View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Purdue E-Pubs

Purdue University
Purdue e-Pubs

ECE Technical Reports | Electrical and Computer Engineering

5-1-2006

Distributed Hashing for Scalable Multicast in Wireless Ad Hoc Networks

Saumitra M. Das
Purdue University

Himabindu Pucha
Purdue University

Y. Charlie Hu
Purdue University

Follow this and additional works at: http://docs.lib.purdue.edu/ecetr

Das, Saumitra M.; Pucha, Himabindu; and Hu, Y. Charlie, "Distributed Hashing for Scalable Multicast in Wireless Ad Hoc Networks" (2006). *ECE Technical Reports.* Paper 336.
http://docs.lib.purdue.edu/ecetr/336

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

Distributed Hashing for Scalable Multicast in Wireless Ad Hoc Networks

Saumitra M. Das
Himabindu Pucha
Y. Charlie Hu

TR-ECE-06-10
May 1, 2006

School of Electrical and Computer Engineering
1285 Electrical Engineering Building
Purdue University
West Lafayette, IN 47907-1285

Contents

Abstract

Several multicast protocols for mobile ad hoc networks (MANETs) have been proposed that build multicast trees using location information available from GPS or localization algorithms and use geographic forwarding to forward packets down the multicast trees. These stateless *multicast protocols carry encoded membership, location and tree information in each packet. Stateless protocols are more efficient and robust than stateful protocols (ADMR, ODMRP) as they avoid the difficulty of maintaining distributed states in the presence of frequent topology changes in MANETs. However, stateless location-based multicast protocols are not scalable to large groups because they encode group membership in the header of each data packet, i.e. they incur a per-packet encoding overhead. Additionally, such protocols involve centralized group membership and location management, either at the tree root or the traffic source.*

In this work, we present the Hierarchical Rendezvous Point Multicast (HRPM) protocol which significantly improves the scalability of stateless location-based multicast with respect to the group size. HRPM incorporates two key design ideas: (1) hierarchical decomposition of multicast groups, and (2) use of distributed geographic hashing to construct and maintain such a hierarchy efficiently. HRPM organizes a large group into a hierarchy of recursively organized manageable-sized subgroups in an effort to reduce per-packet encoding overhead. More importantly, HRPM constructs and maintains this hierarchy at virtually no cost using distributed hashing; distributed hashing is recursively applied at each subgroup for group management and avoids the potentially high cost associated with maintaining distributed state at mobile nodes. The hierarchical organization and the distributed hashing property also allows HRPM to scale to large networks and large numbers of groups.

Performance results obtained via detailed simulations demonstrate that HRPM achieves enhanced scalability and performance. Coupled with its leverage of stateless geographic forwarding, HRPM scales well in terms of the group size, the number of groups, the number of sources, as well as the size of the network. In particular, HRPM maintains close to 95% multicast delivery ratio while incurring on average 5.5% per packet tree-encoding overhead for up to 250 group members in a 500-node network. Furthermore, it achieves a steady 95% delivery ratio while incurring nearly constant overhead as the number of groups increases from 2 to 45, while keeping the total number of receivers constant at 180, in a 500-node network. Lastly, it steadily achieves above 90% delivery ratio as the network scales up to 1000 nodes with up to 30% group members. As a reference, we also compared HRPM to ODMRP, a state-of-the-art topology-based multicast protocol that is scalable to large groups. HRPM performs comparably to ODMRP across a wide range of group sizes. More over, HRPM outperforms ODMRP when the network size, the number of groups, or the number of sources increases.

Keywords: *Simulations, location based multicast, MANETs, scalability, distributed hashing.*

1 Introduction

A mobile ad hoc network (MANET) consists of a collection of wireless mobile nodes dynamically forming a temporary network without the use of any existing network infrastructure or centralized administration. In such a network, since nodes are often not within the radio transmission range of each other, each node operates not only as a host, but also as a router, forwarding packets for other mobile nodes.

Multicast is a fundamental service for supporting collaborative applications among a group of mobile users [10]. Different from in the wired Internet, multicast in MANETs is faced with a more challenging environment. In particular, multicast in MANETs needs to deal with node mobility and thus frequent topology changes, variable quality wireless channel, constrained bandwidth, and low memory and storage capabilities of nodes. On the other hand, unlike in the Internet, nodes in a MANET can be modified at the network layer to provide group communication support. This reduces the need for overlay-based group communication that has been popular in the Internet.

