Retransmission and Backoff Strategies for Wireless Broadcasting

Jesus Arango [a] Alon Efrat [c,*] Srinivasan Ramasubramanian [d]
Stephen Pink [b] Marwan Krunz [d]

[a] *Cisco Systems, San Jose, CA, USA*

[b] *Computing Department, Lancaster University, UK*

[c] *Department of Computer Science, University of Arizona, USA*

[d] *Department of Electrical and Computer Engineering, The University of Arizona, USA*

Abstract

This article defines a wireless broadcasting algorithm as having two components: A retransmission strategy and a backoff strategy. Several strategies are proposed in this article and a comparative analysis is presented between existing algorithms and the strategies proposed herein. Simulation experiments and analysis are used to study or demonstrate the properties and performance of specific strategies or to obtain results of a more general nature. Strategies are also evaluated with respect to their impact on routing protocols that rely on broadcasting to perform path discovery. The purpose of this evaluation is to determine which strategies result in more stable routes.

The second part of this article analyzes the problem of broadcasting when nodes are assumed to be arranged on a strip. Such arrangement occurs in vehicular broadcasting applications. We present the *Strip Broadcasting* retransmission strategy that can be modeled as a one-dimensional problem to significantly reduce the number of retransmissions. Analysis and simulation results are presented to analyze the properties of the algorithm.

Key words: broadcasting, flooding

* Corresponding author: Department of Computer Science, The University of Arizona, USA, alon@cs.arizona.edu

1 Introduction

[1] One way of performing network-wide broadcasting is by "flooding" the network with the broadcast message. Flooding is carried out by having each node retransmit the broadcasted message after receiving it for the first time.

Flooding is important because it is the basis for performing route discovery in mobile ad-hoc networks (MANETs). Link-state routing protocols also rely on flooding for distribution of link-state information. There are other broadcasting techniques that send out-of-band messages to build distribution trees. However, schemes that do not require out-of-band messages continue to be the best general-purpose solution for path discovery and link-state routing as their performance does rely on assumptions about mobility or broadcast frequency [2] .

Flooding is an unreliable operation with no acknowledgment mechanism in place; this is not a major concern for path discovery or link-state routing as %100 reliability is not required. Flooding is able to distribute broadcast messages to as many nodes as possible using very little effort. Analysis of some of the broadcasting strategies presented in this article show that under reasonable assumptions that reliability can be at least as good as in flooding. If such analysis is not possible for a particular case, then simulations will be used to show that their reliability is comparable to that of flooding.

This article only considers broadcasting algorithms that do not require out-of-band transmissions such as *hello messages*. The cost of transmitting "hello" messages often cannot be justified. Thus, out-of-band overhead narrows the application scope and make it challenging to conclusive compare other broadcasting algorithms that do not use out-of-band messages.

Flooding generates more overhead than necessary because, depending on the node density, many or most retransmissions are *redundant*. A retransmission is said to be redundant if all the neighbors of the transmitting node have already received the message. If a transmission is *non-redundant* then its additional coverage is not null [3] , and in this case the *additional coverage* of a retransmission is the percentage of the transmission range that has not been covered by neighbor nodes.

Redundancy can be reduced (resp. eliminated) when retransmitting occurs only if the additional coverage of the retransmission is large enough (resp non-zero) to warrant the additional overhead . This can be achieved if each

[1] This paper extends our preliminary works that were published in [1–3]
[2] How often broadcast operations are performed
[3] This is not necessarily true in the opposite direction

node knows exactly the locations of many of the other nodes — e.g. its neighbors, but in a dynamic network it is quite beneficial to avoid maintaining this information. Instead, each node u receiving the first duplicate of the broadcast message would postpone its retransmission for a short *backoff time*, and would drop this message completely if within this backoff time other retransmission made by other nodes made u's retransmission redundant. Otherwise u would retransmit once the backoff time has elapses. In this paper we study different method by which partial information information "sensed" by u can suggest that this retransmission is redundant. For example, if the location of the transmitting neighbors is known, then the additional coverage can be deterministically computed. If the precise location is unknown, then other information such as distance or angles between neighbors or number of duplicates received can be used to estimate the *expected additional coverage*. Different technologies such as GPS, *angle of arrival* (AOA) and *received signal strength* (RSS) can be used to gather such information.

This *hold and suppress* approach to broadcasting is by no means novel. However, this article makes several contributions in the area of wireless broadcasting. Namely, we redefine a broadcasting algorithm as being composed of two strategies. The *retransmission strategy* refers to how it is decided if a scheduled retransmission is canceled. The *backoff strategy* determines the manner in which the backoff time is chosen. Most of the previous work implicitly falls only under the category of retransmission strategy. We also propose several retransmission and backoff strategies and a comparative analysis is presented between existing algorithms and the strategies proposed herein. Simulation experiments and analysis are used throughout this work to study or demonstrate the properties and performance of specific strategies or to obtain results of a more general nature. Strategies are also evaluated with respect to their impact on routing protocols that rely on broadcasting to perform path discovery. The purpose of this evaluation is to determine which strategies result in more stable routes.

By considering the distribution of nodes in specific applications it is possible to design more efficient retransmission strategies. The second part of this paper analyzes the problem of broadcasting when nodes are assumed to be arranged on a strip. Such arrangement occurs in vehicular broadcasting applications. We present the *Strip Broadcasting* (SB) retransmission strategy that can be modeled as a one-dimensional problem to significantly reduce the number of retransmissions. A whole new range of vehicular information services can be made possible by relaying information using vehicle-to-vehicle communications (V2VCOM). The first class of services that comes to mind is traffic alerts about different upcoming situations such as accidents, construction zones, or traffic jams.

With the exception of some cases such caravans or convoys, communication

between specific vehicles is rarely an issue. Broadcasting is a more natural communication primitive for this type of environment. Accordingly, the goal is to relay information between vehicles for a certain distance or for a given number of relay hops. Omnidirectional transmissions are used because the road and the vehicles it contains can have an arbitrary direction with respect to the transmitter's frame of reference.

Previous algorithms are suboptimal in vehicular environments because they perform considerably more retransmissions than necessary. For example, vehicle b shown in Figure 1 has received duplicates from both a and c. Existing broadcast algorithms would require a retransmission by b because the shaded areas in its range of transmission have not been covered. However, the strip of road in the figure is entirely covered by previous retransmissions.

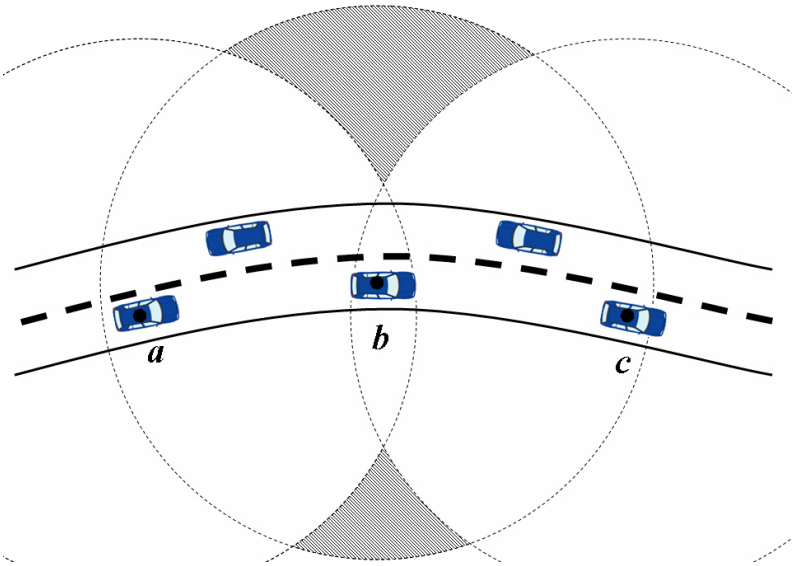

Fig. 1. Redundant retransmission.

The rest of this article is organized as follows. Section 2 presents a summary of the related work. Section 2 describes the simulation environment used to obtain the experimental results presented throughout this article . Section 3 presents flooding as a combination of retransmission and backoff strategies, and provides interesting experimental results about the relationship between backoff, collisions and reliability. Section 3 proposes new retransmission strategies, and Section 3.7 provides a comparative simulation analysis. Backoff strategies are studied in Section 4. Section 5 presents an interesting study on the effect that retransmission and backoff strategies have on the performance of path discovery. Section 6 presents the SB algorithm and provides detailed analysis of its performance. Section 7 presents the simulation results of the SB algorithm and Section 8 offers some concluding remarks.

2 Related Work

Sze-Yao et al. [4] observed that serious redundancy, contention, and collision could exist if flooding is done blindly. Collectively, they refer to these problems as the *broadcast storm* problem. As a solution, they introduce several retransmission strategies, including the counter-based, distance-based, location-based, and cluster-based schemes. Williams and Camp [5] present an analysis of existing broadcasting schemes and is an excellent reference on the topic of wireless broadcasting.

Paruchuri [6] introduced a retransmission strategy that uses a hexagonal lattice to determine which nodes have to retransmit. Cartigny, et al. [7] presented several stochastic algorithms where nodes forward messages with a certain probability. These probabilities are calculated differently for each algorithm using information such as node density and distance between nodes.

Pagani and Rossi [8] presented the reliable broadcast protocol designed for mobile ad-hoc networks. It ensures that all the hosts in the network receive the same messages and provides an *exactly once* message delivery semantics. *Minimum connected dominating sets* (MCDS) can be used to build distribution backbones and several distributed approximation algorithms have been proposed [9,10].

Zhang and Jiang presented one-dimensional analytical results [11] for two broadcasting algorithms originally intended for two-dimensional or three-dimensional arrangement of nodes. While their analysis is interesting, our simulation and theoretical analysis shows that the algorithm presented in this article is more efficient for environments that lend themselves to be modeled in one-dimension.

