

# A Power-Aware Broadcasting Algorithm

Hugo Miranda      Simone Leggio  
Luís Rodrigues      Kimmo Raatikainen

DI-FCUL

TR-06-5

April 2006

Departamento de Informática  
Faculdade de Ciências da Universidade de Lisboa  
Campo Grande, 1749-016 Lisboa  
Portugal

Technical reports are available at <http://www.di.fc.ul.pt/tech-reports>. The files are stored in PDF, with the report number as filename. Alternatively, reports are available by post from the above address.



# A Power-Aware Broadcasting Algorithm\*

Hugo Miranda

Luís Rodrigues

Universidade de Lisboa

Departamento de Informática

Portugal

{hmiranda,ler}@di.fc.ul.pt

Simone Leggio

Kimmo Raatikainen

University of Helsinki

Department of Computer Science

Finland

{simone.leggio,kimmo.raatikainen}@cs.helsinki.fi

April 2006

## Abstract

Flooding is an expensive but unavoidable operation in some application scenarios devised for Mobile Ad Hoc Networks (MANETs). In this paper, we present a novel algorithm to reduce the overhead imposed by flooding operations. The algorithm improves previous results by using a distributed function to elect the nodes that will provide the highest additional coverage to previous retransmissions. The algorithm does not require any signalling or imposes special requirements on the participating devices.

## 1 Introduction

The decentralised nature of Mobile Ad hoc NETWORKS (MANETs) implicitly suggests that many services, like the location of some resource, can only be implemented using a mechanism that, in the worst case, requires the delivery of a message to every participant. A typical example is route discovery. It is not uncommon to find routing protocols that, to learn a route between two endpoints, flood the network with a route discovery message. Typical examples of protocols presenting this behaviour are the Dynamic Source Routing (DSR) [6] and the Ad hoc On-demand Distance Vector (AODV) [9] protocols. However, the use of flooding is not an exclusive of routing protocols. Reputation systems [3] and distributed caching [8] are examples of other services that may also require the paging of an unknown number of nodes.

In some cases, the underlying infrastructure may provide the tools to efficiently broadcast messages to every participant. This is the case, for example, of spanning trees provided by multicast routing protocols for ad hoc networks. This paper addresses the case where such support is not available. In this case,

---

\*This work was partially supported by the Middleware for Network Eccentric and Mobile Applications (MiNEMA) programme of the European Science Foundation and by FCT project Probabilistically-Structured Overlay Networks (P-SON), POSC/EIA/60941/2004 through FCT and FEDER.

the most common implementation of broadcast is to flood the network. In flooding, each node keeps a record of the messages recently received. When a node receives a message for the first time, the node adds some unique identification of the message to its record and retransmits it. This simple algorithm creates a large number of redundant messages because many of the nodes listening to the retransmission have also listened a previous retransmission or will receive different copies, each transmitted by a different node.

Flooding is an expensive and undesirable operation. Independently of the added value of each particular retransmission, it consumes resources at the sender. Furthermore, receivers also spend a non-negligible amount of energy at the reception [4] and CPU time to decide if the message should be discarded or retransmitted. Finally, most of the Medium Access Control (MAC) protocols for ad hoc networks lack a coordination function able to completely avoid collisions of concurrent transmissions. The IEEE 802.11 Distributed Coordination Function (DCF) for example, does not require any coordination between the participants in a MANET for sending broadcast messages [2].

This paper proposes a novel algorithm to reduce the waste of resources of the nodes and of the bandwidth required by flooding in MANETs. This is a challenging problem because we want to minimise the signalling overhead and we assume that no hardware support is available (e.g. nodes do not use a GPS receiver to become aware of their location). In particular, our algorithm only expects that nodes are able to retrieve the power level with which each message is received. This is a realistic assumption as most of the drivers for wireless network cards are already capable of providing this information.

Reducing the number of transmissions associated with a flood is not a new problem. It has been exposed for example in [12] and we can find applications of some mechanisms that address it in domains like ad hoc routing protocols [5] and on the distribution of code updates for sensors in [7].