Numerous multicast protocols have been proposed for multicast in MANETs. These include traditional tree- or mesh-based protocols such as MAODV [37], ADMR [19], ODMRP [27], overlay-based protocols such as AMRoute [47], PAST-DM [15], and back-bone-based protocols such as MCEDAR [38], and more recent stateless protocols such as DDM [20], HDDM [16], and RDG [30]. These multiple protocols either rely on underlying unicast routing schemes (e.g., [47, 20]) or expend great effort to maintain a distributed multicast routing structure (e.g., [19, 27]). Both factors affect the scalability of these protocols.

Recently, several location-based multicast protocols for MANETs have been proposed [2, 7, 32] which neither assume any unicast routing scheme nor build any distributed multicast routing structure. These protocols build multicast trees using location information available from global positioning systems such as GPS [44] and use geographic forwarding to forward packets down the multicast trees. Sharing the stateless nature of geographic forwarding, these protocols are *stateless*, as they carry encoded membership and location as well as tree information in each packet, so that the multicast membership and routing state do not have to be distributed as in traditional multicast protocols such as ADMR or ODMRP. Stateless protocols are more efficient and robust than stateful protocols as they avoid the difficulty of maintaining distributed states in the presence of frequent topology changes in MANETs.

However, because of their stateless nature, previous location-based multicast protocols suffer from limited scalability in terms of the group size. Conceptually, stateless location-based multicast protocols are not scalable to large groups because they encode group membership in the header of each data packet.

In fact, previous location-based protocols are explicitly proposed for small groups.

In this paper, we study the scalability aspect of location-based multicast, in particular, the group (membership and location) management in location-based multicast protocols. A well-known general approach to reducing the load of managing a large multicast group is to partition the large group into hierarchically organized subgroups of manageable sizes. The immediate consequence of distributing membership management is that the protocol becomes stateful. The key question here is therefore whether there is a way to leverage the concept of hierarchical membership management *without* incurring the high cost associated with maintaining distributed state at mobile nodes?

We present the Hierarchical Rendezvous Point Multicast (HRPM) protocol which meets the above criterion and significantly improves the scalability in the group size of previous location-based multicast protocols. HRPM leverages two key techniques: distributed *mobile geographic hashing* and *hierarchical decomposition of multicast groups*. Given a data item and a location, mobile geographic hashing maps (routes) the data item to the node whose geographic location is *currently* closest to the location. Thus mobile geographic hashing allows multicast group members to agree upon a fixed rendezvous point (RP) (and the current node associated with it) as the group manager (root) without incurring any overhead, for example in keeping track of an otherwise mobile group root. This in turn allows the multicast protocol to maximally leverage stateless geographic forwarding for efficient group membership and location management. More importantly, a rendezvous point group management enabled by mobile geographic hashing can be recursively applied to enable a hierarchically organized set of manageable sized RP-based subgroups such that multicast inside each subgroup satisfies a per packet tree-encoding overhead constraint. Group management under such a hierarchy is extremely lightweight as the RP subgroup roots are effectively "stationary". Thus although conceptually the membership state in HRPM is distributed among the subgroup roots, but since they are virtually stationary, HRPM effectively avoids the high cost associated with maintaining distributed state at mobile nodes.

We first study the performance of HRPM as compared to previously proposed location-based multicast protocols. The results demonstrate that for large groups (up to 250 members experimented with), HRPM significantly improves the scalability of previous location-based multicast protocols.

We then compare HRPM to ODMRP, a topology-based multicast protocol that is scalable to large groups. In this comparison, we find that HRPM is comparable to ODMRP in performance as the group size increases. However, HRPM significantly outperforms ODMRP as the network size is increased (up to 1000 mobile nodes experimented with). In addition, HRPM outperforms ODMRP when a large number of groups (up to 45 experimented with) or a large number of sources per group exist.

In summary, leveraging stateless geographic forwarding for data delivery and distributed hashing for group and location management allows HRPM to scale well in terms of the group size, the number of groups, the number of sources, as well as the size of the network.

The rest of the paper is organized as follows. Section 2 formulates the location-based multicast problem. Section 3 presents the detailed design of HRPM. An analysis of key design parameters of HRPM is presented in Section 4. Section 5 presents the simulation studies. Section 6 summarizes related work and finally, Section 7 concludes the paper.

