Specification and Performance Evaluation of Two Zone Dissemination Protocols for Vehicular Ad-hoc Networks

Jeppe Brønsted* and Lars Michael Kristensen†
Department of Computer Science, University of Aarhus
Aabogade 34, 8200 Århus N, Denmark, +45 8942 5600
{jb,kris}@daimi.au.dk

Abstract

Vehicular ad-hoc networks is an emerging research area focussing on communication infrastructures that support vehicles and road-signs in distributing road-state data such as information about hazardous road conditions ahead, approaching emergency vehicles, and traffic delays. Vehicular ad-hoc networks combine the areas of sensor networks (data acquisition) with mobile ad-hoc networks (highly dynamic topology and lack of pre-existing infrastructure). One of the main challenges of vehicular ad-hoc networks is the data dissemination protocols capable of distributing road-state information among vehicles. This paper presents two candidates for dissemination protocols: a zone flooding protocol and a zone diffusion protocol. The two protocols combine ideas from sensor networks and geocasting to ensure that data is aggregated and distributed only in a bounded geographical area. We present a comparative simulation study of the two protocols evaluating their relative performance using conventional metrics (such as network load) as well as application-specific metrics (such as awareness). The simulation study has been conducted using the Network Simulator 2 (NS-2) and has highlighted key properties of the two protocols that can be used as a basis for selecting the most appropriate protocol.

1 Introduction

Ad-hoc wireless communication among vehicles enables a multitude of applications ranging from improved traffic safety to road maintenance and high-speed tolling, and promises to significantly change the transportation sector in near future [1, 2]. The new applications rely on the acquisition and processing of sensor data and the dissemination of data via infrastructure-less wireless communication. The area of vehicular ad-hoc networks thereby combines the areas of sensor networks and ad-hoc communication.

A main challenge of vehicular ad-hoc networks is the protocols for dissemination of sensor data among vehicles. Information about a phenomenon observed by the sensors must reliably reach the vehicles that may be affected by this phenomenon in due time, so that drivers can react without creating dangerous situations. Furthermore, the protocols must be able to handle high vehicle density and mobility and at the same time be robust to sparse network connectivity.

Previous work [14] has shown that the performance of advanced dissemination protocols [23, 26] is highly sensitive to the parameters chosen and to the traffic scenarios. These advanced dissemination protocols typically improve performance by exploiting properties of the environment that may not always be available. The effect is that these protocols have good performance under some conditions, but poor performance under other conditions. One way to avoid this is reduce the assumptions about the environment and use light-weight protocols (both in terms of protocol complexity, parameters, and internal state) which provide reasonable performance in all traffic scenarios. These light-weight protocols will never reach the high performance of the advanced protocols when these operate under optimal conditions, but will perform reasonably under most conditions arising in practise. Furthermore, light-weight protocols typically use less computational resources and are easier to implement.

The work presented in this paper has been developed in the context of the LIWAS research project[17]. The LIfe WArning System (LIWAS) is a traffic warning system for informing drivers about hazardous driving conditions such as ice, water, and snow on the road being

*Supported by ISIS Katrinebjerg Software
†Supported by the Carlsberg Foundation.

0-7695-2559-8/06 $20.00 © 2006 IEEE

approached. LIWAS units are embedded systems with sensors capable of measuring a wide range of physical phenomena such as light reflection, air temperature, and dew point. LIWAS units are mounted on vehicles and alongside roads to detect the condition of the road. The sensed data is combined into road classifications and distributed to other LIWAS units to provide information to drivers. It is not only information about where a road is icy that is distributed, but also information about where the road is *not* icy. A driver can thus be certain about whether the road ahead is safe before initiating an overtaking.

The communication infrastructure supporting the LIWAS system can be realized in many ways ranging from a centralized architecture based on, e.g., GSM or GPRS networks combined with Web servers [6] to a decentralized architecture based on ad-hoc networking and multi-hop communication between vehicles. The two architectures have their pros and cons. The centralised architecture has an advantage in coverage, but a potential problem with scalability. The decentralized architecture has an advantage in scalability, but may have a potential problem when the density of the vehicles equipped with LIWAS units is low. In this paper we focus on protocols suited for the decentralised architecture.

The contribution of this paper is twofold. The first contribution is the specification of two protocols for data dissemination in vehicular ad-hoc networks. A Zone Flooding (ZF) protocol and a Zone Diffusion (ZD) protocol. They are both designed to be light-weight protocols, robust to varying network density and mobility. The ZF protocol is a variant of flooding and constitutes a very simple protocol while the ZD protocol exploits properties of the data to optimize data dissemination by means of aggregation. Both protocols can be used for the LIWAS system, but also for other traffic information systems dealing with information about the road and/or vehicles.

The second contribution is a comparative simulation study evaluating the protocols through simulation of various mobility scenarios. The comparison is done in terms of conventional metrics such as network load and through application specific metrics that expose properties relevant to applications using the data dissemination protocols. The simulations have been conducted using the network simulator NS-2 [24].

The rest of the paper is structured as follows. Section 2 provides a brief survey and comparison of related work. Section 3 contains the specifications of the two zone dissemination protocols. Section 4 presents the simulation model and defines the performance metrics. Section 5 presents the evaluation results for the two protocols. Finally, in section 6 we sum up the conclusions and discuss future work.

2 Related Work

Several protocols for data dissemination in vehicular ad-hoc networks can be found in the literature. They can roughly be divided into three categories: unicast, flooding, and diffusion.

Traditional ad-hoc network routing protocols [9] or position based routing protocols [19, 27] can be used to establish general unicast communication in a vehicular ad-hoc network. A service discovery mechanism is then required to let nodes know where to get the needed information [8, 21]. There is, however, an overhead in maintaining the service discovery mechanism, neighbour lists, and routing tables that introduces latency and diminished network capacity making this method infeasible for most safety critical applications.

The other two methods (flooding and diffusion) rely on the observation that the importance of sensed information about a particular location decreases with the distance to that location. Data therefore only needs to be disseminated in the vicinity of its origin. This is the case for most safety applications, but not for e.g. infotainment [4] or environmental applications, where all data comes from or is collected at a central location.

Flooding can be used to disseminate data in a certain area which can be determined in different ways. The work presented in [5] and [20] uses hop-count to limit forwarding of packets whereas in [13] the area is implicitly defined by an application specific interest rate function. Before a node forwards a received packet it uses the interest rate function to determine whether the amount of interest its neighbours have in the packet is above a certain threshold. Geocasting[12] assumes that nodes can determine their geographical location (e.g. using a GPS device[16]) and stop forwarding when packets leave a predetermined geographical area. This method is used in our Zone Flooding protocol described in the next section.