The paper is organised as follows. Section 2 explains the rationale beyond the proposals presented in the related work. Section 3 describes our algorithm and highlights its contributions to the state of the art. The results of extensive simulations comparing different algorithms are presented in Section 4. Conclusions are presented in Section 5

## 2 Decreasing the Number of Senders in a Flood

In a MANET, the number of nodes required to retransmit a message so that it gets delivered to every participant depends on several factors like the transmission range of the devices, the location of the source, the size of the region covered by the nodes or the geographical distribution of the nodes. Furthermore, these factors may be permanently changing, even between consecutive messages. Algorithms that aim at reducing the number of retransmissions required by a broadcast try to devise one minimal subset of participants that are required to retransmit so that every node in the network receives at least one copy. Due to the movement of the nodes and to the different location of the sources, it is assumed that the subset changes for every message. Furthermore, the algorithms must refrain from using an excessive amount of control information, which could result in an amplification of the problem for example by increasing the bandwidth consumption. Approaches satisfying these constraints consisted

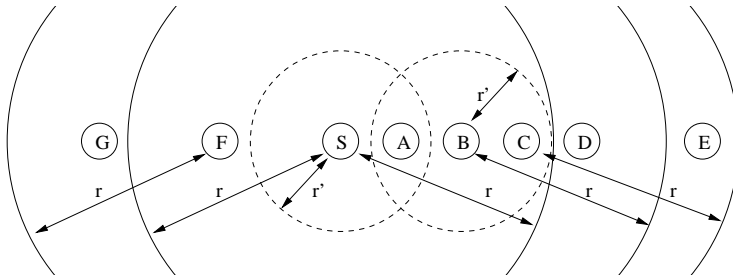


Figure 1: Disposition and transmission range of some nodes

in having each node to forward the first received copy of every message with some probability  $p < 1$  [5, 11, 12]. Although efficiently reducing the number of messages, these algorithms do not take advantage of additional information that can be extracted locally at each node and which can further reduce the number of redundant forwardings. For example, probabilistic approaches do not scale well with the density of the network: independently of the number of nodes within transmission range of each other, approximately  $p\%$  of the nodes retransmit.

Figure 1 represents a region of a MANET with a source  $S$  of a broadcast message and its neighbours. The figure assumes that the transmission range of all nodes is  $r$ . The transmission range of nodes  $S, B, C$  and  $F$  is represented by circles. As it can be seen, the region covered by a retransmission of a broadcast initiated by  $S$  grows with the distance of the retransmitting node to  $S$ . In our example, the most efficient retransmission would be performed by node  $C$ . Elsewhere [12], it was shown that if all nodes transmit with equal power, a retransmission can increase between 0% and 61% the space covered by the previous transmission, depending on the location of the retransmitting node. In the example of Figure 1, the minimal number of transmissions required by a message broadcasted by node  $S$  is three: the message is delivered to nodes  $A, B, C$  and  $F$  by the first transmission and nodes  $C$  and  $F$  could retransmit to deliver it respectively to  $D$  and  $E$ , and  $G$ . We emphasise that, in runtime, nodes do not have access to the information required for following the same rationale presented above.

In the “counter-based scheme” described in [12], each node receiving a message for the first time waits for a random time  $t$  before retransmitting. During the wait period, the node counts the number of retransmissions it listens. If the number of retransmissions exceed a predefined threshold, the message is not retransmitted by the node. The scheme effectively bounds the number of retransmissions of a message in the neighbourhood of every source. However, the random selection of the nodes that will retransmit fails to guarantee the additional coverage of the retransmissions. As an example, consider the case of Figure 1. If the threshold is globally set to two and nodes  $B$  and  $F$  are those with a smaller random delay, four retransmissions will be required for reaching all nodes presented in the figure.