2 Preliminaries

The multicast problem deals with transmission of information from a node to all members of a group. If we denote $G = (V, E)$ as the un-directed graph of the topology of a MANET where V is the set of mobile nodes and E is the set of wireless links, the multicast problem is to deliver a message to a subset $V_G \subset V$ while optimizing certain application specific metric such as bandwidth cost or delay. In a MANET with positioning systems such as GPS [44], each node can determine its own geographic location. Such location information has been previously leveraged to improve the scalability of unicast routing [21] via stateless geographic forwarding. Stateless protocols are more efficient and robust than stateful ones as they avoid the difficulty of maintaining distributed states in the presence of frequent topology changes. Similar to unicast, location information can be exploited to provide location-based multicast. To maintain the stateless nature, these protocols encode the membership as well as tree information in each packet so that membership/forwarding state are not distributed as in multicast protocols such as ADMR or ODMRP. In the following, we discuss the three components of a location-based multicast protocol.

Group Membership and Location Management An efficient scheme for the management of group membership and locations is critical to the efficiency and scalability of location-based multicast, since nodes are continuously moving in a MANET. To manage the group membership, group members can multicast their membership/locations to all other group members [7], or send their updates to an agreed-upon root so that the group members can then contact the root to obtain updated information. Moreover, either the location of the group members [7] or of all the nodes in the network [2] are required depending on the nature of the multicast tree used.

Multicast Tree Construction Once the group membership and location information are obtained, the source of the multicast can construct a multicast tree, using either an overlay tree [7] consisting of only group member nodes or a physical tree [2] consisting of group member nodes and other nodes en-route between the member nodes. Many graph algorithms exist for the construction of such multicast trees. These tree construction algorithms exploit the correlation between geometric distance and network distance (number of routing hops) that longer geometric distance implies more network hops [7], and use geographic distances between nodes as edge weights.

Data Delivery The data delivery mechanism depends on the nature of the tree and the location/member management scheme used. A physical tree can be efficiently encoded in the header of a data packet. Such data packets can be delivered via source routing [2] as the tree contains all the intermediate nodes. In case of an overlay multicast tree, based on the group/location management scheme, there can be two approaches to data delivery: (1) If the locations of the group members are known only to the source of the multicast tree, the destinations and the locations of the group members need to be encoded in the packet header at the source. (2) If every group member knows every other group member's location, only the destinations are encoded in the packet header (since each intermediate overlay node can fill in the locations and decide how to forward the packet). This reduces the per packet encoding overhead. However, this requires intermediate overlay nodes in the tree to acquire such location information via other means, for example, updates from the destination nodes directly. Moreover, in case of an overlay multicast tree, as the tree members may not be within direct reach of each other, geographic forwarding is needed to deliver data packets along the overlay links.

In this paper, we use a greedy geographic forwarding algorithm as the routing protocol. Each node periodically announces its IP address and location to its one-hop (within the radio transmission range) neighbors by broadcasting BEACON packets. Each node maintains the IP and location information of its neighbors. Each packet being routed contains the destination address in the IP header and the destination's location (x- and y-coordinates) in an IP option header. To forward a packet, a node consults its neighbor table and forwards the packet to its neighbor closest in geographic distance to the destination's location. Note that the above greedy geographic forwarding can lead to a packet reaching a node that does not know any other node closer to the destination than itself. This indicates a hole in the geographical distribution of nodes. Recovering from holes can be achieved using *face-routing* (first proposed in [4] and extended in GPSR [21] and GOAFR+ [24]).

3 Hierarchical Rendezvous Point Multicast

In this section, we describe the design of HRPM. HRPM incorporates two key design concepts: (1) Use of hierarchical decomposition of multicast groups, and (2) Leveraging geographic hashing to construct and maintain such a hierarchy efficiently.