A general problem with flooding protocols is that they tend to have a lot of redundant transmissions which causes several problems [23]. This can be remedied by letting the nodes attempt to estimate whether a potential retransmission will be redundant [5, 23] or by lowering the transmission power [20]. In the Zone Flooding protocol we limit the amount of redundant transmissions by ensuring that a node transmits a packet at most once. More advanced mechanisms have been left out to keep the protocol light-weight.

Another method for dissemination of information in the neighbourhood of a source is to use a technique known as diffusion [15, 22]. Each node maintains a view of its surroundings and periodically broadcasts that view. Each time a view is received that view is aggregated with the local one. We use this approach in the Zone Diffusion protocol. In [22] the authors compare different aggregation algorithms using application specific metrics similar to ours, but do not try to estimate the relationship between network load and protocol performance as we do. In [15] the amount of sensed data is small compared to our scenarios making frugal use of network capacity less important.

3 Zone Dissemination Protocols

As pointed out in the previous section both our protocols rely on the observation that the relevance of information about the road at some location decreases with the distance to that location.

3.1 The Zone Flooding Protocol

The Zone Flooding protocol is a variant of basic flooding with three modifications to limit the dissemination of packets. It can be seen as a special case of flooding-based geocasting [12] in the sense that the source is located inside the geocast zone. Traditionally the problem with flooding-based protocols [23] is that they congest the network with hordes of packets. To alleviate this problem we use several techniques to limit the forwarding of packets.

A *hop-count* is embedded in every packet and decremented when the packet is forwarded. When the hop-count reaches zero the packet is discarded. This has the effect that the packet only reaches nodes in the part of the network that is within a certain hop-count radius from the originator of the packet. It is, however, possible that nodes near the originator forward a packet multiple times.

To avoid that a node forwards a packet more than once, we use *sequence lists* as in [11, 25] to detect whether a packet has been received before. Packets should only be forwarded upon the first reception. Each node maintains a *sequence number* that is incremented every time a new packet is created by the node. The sequence number is embedded in every packet originating from the node. Every node also maintains a sequence list, mapping other nodes to their last known sequence number. When a packet is received the sequence number for the originator is updated. If the sequence number contained in the packet is the same or lower than the sequence number in the sequence list, the packet has been received before and should thus not be forwarded. If the sequence number in the packet is strictly lower than the one in the sequence list, the packet being received must have been overtaken. If, however, the sequence number in the packet is greater than the sequence number in the sequence list it is known that the packet is being received for the first time and therefore should be forwarded. The amount of memory used by this mechanism can be limited by for each entry in the sequence list noting the time at which the last packet was received. When a packet is received multiple times it always happens inside a short period of time. A copy of a packet can in practise only be delayed shortly because when a packet is received by an intermediate node it is either dropped or forwarded immediately. Therefore, the sequence list can be cleaned up periodically by removing the oldest entries.

To further limit the dissemination of packets the concept of a *flooding zone* is introduced. In every packet a flooding zone is embedded specifying a geographic destination area. In the current implementation of the protocol the flooding zone is a rectangle, but other shapes are possible. When a node receives a packet it checks (e.g. using a GPS device) whether its current position is inside the flooding zone and discards the packet if that is not the case. The effect is that packets are only delivered to nodes in a certain geographical area.

Figure 1 illustrates the operation of the Zone Flooding protocol. The white source node broadcasts a packet that is forwarded by other nodes until it reaches a node outside the flooding zone.

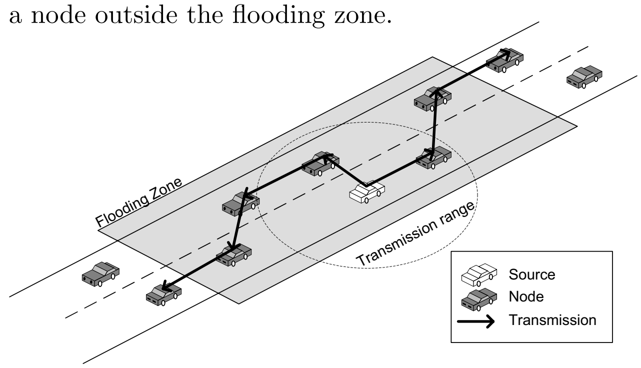

Figure 1. The Zone Flooding Protocol.

When limiting the forwarding of packets by using the flooding zone, the hop-count limitation almost never gets effectuated because packets move out of the flooding zone before they reach the hop-count limit. However, since nodes are constantly moving, it is possible (albeit unlikely) that a packet would be forwarded infinitely inside the zone provided that new nodes keep entering the zone at an appropriate rate. Hop-count is therefore necessary to ensure correctness of the protocol.

Proceedings of the 39th Annual Simulation Symposium (ANSS'06)
0-7695-2559-8/06 $20.00 © 2006 IEEE

To summarise, the techniques described above limits the forwarding of packets in three ways: No packets will reach nodes outside a certain hop-count radius from the source, no packets will be forwarded more than once by each node, and no packets will be forwarded by nodes outside the flooding zone.

The pseudo code for the Zone Flooding protocol can be found in algorithms 3.1 and 3.2 and consists of two parts: BROADCAST_LOOP and RECEIVE. The primitives used are described in appendix A. BROADCAST_LOOP is called when the system is started and RECEIVE is called every time a packet is received.

Algorithm 3.1: BROADCAST_LOOP(*bcastInterval*)

```
while true
  do { classification ← GET_CLAS()
       pos ← GET_POSITION()
       zone ← NEW_ZONE(pos)
       seqNumber ← seqNumber + 1
       packet ← NEW_PACKET(classification, zone,
                           seqNumber)
       BROADCAST(packet)
       SLEEP(bcastInterval)
```

Algorithm 3.2: RECEIVE(*packet*)

```
if packet.hopcount ≤ 0
  then return
senderSeqNumber ← SEQ_LIST(packet.sender)
if senderSeqNumber ≤ packet.seqNumber
  then return
pos ← GET_POSITION()
if IS_OUTSIDE(packet.zone, pos)
  then return
SEQ_LIST(packet.sender) ← packet.seqNumber
DEC_HOPCOUNT(packet)
BROADCAST(packet)
return
```

3.2 The Zone Diffusion Protocol

The Zone Diffusion protocol is based on *data aggregation* which is a commonly used technique in sensor networks [18, 22, 28]. Each node maintains an *environment representation* (ER) representing the surrounding environment. The ER is updated every time data arrives from the sensors. To disseminate data the ER is periodically broadcasted. When an ER is received from another node it is aggregated with the local ER by merging the information in the received ER that intersects with the area covered by the local ER. Contrary to the Zone Flooding protocol, packets are never forwarded. However, data about the local environment is indirectly forwarded to other nodes since nodes periodically broadcasts their ER. The protocol is thus data-centric as opposed to node-centric.