Michael and Nakagawa [12] propose a MAC-based broadcast protocol that focuses on channel allocation by implementing a slot reservation scheme. Sun *et.al.* proposed a broadcasting protocol [13] which reduces the number of retransmissions in vehicular environments but requires GPS. Wu *et.al.* [14] present analytical models to study the spatial propagation of information in a highly mobile vehicle-to-vehicle ad-hoc network. They show that information propagation depends on vehicle density, average vehicle speed and relative speed among vehicles.

Simulation Environment

All simulations were written in ns-2 [15] version 2.28, a discrete event simulator with extensive support for wireless networks. Unless otherwise specified in each section, the simulation environment is as follows. All simulations conform to the 802.11 standard. The data rate and the basic rate are both set to 11 Mbps.

The transmission range is 250 meters and the default network size is 2500×750 square meters. All broadcast messages have a payload of 128 bytes.

Nodes are randomly distributed with uniform probability. All node densities for two-dimensional models are expressed as number of nodes per *range unit*, where the range unit is defined as the area of a circumference whose radius is equal to the range of transmission R. For a one-dimensional model, the density is the average number of nodes in a segment of length R. With a few exceptions noted such as Section 5, there is no mobility, as the type of broadcasting protocols studied in this article are not adversely affected by mobility.

3 Backoff Strategies for Flooding

Flooding is often enhanced by waiting for a short and uniformly distributed random backoff time before forwarding a message in an effort to reduce the number of collisions. As such, flooding can be expressed as a typical hold-and-suppress broadcast with a very simple retransmission strategy where messages are never suppressed. Consequently, if the distribution is uniform, this algorithm only has one possible parameter: The *maximum backoff time*.

The purpose of the timer at the MAC for pure flooding is to reduce collisions. However, hold-and-suppress broadcasting algorithms employ backoff timers at the application layer primarily as a mechanism to realize a given retransmission strategy. Note also that in transport-level or application-level implementations of flooding and broadcasting strategies the backoff time is separate from and additional to whatever backoff mechanisms are in place at the medium access control layer. That being said, one could surely consider the possibility of implementing network-wide broadcasting at the MAC layer. Our simulations in particular deal with application-level broadcasting on top of an 802.11 network. Therefore there are two independent backoff mechanisms in place. The backoff intervals in the broadcasting algorithm are much longer than the backoff times in 802.11.

Figure 2 plots the backoff time at the application layer necessary to achieve %100 *coverage* using pure flooding, as a function of the node density. Coverage is a reliability measurement defined as the fraction of nodes that receive the broadcasted message. To conduct this experiment 25 connected topologies were generated for each simulated node density using random uniform node distribution. Each generated topology was checked for connectivity and disconnected topologies were discarded. The process was repeated until 25 connected topologies were gathered for each density. Starting with a value of zero, the backoff time is incremented until %100 coverage is obtained for all

25 topologies.

The results in Figure 2 are quite conservative as only one message was broadcasted at a time and no other traffic was present. Smaller densities, which are more common in practice, require a larger backoff time than higher densities, and values of at least 10 ms are commonly used [16,15].

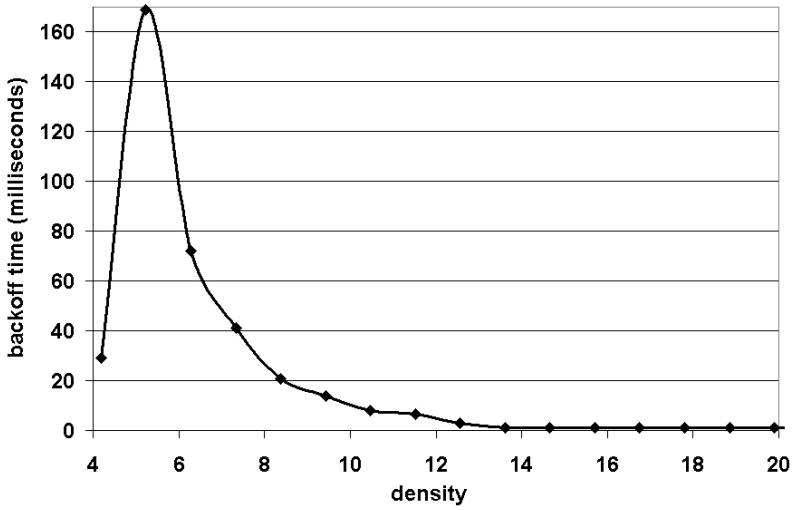

Fig. 2. Necessary backoff time to achieve %100 coverage with standard flooding as a function of node density.

Intuitively, one might think that flooding would be less reliable as the density increases, because higher densities have higher collision rates. However, broadcasting at the MAC layer is implemented with only carrier sensing (at the transmitter) and virtual carrier sensing (using RTS/CTS exchange) is not performed. At high loads, the carrier sensing mechanism at the MAC layer is sufficient for the nodes to back-off. In addition as the node density is high, every node has multiple opportunities to receive the broadcast successfully from at least one of the neighboring nodes. Therefore, application-level timers for broadcast can be neglected at higher node densities. In contrast, at lower node densities, the MAC layer would transmit the packet as soon as the packet is received. This is due to the fact that carrier sensing is not effective at forcing nodes to back-off from transmission at lower node densities. In addition, as a node has a smaller number of neighbors (due to lower node density) through which the broadcast can be successfully received, a good number of nodes do not receive the broadcast due to the collision resulting from the transmission from multiple nodes. In order to separate near-simultaneous transmissions of broadcast messages from nodes that received the messages at the same time, the application layer has to introduce random back-off timers at the application layer at lower node densities (as depicted in the figure 2).

An explanation of this phenomenon, hinged on suggestions provided by the anonymous reviewers of this paper, is as follows: If the density of nodes is low,

the probability of a collision is rather small and may be partially neglected. On the other extreme, in a very dense environment, enough many nodes can receive the message, despite the large number of collisions. However, for intermediate density values, we expect to find clusters of many nodes occupying a small region, and connected to other clusters with small "bridges", with only a few nodes in each bridge. If many transmissions occur within the cluster in a short period of time, the collisions that they would cause will not suffice to jeopardize connectivity within the cluster itself (as indicated in the high-density scenario just discussed). Thus these transmissions might cause a large enough number collisions with the transmissions of the few nodes in the bridges, hence leaving the cluster disconnected from the remaining nodes. We are leaving it for future study to validate this explanation.

Proposed Forwarding Strategies for Static Nodes

This section presents new forwarding strategies that we are proposing. For the ones we were able to, we also prove analytical certain algorithmic properties. *Ideal network conditions* are usually assumed, meaning that there are no collisions or transmission errors. This simplified model is appropriate since the purpose of our analysis is to highlight certain intrinsic properties of the algorithms, rather than to accurately reflect simulation or real life results. However, the simulation results in Section 3.7 provide a comparative analysis of all these strategies.

The *expected additional coverage* of a node is small when the duplicates it receives come from transmitting nodes that are sufficiently distant from each other and/or very close to the receiving node. The goal of a retransmission strategy is to avoid transmitting when these node configurations occur. Several measurement technologies have been proposed to detect these scenarios. These include GPS, received signal strength (RSS) and angle of arrival (AOA). Even when All these techniques have either serious restrictions or accuracy issues. But the major problem is that they are simply not readily available in most off-the-shelf laptop and hand-held devices. Many wireless adapters have hardware support for reading the RSS, but there is usually no device-independent interface provided by the operating system. The different strategies we propose are tailored to the availabilities, or lack of availabilities of these devices.

3.1 Adjacency Strategy

This heuristic addresses the case at which we cannot measure positions, distances, or angles. Instead once a node u that (after a backoff period) transmit a duplicate of the message, append to to the duplicates the i.d.'s of all the nodes that also have transmitted the message while u was in its backoff pe-

riod, and were heard by u. Once another node v receive the duplicate, it schedule a retransmission, wait for a backoff period, but it would canceled the retransmission if duplicates have been received from at least k neighbors that did not hear each other. This is obtained by building an *neighbor adjacency graph*. Assume two duplicates are received from neighbors u and v, in that order. If the duplicate from v reports that v heard the message from u then an undirected edge between u and v is added to the graph. Then we check the cardinality k of the maximally independent set in this graph. This goal is computationally intractable for large graphs, but is be handled relatively efficiently in our setting. Using standard packing arguments, one can show that $k \leq 5$. Moreover, Our simulations for uniformly distributed nodes show that with an independent set of size 3 the average coverage is at least 99.7% for all densities.

In order to reduce the computational overhead of our algorithm, we also propose a modified strategy called *Adjacency-2.5*. Scheduled retransmissions are canceled if there exists an independent set of size two and at least one additional node that is adjacent to only one (but not both) of the nodes in the independent set. Simulation results indicate that the average coverage is at least 97.5% for all densities, while providing overhead savings that are comparable with other algorithms that rely on location information.

3.2 Angular Strategy

Next we propose a retransmission strategy where nodes use angle of arrival (AOA) information to make forwarding decisions. Assume that nodes can determine the angle of arrival of incoming messages with respect to some local frame of reference. Such angular measurements will in most likelihood be subject to a certain degree of noise or error. One can show that If the clockwise angle distance between every pair of adjacent duplicates received by a node u is at most θ_{thr} then the additional coverage of u is bounded from above by a function of θ_{thr}, and approaches zero as θ_{thr} approaches $\pi/6$.

Figure 3 illustrates the coverage of the angular strategy as a function of node density for several values of θ_{thr}. The idea is to give an insight into how to parameterize θ_{thr} in order to achieve a given threshold of reliability. Fifty connected topologies were generated for each node density using a uniform random node distribution. The values shown for each density corresponds to the average of all fifty executions.