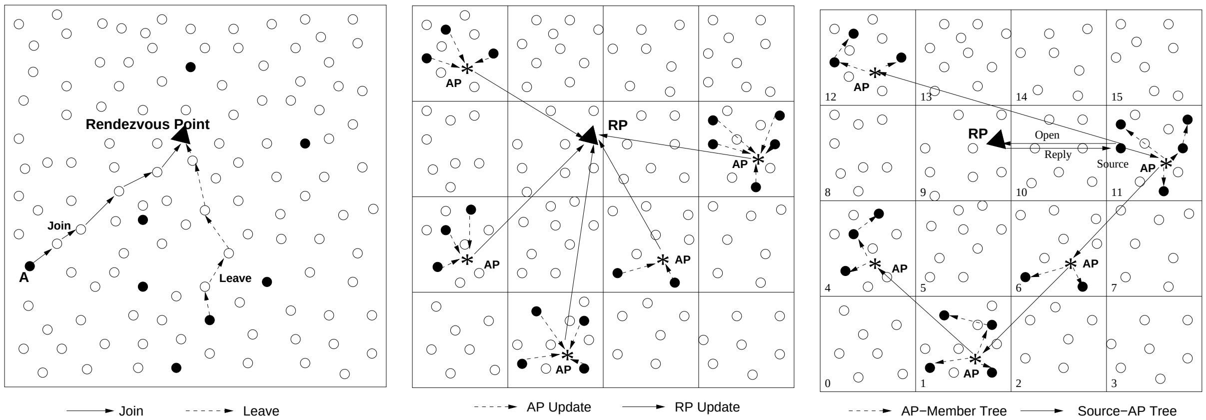
Hierarchal routing [22] is a well known approach to reducing the protocol states in a large scale network. The per-packet encoding overhead of a stateless location-based multicast protocol grows with the group size as $O(G)$, where G is the multicast group size. So, an increase in G severely limits the usability of such protocols. The main design goal of HRPM is to limit the per-packet overhead to an application-specified constant (ω), irrespective of the increase in G. The value of ω is a parameter of HRPM and can be adjusted based on the amount of overhead that can be tolerated by an application. To achieve this, HRPM recursively partitions a large multicast group into manageable sized subgroups in which the tree-encoding overhead satisfies the ω constraint. This partitioning is achieved by geographically dividing the MANET region into smaller and smaller *cells*. Such cells form a hierarchy with the root representing the entire region. Every cell in the hierarchy has an AP (*Access Point*), and the entire region has an RP (*Rendezvous Point*). All members in a leaf cell of the hierarchy form a subgroup and are managed by that cell's AP. Groups of APs are managed recursively, i.e., by the APs of their parent cells. ω is an application parameter and we discuss how HRPM adjusts the hierarchy to meet this ω constraint in Section 4.

Central to the design of HRPM is the fact that both RPs and APs are logical entities. If such a logical entity is associated with a specific node (IP address), keeping track of the RP/AP would require an external location service or some flooding-based mechanism due to mobility in MANETs. This can potentially incur high overhead. To avoid such overhead, HRPM disassociates the RP/AP from any specific node by adopting the concept of *geographic hashing* previously proposed for data storage in static sensor networks [36]. Given a data item and a location, geographic hashing maps (routes) the data item to the node whose geographic location is closest to the location. Since in MANETs different mobile nodes can become the closest to a fixed location over time, *mobile geographic hashing* in HRPM extends geographic hashing via a continuous handoff process which ensures the data item is always stored on the node currently closest to the location. Thus if the members of a group/subgroup use an agreed-upon hashing function to hash the multicast group identifier and obtain the RP/AP location for the group/subgroup, all group management messages can be routed to the RP/AP by leveraging geographic forwarding.

In the following, we describe the details of HRPM group management, tree construction, and data delivery.

3.1 Group Management

We first introduce the concept of rendezvous point group management (RPGM) assuming a flat geographic domain. We then introduce hierarchical domain decomposition of a multicast group and describe how to apply RPGM recursively in a hierarchy of subdomains.



(a) Rendezvous point group management in HRPM ($d = 1$)

(b) Location updates in HRPM ($d = 4$)

(c) Data delivery in HRPM ($d = 4$)

Figure 1: Group management, tree construction, and data delivery in HRPM.

3.1.1 Rendezvous Point Group Management

Rendezvous point group management allows multicast group members to leverage geographic hashing for efficient group management. Figure 1(a) shows RPGM in a flat geographic domain. Any node that wants to join a multicast group first hashes the group identifier to obtain the RP's location in the physical domain of the network using a hash function:

$$H(GID) = \{x, y\} \; where \; x, y \in MANETregion$$

This hashing function takes as input the group identifier (GID) and outputs a location (x- and y-coordinates) contained in the region. Note that we assume that this is a well known hash function that is known by nodes that enter the network through external means or using some resource discovery process.

After obtaining the hashed RP location for the group it wants to join, the node sends a JOIN message addressed to this hashed location. This JOIN message is routed by geographic forwarding to the node that is currently closest to the hashed location in the network. This node is the designated RP at this time. Since there is only one such node at any given time, the JOIN messages from all the group members converge at a single RP in a distributed fashion without global knowledge. Figure 1(a) depicts the JOIN message from node A being routed to the RP. Similarly, LEAVE messages are also routed to the RP.