The “counter-based scheme” has similarities with “GOSSIP3( $p, k, m$ )” [5]. In this scheme, nodes retransmit immediately if the source of the message is less than  $k$  hops away or with some probability  $p$  otherwise. The parameter  $m$

dictates the minimal number of retransmissions that a node which decided not to retransmit should listen in a short period of time after receiving the message for the first time. If less than  $m$  messages are listened by the node, it will retransmit the message. Like in other schemes presented in [5], the reduction of the number of messages is mostly achieved by reducing the number of forwarding nodes using random number generators.

Preventing nodes too close to a previous sender (and therefore providing a negligible contribution to the coverage) from retransmitting is addressed in the “distance-based scheme” [12]. Here, nodes will fail to retransmit if the power with which the original message or some retransmission is received is above some threshold. As in the “counter-based scheme”, the wait period is randomly defined. Using some of the propagation models that have been extensively studied in the literature (see for example [10]), it is possible to establish a relation between the reception power and the distance to the source. Therefore, it can also be said that this model prevents nodes within a minimal distance from a previous transmitting node from forwarding the message. In Figure 1, this distance is presented as  $r'$  and the radius illustrated using dashed circles for nodes  $S$  and  $B$ . Again, the randomness of the delay can also lead to non-optimal solutions. This would be the case if the timer of node  $B$  expires before the one for node  $C$ . In this case, node  $D$  would be required to retransmit the message, adding one message to the optimal solution.

### 3 Pampa

In this section, we describe a novel algorithm for increasing the efficiency of message flooding in MANETs. The algorithm, named Pampa (Power-Aware Message Propagation Algorithm) improves the efficiency of previous proposals by removing some of the randomness associated with the decision on the nodes that will retransmit a message.

As mentioned before, the increment in the coverage provided by a retransmitting node may vary between 0% and 61% depending of its location with respect to the previous transmission. The “distance-based scheme” defined a threshold to prevent nodes providing a small coverage from retransmitting. Pampa also uses the power level of the reception to estimate the distance to the source. The novelty of our algorithm is that nodes are sorted by distance with nodes more distant to the source retransmitting first. Like in the “counter-based scheme”, only a limited number of nodes will retransmit. The algorithm is defined as follows:

- S1.** Initialise a counter  $c = 1$  when a broadcast message  $msg$  is heard for the first time. Define  $hold_{msg} = delay(p_{msg})$ , where  $p_{msg}$  is the reception power of the transmission.
- S2.** Wait  $hold_{msg}$  time slots. Then submit  $msg$  for transmission and wait until the transmission actually starts. While in the hold period, interrupt the waiting and perform S4 if  $msg$  is listened again.
- S3.** The message is on the air. The procedure exits.
- S4.** Increment  $c$ . If  $c < C$ , resume the interrupted waiting in S2. Otherwise, cancel the transmission of  $msg$ .

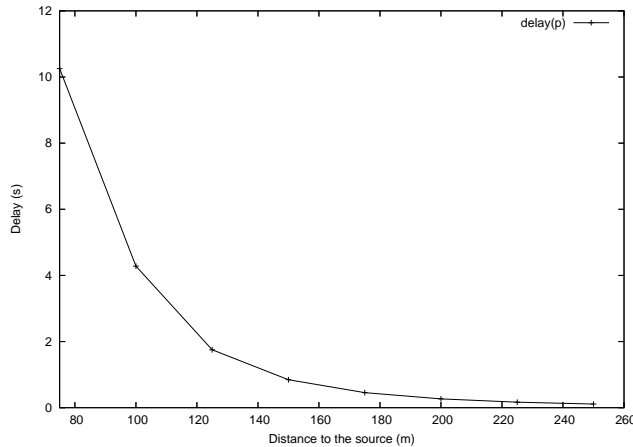


Figure 2: Function *delay*

The algorithm uses a function *delay* which gets the reception power of a transmission and returns a delay. This function is expected to map an increasing distance to the source (corresponding to a smaller reception power of the message) in a smaller return value. We estimate that a *delay* function that varies linearly with the distance to the source would provide the best results.

