast will reach sufficient additional coverage area.
- Neighbor Knowledge Methods: These methods maintain state on their neighborhood, via Hello packets, which is used in the decision to rebroadcast.

We have decided to compare our solution with the Simple Flooding [10, 7], the Location-Based scheme (LB Flood)[9], Scalable Broadcast Algorithm (SBA)[12] and Ad Hoc Broadcast Protocol (AHBP) [13]. We deliberately have omitted choosing a protocol from the probability based methods since their performance is worse than the area based and neighbor knowledge methods. The simple flooding is the simplest yet most robust technique when it comes to highly dynamic network topologies. It also provides a worst case measure for many performance metrics including the end to end delay, delivery ratio and the number of retransmissions. The Location-Based scheme was chosen to represent the Area Based methods. In general, the Location-Based scheme was shown to be more robust than the Distance-Based scheme [9]. Neighbor Knowledge is the largest category and therefore two protocols were chosen from this category: SBA and AHBP. The following discussion justifies this choice.

The neighbor knowledge protocols can be classified by whether a node makes a local decision to retransmit a broadcast packet. A node that uses Flooding with Self Pruning, SBA, or LENWB makes this local decision. A node that uses Dominant Pruning, Multipoint Relaying, AHBP, or CDS-Based Broadcasting is told (either via the packet or via a previously sent control packet) whether it needs to retransmit a broadcast packet.

We chose SBA to represent the protocols that make local decisions on whether to rebroadcast. We chose AHBP to represent neighbor knowledge protocols that do not make a local decision on whether to rebroadcast. AHBP uses a more efficient algorithm for selecting next hop rebroadcasting nodes than Dominant Pruning. In addition, AHBP appears to benefit from the three ways in which it differs from Multipoint Relaying. Next, we will describe each study.

4.1 Description of Studies

As we have mentioned before, we have conducted three different studies over four different protocols in addition to our proposed approach to evaluate and compare our approach. All three studies are conducted using the NS2 simulator. While our three studies vary some network parameters, the ones outlined in Table 1 remain constant for all simulations⁶. We assume that there is only one source of broadcast in the network. In other words we have studied the single source broadcasting scenario. In all our studies we measure the following four metrics:

- Number of retransmitting nodes: this is the average number of nodes that forward the packet in a broadcast session. This metric is computed as $\frac{\text{total number of packets sent during the simulation}}{\text{total number of packets generated by the source node}}$ and it will be a good measure of algorithm efficiency in terms of selecting the minimum number of rebroadcasting nodes.

⁶We have used 1 second intervals for “hello” packets for AHBP and SBA and our own solutions.

Table 1: Simulation parameters common to all studies.

Simulation Parameter	Value
Simulator	NS-2 (2.34)
Network Area	400 x 400 meter
Node Transmission Range	100 meters
Data Packet Size	256 bytes
Channel Bandwidth	1 Mbps
Simulation Time	1000 seconds

- End to end delay: This is the average time that takes for a packet to reach the farthest node from the source of the broadcast. It is measured in seconds.
- Delivery ratio: This metric is computed as $1 - \frac{\text{total number of dropped packets}}{\text{total number of received packets}}$. A high delivery ratio is a reliability indicator for a protocol.
- Coverage: This metric is perhaps the most important metric of all. It represents the number of nodes that have received at least one copy of every single packet. It is computed as $\frac{\sum_i (\text{number of single copies of each packet received at node } i)}{\text{total number of packets generated at source} \times \text{total number of nodes in the network}}$

4.1.1 Study 1 - Effect of Density

Our first experiment studies the effect of network density on the four metrics we have defined above. In this study we increase the number of nodes in the network by 10 nodes at each simulation starting from 20 nodes and ending in 100 nodes. The average speed of each node is set as 10 m/s with average pause time of 10 seconds per node.

4.1.2 Study 2 - Effect of Congestion

In our second experiment we investigate the effect of network traffic on the four metrics. For this, we increase the packet origination rate the the source node from 10 packets per second to 70 packets per second in steps of 10 while fixing the number of nodes in the network at 60 nodes. The average speed of each node is set to 10 m/s with an average pause time of 10 seconds.

4.1.3 Study 3 - Highly Variant Networks

Our last experiment studies the robustness of each protocol based on the values of the four metrics in a highly variant scenario. In this study we intend to simulate a scenario where the network starts at a quiet steady and sparse topology with low traffic and changes to a more dense and dynamic topology while increasing the traffic. We have used four trials to implement this scenario. The four trials are designed so that Trial 1 takes a combination of the least severe conditions and Trial

Table 2: Study 3 Trial Simulation Parameters.