Note that computing the hashed location assumes that all nodes know the approximate geographic boundaries of the network. Such boundary information may be pre-configured at nodes before deployment or discovered using some simple protocol. This assumption is consistent with the literature [28, 36].

3.1.2 Virtual Hierarchical Organization

To apply rendezvous point group management described above hierarchically, HRPM partitions the geographic domain into d^2 equal-sized square sub-domains called *cells*, where d is the decomposition index. The partition can be repeated recursively until each cell consists of a manageable sized subgroup of members. For the ease of explanation, we restrict our following discussion of HRPM to two levels, as shown in Figures 1(b) and 1(c). We defer the description of how HRPM dynamically adjusts the d value according to the group size and why a two-level hierarchy is sufficient till Section 4.

In case of a two-level hierarchy, the members of each subgroup, i.e., in each leaf cell of the hierarchy, choose an AP in the cell using the same geographic hashing of the group identifier except the hashed location is scaled to be inside the cell. The APs then coordinate with the RP for the group. We extend the hash function for locating APs as well as the RP for a particular multicast group as:

$$H(GID, d, myLoc) = \{x, y\} \; where \; x, y \in Cellregion$$

where d is the decomposition index and $myLoc$ is the current location of the node invoking the function. Figure 1(b) depicts the network partitioning for $d = 4$ in which case the region is divided into 16 cells. Note that for the special case of $d = 1$ (Figure 1(a)), only one cell exists in the region and the function outputs the hashed location of the RP.

3.1.3 Hierarchical Rendezvous Point Membership Management

To join a hierarchically decomposed multicast group, a node first generates the hashed location for the RP and sends a JOIN message to the RP, same as in the flat domain scenario. After receiving the value of the current decomposition index d of the hierarchy from the RP, the joining node invokes the hashing function with d and its current location to compute the hashed location of the AP of its cell. The node then starts periodically sending LOCATION UPDATE packets to its AP. Such location updates are soft-state and serve as a subgroup membership update, i.e., if an AP stops receiving location update from a member, it assumes the member has migrated to another cell.

Upon receiving (or not receiving) a location update from each member, the AP summarizes the membership inside its cell as non-empty (or empty) and further propagates to the RP whenever the membership switches between empty and non-empty. The cells in which no group members exists do not have any active APs and consequently no updates from these cells are sent to the RP, further reducing the update congestion at the RP, as shown in Figure 1(b).

The state the RP needs to keep about the group is just a bit vector of d^2 bits with each bit representing whether a member exists in a particular cell or not. Thus the RP can easily encode a large number of APs. For example, 256 APs from 256 cells can be encoded in 32 bytes. Thus for a large multicast group, a two-level HRPM reduces the state required at the RP to d^2 bits while requiring the (leaf) AP in each cell to only maintain the addresses and locations of $\frac{G}{d^2}$ nodes on average where G is the original size of the multicast group.

The frequency of location update determines the accuracy of the knowledge at the RP/APs and consequently the accuracy of the multicast tree. We use threshold-based updates where each node initiates a LOCATION UPDATE whenever it moves 100m from the location of the last update. This is similar to the strategies used in location services for MANETs (for example, [28]).

When a node moves into a new cell, it does not immediately send an update to the new AP. Its previous AP can continue to route data using geographic forwarding. When the node moves a certain distance (i.e. 100m) from the location of its last update, it will send a new update to the AP in the new cell.

Note that the group management architecture of HRPM needs to also deal with the situation when nodes of a group are close to each other, i.e. there is locality in the group membership. In such a case, extra overhead is incurred in sending control messages to an RP that may be far away from the cluster of group members. Fortunately, a hierarchy is useful in this scenario as well since a group with locality will send updates primarily to a small set of APs in the clustered cells where the group members are located. The RP is only sent one update from each AP indicating the existence of members in its cell. Each source only needs to retrieve a bit vector from the RP once to perform data delivery which will be done locally through the nearby APs. Thus, when group membership has geographic locality, HRPM incurs minimal overhead in using an RP. We believe this small overhead is justified given the overall overhead reduction made possible by using a virtual hierarchy.

3.1.4 Hierarchy Maintenance

As nodes move, the RP or an AP for a particular group may change as some other node becomes the closest to the hashed location of the group identifier. Thus a continuous handoff protocol is required to maintain geographic hashing. The current RP/AP on the receipt of any BEACON packet (used in geographic forwarding) checks whether this neighbor is currently closer to the hashed location. If so, the current RP/AP performs a handoff procedure that transfers the state of the multicast group/subgroup to the neighbor. This neighbor now becomes the RP/AP. Note that this process is transparent to the multicast group members.