Contrary to “distance-based” and “counter-based” schemas, in Pampa, the moment of each node to forward a message is locally determined from its distance to the sender. In the absence of abnormal effects on the signal propagation, our model guarantees that the limited number of nodes performing a retransmission will be those that provide the higher possible additional coverage. In the example presented in Figure 1, node *C* would be the first to retransmit, delivering the message to both nodes *D* and *E*. Although slightly later, node *F* would also be required to retransmit and therefore, guarantee the coverage of node *G*. Pampa reduces the probability of requiring four instead of the minimal three transmissions for delivering the message to every node in the figure.

**Delay Assignment** The behaviour of the reception power with the distance varies with the environment where the network is being used. Therefore, it is not possible to accurately estimate the distance to the source. Furthermore, even simple propagation models require complex computations unsuitable to be applied on every message by the mobile devices. In our tests, we defined a simpler delay function that multiplies the reception power by a constant  $k$  to return the number of seconds that the node should wait before retransmitting. Analysis of an adequate value for  $k$  was performed using the Two Ray Ground propagation model as defined in the *ns-2* network simulator version 2.28. We have found  $300 \times 10^6$  to be an adequate value for obtaining distinct wait times for nodes close to each other. The behaviour of the function  $delay(p) = 300 \times 10^6 \times p$  for the Two Ray Ground propagation model is presented in Figure 2.

As expected, the function follows the logarithmic decay of the reception power of a message. For short distances, the function returns excessively large delay values. However, nodes at these distances from the source have a large probability of not needing to retransmit. The buffers occupied by messages

holding for large periods may be freed as soon as the threshold number of replicas listened is reached. Section 4 will show that these excessive delays do not significantly influence the propagation of the message.

## 4 Evaluation

We have implemented the “count-based”, “distance-based” and Pampa algorithms in the *ns-2* network simulator v. 2.28. Each implementation was tested with different parameters. For the “count-based” and Pampa, we tested different thresholds for the number of messages after which a retransmission is discarded. The “distance-based” scheme was tested with different threshold distances. Each schema had one parameter immutable for all simulations. The maximum random delay used by “count-based” and “distance-based” schema was set to 0.75s. The constant in the *delay* function used in Pampa was set to  $300 \times 10^6$ .

Different node densities have been experimented by keeping the number of nodes and the transmission range stable while changing the size of the simulated space. Nodes were configured to emulate a 914MHz Lucent Waveland DSSS radio interface running an IEEE802.11 protocol at 2Mb/s. Network cards present a transmission range of 250m using the Two Ray Ground propagation model. Six simulated regions were tested: 250mx250m, 500mx250m, 1000mx500m, 1000mx1000m, 2000mx1000m and 2000mx2000m. In practice, these provided node densities between  $625m^2/node$  and  $40000m^2/node$ .

At the beginning of the simulation, nodes are uniformly deployed over the simulated region. In one set of tests, nodes do not move for the entire duration of the simulation. These tests have been named “Speed 0”. In the remaining sets nodes move, with no pause time, accordingly to the random waypoint movement model [6]. Random speeds were defined between 4m/s and 6m/s for one set and 9m/s and 11m/s for the other. These sets have been named respectively “Speed 5” and “Speed 10”.

For each simulated region and movement speed, 100 movement files have been created. On all tests, 1000 flood messages are generated at a pace of one message per second. The size of each message is of 1000 bytes. The source of each message is randomly selected. Each point in the figures presented below averages the result of the 100 independent runs. Each test was performed for the three algorithms using the same movement and traffic files.

The delivery ratio of Pampa is compared with the “count-based” scheme for different thresholds in Figure 3. The figure shows that for high node densities, all variations of both schemas are capable of delivering every message to all nodes. As the node density decreases, the differences between both schemas become visible, particularly in the cases where the message threshold is lower. For a threshold of one message, the delivery ratio of the “count-based” schema begins to decay at a much faster pace than Pampa. We attribute this behaviour to the randomness associated with the node selection in “count-based”. In Pampa, the nodes forwarding the message have an higher probability of reaching more distant locations. When the simulated space is of 2000mx2000m ( $40000m^2/node$ ) there are network partitions. The benefits of using Pampa can be more clearly observed in these extreme conditions: for the same thresholds, Pampa always presents an higher delivery ratio.