Trial	1	2	3	4
Number of Nodes	40	60	70	100
Average Speed (m/s)	1	10	10	20
Average Pause Time (seconds)	600	60	10	10
Packet Origination Rate (pps)	10	40	80	80

4 takes a combination of the most severe conditions. Table 2 shows the specific parameters of each trial.

4.2 Results

In this section we present and explain the results of our studies. We compare two of our solutions namely FDMA and TDMA over FDMA with other techniques.

4.2.1 Study 1 - Effect of Density

Figure 6 shows the average end to end delay (the time it takes for a node to receive the packet) as the network density increases. The end to end delay has an inverse relationship with the number of nodes in the network. Note that this is not the same as the maximum end to end delay which is defined as the time it takes for the last node to receive the packet. The inverse relationship is due to the fact that as the network density increases the average number of one hop neighbors also increases, this results in a decrease in the average end to end delay. Another observation that we have made in this study was that the maximum end to end delay was almost constant regardless of the network density. As it is demonstrated both of our solutions result in the least average end to end delays.

Figure 7 shows the delivery ratio as the network density increases. The delivery ratio is an indicator of the number of collisions in the network. A unit delivery ratio indicates zero collisions in the network. The delivery ratio for all protocols except for the simple flooding and the location based are almost one regardless of the density. For simple flooding and location based scheme the delivery ratio would decrease since as the density increases the number of overlapping retransmitting nodes would also increase since the retransmitting nodes are chosen either at random (in case of simple flooding) or based on the geographic coverage (in case of location based scheme) and not the neighborhood knowledge which attempts to avoid collisions with other neighbors. Our solutions are both showing close to one delivery ratios. Based on our analysis of the trace files we have found that the very few collisions that had occurred in case of our solutions were due to either collisions with “hello” packets or with data packets with different sequence numbers. In other words there was no collision between two packets with the same sequence number.

Figure 8 shows the percentage of covered nodes with respect to network density. This metric represents the number of nodes that are covered by the broadcast protocol and is an indicator of the

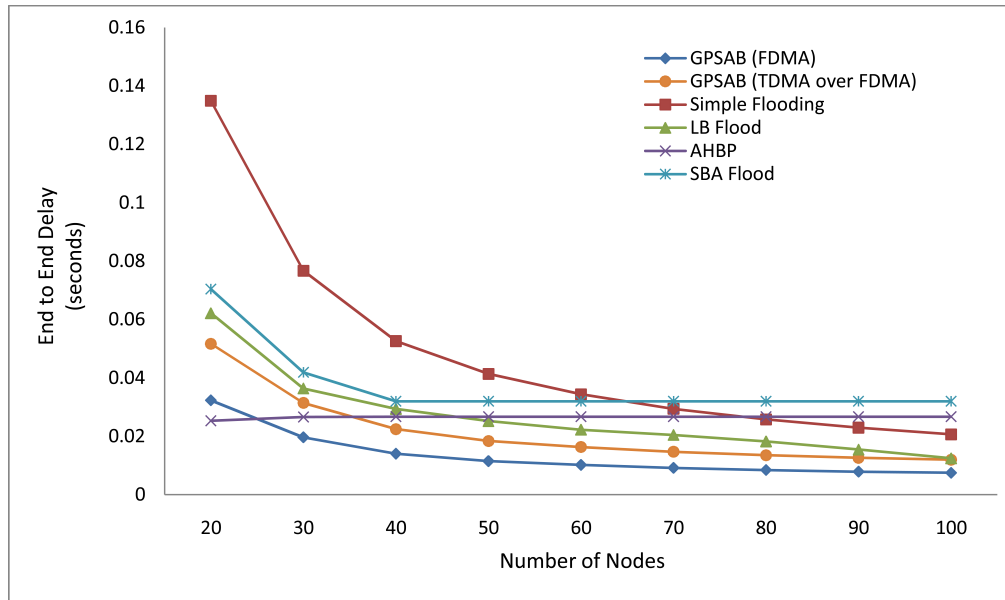


Figure 6: End to end delay VS. Total number of nodes in the network.