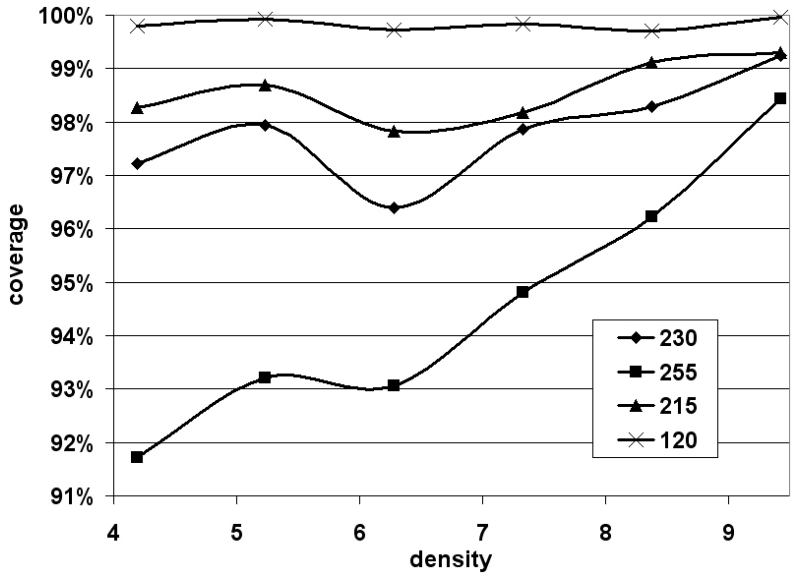

Fig. 3. Angular strategy. Coverage vs. density for several values of θ_{thr}. Angles are shown in degrees.

3.3 k-sector Strategy

This strategy is quite similar to the angular strategy of Section 3.2. The k-sector partitions the receiver's frame of reference into k equal-sized sectors, each with angular size of $2\pi/k$. A Scheduled transmission is canceled when at least one message duplicate is received from each sector.

Note again that this scheme does not require it only requires it can also be realized using (approximate) angle of arrival (AOA) technology, which is a desired property in environments where GPS cannot be used (e.g. inside buildings). Niculescu and Nath [17] were able to obtain angle of arrival measurements using 802.11 with expected error of 22 degrees for a departmental indoor setting.

Analytical performance of the k-sector Strategy

The main goal of all algorithms presented in this paper is to improve the total number of retransmission, while obtaining good coverage. The k-sector Strategy suggests that good coverage is usually obtained. It is interesting to see how many messages in total are required to obtain the same quality of coverage that flooding guarantees, that is %100 of the area covered by the transmission range of at least one node. We define the depth $depth(p)$ of a point p as the number of retransmission of the message that p can hear. Obviously the smaller p is, the more efficient is our algorithm. For some setting all nodes have to retransmits in order to obtain a %100 coverage, or even just to obtain connectivity. This is the case if the nodes are far apart from each other along

a line. Yet in this case $depth(p)$ is probably a small constant, independent of which algorithm is used. More interesting is the case that a large number m of nodes all lie inside a small area D, say a disk of radius R, where R is the transmission range of a node. Now redundancy is significant. Even in this case an adversary might place all m nodes along a line but very close to each other, so if %100 coverage is required, than each node must retransmit, and $depth(p) = m$ for some point $p \in D$.

Let $depth(D) = \max_{p \in D} depth(p)$. We next show that the 4-sector broadcast scheme, under quite wide assumptions (defined formally below), yields a %100 coverage, but with very hight probability $depth(p) = O(\log^2 m)$.

Assume $R = 1$. We approximate the unit disk with an axis-parallel square (the analysis can be extended to convex n-gon for better approximation). The 4-sector scheme partitions the receiver's frame of reference into 4 quadrants, NW, NE, SE, SW. A scheduled retransmission of a node u is canceled only if u has receive at least one node in each quadrant. Note that in this case the coverage region of u must be contained inside the union of the coverage regions of these nodes, hence %100 converge is always guaranteed. Thus we are left with bounding the number of nodes that actually retransmits, and bounding $depth(D)$.

Consider a single message $\mathcal{M}$ that is simultaneously heard by all m nodes in D. We label these nodes $s_1 \dots s_n$, according to their vertical position (y-axis) such that the lowest one is labeled s_1 and the highest s_m. Ties are assigned arbitrarily. The order of the nodes along the x-axis defines a permutation of $(1, 2, \dots n)$. We denote this permutation as $(s_{i_1}, \dots s_{i_m})$.

Theorem 1 *If every permutation is equally likely, then the expected number of copies of $\mathcal{M}$ transmitted inside D is $\leq 4 \ln^2 m$.*

Proof: For simplicity we assume the range of transmission is an axis-parallel unit square. We say that s_i is *SE-dominator* if among all the nodes below it and to its right, s_i has the shortest backoff time. *SW-dominator*, *NE-dominator* and *NW-dominator* are symmetrically defined. Note that s_i will transmit **only** if it is a dominator with respect to one or more directions.

We now bound the probably that s_i is not SE-dominator. Note that s_i has $i-1$ nodes below it (i.e. their y-coordinate is smaller than the y-coordinate of s_i). Consider the projections of $(s_1 ... s_n)$ on the x-axis. Since all permutations of the projected points are equably likely, the probability that exactly j of the points of $\{s_1 \dots s_{i-1}\}$ are to the right of s_i (for $0 \leq j < i$) is $1/(i-1)$. In this case the probability that s_i is SE-dominator is $1/(j+1)$. To see why this is correct, recall that all nodes pick their backoff time simultaneously. Consider only s_i and the j points below it and to its right. The probability that s_i picked the smallest backoff time in this subset of $j+1$ nodes is therefor $1/(j+1)$.

Hence the probability that s_i is SE-dominator is:

$$\frac{1}{i-1}\sum_{j=1}^{i}\frac{1}{j+1} \tag{1}$$

Accordingly, the probability that s_i is dominator in one or more directions is therefore at most 4 times expression (1):

$$\leq \frac{4}{i-1}\sum_{j=1}^{i}\frac{1}{j+1} \;\leq\; \frac{4}{i-1}\ln m \tag{2}$$

Finally, the expected number of messages sent is obtained by summing expression (2) of all s_i:

$$\leq 4\sum_{i=1}^{m}\frac{\ln m}{i} \leq 4\ln^2 m$$

Finally, the $O(\log^2 m)$ bound on depth can be derived from the number of retransmissions occurring in D, and the number of retransmissions occurring in a constant number of neighboring unit disks, so their sum gives the desired bound.

3.4 Counter-Based Scheme

In this scheme, nodes may cancel a scheduled message retransmission of a broadcast packet p whenever the number of message duplicates $dups(p)$ reaches a threshold k. The adventurous of this method is that it does not require any knowledge of the distance or direction retransmitting nodes. Figure 4 shows the coverage vs. density for several values of k. Fifty connected topologies were generated for each node density using a uniform random node distribution. The values shown for each density corresponds to the average of all fifty executions. Note that k needs to be at least 5 to obtain a coverage of at least 98% for all densities.

3.5 Distance-Based Scheme

Again assume that the transmission range $R = 1$. This scheme is design to address the scenario where a node u receiving a message from a node v can provide a rough estimate of the distance $d(u, v)$ between them. In particular, it can estimate only whether $d(u, v)$ is smaller than some threshold α (it is always

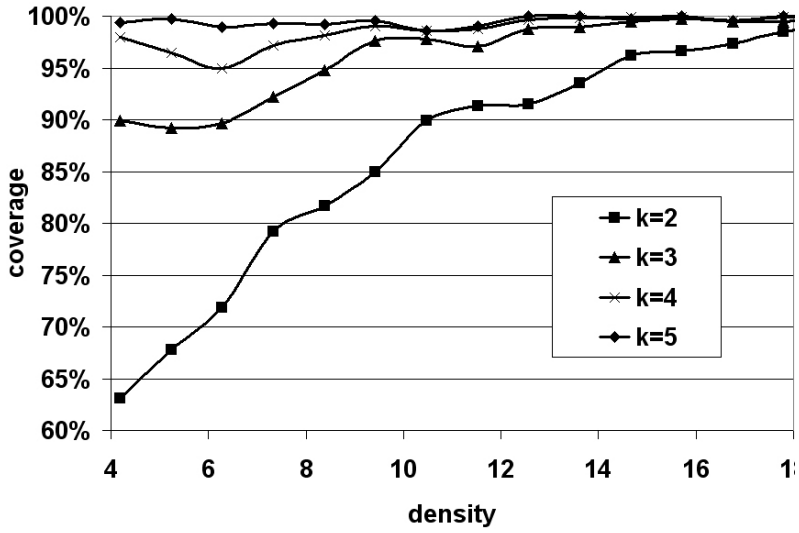

Fig. 4. Counter-based scheme. Coverage vs. density for various values of k.

assumed that $d(u, v) \leq 1$). This scheme can be used solely, or with combination with other one, e.g. the counter-based scheme. a scheduled retransmission is canceled if a node receives a message duplicate from another node that is less than α distance apart. This algorithm is based on the observation that the expected additional coverage is very small when a duplicate is received from a neighbor that is very close. This algorithm has the nice property that are similar to the algorithm of Section 3.3, i.e. coverage of %100 is guaranteed, while a simple packing argument shows that the maximum depth (number of duplicates heard by a point) can be bounded.