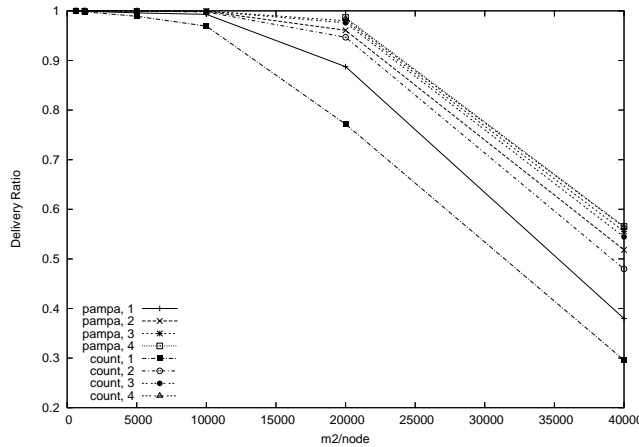


Figure 3: Delivery ratio (Speed 0)

In an attempt to improve the delivery ratio of the larger simulated regions, we have also run similar tests using a threshold of four messages. As shown in the figure, increasing the threshold did not improve the delivery ratio. On the other hand, Figure 5 confirms that the number of messages sent increased. It should be noted that although presenting higher delivery ratios, Pampa requires the transmission of a comparable number of messages. The small differences found for lower node densities can be attributed to the additional gains in reliability. The low reliability presented when schemas use a threshold of one and the increased number of messages transmitted with thresholds of four suggest that these schema are not adequate for further analysis in the rest of the paper and will be omitted.

The delivery ratio of the “distance-based” scheme with thresholds corresponding to 125m, 175m and 225m (respectively, 50%, 70% and 90% of the transmission range) are compared with Pampa in Figure 4. The figure shows that the “distance-based” scheme is particularly sensitive to its threshold. Like in “count-based”, the “distance-based” scheme also exhibits less tolerance to lower node densities than Pampa. Due to their extremely low reliability results, thresholds 175m and 225m are omitted from the remaining of the evaluation. The number of messages required by the “distance-based” schema is presented in Figure 5 and follows a similar pattern to that of the remaining schema.

A comparison between Figures 3, 4 and 6 shows that the performance of all schema improves with the movement of the nodes for the lowest densities. We attribute this unexpected behaviour to a reduced number of partitions occurring due to the concentration of nodes at the centre of the simulated space, a well-known effect of the random waypoint model [1]. To support this conclusion we counted the average number of nodes receiving each message in the 2000m $\times$ 2000m tests. The values were of 4.30, 5.91 and 5.92 respectively for tests in Speeds 0, 5 and 10. An interesting result from the figure is that with the additional gain in the node density, the “Distance-based” scheme improves its performance more than both schema with a threshold of 2. It should also be noted that Pampa augments its gains over the corresponding “count-based” schema.

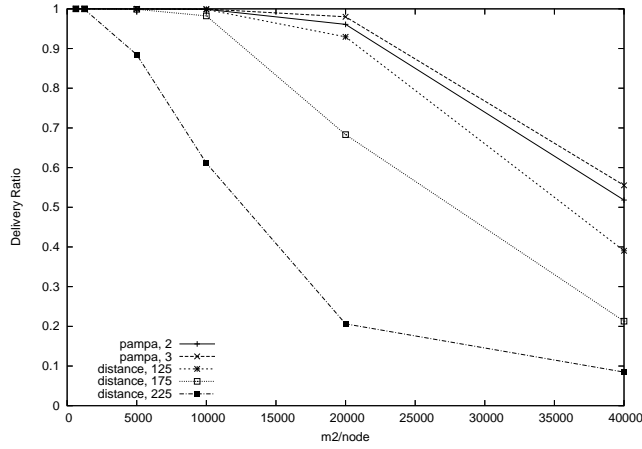


Figure 4: Delivery ratio (Speed 0)