In rare instances, messages sent to the RP/AP from different nodes may not converge at a single node. This could be due to the loss of BEACON packets which causes inconsistencies in the view each node has of its neighborhood. This convergence problem is solved as follows. When the first group management message *for a new group* g arrives at a node A which discovers that it is the closest to the hashed location, the node initiates a *converge* operation for the packet by buffering the packet and initiating an expanding ring broadcast search for any other node that also thinks it is the current RP/AP node for the group g. This search is limited to two hops since any other potential RP/AP node is expected to be close by due to the geographic hash. If another node is located that is acting as an RP/AP node for the group g, the current node A relays the buffered packet to this RP/AP node for further processing along with its own fresh location so that the current RP/AP node can perform handoff if appropriate to A. In this way consistency of RP/APs are maintained on the rare occasions that convergence does not occur.

3.1.5 Adaptivity and Per-group Architecture

Another important design choice of HRPM is *adaptive per-group hierarchies*, i.e. each group operates with its own virtual hierarchy based on its group size. Note that each group automatically has logically and potentially physically separate nodes serving as RPs and APs. The per-group hierarchy architecture is motivated by the fact that depending on the group size G there exists a tradeoff between the level of hierarchical partitioning required and the path length traveled by the location updates. The larger the number of levels in the hierarchy, the more detours location updates and data packets need to take to reach the RP. For small groups, since the amount of aggregation required is low and there is no hot spot at the RP, the hierarchy imposes overhead without adequate gain in performance. For large groups, increasing the levels of the hierarchy results in lower congestion at the RP and reduced encoding overhead in data packets. In summary, HRPM uses per-group hierarchy construction to allow choosing suitable hierarchy heights for groups of different sizes.

As will be discussed in Section 4, HRPM uses the RP to coordinate the construction of dynamic per-group hierarchies according to the changing group size. However, as will be explained in Section 4, the hierarchy height rarely needs to be increased to beyond two levels.

3.2 Tree Construction and Data Delivery

HRPM provides a framework for scalable group management in location-based multicast in which any tree construction algorithm of choice can be utilized based on the application metrics. For the performance study in this paper, we assume the use of a specific overlay tree construction algorithm that minimizes the bandwidth cost. The source of the multicast uses geographic distances between the multicast group members as edge weights to build an overlay graph, and then a minimum spanning tree of the overlay graph (i.e. an overlay tree) is built, using MST algorithms (e.g. Prim's [8] or Kruskal's [8]). In Section 4.3, we evaluate different tree construction algorithms and show that such an overlay MST makes the best tradeoff between bandwidth efficiency, computational cost, and location management overhead.

To send a data packet, the source sends an OPEN SESSION message to the RP and receives the membership group vector from the RP. The membership vector is of size d^2 bits, with a bit '1' for each cell that contains any group members. This vector is cached by the source. The RP differentially updates (sending only the changes) the source whenever the RP receives a change in membership notification from an AP. Once the group vector is received, the source can build a virtual overlay tree (the $Src \rightarrow AP$ tree) by assuming each active AP as a vertex in a topology graph. The tree is *virtual* since the source does not need to know the actual AP node in each cell; it just needs to hash the GID in the AP's cell to put in a virtual vertex in the topology graph.

Multicast data packets are delivered down the $Src \rightarrow AP$ tree. The per-packet encoding overhead is limited to a constant of d^2 bits. Once a data packet reaches an AP, the AP constructs an $AP \rightarrow Member$ overlay tree this time using member node identifiers and their actual locations. The AP then encodes the list of destinations and their locations under each branch of the overlay tree in each data packet sent along that branch. On average, the number of group members in a cell is $\frac{G}{d^2}$ where G is the group size. The packet then is delivered to the nodes down the tree, with each node recomputing a tree of the remaining destinations in the list. Note that the size of this multicast header reduces as the packets travel down the tree and the height of the remaining multicast tree reduces.

Figure 1(c) shows an example of data delivery in HRPM for a multicast group which only has group members in cells 1, 4, 6, 11 and 12. A multicast source receives a group vector with bits 1, 4, 6, 11 and 12 set from the RP since only those cells contain group members and consequently active APs. It then constructs a virtual topology graph containing all the active APs, and builds a $Src \rightarrow AP$ multicast tree containing the active APs. Multicast data packets are first transmitted down the $Src \rightarrow AP$ tree to reach the active APs, and then further down each individual $AP \rightarrow Member$ tree constructed by each active AP.