Theorem 2 *Let D be a unit disk. Then the number of nodes inside D that transmit a duplicate is at most $\frac{1}{6}\pi\sqrt{3}(1+\frac{\alpha}{2})^2$.*

Proof: Refer to Figure 5.Let p be a point, and consider the unit disk D centered at p, and let m be the number of transmitting nodes in D. We need to bound the number of nodes in D that transmit a duplicate. Consider two such nodes s_1 and s_2. Assume WLOG that s_1 has transmitted first. Since s_2 also transmitted, then $d(s_1, s_2) > \alpha$. Placing two disks of radius centered at s_1, s_2 and having radii $\alpha/2$, there disks must be disjoint. In general, all disks centered at transmitting nodes in D and having radii $\alpha/2$ must be pairwise disjoint. On the other hand, they are all contained in a disk D' of radius $1 + \alpha/2$ centered at p, Hence their total area is bounded by the area of D', and $m \cdot \pi(\alpha/2)^2 \leq \pi 1 + \frac{\alpha}{2})^2$, or

$$m \leq (1 + 2/\alpha)^2. \tag{3}$$

To improve this bound, we use previous results on the *packing density*. The problem of packing congruent disjoint disks in a disk is an old and difficult problem and [18] is a good reference. It was shown that the limit of

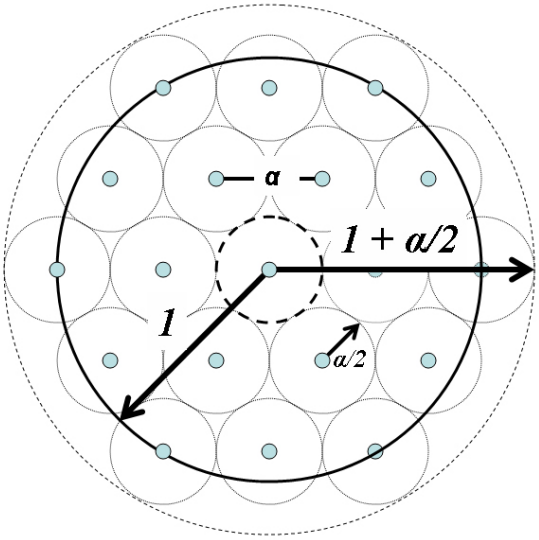

Fig. 5. the maximum number of discs of radius $\frac{\alpha}{2}$ that fit into a disc of radius $1+\frac{\alpha}{2}$ without overlapping.

the packing density as the radius of the enclosing circle tends to infinity, is $\frac{1}{6}\pi\sqrt{3} \approx 0.90689$, and by plugging this bound by the inequality (3) we obtain the claimed bound on m. **QED**.

Comment: It is interesting to note that while this threshold needed for most scenario are probably smaller, the bound provided by $k(\alpha)$ guarantees zero additional coverage.

3.6 Position-Based Scheme

There are several retransmission strategies that can make use of position information. Nodes are assumed to be equipped with GPS or other positioning devices and each message must bear the coordinates of the transmitting node.

One possible strategy is to compute the additional coverage according to the coordinates contained in the duplicates and cancel the retransmission whenever the additional coverage is zero, or falls below a predetermined threshold AC_{thresh}. Computing the additional coverage among many nodes is complicated as it requires additions and subtraction among all possible range intersections. One possibility is to approximate the range of each sending neighbor into a grid, by marking each squared as covered or not covered.

A second strategy is to cancel the retransmission if the local node lies inside the convex polygon defined by the location of any subset of k sending neighbors. There is a very high likelihood that the additional coverage is negligibly small if the local node lies within said polygon.

The parameter(s) of each retransmission strategy were set to achieve at least 98% coverage across all densities, and their overhead is compared in this section. Accordingly, $\theta_{\text{thr}} = 210°$ for the angular strategy, $k = 5$ for the counter-based scheme, an independent set of size 3 is used for the adjacency scheme, $\alpha = 0.11$ for the distance-based scheme, and 3 sectors were used for the k-sector scheme. For each density, 200 random (uniformly distributed) topologies were generated and the average overhead was computed for each strategy.

Figure 6 shows each strategy's overhead as a function of density. The angular strategy has the lowest overhead, in part due to the strong correlation between angular distance a additional coverage, but also because θ_{thr} is a real number that, unlike the parameters of other strategies, provides a finer granularity.

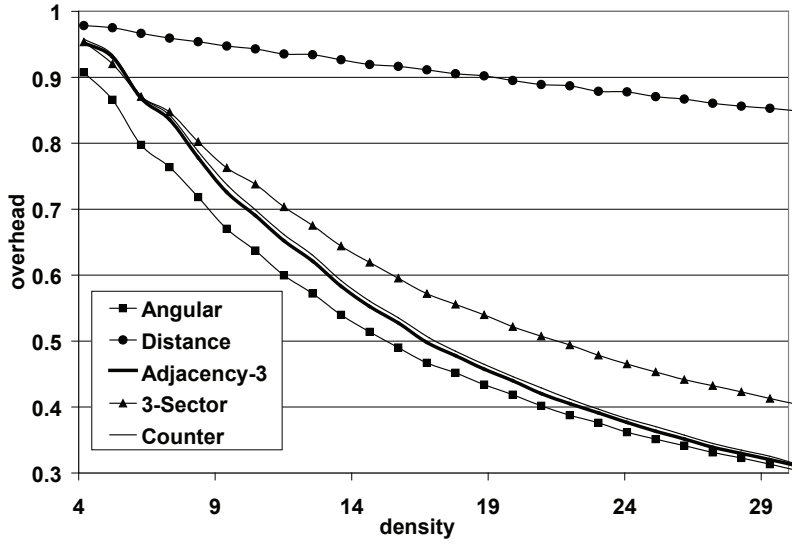

Fig. 6. Overhead vs. density comparison between retransmission strategies.

4 Backoff Choices

Timer-based contention has been previously studied in the context of location-based routing [19,20]. However, the backoff procedure by which nodes postpone retransmissions in network-wide broadcasting has not received much attention in the past. We introduce the concept of *backoff strategy* and define it as having three components:

(1) *The backoff magnitude*: How much time to wait. Usually specified as a maximum or average value.
(2) *The backoff function:* How are backoff times assigned to different nodes.

(3) *Reset strategy*: Nodes that have significantly reduced their additional coverage during the last backoff period could be rewarded with additional backoff time.

The backoff function could be as simple as a uniform random distribution of backoff times, but other alternatives could be considered. For example, backoff times could be assigned as a function of receive power where the function approximates a linear backoff decrease with respect to distance. The advantage of this backoff function is that it significantly reduces the average delivery latency. The message travels as concentric waves or rings emanating from the source and rapidly reaching the entire network. Messages are then delivered by the nodes on these rings to their local neighborhood.

Figure 7 compares the latency of a random backoff time distribution to a backoff function that assigns shorter times to weaker signals (distant nodes). We call this function *inv-linear* because the backoff times change with receive power in a way that approximates a linear decrease in backoff time with respect to distance. See section 5 for details of how this and other functions are defined. Also shown is the latency for the symmetrically opposite function (*linear*) where receive power is used to approximate a linear increase in backoff times with respect to distance.

Five different retransmission strategies are used to compare the backoff functions. 200 connected topologies were generated using a fixed node density. Each combination of retransmission and backoff function was executed on the 200 topologies and the averages are reported in the figure. The results of all the retransmission strategies were then averaged together to get an overall average for each backoff function.

The inv-linear function cuts the average delivery latency for all strategies by more than half with respect to a random distribution. In contrast the linear function results in an excessive increase in latency. One last observation is that the strategy chosen does not seem to have any effect on latency.

An interesting finding is that the choice of backoff function affects the overhead and that such effect varies with each retransmission strategy. This is due to an effect we call *directional diversity*. Think of a broadcast operation as the ripples formed when dropping a small rock into a pool of water and emanating from a single point. If the backoff function is such that the broadcast message (the ripples) travels slowly, then the nodes will only receive duplicates from the direction of the source, resulting in a reduced number of messages received. If the message travels faster, then (similar to the ripples), they will bounce off objects (the nodes) and travel in the opposite direction before the backoff period of inner nodes is over. The number of duplicates received is increased because the now arrive from all directions.

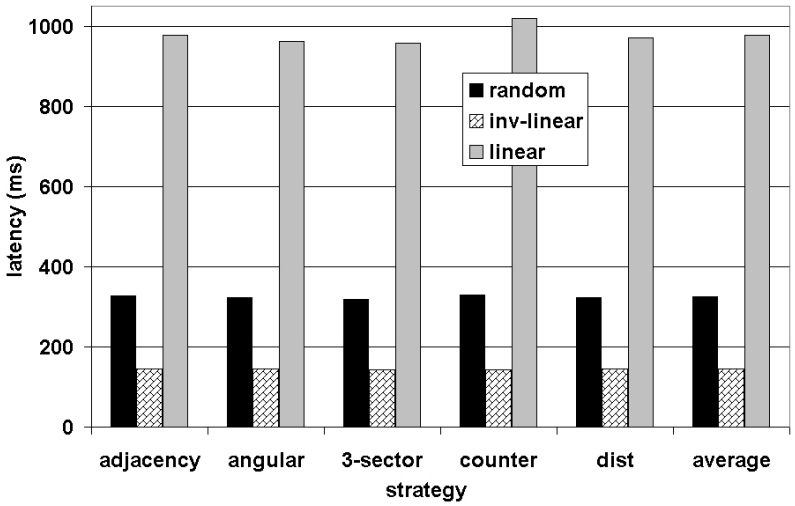

Fig. 7. Average latency vs. backoff function for several retransmission strategies. The Average for all retransmission strategies is also shown.

Figure 8 illustrates for different retransmission strategies the variation in overhead with respect to the choice of backoff function. The values illustrated in the figure were obtained in a manner identical to that of the previous experiment. In all the strategies except the distance strategy, the overhead for the slowest function (linear) is much higher when compared to random and inv-linear functions. There is very little directional diversity with the linear function, resulting in fewer duplicates.

The existence of directional diversity is corroborated by the overhead difference between the random and inv-linear functions for the angular and 3-sector strategies. By definition, the overhead of these two strategies depends on the diversity of the angle of arrival of the duplicates. And indeed, the faster propagation of the inv-linear function results in a clear overhead reduction. For the other strategies there is not much difference because they depend more on the number of duplicates rather than their direction of arrival.

The backoff magnitude is another important part of the backoff strategy. Figure 9 shows how the coverage and the latency vary as the maximum backoff time increases. The values shown represent the average for three different retransmission strategies using the inverse-linear backoff function and a node density of 15.7. The strategies used are 3-sector, Adjacency and Angular. Note that the latency has been normalized with respect to its maximum value (171 ms) in order to use a common y-axis and plot both function in the same figure. The reader would agree that an appropriate backoff time should be somewhere between 100ms and 200ms. After that the latency continues to increase linearly without really gaining any coverage improvement.