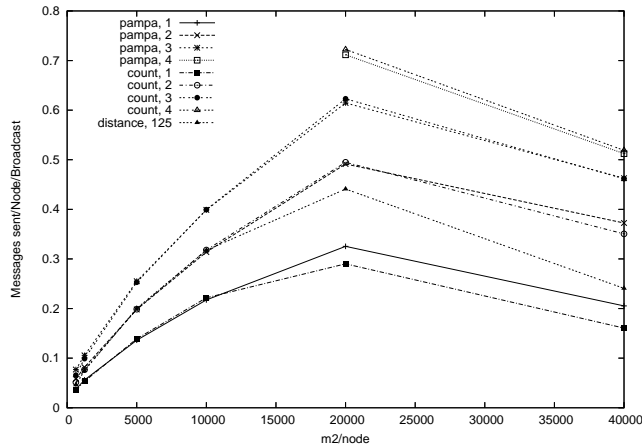


Figure 5: Transmissions per broadcast per node (Speed 0)

To evaluate the gain in coverage provided by each retransmission with Pampa, we present in Figure 7 the average number of hops travelled by a message before being delivered for the first time to each node. The figure shows that Pampa delivers messages in less hops than the remaining schema. Therefore, we conclude that Pampa can cover an higher number of nodes with each retransmission. The figure also shows that contrary to “count-based”, the number of hops in Pampa is independent of the threshold. We attribute this behaviour to the randomness of the scheme: when the threshold is lower, the probability of having the most adequate hops selected is reduced and some nodes will only receive the message after an additional retransmission. The results for 2000mx2000m have been omitted because the low delivery ratio presented do not allow the interpretation of this metric.

To conclude the evaluation, we show in Figure 8 the average time interval between the initial transmission of a message and its delivery to each node. Results show that messages in Pampa have only a negligible additional latency

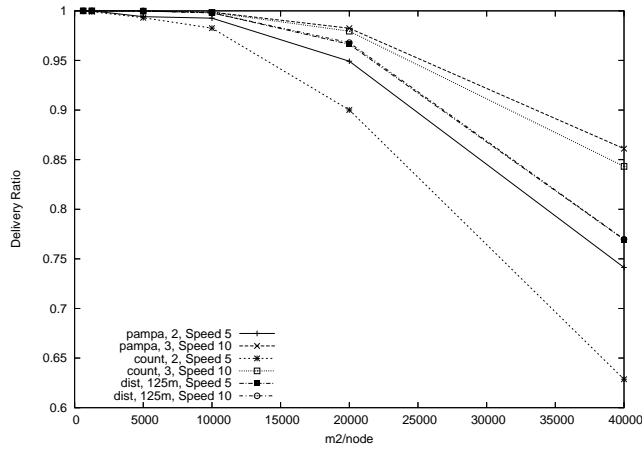


Figure 6: Delivery rate for tests with Speeds 5 and 10

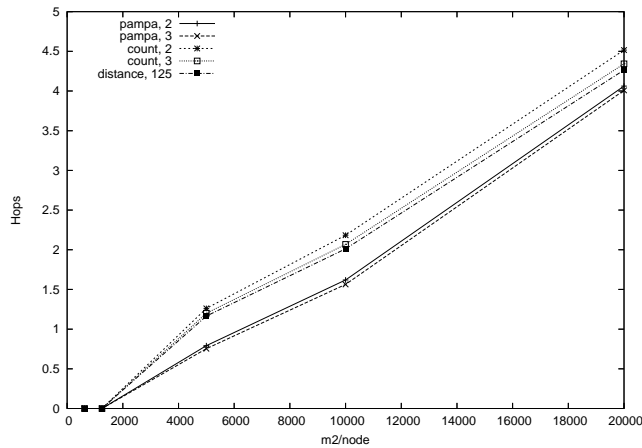


Figure 7: Hops to the source of the message (Speed 0)

when compared with the remaining schema. An important conclusion derived from these results is that, as expected, the large values returned by the function *delay* in Pampa when nodes are particularly close to the source do not negatively influence the performance of the protocol.