Since the primary focus of this paper is on multicast routing and group management we do not address reliability and security issues in this paper due to lack of space. As with all multicast protocols, malicious operation of nodes or failure of nodes can cause service disruptions. Mechanisms to deal with these problems are part of our future work.

3.3 Dealing with sparse topology

A fundamental problem that has been researched with regard to geographic forwarding is the occurrence of local maxima while greedily forwarding packets. In such a situation, a packet is received by a node whose transmission range does not cover the destination location yet does not know of any other neighbor closer to the destination location than itself. Local maxima is more likely to occur in sparse network deployments. Even when the overall deployment is not sparse, certain regions of the network may be sparse due to non-uniform node distribution. Local maxima is also referred to as a *hole* in the literature.

To enable geographic routing when local maxima occurs, *face-routing* was proposed (first in [4] and extended in [21], [24]) to route along the face of a planarized topology surrounding the hole graph until greedy forwarding can be invoked again or the destination is reached. Recent work [34] has extended the algorithms for face routing to take into account realistic radio transmission characteristics such as asymmetric links and non-ideal ranges. Such schemes can also be easily incorporated into the geographic forwarding component of HRPM.

Similar to previous geographic unicast routing protocols, HRPM also needs to deal with *holes* in the network topology. Our implementation of HRPM uses GPSR [21] as the underlying geographic forwarding protocol to recover from holes. Holes can occur in the following two cases during the operation of HRPM.

1. Routing to a node This scenario occurs when the $AP \rightarrow Member$ overlay tree is being traversed for data delivery to the individual group members. In this case, the problem is similar to that faced by unicast geographic routing protocols and thus the normal protocol operations of GPSR (i.e. distributed planarization followed by face traversal) are used to route the packet to the destination node thereby avoiding the hole. This is expected to work unless the network is partitioned.

2. Routing to a hashed location Routing to a hashed location in HRPM occurs during the routing of JOIN, LEAVE and LOCATION UPDATE messages to the RP/AP and during data delivery to the APs using the $Src \rightarrow AP$ tree. Holes that occur whenever a message is routed to a hashed location have to be dealt with differently from when a message is being routed to a specific node. In the latter case, face routing is triggered whenever a node does not have the destination node in its table and does not know a neighbor closer to the destination node. On the other hand, dealing with a hole while routing to a hashed location is more complicated, since when a node encounters a hole, it needs to distinguish whether the hole is en route from the sender to the hashed location, or the hashed location is inside the hole.

We modify HRPM to deal with local maxima when routing to a hashed location as follows. A node X that detects a local maxima stores the sequence number of the packet and starts face routing (perimeter forwarding mode in GPSR). During perimeter forwarding, the packet may be switched back to greedy mode (if a node discovers itself to be closer to the hashed location than the point of entry into the current face). If this happens, the packet will continue to be routed normally. If the packet traverses around the face and comes back to X, then X becomes the rendezvous point. All subsequent packets are routed in this manner and are expected to reach the current rendezvous point despite sparse topology.

3.4 Other communication primitives

In this paper, we focus on a design of HRPM for enabling multicast operation. Apart from multicast, anycast and manycast [6] are also useful communication primitives for mobile ad hoc networks. The group and location management architecture of HRPM can be easily leveraged for manycast and anycast services. For example, HRPM can be extended to provide manycast service which delivers data to any k of G group members by constructing a tree consisting of k group members at the source of the manycast. In the non-hierarchical case, this k-member tree can be trivially constructed. In the hierarchical case, k APs (with group members) are selected to forward the message to. These selected APs then deliver the message to at most 1 group member in each of their cells.

HRPM can also be easily extended to provide anycast service. In this case, each node needs to contact its AP in the lowest level of the hierarchy for that group. The AP checks if it can locate a group member in its *cell*. If one exists, it is notified, otherwise the anycast request is forwarded up the hierarchy to a higher level AP. This is done recursively till an anycast recipient is found. Note that this architecture allows for the anycast request to travel to a nearby anycast group member and exhibits good locality properties.

4 Analysis

In this section, we analyze the depth of the HRPM hierarchy and the choice of the decomposition index d.

4.1 Choice of d and Hierarchy Depth

We first show how HRPM chooses the decomposition index d that satisfies certain per-packet encoding overhead constraint. We then show that a two-level HRPM hierarchy is sufficient to support a very large multicast group. To simplify the analysis, we assume a random uniform distribution of N nodes in the geographic domain, the existence of G group members, and that the cells have about the same number of group members. For simplicity, we assume the MANET region to be a square of side length l, and each cell to be a square of side length k.