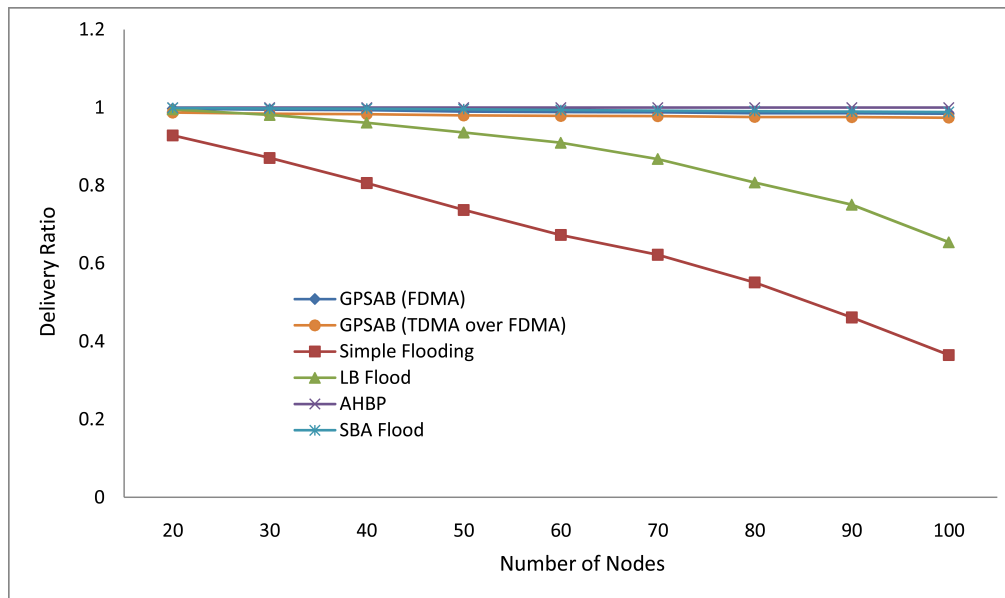


Figure 7: Delivery ratio VS. Total number of Nodes in the network.

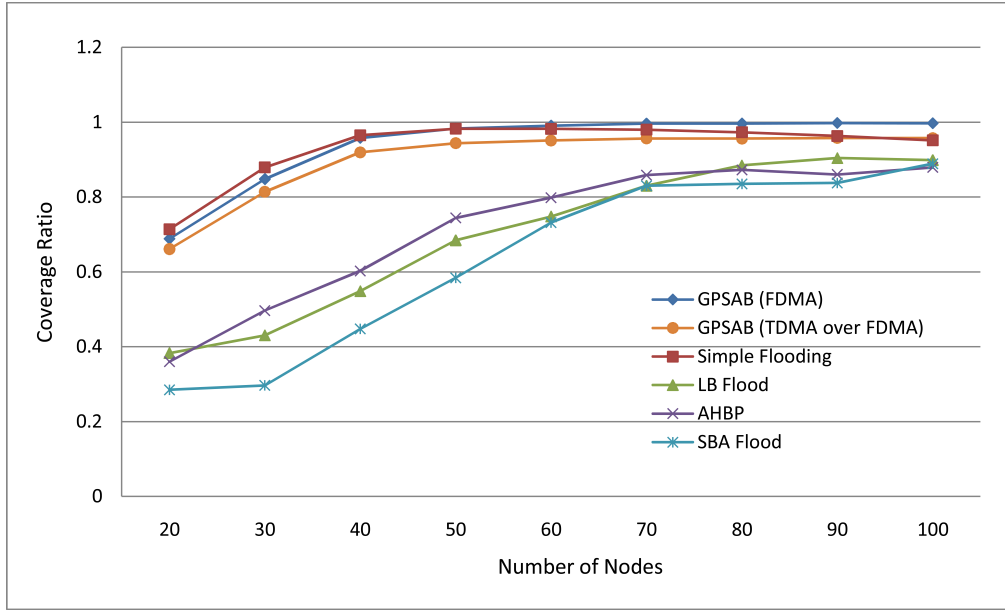


Figure 8: Node coverage VS. Total number of Nodes in the network.

reliability of the protocol. As one would expect the simple flooding would be the most reliable one but the results show that our two solutions result in almost the same coverage as simple flooding. The advantage gained by using our protocol in comparison to other protocols is very notable when it comes to reliability.

Figure 9 shows the average number of retransmitting nodes for each protocol as the number of nodes is increased. With the exception of simple flooding, the number of retransmitting nodes has a logarithmic relationship with the total number of nodes in the network. This is mainly because of the fact that these protocols tend to select a small fraction of nodes that would cover all nodes in the network. This set is proportional to the height of the Breadth First Search (BFS) tree which is proportional to $\log n$ for a random graph of size n .

As it is shown AHBP and SBA choose the least number of forwarding nodes. This is reasonable since they use neighborhood information to approximate a minimum connected dominating set (MCDS). Comparing to AHBP, SBA, and the Location Based approach both GPSAB (FDMA) and GPSAB (TDMA over FDMA) choose more retransmitting nodes, however, the trend is still logarithmic. Another observation is that the added number of retransmitting nodes has resulted in improved node coverage (reliability) while the number of collisions is still very low and the end to end delay is the lowest comparing to the other protocols therefore the added benefit is worth the increase in the number of retransmitting nodes.