## 5 Conclusions

In Mobile Ad Hoc Networks, broadcasting a message to every node is an operation that consumes a non-negligible amount of resources at all participants. However, broadcast is frequently referred in the literature as a mechanism to be used by protocols at different levels of the networking stack. The most simple implementation of broadcast consists in having each node to retransmit each message after receiving it for the first time. This implementation, usually referred as flooding, creates a large redundancy of messages in the network and

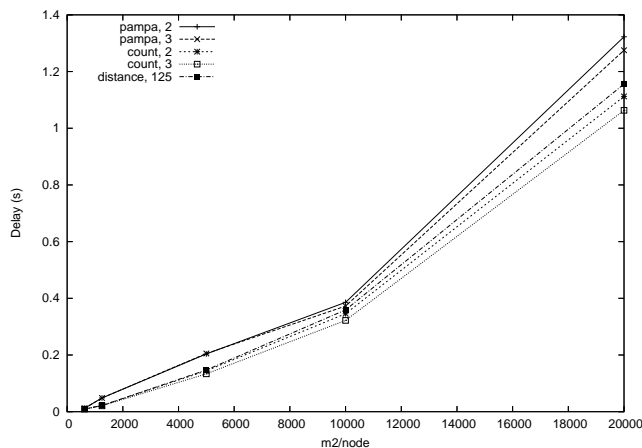


Figure 8: Average latency (Speed 0)

unnecessarily wastes resources at the participating nodes.

This paper presented an innovative mechanism that uses information locally available at each node to reduce the redundancy of the broadcast operation. The algorithm was compared with previous proposals and it was shown that it presents better performance.

## References

- [1] C. Bettstetter, G. Resta, and P. Santi. The node distribution of the random waypoint mobility model for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(3):257–269, 2003.
- [2] IEEE Computer Society LAN MAN Standards Committee. Wireless LAN medium access control (mac) and physical layer (phy) specifications. ANSI/IEEE Std 802.11, 1999 Edition, 1999.
- [3] F. Perich et. al. In reputation we believe: query processing in mobile ad-hoc networks. In *Proc. of the 1st Conf. on Mobile and Ubiquitous Systems: Networking and Services (MOBIQUITOUS2004)*, pages 326–334, 2004.
- [4] L. Feeney and M. Nilsson. Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proc. of the 20th Conf. of the IEEE Computer and Communications Societies (INFOCOM 2001)*, volume 3, pages 1548–1557, 2001.
- [5] Z.J. Haas, J.Y. Halpern, and Li Li. Gossip-based ad hoc routing. In *Proc. of the 21st Conf. of the IEEE Computer and Communications Societies (INFOCOM 2002)*, volume 3, pages 1707–1716, 2002.
- [6] D.B. Johnson and D.A. Maltz. *Mobile Computing*, chapter Dynamic Source Routing in Ad Hoc Wireless Networks, pages 153–181. Kluwer Academic Publishers, 1996.

- [7] P. Levis, N. Patel, D. Culler, and S. Shenker. Trickle: A self-regulating algorithm for code propagation and maintenance in wireless sensor networks. In *Proc. of the 1st USENIX/ACM Symp. on Networked Systems Design and Implementation (NSDI 2004)*, 2004.
- [8] S. Lim, W.-C. Lee, G. Cao, and C.R. Das. A novel caching scheme for improving internet-based mobile ad hoc networks performance. *Elsevier Journal on Ad-Hoc Networks*, 4(2):225–239, 2006.
- [9] C.E. Perkins and E.M. Royer. Ad-hoc on-demand distance vector routing. In *Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, 1999.
- [10] T. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall PTR, 2nd edition, 2001.
- [11] Y. Sasson, D. Cavin, and A. Schiper. Probabilistic broadcast for flooding in wireless mobile ad hoc networks. In *Proc. of the Wireless Communications and Networking Conf. (WCNC 2003)*, volume 2, pages 1124–1130. IEEE, 2003.
- [12] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu. The broadcast storm problem in a mobile ad hoc network. *Wireless Networks*, 8(2/3):153–167, 2002.