The ER is divided into *cells* of equal size each representing an atomic part of the road. When a node receives information concerning a cell it already has information about, the information is combined according to a data combination policy. One policy could be to make a conservative estimate: if one node thinks the cell is dry and another node classifies it as icy then the cell is to be considered icy. Other policies are also possible. The actual choice of policy is, however, out of the scope of this paper. In the implementation of the protocol we just store in each cell whether the node has received information about that cell at all and so the protocol can accommodate different policies depending on the specific application.

Figure 2 illustrates the operation of the Zone Diffusion protocol. Node A sends it's ER (depicted over it with a white car in it) to node B. Node B aggregates the received ER with its own and thus learns about icy cells up ahead (marked with "ICY" in bold font and capital letters).

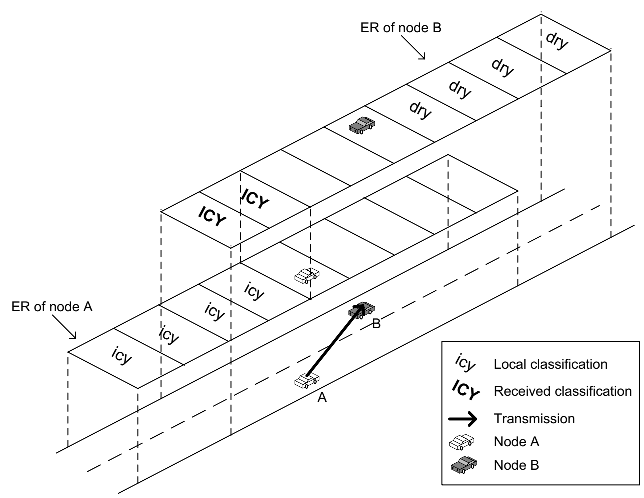

Figure 2. The Zone Diffusion Protocol.

The pseudo code for the Zone Diffusion protocol consists of three parts UPDATE_LOOP, BROADCAST_LOOP, and RECEIVE given in algorithms 3.3, 3.4, and 3.5. UPDATE_LOOP and BROADCAST_LOOP is called when the system is started and RECEIVE is called every time a packet is received. UPDATE_LOOP ensures that the ER is continuously updated with road classifications from the sensor of the vehicle. BROADCAST_LOOP periodically broadcasts the ER and RECEIVE handles incoming ERs from other nodes, compares them with the local one, and combines the intersecting cells.

Algorithm 3.3: UPDATE_LOOP()

while *true*
  do $\begin{cases} \textit{classification} \leftarrow \text{GET\_CLAS}() \\ c \leftarrow \text{GET\_CELL}() \\ ER.at(c) \leftarrow \text{POLICY}(ER.at(c), \textit{classification}) \end{cases}$

Algorithm 3.4: BROADCAST_LOOP(*bcastInterval*)

while *true*
  do $\begin{cases} \text{BROADCAST}(ER) \\ \text{SLEEP}(\textit{bcastInterval}) \end{cases}$

Algorithm 3.5: RECEIVE(*packet*)

$\textit{commonCells} \leftarrow \text{COMMON\_CELLS}(ER, packet.ER)$
for each $c \in \textit{commonCells}$
  do $\begin{cases} ER.at(c) \leftarrow \text{POLICY}(ER.at(c), \\ \qquad packet.ER.at(c)) \end{cases}$
return

4 Simulation Model

To evaluate the protocols simulations have been conducted using the discrete event simulator NS-2 [24]. The setup has been chosen so that it resembles a realistic scenario for the LIWAS system - a straight section of a road, 2000 meters long and 10 meters wide, with vehicles moving in both directions. An overview of the road scenario is shown in figure 3. Ideally the scenario would just consist of two lanes of vehicles entering the road section at the one end, and leaving it at the other. However in NS-2 [24] it is not possible to add or remove nodes once the simulation has started. The ideal situation is obtained by turning the nodes around and resetting them at both ends of the road section thereby making them behave as new nodes.

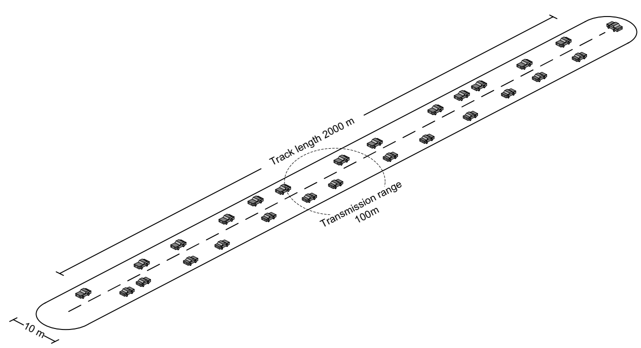

Figure 3. The simulation scenario.

Three classes of mobility, low velocity, medium velocity, and high velocity, each corresponding to a typical traffic situation, were generated using the FreeWay tool [3]. For each class the node velocity changes randomly every 5 seconds according to a Gaussian distribution having the effect that overtaking occurs once in a while. The average node velocity and the velocity variance corresponding to the mobility classes are listed in table 1.

Class	Average velocity	Variance
Low	14m/s(50km/h)	1
Medium	22m/s(80km/h)	1
High	36m/s(130km/h)	1

Table 1. Mobility classes.

Each node is equipped with an IEEE 802.11 radio operating in broadcast mode meaning that there is no channel reservation or acknowledgements. The transmission range is set to 100 meters and the bandwidth is 1 Mbit/s. We use the two-ray-ground radio propagation model that comes with the NS-2 simulator.

Each simulation was run for 200 seconds of simulation time with 100, 200, and 300 nodes. The two protocols were each simulated with broadcast intervals ranging from 0.01 second to 56 seconds.

For the Zone Flooding protocol the size of the flooding zone is set to 2000 by 10 meters, the packet size is set to 64 bytes, and the hop-count is 200. As mentioned in section 3.1, the hop-count only ensures that packets will not travel infinitely inside the flooding zone.