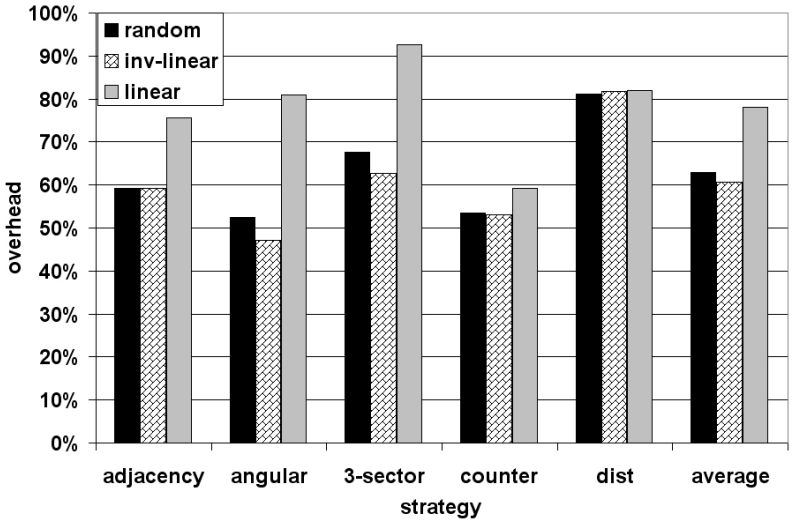

Fig. 8. Overhead for several backoff functions and retransmission strategies. The Average for all retransmission strategies is also shown.

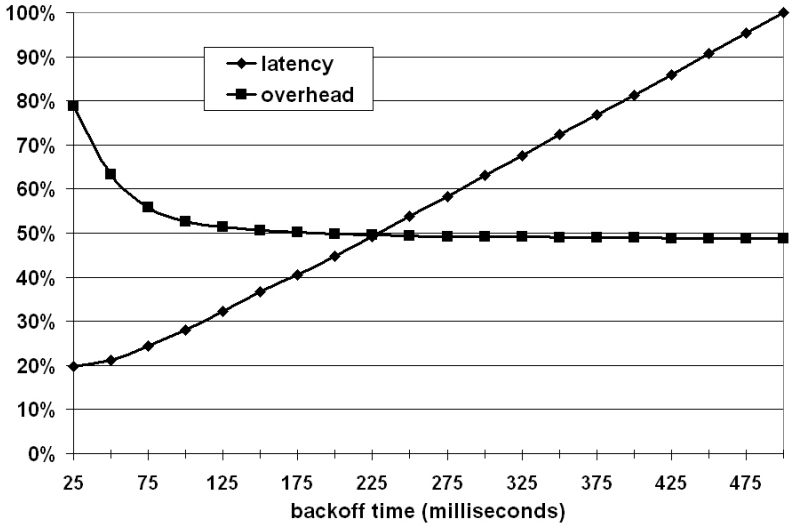

Fig. 9. Overhead and latency as a function of the maximum backoff time. The values represent the average for three different retransmission strategies using the inverse-linear backoff function.

5 Routing

Flooding is primarily used in routing. It is the basis for performing path discovery in on-demand routing protocols [21,22] that are commonly used in multi-hop wireless networks. This section studies the effects that different retransmission and backoff strategies may have on path discovery.

Path discovery from source s to destination d works by having s broadcast a *route request* (RREQ) that will eventually arrive at one or more nodes with a valid routing entry for d (possibly d itself). These nodes respond with a *route*

reply (RREP) that follows the reverse path of the *RREQ*. Node s will choose the RREP with the smallest hop count and create the corresponding routing entry for d.

Note that an intermediate node i with a routing entry for d would only respond to the first RREQ duplicate it receives. Accordingly, the resulting path is (s, i, d) for some intermediate node i, where the sub-path (s, i) is determined by the first RREQ duplicate received by i. The first RREQ to arrive at i will travel on a path that is either shorter than that of other duplicates, or a path whose nodes collectively and coincidentally selected shorter backoff times.

In section 4 we proposed a backoff strategy where nodes that are further away from the transmitter select a smaller backoff time. From the broadcast protocol's point of view, such strategy reduces the delivery latency as well as the overhead. From the routing protocol's point of view, our initial hypothesis is that this backoff strategy could lead to fragile routes that have a short duration. The first RREQ to arrive at i will travel through a path that has fewer nodes spread further apart but weaker links (power-wise). The slightest movement of nodes in the opposite direction could break a link and affect the lifetime of the route. We performed simulation experiments to test this hypothesis and the results are presented in this section.

Our second hypothesis is that in mobile ad-hoc networks the smaller end-to-end delays of shorter routes rarely compensates for their reduced route lifetime, and that hop-count is often a lousy metric. Accordingly, we propose to speed up RREQ propagation on those paths with stronger links by using a backoff time that is a function of the reception power. Higher reception power should result in lower backoff delays. In particular, we propose a function of the form:

$$t\left(P_r\right) = t_{max}\ \lambda^{\left(P_{\text{thr}}/P_r\right)^{1/\beta}-1} \tag{4}$$

where P_r is the reception power, $\lambda > 1$, and P_{thr} is the receive power threshold [4]. Let P_t be the transmission power, then:

$$P_t \geq P_r \geq P_{\text{thr}}.$$

β is the *path loss exponent*. The reception power decreases non-linearly with distance, and the degree of decrease is determined by the path loss exponent. The function:

[4] The minimal reception power required to decode the signal

$$\mu(P_r) = \left(\frac{P_{\text{thr}}}{P_r}\right)^{1/\beta} \tag{5}$$

is a transformation of the receive power into a normalized value in the interval [0,1] that increases linearly with distance. This transformation is used in (4) to obtain a adequate mapping between receive power and backoff time, but it is not an attempt to measure distance. Speeding up the broadcast operation on stronger links does not depend on the accuracy of measuring distance with the received signal strength. The backoff function $t\,(P_r)$ as defined in (4) will always result in a smaller backoff time for a stronger signal. There could be variability of the receive power with respect to time, but that is a different issue.

The parameter λ determines the *convexity* of the backoff function. Figure 10 plots

$$t\,(\mu) = t_{max}\ \lambda^{\mu-1}$$

for three different values of λ. Note that $t_{max} = 1$ and $0 \leq \mu \leq 1$. For our simulations we make $\lambda = 100$.

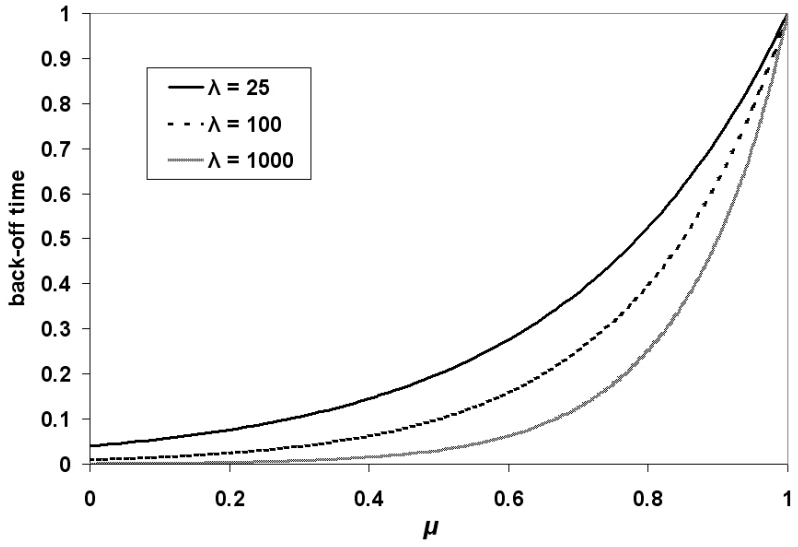

Fig. 10. Convex backoff function for path discovery using flooding with different values for the λ parameter.

Where concave functions prefers smaller backoff time to more remote nodes, and increases the additional coverage while reducing robustness. We thought it was would interesting to consider several other "natural" alternatives, so simulations were conducted to evaluate the path discovery process using the following backoff functions which are also illustrated in Figure 11:

$$t(\mu(P_r)) \leq \begin{cases} [0, t_{max}] & \text{random} \\ t_{max}\ \lambda^{\mu(P_r)-1} & \text{convex} \\ t_{max} - t_{max}\ \lambda^{-\mu(P_r)} & \text{concave} \\ t_{max}\ \mu(P_r) & \text{linear} \\ t_{max} - t_{max}\ \mu(P_r) & \text{inv. linear} \end{cases} \tag{6}$$

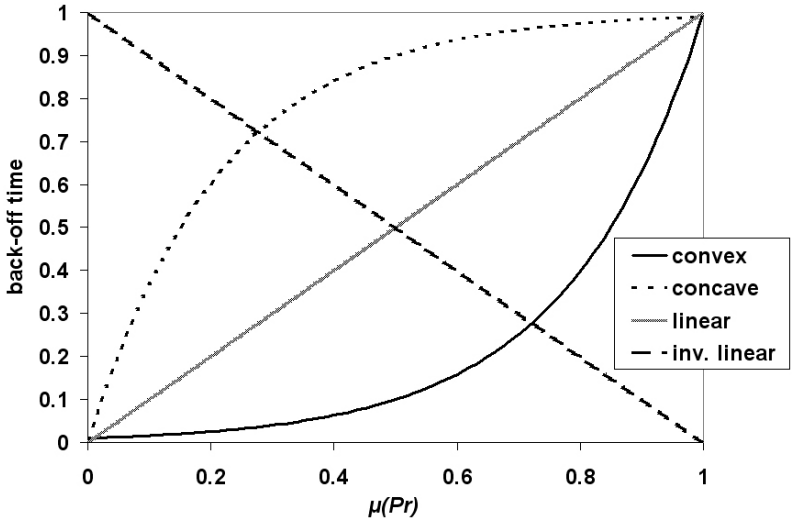

Fig. 11. Backoff functions for path discovery.

Figure 12 shows the average path duration for each backoff function in (6) under five different retransmission strategies. The figure also shows the average path duration for each backoff function, obtained by averaging the results of all five retransmission strategies. Each retransmission and backoff strategy was simulated on 200 connected topologies that were generated for this experiment. Each individual run was 300 seconds of simulation time.