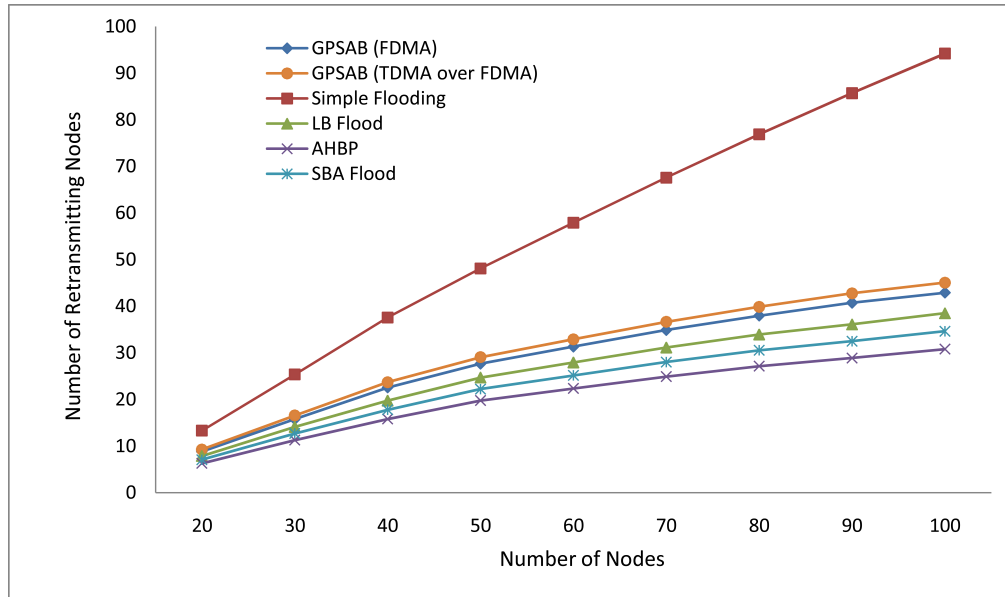


Figure 9: Retransmitting nodes VS. Total number of Nodes in the network.

4.2.2 Study 2 - Effect of Congestion

Figure 10 shows the relationship between the end to end delay and the packet origination rate. As it is indicated in the figure all protocols with the exception of our solutions are depicting a correlation between end to end delay and the packet origination rate. This is perhaps due to the collisions that are caused because of network congestion. Our two solutions aim at avoiding collisions and therefore the end to end delay is almost independent of the packet origination rate. We would like to emphasize that this independence is bounded by the size of the timeslot used in our solutions. As the size of the timeslot gets bigger and the packet origination rate gets higher the probability that the timeslots for two consecutive broadcast packets overlaps becomes higher. Therefore we would expect our solutions to depict the same behavior as the other protocols at higher packet origination rates.

In our second set of experiments we studied the effects of traffic on our set of performance metrics. Figure 11 shows the number of retransmitting nodes as the packet generation rate is increased from 20 packets per second to 70 packets per seconds. With the exception of simple flooding as it is expected, the number of retransmitting nodes is almost constant for all protocols. This is because the area and the number of nodes remain constant. For Flooding, the number of retransmitting nodes drops as the network becomes congested, which directly illustrates the effect of collisions and queue overflows in congested networks.