To enable comparison, the environment representation for the Zone Diffusion protocol is set to have the same size as the flooding zone for the Zone Flooding protocol. Each cell in the environment representation is 10 by 10 meters and can hold one of 64 values. The packet size is 224 bytes which is enough to hold information about 200 cells and some additional auxiliary information. Table 2 provides an overview of the simulation parameters.

4.1 Performance Metrics

The protocols have been evaluated both in terms of general protocol performance metrics and in terms of application specific performance metrics.

To estimate the load placed on the network by the protocols we record the amount of packets sent, received, and dropped. These figures can be measured in any network and thus enables comparison with protocols not related to traffic warning systems. It should be noted that for connection-oriented protocols the limitation of dropped packets is typically handled by a MAC layer protocol. This implies that direct comparison with connection-oriented protocols is not appropriate.

General parameters		Zone Flooding parameters	
MAC protocol	IEEE 802.11	Flooding zone size	2000 m × 10 m
Propagation model	Two ray ground	Packet size	64 bytes
Transmission range	100 m	Hop-count	200
Simulation duration	200 secs	Zone Diffusion parameters	
Broadcast interval	[0.01 .. 56] secs	ER size	2000 m × 10 m
Node count	100, 200, 300	Packet size	224 bytes
		Cell size	10 × 10 m

Table 2. Simulation parameters.

Besides the general metrics, the protocols have been evaluated according to traffic warning system specific metrics. One goal of a traffic warning system is to provide information to drivers about the road ahead and the two application specific metrics investigated relate to this goal. First we will intuitively introduce the concepts of *Information Distance* and *Awareness Percentage*, and then we will go into further details about how the figures are determined.

Information Distance When a node learns something about a phenomenon further up the road for the first time, the Information Distance is the distance from the current position to that phenomenon.

Awareness Percentage For a particular location, the Awareness Percentage is the fraction of nodes passing the location that had information about the location before entering it.

To determine the Information Distance and the Awareness Percentage the simulation area is divided into 50 by 50 meters *sectors*. Every sector represents an atomic part of the road; if a sensor somewhere inside the sector classifies the road as being dry, the whole sector counts as being dry. This mimics the fact that if a sensor at some point classifies the road as being dry, there is a high probability that the area around the sensor will be dry as well.

Each time a node receives information about a sector that is has no previous information about, the distance to that sector is the Information Distance. Only information from sectors that the node will eventually enter is counted. In figure 4 information about sector 4 travels from node B to node A - either directly by forwarding of packets in the Zone Flooding protocol or indirectly travelling from one ER to the next in the Zone Diffusion protocol. When node A receives the information, the distance from itself to sector 4 is the Information Distance. It is assumed that the vehicles continuously measures the road implying that if a node knows nothing about a sector immediately before entering it, it will learn something about the sector the instance it enters it, implying that the Information Distance in that situation is 0. The Information Distance is a measure of what warning distance the driver of the vehicle can expect.

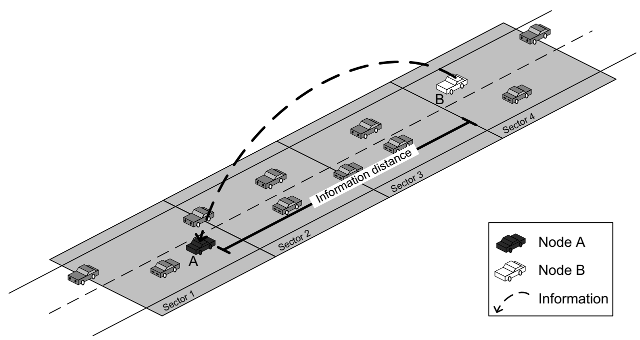

Figure 4. The Information Distance metric.

Let N be the set of nodes and S be the set of sectors. Because the simulation area only consists of a straight section of a road a node n enters a sector s at most once per simulation (recall that nodes are treated as new nodes when they turn at the end of the road). This allows us to define the set of events $E_{\mathbb{S}} \subset N \times S$ so that if node n enters sector s some time during the simulation $\mathbb{S}$ then $(n, s) \in E_{\mathbb{S}}$. The point $p_{\mathbb{S}}(n, s)$ is the position of node n when it first got information about sector s in $\mathbb{S}$ - either receiving it in a packet or measuring it on the road. The function $p_{\mathbb{S}}$ is partially defined on the set $N \times S$. For each event $(n, s) \in E_{\mathbb{S}}$ we define the Information Distance (ID) to be:

$$ID(n, s) = \text{ the shortest distance from } p_{\mathbb{S}}(n, s) \text{ to a point in } s$$

The Awareness Percentage for a sector is the fraction of Information Distances that are above 0 and is therefore only defined on the set of sectors that at some time during the simulation contains a node. This means that if for a sector $s : \{n \mid (n, s) \in E_{\mathbb{S}}\} \neq \emptyset$, then we define the Awareness Percentage (AP) to be:

$$AP(s) = \frac{|\{n \mid ID(n, s) > 0\}|}{|\{n \mid (n, s) \in E_{\mathbb{S}}\}|}$$

5 Performance Evaluation

This section presents the performance evaluation of the protocols. The network load, Awareness Percentage, and Information Distance of the protocols obtained from the simulations are analysed and at the end we sum up conclusions concerning the choice of protocol and parameters.

5.1 Network Load

Figure 5 shows packet statistics for the medium velocity mobility class with 200 nodes. The number of packets sent, received, and dropped is shown as a function of the number of broadcasts per second. For a single packet transmission the number of received packets is the number of nodes in range that receives the packet whereas the number of dropped packets is the number of nodes in range that due to signal interference do not receive the packet. Each number displayed in the graph is a sum over all transmissions in a simulation run. Therefore, as can be seen in figure 5, the number of received packets is larger than the number of sent packets.

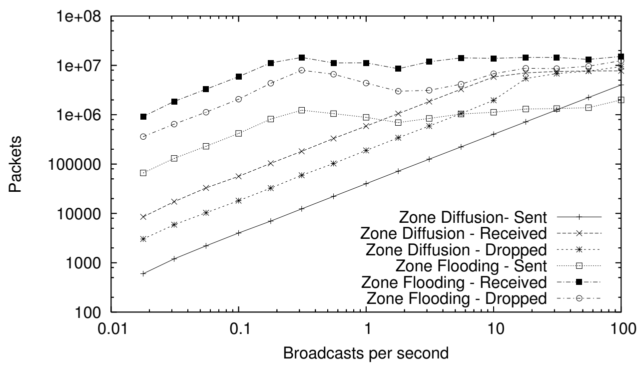

Figure 5. Medium velocity, 200 nodes.