The results indicate that the longer lasting paths are always obtained with the convex backoff function. The concave and linear backoff functions also result in longer path durations than the random and inv-linear backoff functions. However, an important observation is that in most cases the inv-linear backoff function performs better than the random backoff function. This is a significant result because it means we can reduce the latency and overhead without sacrificing path-duration beyond what can be achieved with random backoff.

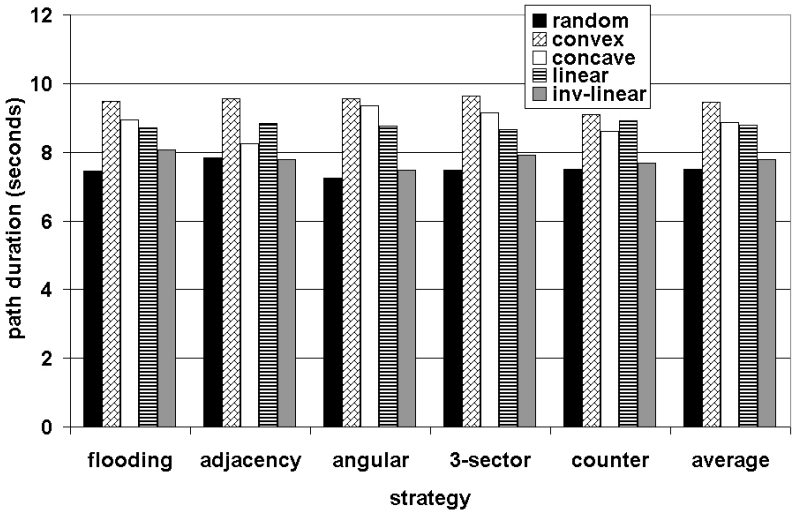

Fig. 12. Duration of first path found with five different backoff functions and retransmission strategies.

6 Strip Broadcasting. Broadcasting Between Moving Nodes

If nodes (vehicles) are assumed to be located on a strip (road) and the range of transmission is carefully chosen with respect to width and curvature of the strip, then in most situations a retransmission is deemed to be redundant once a message has been received from the front and back of the vehicle. Accordingly, we propose the *Strip Broadcasting* (SB) algorithm shown in Algorithm 1, where scheduled retransmissions are canceled once the message is received from both directions. Vehicles could use direction of arrival (DOA) schemes described in the signal processing literature to determine if a message is coming from the front or from the rear.

Algorithm 1 Strip Broadcasting Algorithm

Require: packet p
1: **if** $duplicates(p) = 1$ **then**
2: $\quad schedule\_tx(p)$ ▷ Schedule retransmission
3: **end if**
4: **if** p arrived from the rear and p arrived rom the back **then**
5: $\quad$ Cancel scheduled retransmission
6: **end if**

Most highways of course have curves and multiple lanes with vehicles traveling in both directions. The performance of the algorithm depends on the correct selection of the transmission range and the curvature of the strip. Two possible scenarios are shown in Figure 13. If the transmission range is too small then part of the road might not be covered (a). If it is too big then the road might change directions within the range of the transmitter (b). This may cause some nodes to incorrectly determine the direction of an incoming message.

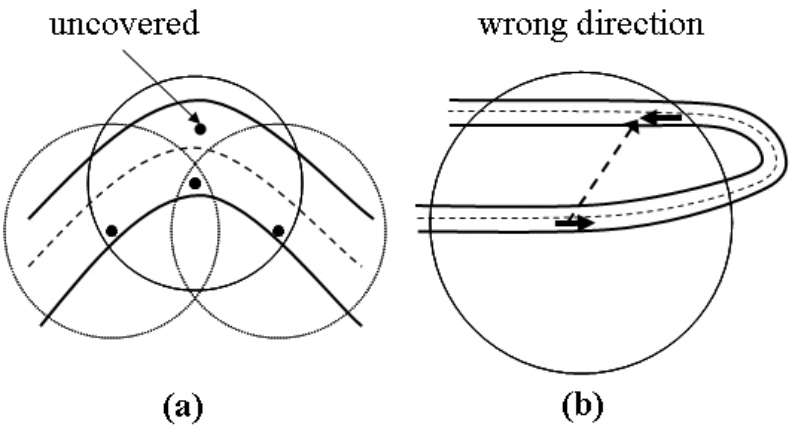

Fig. 13. Strip Scenarios.

6.1 Definitions and Analytical Model

The SB algorithm can be analyzed by modeling the problem as a one dimensional arrangement of n nodes on a segment of length L, as shown in Figure 14. All transmissions are omnidirectional have the same transmission range R. A node that receives a duplicate from the left and from the right does not need to retransmit.

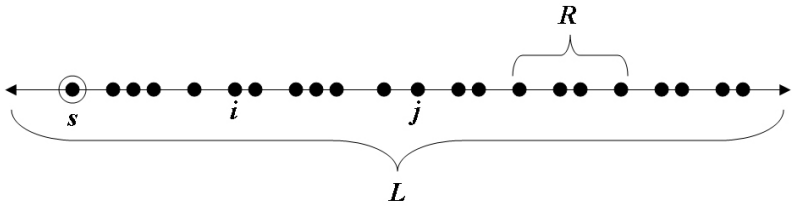

Fig. 14. Analytical Model

Let s be the *source* of the broadcast message, meaning it is the first node to transmit the message. The broadcasting process is symmetrical on both sides of s, thus it is assumed that s is the leftmost node and the model only considers the broadcast process to the right of s. If node x is closer to s than node y, then node x is called a *predecessor* of y, and node y is called a *successor* of x. In Figure 14, j is a successor of i.

For analytical purposes, the communication model is based on ideal network conditions. Transmission errors and collisions are not considered. Both discrete-time and continuous-time analyse are considered. In the discrete model, time advances in discrete slots of length Δ, and the time necessary for a node to retransmit the message to all its neighbors is assumed to be less than Δ. When a node receives the first duplicate, with some probability q it will retransmit the message in the next time slot. With probability $1 - q$ it will wait for the next slot, and the process is repeated until the packet is sent

or the scheduled retransmission is canceled. In the continuous time model, the backoff period is exponentially distributed, and the transmission delay is assumed to be short enough such that there is a negligible probability that two retransmissions overlap in time. This assumption is relaxed later on and analytical results are provided for overlapping retransmissions. Because of the memoryless property of both time models, the waiting time for all nodes is equally distributed regardless of how long ago they received the first duplicate. As before, we define the *depth* at a point p ($depth(p)$) as the number of duplicates of the message heard at p. Again we seek to obtain a 100% coverage, as long as connectivity is preserved.

6.2 Retransmission Analysis

This section presents analytical results about the number of retransmissions performed in a broadcast operation by the SB algorithm under the model defined in Section 6.1. The following lemma is similar to Theorem 1, but in contrast to this Theorem, no assumptions are needed about the locations of the nodes.

Lemma 1 *Consider an interval D of the road of length R, where R is the transmission range, and let m denote the number of nodes inside D. Consider all possible picks of backoff periods. Then the expected number of retransmissions occurring in D is at most $1 + \ln m$.*

Proof: Consider any interval of length R (refer to the interval $[y, z]$ shown in Figure 15). Clearly the more nodes that lie to the right of z, the higher the probability that one of them will transmit, causing some or all of the nodes in $[y, z]$ to drop their message. Hence to obtain an upper bound, we would assume that, excluding the rightmost node in $[y, z]$ (node 3 in Figure 15), no node in $[y, z]$ can reach a node to the right of z.

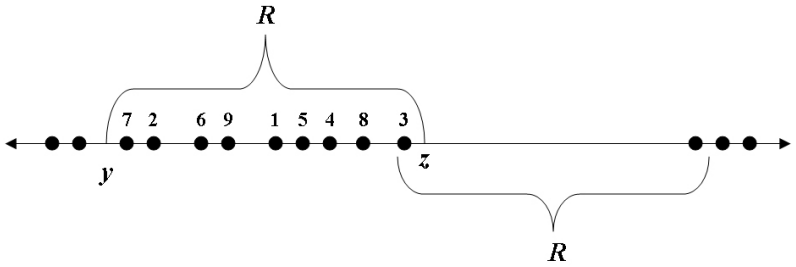

Fig. 15. An empty interval to the left of length at least R represents a wort-case analysis.

Enumerate the nodes in $[y, z]$ according to the order in which they complete their backoff period (breaking ties arbitrarily). Let $(s_1 \dots s_m)$ denote the resulting sequence of nodes, so s_1 is the first node completing its backoff period,

and s_m the last. Let X_i be a random variable associated with s_i. We set $X_i = 0$ if s_i cancels its retransmission since it received a message from both directions. Otherwise (if s_i retransmit) we set $X_i = 1$.

Note that s_i retransmit if and only if its location is the rightmost among the nodes $(s_1 \dots s_i)$. Otherwise, there is a node to its right, which transmits before s_i's backoff period has expired, causing s_i to cancel its retransmission. Since every permutation of $s_1 \dots s_i$ is equally likely, s_i has equal probability $1/i$ of being in any of the possible i locations occupied by $\{s_1 \dots s_i\}$. This is because the memoryless property of either the discrete time or continuous time models makes all node permutations equally possible. As a result, the probability $\Pr(X_i = 1)$ that s_i retransmits is $1/i$. Thus $E[X_i] = \Pr(X_i = 0) \cdot 0 + \Pr(X_i = 1) \cdot 1 = \frac{1}{i}$. The number of retransmissions is $\sum_1^m X_i$, and its expectation is and the expected value of X_i is

$$E\left[\sum_{i=1}^{m} X_i\right] = \sum_{i=1}^{m} E\left[X_i\right] = \sum_{i=1}^{m} \frac{1}{i} \leq 1 + \ln m$$

QED.

The reason that $1 + \ln m$ is only an upper bound, is that messages can be received also from the right of y, and reaching some of the nodes in $[x, y]$. Next we show that for some scenario, the bound is tight.

Corollary 1 *There is a positioning of the nodes along the strip, for which the bound of Lemma 1 is tight, up to a constant.*

Proof: Consider the arrangement depicted in Figure 16. Along this strip nodes are located in intervals containing m nodes, called a *dense set*, followed by intervals containing a single node, called a *singleton*. Once a singleton retransmit, all nodes in the following intervals receives the message, and only once the rightmost node in the dense set retransmit, the message is received by the singleton in the following interval. As easily verified, the inequalities of Lemma 1 is tight. **QED**.