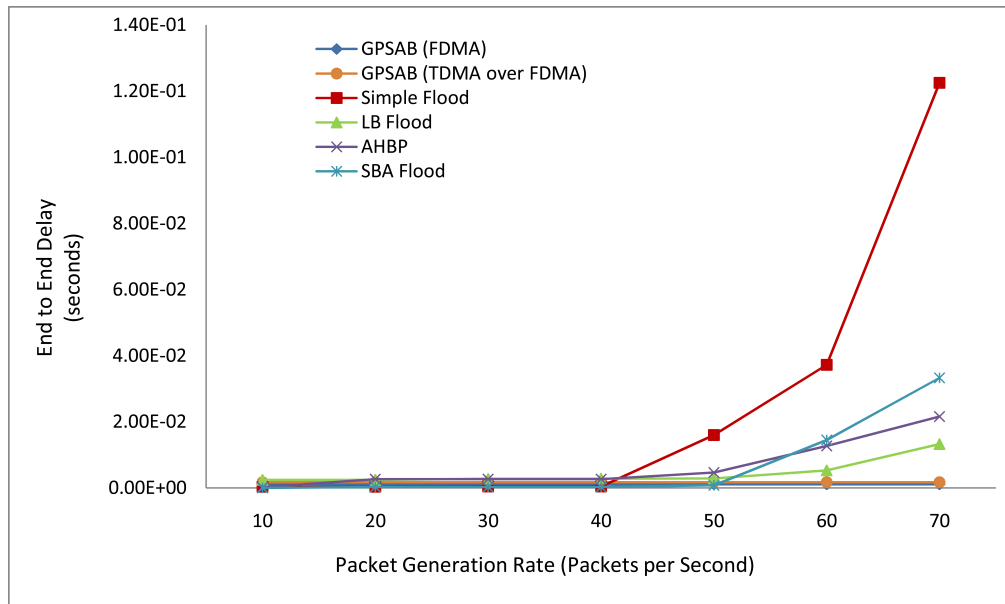


Figure 10: End to end delay VS. Packet origination rate.

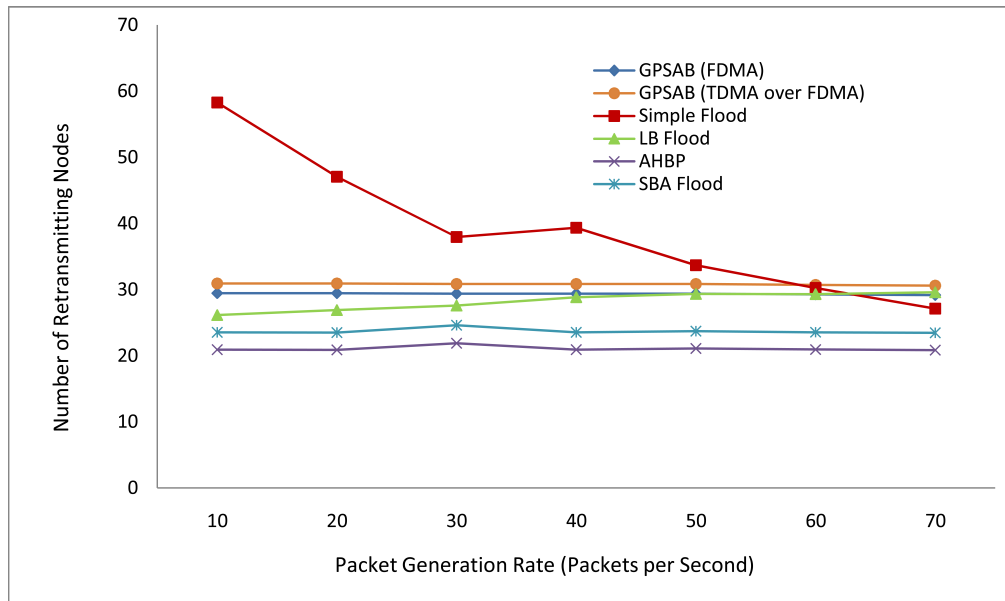


Figure 11: Retransmitting nodes VS. Packet origination rate.

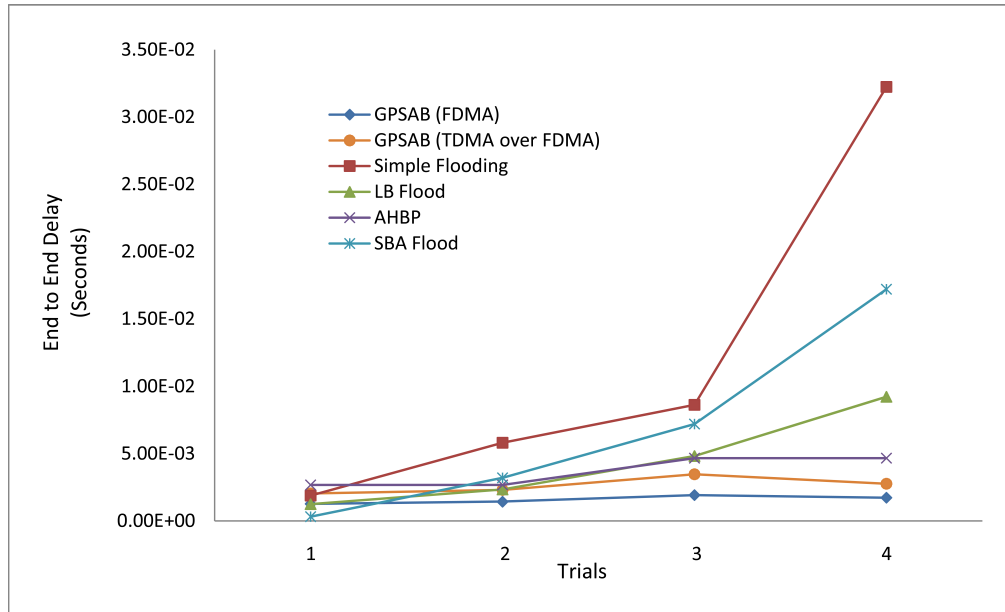


Figure 12: End to end delay VS Per Trial.