Both axes are scaled logarithmically implying that exponential functions are shown as straight lines. The curves for packets sent, received and dropped for the Zone Flooding protocol in the interval 0.17 to 0.3 broadcasts per second are straight lines and therefore exponential functions. The base of the exponential functions is one and therefore there is a linear relation between the number of broadcasts per second and the number of packets sent, received, and dropped. An explanation of this is that when the number of broadcasts per second is low the probability that separate floodings will interfere is low. If a packet originating from a particular flooding is dropped, it is most likely because it collides with another packet from the same flooding. Therefore, when the number of broadcasts per second increases with a constant, the number of packets sent, received, and dropped increases proportionally.

When the number of broadcasts per second exceeds 0.3 the Zone Flooding protocol changes behaviour. Instead of being linear, the number of packets sent, received, and dropped are approximately constant. This indicates that separate floodings start to interfere. Even though more floodings are initiated the amount of packets sent, received, and dropped remains almost constant. The interference between separate floodings causes packets to be dropped and since a node has to receive a packet before it can forward it, the number of nodes that sends packets originating from a particular flooding decreases. When fewer packets are sent, fewer packets get dropped. As mentioned, the figures remain approximately constant and that means that the average number of receivers per flooding decreases approximately as fast the number of broadcasts per second increases. Hereby, the protocol continues to have reasonable performance in spite of increased network load and thus the effect resembles a crude form of congestion containment.

The Zone Diffusion protocol has no forwarding of packets and the number of sent packets therefore only depends on the number of broadcasts per second. At more than 10 broadcasts per second an increased number of dropped packets can be observed indicating that the network has reached its maximum capacity. The number of received packets decreases accordingly as a consequence.

With less than 0.3 broadcasts per second the number of packets sent, received, and dropped for the Zone Flooding protocol are about a factor 100 larger than the corresponding numbers for the Zone Diffusion protocol.

The packet statistics for the other mobility classes are similar although as the node count increases the number of broadcasts per second for which the protocols change behaviour decreases.

To summarise, the network load of the Zone Flooding protocol is linear in the number of broadcasts per second until a certain threshold after which it is constant. The network load of the Zone Diffusion protocol is linear for all broadcasts per second and is a factor 100 less than the network load of the Zone Flooding protocol when the number of broadcasts per second is low.

5.2 Awareness Percentage

As described in section 4.1, the Awareness Percentage is associated with a sector, and the sector for which the Awareness Percentage is considered in the rest of the paper is the one in the centre of the road section. This sector is chosen because flooding zones and ERs centred at this sector are fully contained in the simulation area and thus the boundaries of the scenario have no effect on the Awareness Percentage of this sector.

Figure 6 shows the Awareness Percentage for the two protocols as a function of the number of broadcasts per second in the medium velocity mobility class with 200 nodes.

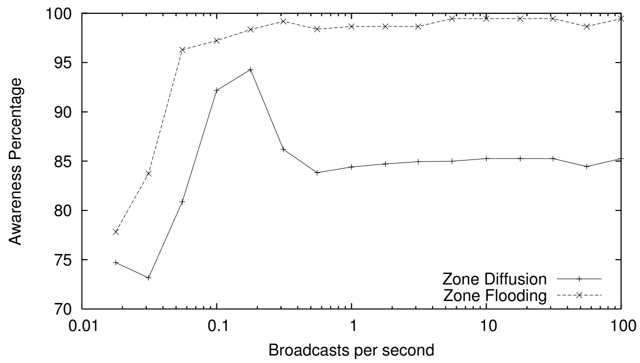

Figure 6. Medium velocity, 200 nodes.

When comparing the protocols' Awareness Percentage for a given number of broadcasts per second the Zone Flooding protocol outperforms the Zone Diffusion protocol in most cases. As mentioned in the previous section the Zone Flooding protocol sends about a factor 100 more packets than the Zone Diffusion protocol when the network is not congested and therefore much more information can be exchanged. In figure 7 it can be seen that the difference between the two protocols is not always as significant as in figure 6.

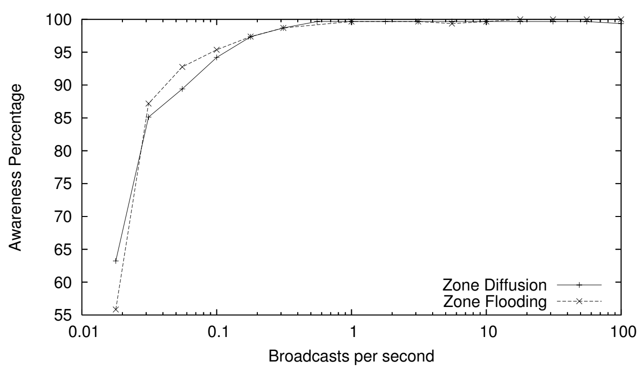

Figure 7. Low velocity, 300 nodes.

The Awareness Percentage that the protocols are able to achieve remain fairly constant across most velocity classes and node densities, indicating that the protocols are indeed robust to varying network density and mobility.

To investigate the advantage the Zone Flooding protocol has by using more network capacity than the Zone Diffusion protocol, we investigate how the protocols perform in terms of Awareness Percentage versus number of packets sent (instead of number of broadcasts per second). As seen in figure 8 the Zone Diffusion protocol achieves good performance using significantly fewer packets than the Zone Flooding protocol implying that there is a trade-off between high Awareness Percentage and low network utilisation. Where high Awareness Percentage is the primary goal, the Zone Flooding protocol should be used and when low network utilisation is the goal the Zone Diffusion protocol is to be preferred.

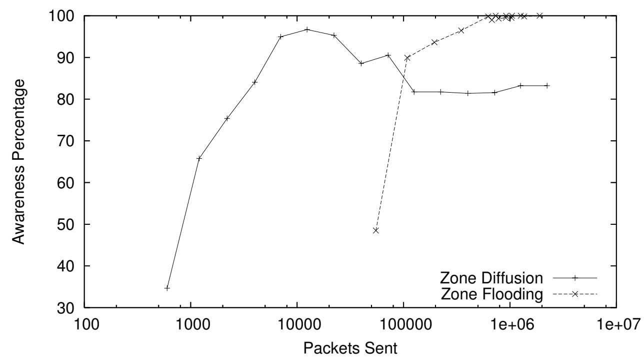

Figure 8. High velocity, 200 nodes.