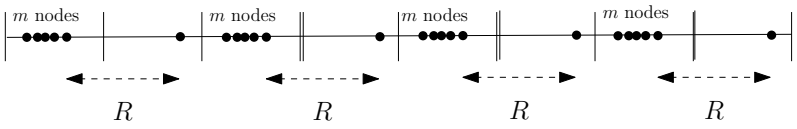

Fig. 16. Example of a node arrangement whose expected number of retransmissions is $1 + \ln m$.

Discussion: A fundamental assumption of these theoretical results is that no two nodes decide to retransmit at the same time. A practical interpretation of this assumption is that there are no *concurrent retransmissions*. Two retransmissions are said to be concurrent or *overlapping* if the nodes involved are

within range of each other and if the difference between their backoff times is less than the *retransmission delay*. The retransmission delay is the time interval from the moment the sender completes the backoff period to the moment the receiver determines it can cancel its own scheduled retransmission.

Concurrency increases the overhead significantly. Note that n concurrent retransmissions can increase the overhead by up to $n - 1$ additional retransmissions. Consider a set n nodes within range of each other where $n - 1$ nodes complete their backoff period while a retransmission from their rightmost neighbor is taking place. There are $n - 1$ nodes that will not be able to cancel their retransmission even though they all have a neighbor on the right that started transmitting first.

Retransmissions can be concurrent without being physically transmitted at same time, which would amount to having a collision on a shared channel. This happens when the broadcast algorithm is implemented above the MAC layer. There can be nodes that have decided to retransmit the message, but these packets are still on the system, perhaps traveling down the protocol stack.

The problem with collisions is that the number of retransmissions is not only increased by the concurrency but also because there is a potentially large set of nodes that will not be able to receive some or all of these messages. This disrupts the dynamics of the broadcast operation, resulting in an even larger number of retransmissions.

7 Simulation Results

The experimental results presented in this section evaluate the performance properties of the algorithm. Such properties include retransmission overhead, coverage (reliability), scalability and latency. The SB algorithm was implemented at the application level, meaning that the backoff period of the SB algorithm takes place outside of the MAC layer. Once the backoff period is complete, the packet travels down the protocol stack and is handed to the MAC layer, where an additional backoff period is used as a collision avoidance mechanism. The MAC protocol used in our simulations is 802.11. For broadcast messages, 802.11 is reduced to a simple slotted CSMA protocol.

7.1 Comparison with the counter-based algorithm

We first show that in one-dimensional arrangements the SB algorithm performs better than algorithms originally designed for two-dimensional arrangements. We perform a comparison with the counter based algorithm because a retransmission strategy based on a counter is not intrinsically related to the dimensionality of the problem and because the counter-based algorithm performs very well in two dimensions.

Both algorithms were evaluated for nodes randomly arranged in a one dimensional segment whose length is 20 times the transmission range. Fifty node distributions were generated for each density and an average was obtained among the 50 results. Backoff times are uniformly distributed between 0 and 200 milliseconds for both algorithms. In each run, a message is broadcasted from the leftmost node.

Figure 17 shows the coverage vs. density for the SB and counter-based algorithms for one-dimensional node arrangements. The SB algorithm achives a perfect coverage for all densities. With the counter-based algorithm, the coverage falls as far as 86% even when the counter threshold is as high as 4.

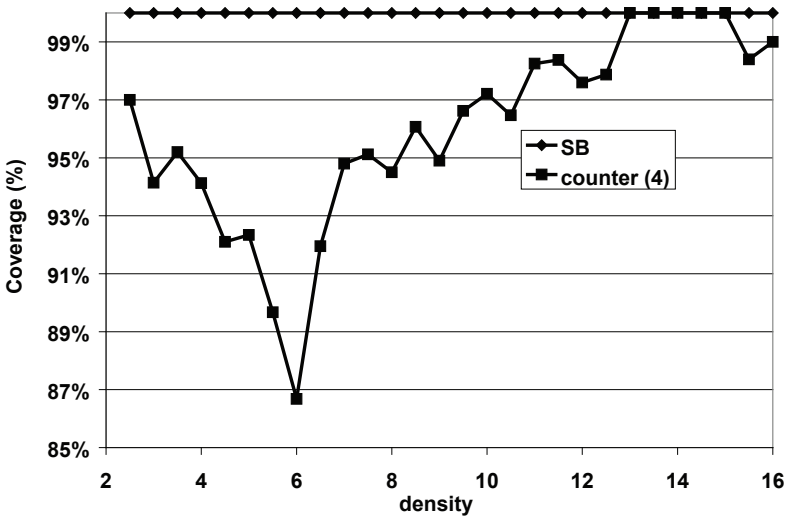

Fig. 17. Coverage vs. density for the SB and counter-based algorithms for one-dimensional node arrangements.

Figure 18 shows the overhead vs. density for the same experiment. The SB algorithm has less overhead across the entire range of densities. The SB algorithm is both much more reliable and efficient than the counter-based algorithm.

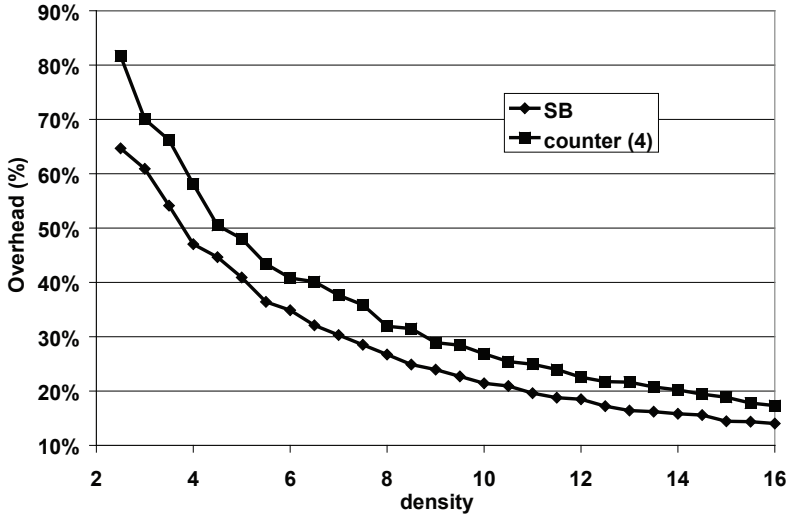

Fig. 18. Overhead vs. density for the SB and counter-based algorithms for one-dimensional node arrangements.

7.2 Depth, density and backoff period

Figure 19 shows the depth as a function of the node density when the backoff period is exponentially distributed. Results are shown for four different expected values of the backoff period (150 ms, 250 ms, 500 ms, 16 sec). The exponential regression of each dataset is also shown.

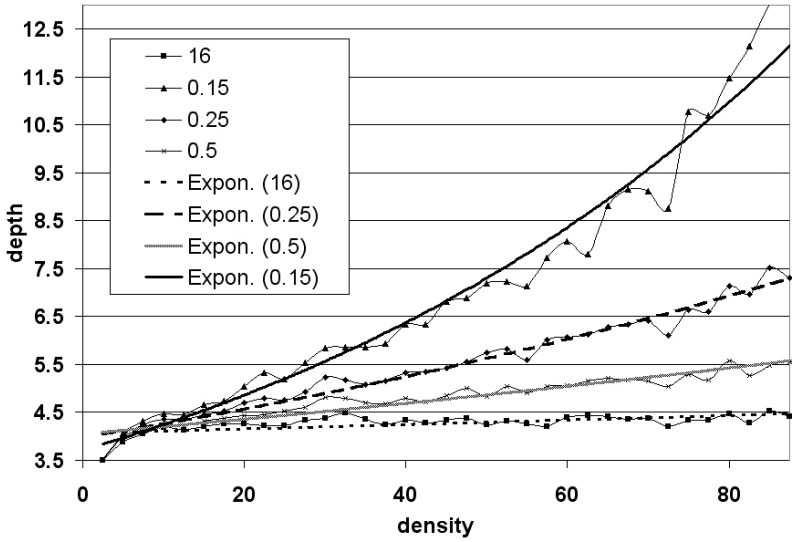

Fig. 19. Depth vs. density for different mean values of exponentially distributed backoff time.

If there are no overlapping retransmissions then the depth is 4 and does not depend on the density or the mean backoff period. However, the probability of concurrent retransmissions increases with shorter mean backoff periods or higher densities. This results in depth values that are considerably higher than

our theoretical result of 4. Note that the results for the mean backoff period of 16 seconds confirms that the depth approaches four when the probability of concurrent retransmissions is small.

7.3 Latency and Scalability

Figure 20 compares the scalability of the SB protocol to the standard flooding protocol. The average delivery latency of both protocols is compared as the number of simultaneous broadcasts operations is increased in each iteration. The average delivery latency is the average duration of time that it takes for a node to receive the first message. The SB protocol has a mean backoff period of 150 milliseconds, and it is compared with two configurations of the flooding protocol. One with a mean backoff period of 150 milliseconds and another one with no backoff period. The SB algorithm increasingly outperforms both flooding configurations as the number of simultaneous broadcasts increases. When the number of broadcast operations is increased, the adverse effect of collisions has a greater impact on the flooding protocol.