4.2.3 Study 3 - Highly Variant Networks

In our final set of experiment we studied the effects of variability on our set of performance metrics. As was described before this scenario uses four trials and starts at a moderately static network and changes to towards more dynamic scenarios. Figure 12 shows the average end to end delay with respect to each trial. As you can see all protocols with the exception of our solutions indicate an increase in the end to end delay as the network becomes more dynamic. This is mainly because of the collisions caused by the transmitting nodes. The end to end delay of our solutions are almost independent of the severity of the trials. This indicates that the collisions are avoided to a high extent.

Figure 13 shows the percentage of covered nodes at each trial. As you can see the simple flooding and our two solutions start with the highest coverage but as the trials get more violent the coverage ratio decreases for the simple flooding and all protocols result at almost the same coverage with simple flooding (perhaps because they exhibit the same behavior as simple flooding in severe scenarios) while our two solutions are consistently providing close to 100% coverage in all scenarios. This is a very important indicator of how reliable our solutions are when it comes to dealing with a wide range of network scenarios.

Finally Figure 14 shows the number of retransmitting nodes at each trial. The general trend is that the number of retransmitting nodes would increase as the scenario becomes more severe. But the interesting observation is that our solutions are choosing less retransmitting nodes in the more severe trials comparing to other protocols. The results of this study clearly indicate that our solutions are exhibiting the same behavior in different scenarios while they prove to be much more

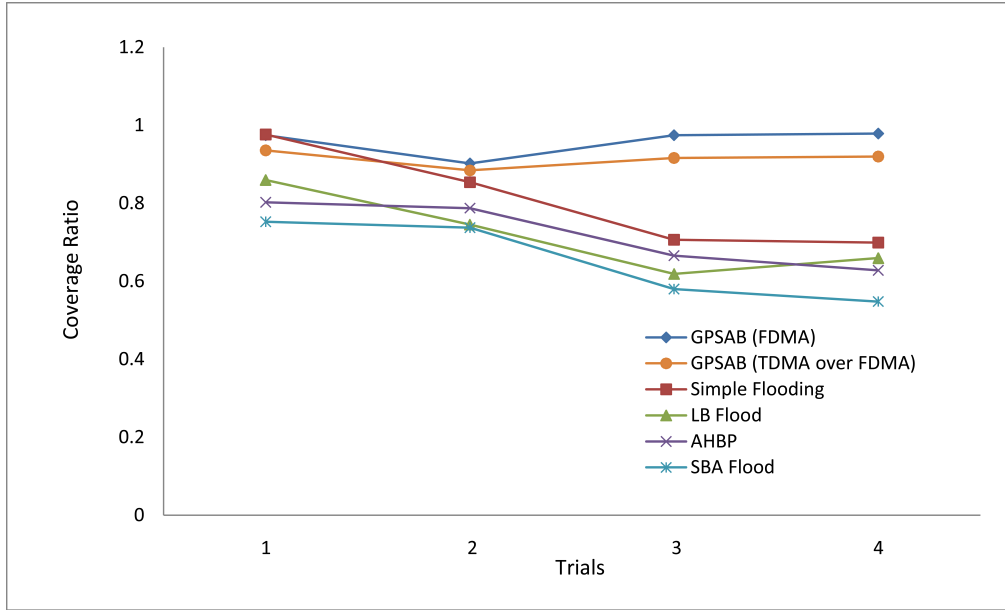


Figure 13: Node coverage Per Trial.

effective when it comes to more severe scenarios.

4.2.4 Comparison of the optimality factor

As we mentioned earlier our two methods would theoretically result in end to end delays that are constantly proportional to the height of the BFS tree for the network graph. In particular the GPSAB (FDMA) would result in worst case end to end delay of $7 \times (BFS(f))$ where f is the farthest node in terms of BFS height from the source of the broadcast, while the GPSAB (TDMA over FDMA) would result in worst case end to end delay of $7 \times 3 \times (BFS(f))$. We refer to the constant factor as the optimality factor since the smaller the factor the more optimal the solution becomes in terms of end to end delay. In order to compare the optimality factors we compute the following ratio which we refer to as *virtual height* for a single simulation scenario: $\frac{\text{Maximum end to end delay occurred}}{\text{single link delay}}$ and we average it over 9 similar scenarios and we compare it with the average value of the maximum BFS height for the 9 similar scenarios.