There are a few exceptions to this pattern as can be seen in figure 9. In some scenarios the Zone Diffusion protocol performs better for *any* number of packets sent and thus no trade-off is present. The circumstances for which there is a trade-off will be further discussed in section 5.4.

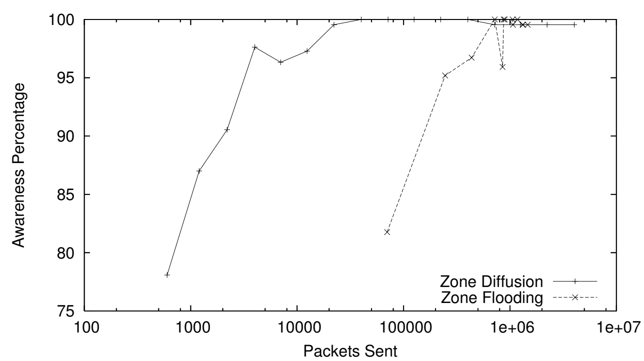

Figure 9. Low velocity, 200 nodes.

In summary the Zone Flooding protocol uses more network capacity than the Zone Diffusion protocol and thereby achieves better Awareness Percentage in most cases. Zone Diffusion provides reasonable performance using less network capacity and therefore there is a trade-off between Awareness Percentage and network utilisation.

5.3 Information Distance

As was the case with the Awareness Percentage, the Information Distance is measured at the sector in the centre of the road section. The Information Distance for the centre sector is specified as an average over all Information Distances for that sector together with a confidence interval of 95%. Assuming that the Information Distances for the sector is distributed according to the Gaussian distribution, the confidence interval is the range in which the actual average (as opposed to the average of the point samples) is located with a probability of 95%.

In figure 10 the average Information Distance is shown as a function of the number of broadcasts per second for the low velocity mobility class with 200 nodes. When comparing the average Information Distance obtained for the two protocols at corresponding broadcasts per second two issues are evident. Firstly, with less than one broadcasts per second the Zone Flooding protocol always achieves the largest average information distance. As was the case with Awareness Percentage this can be explained by the difference in sent packets. Secondly, when the number of

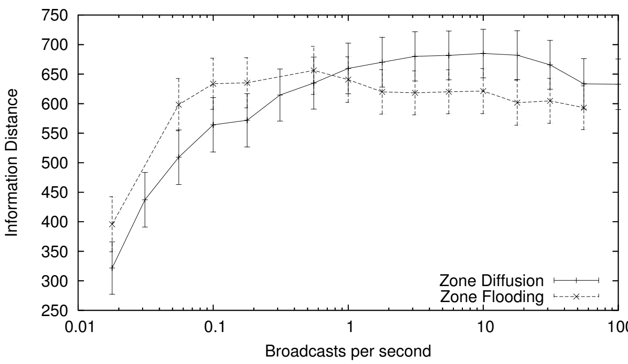

Figure 10. Low velocity, 200 nodes.

broadcasts per second is greater than one, the average Information Distance for the Zone Flooding protocol decreases and the Zone Diffusion protocol becomes superior. In section 5.1 we saw that the number of nodes that receive a particular flooding decreases as the network gets congested. Combined with the fact that the average Information Distance decreases as the network gets congested, we conclude that the nodes that receives a particular flooding in a congested network are the nodes located closest to the origin of the flooding. In other words, when the network gets congested the area of dissemination decreases in size.

Figure 11 and 12 shows the average Information Distance as a function of the number of packets sent. When comparing the two protocols, the behaviour from section 5.2 is repeated. In some cases there is a trade-off between a large information distance and low network utilisation while in most cases the Zone Diffusion protocol outperforms the Zone Flooding protocol.

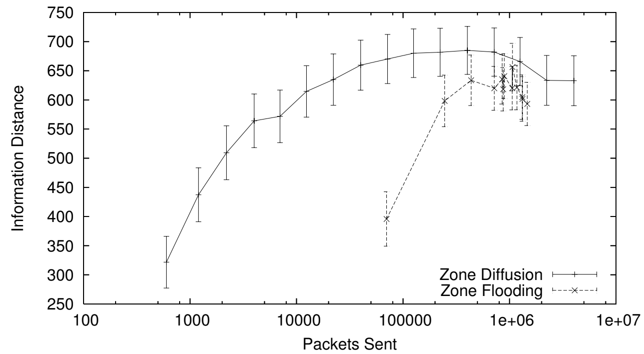

Figure 11. Low velocity, 200 nodes.

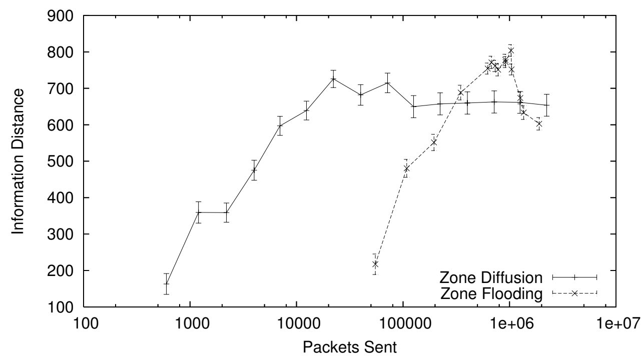

Figure 12. High velocity, 200 nodes.

The Information Distance the protocols achieves remains fairly constant across varying mobility classes and node densities, as was the case with the Awareness Percentage,

5.4 Zone Flooding versus Zone Diffusion

In the previous sections we saw that in some cases there is a trade-off between Awareness Percentage (and Information Distance) and network utilisation. The Zone Flooding protocol achieves better performance than the Zone Diffusion protocol by using more network capacity. This section further explores in which scenarios the trade-off is present and what effect is has on the choice of protocol and parameters. The following analysis considers high Awareness Percentage and low network utilisation as goals. The analysis could easily be extended with the average Information Distance as a goal parameter, but that has been left out due to space limitations.

For a given scenario the problem of choosing a protocol and the number of broadcasts per second such that the Awareness Percentage is maximised and the number of sent packets is minimised can be categorised as a multi-objective optimisation problem. Each candidate solution consists of a pair (protocol, broadcasts per second) with an affiliated Awareness Percentage and a number of sent packets. To identify optimal solutions we use the concept of *Pareto optimality* [7].