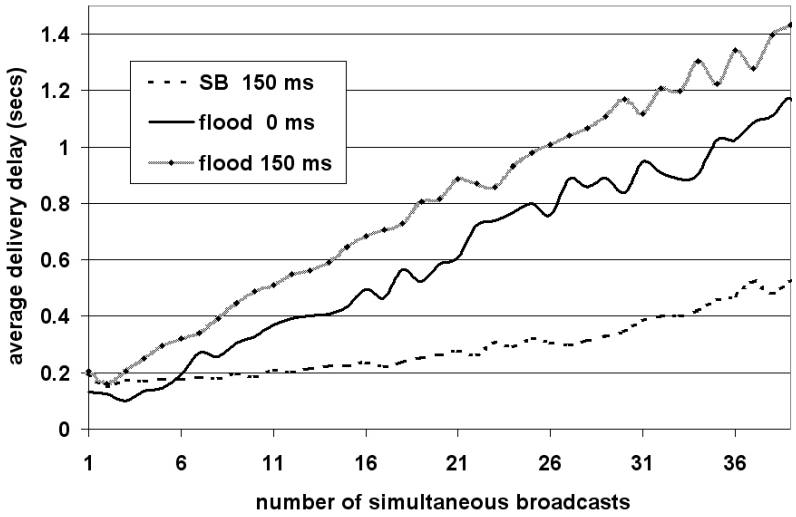

Fig. 20. Comparison of analytical and simulation depth results.

7.4 Memoryless vs Randomness Properties in Backoff Time Distributions

This section presents an interesting experiment comparing the memoryless property that the exponential distribution exhibits with the maximum randomness or spread that the uniform distribution provides. We will see that both properties are beneficial and therefore desirable to have in a backoff time distribution. Unfortunately, we cannot maintain both properties.

Note first that all nodes currently waiting to retransmit did not necessarily start waiting at the same time since they could have received the first duplicate from different nodes at different times. The memoryless property gives all nodes equal probability of being the first one to retransmit because the duration a node backoff is independent the duration it has already waited. This has a favorable effect on the number of retransmissions. Without the memoryless property nodes closer to the source would have a tendency to retransmit first as they have waited longer. The retransmissions would then be closer to each other, less nodes are silenced and the broadcast operation propagates slowly.

On the other hand, a uniform discrete distribution has less propensity for collisions because its probability density function is more evenly spread across the range of values. In other words, a uniform distribution is as random as it gets. This too has a favorable effect on the number of retransmissions. More randomness means fewer collisions between duplicates that will potentially silence more nodes.

Figure 21 shows that having less propensity to collisions proved more favorable than having the memoryless property. The uniform distribution results in smaller depths across the simulated density spectrum. A mean backoff period of 150 milliseconds was used for both distributions.

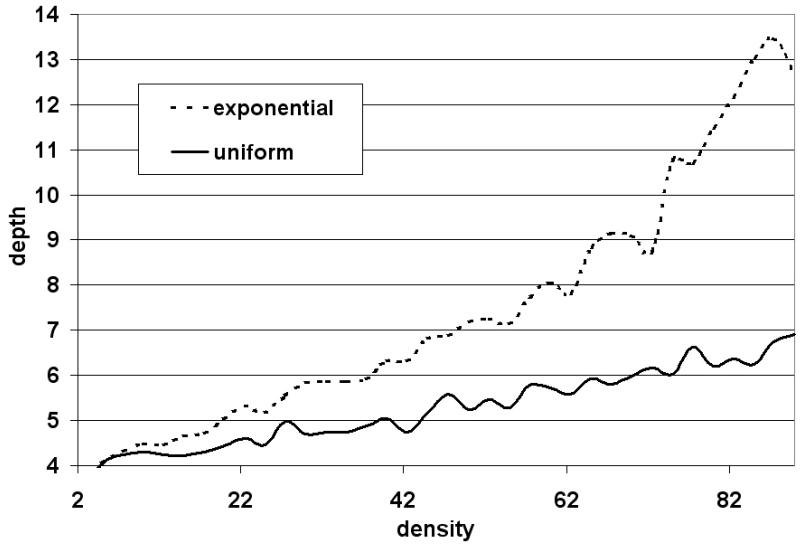

Fig. 21. Exponential versus Uniform backoff period distributions.

8 Conclusions

In this paper, we study the application-level backoff timers necessary for effective broadcasting in a wireless ad hoc network. We observe that the application-level back-off time required to achieve 100% coverage decreases

with increasing node density. At low node densities, smaller application-level backoff timers lead to collisions at the MAC layer, achieving less than 100% coverage. Thus, application-level backoff timers are critical at lower node densities. With this observation, we studied different strategies for setting the application-level backoff timers, and how to use information from the node's neighbors to determine whether the node should participate in the forwarding process.

The Angular retransmission strategy presented in this article has the least overhead among all compared strategies. It is a simpler way to implement a retransmission strategy based on location information. Another advantage of the angular strategy is that it can either be implemented with GPS or angle of arrival (AOA) technology. When reliance on these technologies is not possible, the simplicity and efficiency of the counter-based scheme makes it the best choice.

A good backoff strategy is important for minimizing the latency and taking advantage of directional diversity to reduce the overhead. For path discovery, the backoff strategy must also consider the impact on route lifetime. The invlinear backoff function reduces the latency and overhead without sacrificing path-duration beyond what can be achieved with random backoff.

This article has presented a broadcasting algorithm for applications such as vehicular broadcasting that can be modeled as a set of nodes arranged on a strip and analyzed as a 1-dimensional arrangement of nodes. Under ideal network conditions, the algorithm exhibits a constant overhead independent of the density and therefore scales better than flooding.

The article also analyzes which properties of backoff distributions affect the broadcasting overhead and show that uniformly distributed backoff times is the better choice. Another key finding is that waiting for additional duplicates before deciding to retransmit actually decreases the delivery latency because reducing overhead reduces collisions and speeds up message propagation.

Acknowledgment We would like to thank the anonymous reviewers for their valuable comments. We also wish to thank Mikael Degermark for introducing us to the problem. Alon Efrat and Jesus Arango we partially supported by NSF CAREER grant 0348000.

References

[1] J. Arango, M. Degermark, A. Efrat, and S. Pink, "An Efficient Flooding Algorithm for Ad-hoc Networks," *Proceedings of the Second Workshop on*

Modeling and Optimizations in Mobile Ad Hoc and Wireless Networks (WiOpt 2004), 2004.

[2] J. Arango, A. Efrat, S. Ramasubramanian, M. Krunz, and S. Pink, "Retransmission and Backoff Strategies for Broadcasting in Multi-hop Wireless Networks," *Broadband Communications, Networks and Systems, 2006. BROADNETS 2006. 3rd International Conference on*, pp. 1–10, 2006.

[3] J. Arango, A. Efrat, S. Ramasubramanian, and M. Krunz, "Onroad Vehicular Broadcasting," *Computer Communications and Networks, Proceedings-15th International Conference on*, pp. 49–55, 2006.

[4] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. 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*. New York, NY, USA: ACM Press, 1999, pp. 151–162.

[5] B. Williams and T. Camp, "Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks," in *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, 2002, pp. 194–205. [Online]. Available: citeseer.ist.psu.edu/williams02comparison.html

[6] V. K. Paruchuri, A. Durresi, D. S. Dash, and R. Jain, "Optimal Flooding Protocol for Routing in Ad-hoc Networks," in *IEEE Wireless Communications and Networking Conference*, March 2003.

[7] J. Cartigny, D. Simplot, and J. Carle, "Stochastic flooding broadcast protocols in mobile wireless networks," Technical Report LIFL Univ. Lille 1 2002-03, May 2002.

[8] E. Pagani and G. P. Rossi, "Providing Reliable and Fault Tolerant Broadcast Delivery in Mobile Ad-hoc Networks." *MONET*, vol. 4, no. 3, pp. 175–192, 1999.

[9] I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating Sets and Neighbor Elimination-Based Broadcasting Algorithms in Wireless Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 1, pp. 14–25, 2002. [Online]. Available: citeseer.ist.psu.edu/stojmenovic01dominating.html

[10] J. Wu and H. Li, "A Dominating-Set-Based Routing Scheme in Ad Hoc Wireless Networks," *Telecommunication Systems*, vol. 18, no. 1-3, pp. 13–36, 2001. [Online]. Available: citeseer.csail.mit.edu/wu99dominatingsetbased.html

[11] H. Zhang and Z.-P. Jiang, "Analysis of two ad hoc broadcasting protocols," in *IEEE Wireless Communications and Networking Conference*, Atlanta, GA, March 2004, pp. 21–25.

[12] L. B. Michael and M. Nakagawa, "Non-Platoon Inter-Vehicle Communication Using Multiple Hops," *IEICE Trans. Commun*, vol. E82-B, no. 10, October 1999.

[13] M. T. Sun, W. C. Feng, T. H. Lai, K. Yamada, and H. Okada, "GPS-Based Message Broadcast for Adaptive Inter-Vehicle Communications," in *IEEE VTC*, 2000.

[14] H. Wu, R. Fijimoto, and G. Riley, "Analytical Models for Information Propagation in Vehicle-to Vehicle Networks," in *IEEE VTC*, 2004.

[15] T. V. Project, "The *ns* Manual," Available in html, postscript and PDF, December 2003.

[16] K. Obraczka, K. Viswanath, and G. Tsudik, "Flooding for reliable multicast in multi-hop ad hoc networks," *Wirel. Netw.*, vol. 7, no. 6, pp. 627–634, 2001.

[17] D. Niculescu and B. Nath, "VOR Base Stations For Indoor 802.11 Positioning," in *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking.* New York, NY, USA: ACM Press, 2004, pp. 58–69.

[18] R. Graham, B. Lubachevsky, K. Nurmela, and P. Ostergard, "Dense packings of congruent circles in a circle," *Discrete Mathematics*, vol. 181, no. 1, pp. 139–15, February 1998, a good reference to what is known about packing congruent circles in a circle.

[19] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," *Ad-Hoc Networks*, vol. 1, no. 4, pp. 351–369, 2003.

[20] H. Fler, J. Widmer, M. Ksemann, M. Mauve, and H. Hartenstein, "Beaconless Position-Based Routing for Mobile Ad-Hoc Networks," Department of Computer Science, University of Mannheim, Tech. Rep. TR-03-001, Feb 2003.

[21] D. B. Johnson, D. A. Maltz, and J. Broch, *DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks.* Addison-Wesley, 2001, ch. 5, pp. 139–172.

[22] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," in *2nd IEEE Workshop on Mobile Computing Systems and Applications*, WMCSA. New Orleans, Lousiana, USA: IEEE, February 1999, pp. 90–100.