Figure 15 shows the *virtual height* for the GPSAB (FDMA) and GPSAB (TDMA over FDMA) alongside the maximum height of the BFS trees for our experimental scenarios. As you can see the height of BFS tree would increase until 50 nodes and then it would decrease. The same trend holds for the virtual heights. What is interesting is that the virtual height for the GPSAB (FDMA) is at most 1.5 times the height of the BFS tree which means that the optimality factor would be 1.5 instead of 7 and for GPSAB (TDMA over FDMA) the virtual height is at most 2 times the height of the BFS tree which means that the optimality factor would be 2 instead of 21. We conclude that in reality our solutions are in fact very close to optimal solutions.

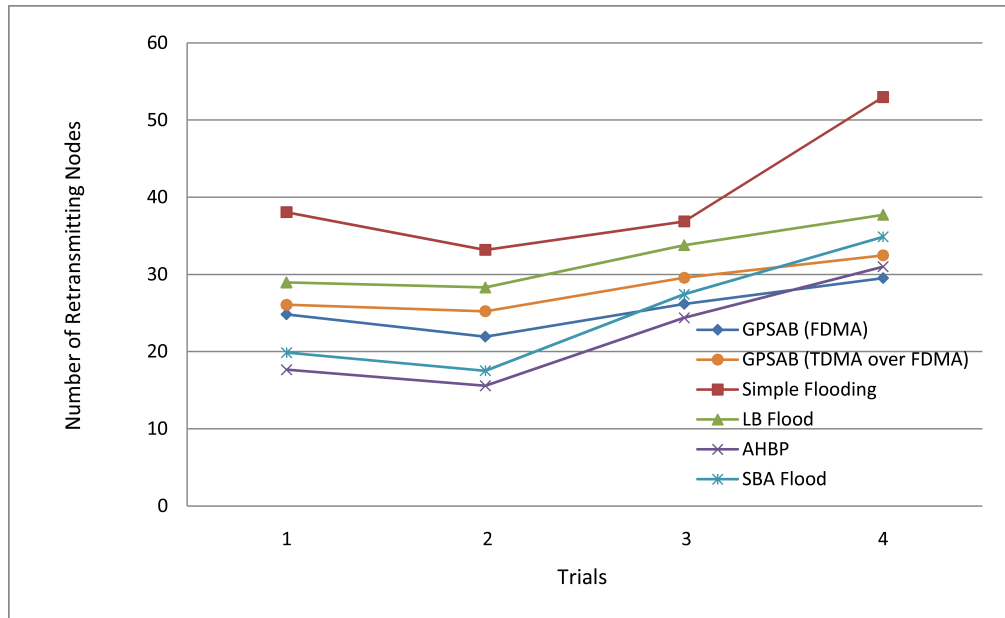


Figure 14: Number of retransmitting nodes Per Trial.

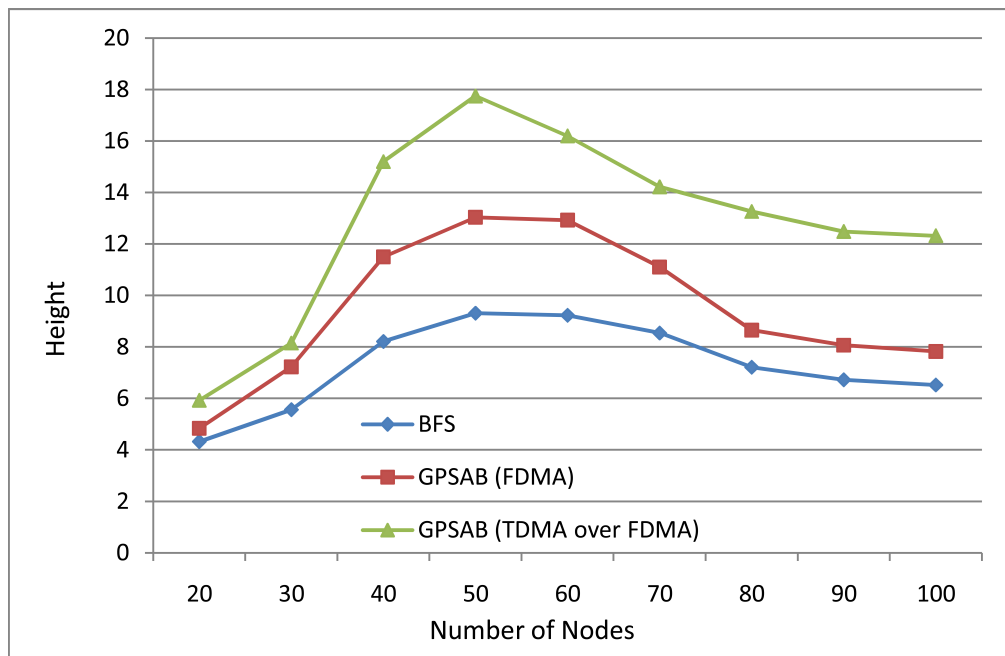


Figure 15: Comparison of the optimality factors.