A solution is Pareto optimal if all other solutions that are better according to one goal parameter is worse according to the other. Pareto optimality can be defined as follows. Let s be a solution in the solution space $S = \{ZF, ZD\} \times \{0.017, ..., 100\}$[1] and $PS(s)$ be the number of packets sent for s and $AP(s)$ be the Awareness Percentage for s. Then a solution $s \in S$ is Pareto optimal if $\forall t \in S$:

$$PS(t) < PS(s) \Rightarrow AP(t) < AP(s) \\ \wedge \\ AP(t) > AP(s) \Rightarrow PS(t) > PS(s)$$

For each scenario (node count, traffic speed) the set of solutions have been plotted and the Pareto optimal solutions have been connected to form a Pareto curve. Figure 13 shows the Pareto curve for the high velocity mobility class with 200 nodes.

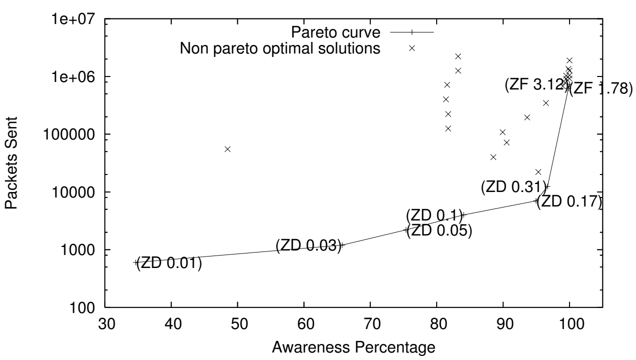

Figure 13. Pareto optimal solutions. High velocity, 200 nodes.

For each solution on the Pareto curve the corresponding protocol (ZD for Zone Diffusion and ZF for Zone Flooding) and the number of broadcasts per second have been noted. As an example it can be seen in figure 13 that the Zone Flooding protocol with 0.03 broadcasts per second (ZD, 0.03) is a Pareto optimal solution for the high velocity mobility class with 200 nodes. Since AP(ZD, 0.03) = 65.8 and PS(ZD, 0.03) = 1200 we know that any solution that has more than 65.8 Awareness Percentage sends more than 1200 packets and any solution that sends less than 1200 packet has less than 65.8 Awareness Percentage.

[1] Note that these values specifies the number of broadcasts per second while the values in table 2 specifies the broadcast interval (seconds/broadcast).

For all mobility classes and nodes densities, it is the case that the Pareto curves are monotone in the sense that the first half of the solutions (the one with the lowest Awareness Percentage) includes the Zone Diffusion protocol and the other half includes the Zone Flooding protocol. This means that if the Awareness Percentage is the primary optimisation factor, then the Zone Flooding protocol should be used whereas if low network load is the prime consideration the Zone Diffusion protocol should be used. However, it is usually not the case that one of the factors is the primary. Usually, there are certain demands for Awareness and limitations on network utilisation. To analyse this question in detail we have determined which is the lowest Awareness requirement for which the Zone Flooding protocol would be best, and similarly, which is the highest packets sent allowance for which the Zone Diffusion protocol should be used. These figures have been derived from the Pareto curves and are listed in tables 3 and 4. The node density is specified as the average number of nodes per square meter in the scenarios. Since we are not aware of theoretical results [10] that allow us to characterise the broadcast capacity of a mobile ad-hoc network, we use packets sent per square meter per second as a measure of network load in table 4.

Node density/Mobility	Low	Medium	High
0.005	None	97.4	94.6
0.010	None	94.3	96.7
0.015	99.7	94.3	95.1

Table 3. Lowest Awareness Percentage requirement to choose Zone Flooding.

Node density/Mobility	Low	Medium	High
0.005	All	0.050	0.021
0.010	All	0.057	0.156
0.015	0.403	0.130	0.167

Table 4. Maximum packets sent allowance requiring Zone Diffusion to be chosen.

As an example we see in table 3 that for the medium velocity mobility class with a node density of 0.005 $nodes/m^2$ the lowest Awareness Percentage requirement that would require us to choose the Zone Flooding protocol is 97.4 - if we can settle with anything less the Zone Diffusion protocol should be used. Similarly, we see in table 4 that if we in the medium velocity mobility class with 0.005 $nodes/m^2$ can settle with anything worse than 0.050 $packets/m^2/sec$ sent the Zone Flooding protocol should be used.

In a few cases (the slow scenarios with 0.005 and 0.010 $nodes/m^2$) all the Pareto optimal solutions in-

0-7695-2559-8/06 $20.00 © 2006 IEEE

clude the Zone Diffusion protocol. In these cases the Zone Flooding protocol should *never* be used. The corresponding entries in the tables are marked with *None* and *All*. Table 3 indicates that the Awareness requirement weakens (excluding the (fast, 0.005) scenario) as the number of nodes increases. When turning to the values in table 4 we again see that the packet requirement weakens as node count increases.

A general conclusion is that the Zone Flooding protocol achieves the best Awareness Percentage in all but a few cases, and the Zone Diffusion protocol sends fewest packets. Furthermore the two goal functions - Awareness Percentage and the number of packets sent - are inversely connected; optimising one lowers the other and vice versa.

However, the Pareto graphs and the associated tables above allows us to conclude that if we can settle with an Awareness Percentage of 94.3 in worst case the Zone Diffusion protocol should *always* - no matter what scenario - be used. A similar conclusion cannot be drawn regarding the number of packets sent since, as we saw before, sometimes the Zone Flooding protocol should never be used. If we however, limit ourselves to only consider the medium and high velocity scenarios we see that if we can accept that our protocol uses 0.167 $packets/m^2/sec$ in worst case then the Zone Flooding protocol should always be used.

6 Conclusions and Future Work

In this paper we presented two light-weight protocols for data dissemination in vehicular ad-hoc networks. The protocols only rely on the assumption that the relevance of information about a particular phenomenon decreases with the distance to that phenomenon and can therefore be used in typical vehicular ad-hoc network applications.

To be able to evaluate the performance of the protocols we defined general metrics that measure the load placed on the network by the protocols and domain specific metrics that characterise how the data is disseminated. The domain specific metrics have general applicability in the area of data dissemination protocols for vehicular ad-hoc networks.

In the evaluation of the protocols, we concluded that the protocols are robust to changes in network density and mobility. The Zone Flooding protocol generally achieves better Awareness Percentage and Information Distance than the Zone Diffusion protocol, but the Zone Diffusion protocol achieves reasonable performance at a much lower network utilisation. In most mobility classes and node densities, there is a trade-off between Awareness Percentage and network utilisation. An analysis of the trade-off revealed that in most applications (where an Awareness Percentage of 94.3 is acceptable) the Zone Diffusion protocols should be used.

By a crude form of congestion containment, the Zone Flooding protocol adaptively decreases the size of the area in which data is disseminated when the networks gets congested. The Zone Diffusion protocol could be improved by adding congestion control - either by limiting the dissemination area or by decreasing the number of broadcasts per second.

As mentioned earlier, the relevance of information decreases with the distance to the source. Some data dissemination protocols [20, 22] reflect this by letting the information resolution decrease with the distance from the originating node. The Zone Diffusion protocol could be extended with such a mechanism.

Based on the evaluation results presented in this paper, the Zone Diffusion protocol will be implemented as part of the general communication infrastructure for the LIWAS system[6]. Once implemented, the conclusions of this paper can be validated by real world experiments.

References

[1] *Proceedings of The First ACM International Workshop on Vehicular Networks*, 2004.

[2] *Proceedings of The Second ACM International Workshop on Vehicular Networks*, 2005.

[3] F. Bai, N. Sadagopan, and A. Helmy. IMPORTANT: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In *Proceedings of IEEE INFOCOM*, pages 825–835, 2003.

[4] R. Bogenberger, W. Kellerer, T. Kosch, T. Reicher, C. Schwingenschlögl, P. Sties, and M. Wagner. Virtual city portal - a multi-network personal information system for automobile users. In *IEEE/ITG International Workshop on Multiradio Multimedia Communications*, 2003.

[5] L. Briesemeister and G. Hommel. Integrating simple yet robust protocol layers for wireless ad hoc intervehicle communications. In *Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS)*, pages 186–192, 2002.

[6] J. Brønsted, K. M. Hansen, and L. M. Kristensen. An infrastructure for a traffic warning system. In *Proceedings for the IEEE International Conference on Pervasive Services*, pages 136–145, 2005.

[7] Y. Censor. Pareto optimality in multiobjective problems. *Applied Mathematics and Optimization*, pages 41–59, Mar 1977.

[8] A. Festag, H. Fussler, H. Hartenstein, A. Sarma, and R. Schmitz. Fleetnet:bringing car-to-car communica-

tion into the real world. In *Proceedings of the 11th World Congress on ITS*, 2004.

[9] H. Füssler, M. Mauve, H. Hartenstein, D. Vollmer, and M. Käsemann. A comparison of routing strategies for vehicular ad-hoc networks. In *Proceedings of MOBICOM*, 2002. Student Poster.

[10] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions of Information Theory*, 46:388–404, 2000.

[11] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353, chapter 5, pages 153–181. Kluwer Academic Publishers, 1996.

[12] Y.-B. Ko and N. H. Vaidya. Flooding-based Geocasting Protocols for Mobile Ad Hoc Networks. *Mobile Networks and Applications*, 7:471–480, 2002.

[13] T. Kosch, C. Schwingenschlögl, and L. Ai. Information dissemination in multihop inter-vehicle networks. In *Proceedings of the International Conference on Intelligent Transportation Systems*, pages 685–690, 2002.

[14] L. M. Kristensen and K.-D. Nielsen. On the application of zone flooding in a traffic warning system. Technical report, Department of Computer Science, Univerity of Aarhus, 2004.

[15] H. R. Lars Wischhof, André Ebner. Sotis - a self-organizing traffic information system. In *Proceedings of the IEEE Vehicular Technology Conference Spring*, pages 2442–2246, 2003.

[16] A. Leick. *GPS Satellite Surveying*. John Wiley and Sons, 2004.

[17] LIWAS: Life Warning Systems. `http://www.liwas.dk`.

[18] S. Madden, M. J. Franklin, J. Hellerstein, and W. Hong. Tag: a tiny aggregation service for ad-hoc sensor networks. In *Proceedings of the Fifth Symposium on Operating Systems Design and implementation*, pages 131–146, 2002.

[19] M. Mauve, J. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad-hoc networks. IEEE Network Magazine, 2001.

[20] L. B. Michael. Adaptive layered data structure for inter-vehicle communication in ad-hoc communication networks. In *Proceedings of the 8th World Congress on Intelligent Transportation Systems*, 2001.

[21] R. Morris, J. Janotti, F. Kaashoek, J. Li, and D. Decouto. Carnet: A scalable ad hoc wireless network system. In *Proceedings og the 9th ACM SIGOPS*, pages 61–65, 2000.

[22] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode. Trafficview: A scalable traffic monitoring system. In *Proceedings of the 2004 IEEE International Conference on Mobile Data Management*, pages 13–26, January 2004.

[23] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu. The Broadcast Storm Problem in a Mobile Ad Hoc Network. In *Proceedings of ACM/IEEE MobiCom'99*, pages 151–162, 1999.

[24] The network simulator. `www.isi.edu/nsnam/ns/`.

[25] C. Perkins, e. Belding-Royer, and d. Das. Ad hoc on-demand distance vector (aodv) routing. `http://www.ietf.org/rfc/rfc3561.txt`, 2003. rfc-3561.

[26] 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*, pages 194–205, 2002.

[27] H. Wu and M. H. Richard Fujimoto, Randall Gunsler. Mddv: A mobility-centric data dissemination algorithm for vehicular networks. In *Proceedings of The First ACM International Workshop in Vehicular Networks*, pages 47–56, 2004.

[28] M. Yarvis, W. Conner, L. Krishnamurthy, J. Chhabra, B. Elliott, and A. Mainwaring. Real-world experiences with an interactive ad hoc sensor network. In *Proceedings of the International Workshop on Ad Hoc Networking*, pages 143–152, 2002.

A Algorithm primitives

Primitive	Description
BROADCAST	Broadcasts the argument
COMMON_CELLS	Returns a collection of cells that exists in both environment representations
DEC_HOPCOUNT	Given a packet it decrements the hop-count of the packet
GET_CELL	Returns the current cell
GET_CLAS	Returns a classification of the road
GET_POSITION	Returns the current position
IS_OUTSIDE	Returns true if the given position is outside the given zone
NEW_PACKET	Constructs a new packet containing the arguments.
NEW_ZONE	Given a position it returns a zone having the position at its centre
POLICY	The given classifications are combined and return according to the combination policy
SEQ_LIST	Given a node identity, it returns the stored sequence number for that node. If no sequence number is stored, infinity is returned
SLEEP	Waits for a given interval
packet	A data structure for a Zone Flooding packet containing fields: *hopcount* (an integer) *seqNumber* (an integer) *classification* (an integer) *zone* (a zone represented by two corners)
ER	A data structure for an ER. Maps cells to classifications (*ER.at(pos)*)