5 Conclusions

Our protocol has the following advantages:

- It is very robust to changing network environments (even more robust than simple flooding).
- it is efficient in terms of power consumption since each node would only transmit at most one copy of every packet therefore it is usable in sensor network and in general in scenarios where power is a scarce resource.
- it would implement a very low end to end delay broadcast which means that it can be used in many delay bound applications where a certain QoS needs to be satisfied in terms of end to end delay.
- it is practical and in fact can be implemented at software level without any need of modifying the hardware. In fact frequency hopping for the FDMA part has been implemented and used before[2]. Another important advantage is that our protocol is completely distributed and would use minimal information (GPS location and 1 hop neighborhood knowledge) to broadcast the packets.
- In practice the optimality ratio is very small (around 2 to 3) which makes the achieved end to end delay very acceptable as the experimental results verify.

We intend to implement our protocols on a set of MICA 2 sensors to evaluate its performance on a real world scenario. We also intend to explore the use of our protocol and its impact on the discovery phase of a set of Ad Hoc routing protocols.

References

- [1] A. S. Ahluwalia and E. H. Modiano. On the complexity and distributed construction of energy-efficient broadcast trees in wireless ad hoc networks. *IEEE Transactions on Wireless Communications*, 4(5):2136–2147, September 2005.
- [2] Paramvir Bahl, Ranveer Chandra, and John Dunagan. SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 216–230, New York, NY, USA, 2004. ACM.
- [3] Zhenming Chen, Chunming Qiao, Jinhui Xu, and Taekkyeun Lee. A constant approximation algorithm for interference aware broadcast in wireless networks. In *Proceedings of IEEE INFOCOM*, pages 740–748, 2007.
- [4] I. Chlamtac and S. Kutten. On broadcasting in radio networks - problem analysis and protocol design. *IEEE Transactions on Communications*, 33:1240–1246, 1985.

- [5] Rajiv Gandhi, Arunesh Mishra, and Srinivasan Parthasarathy. Minimizing broadcast latency and redundancy in ad hoc networks. *IEEE/ACM Transactions on Networking*, 16(4):840–851, August 2008.
- [6] Scott C.-H. Huang, Peng-Jun Wan, Xiaohua Jia, Hongwei Du, and Weiping Shang. Minimum-latency broadcast scheduling in wireless ad hoc networks. In *IEEE Infocom*, pages 733–739, 2007.
- [7] J. Jetcheva, Y. Hu, D. Maltz, , and D. Johnson. A simple protocol for multicast and broadcast in mobile ad hoc networks. Internet-Draft Internet Draft: draft-ietf-manetsimple- mbcast-01.txt, Internet Engineering Task Force, 2001.
- [8] Reza Mahjourian, Feng Chen, Ravi Tiwari, My Thai, Hongqiang Zhai, and Yuguang Fang. An approximation algorithm for conflict-aware broadcast scheduling in wireless ad hoc networks. In *ACM MobiHoc*, pages 331–340, May 2008.
- [9] Sze-Yao Ni, Yu-Chee Tseng, Yuh-Shyan Chen, and Jang-Ping Sheu. The broadcast storm problem in a mobile ad hoc network. In *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 151–162, New York, NY, USA, 1999. ACM.
- [10] Katia Obraczka, Kumar Viswanath, and Gene Tsudik. Flooding for reliable multicast in multi-hop ad hoc networks. *Wirel. Netw.*, 7(6):627–634, 2001.
- [11] I. Papadimitriou and L. Georgiadis. Minimum-energy broadcasting in multi-hop wireless networks using a single broadcast tree. *Mobile Networks and Applications*, 11(3):361–375, 2006.
- [12] Wei Peng and Xi-Cheng Lu. On the reduction of broadcast redundancy in mobile ad hoc networks. In *MobiHoc '00: Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, pages 129–130, Piscataway, NJ, USA, 2000. IEEE Press.
- [13] Peng Wei and Lu Xicheng. Ahbp: An efficient broadcast protocol for mobile ad hoc networks. *Journal of Computer Science and Technology*, 16(2):114–125, 2001.
- [14] Brad Williams and Tracy Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In *MobiHoc '02: Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*, pages 194–205, New York, NY, USA, 2002. ACM.