In a two-level HRPM hierarchy, the $Src \rightarrow AP$ tree rooted at the source has maximally d^2 members (due to d^2 cells) and the per-packet encoding overhead is $d^2/8/f$ bytes where f is the average fan-out of the overlay tree at the root. Each $AP \rightarrow Member$ tree has on average $\frac{G}{d^2}$ members and thus the per-packet encoding overhead is at most $C \cdot \frac{G}{d^2}/f$ bytes, assuming that the cost of encoding the node identifier and locations is C. Note that as the data packet descends either type of overlay trees, the tree-encoding overhead decreases as the remaining subtree becomes smaller and smaller. Since the nodes within each cell are assumed to be uniformly distributed, i.e., similar to the APs, the overhead in the two kind of trees are expected to decrease in a similar fashion, and thus we can focus on comparing the very first packet(s) departing the tree roots.

Since the design goal of HRPM is to limit the per-packet encoding overhead, for example, to be less than ω bytes (or a fixed percentage of the payload), the partitioning of the network region into cells is governed by two constraints. The first constraint requires that the worst case encoding overhead in the $AP \rightarrow Member$ tree, $C \cdot \frac{G}{d^2}/f$, be less than ω bytes. Assuming a worse case fan-out from the tree root of 1, the constraint becomes

$$C \cdot \frac{G}{d^2} \leq \omega \tag{1}$$

The second constraint dictates that the worst case encoding overhead in the $Src \rightarrow AP$ tree, $\frac{d^2}{8}/f$, is also less than ω bytes. Thus the constraint becomes

$$\frac{d^2}{8} \leq \omega \tag{2}$$

In HRPM, the RP selects a particular decomposition index d based on the group size G and the MANET region side length l subject to the above constraints. Since all group JOIN and LEAVE messages reach the RP, it knows the group size G. The RP evaluates equation (1) to choose a d value that is just large enough to satisfy the constraint. It then checks if this value of d satisfies equation (2). In this case, HRPM forms a two-level hierarchy with decomposition index d. As an example, consider a multicast group of size 125. Using equation (1) and $\omega = 96$ bytes (20% of 512 bytes), we have $d = 3.95 \approx 4$. As this value of d satisfies equation (2), HRPM will divide the network into 16 grids, with the RP having a constant encoding overhead of 2 bytes.

When the multicast group grows to be large enough that no choice of d can satisfy both equations (1) and (2) for a particular ω, HRPM increases the level of the hierarchy to 3 or higher. Effectively, the depth of the hierarchy should be the smallest h that satisfies equation (2) and

$$C \cdot \frac{G}{d^{2(h-1)}} \leq \omega \tag{3}$$

In a depth h hierarchy, the top level remains a $Src \rightarrow AP$ tree, followed by $(h-2)$ levels of $AP \rightarrow AP$ trees, and the bottom level consists of $AP \rightarrow Member$ trees.

Based on the above analysis, for a reasonably small ω, a two-level hierarchy can support multicast groups that are larger than any deployable MANETs today. For example, assume the per packet overhead is restricted to be below 20% of the payload size of 512 bytes, i.e., around 100 bytes. Since 12 bytes are needed to encode a node identifier and its x- and y-coordinates, $C = 12$. Using equation (2), the maximum d that can be supported by the RP is 27 for $\omega = 96$. Substituting this value of d in equation (1) results in $G \approx 5800$, i.e., a two-level HRPM hierarchy can support up to 5800 group members while limiting the per-packet encoding overhead to be under 20%. Note due to the fan-out at the tree roots and the shrinking tree size during tree descent, the average per-data packet overhead is expected to be much lower than ω.

When the RP decides to adjust d due to changes of the group size, it multicasts a NOTIFY message containing the new d value to all member nodes, i.e., via the current hierarchy. Upon receiving such a message, each member node generates the hashed location for its new AP and starts sending updates to that AP. The new APs then send the aggregated membership to the RP.

4.2 Tradeoff between Encoding Overhead and Delay

There exists a tradeoff between the number of partitioned cells and the detours in the tree and consequently the delay in data delivery. In general, the more partitions, i.e., the larger the d, the longer the average detours data packets will take before reaching group members. HRPM chooses the minimum d that satisfies both equations (1) and (2) to improve the forwarding cost and delay while satisfying the ω constraint. We experimentally evaluate this tradeoff in the next section.

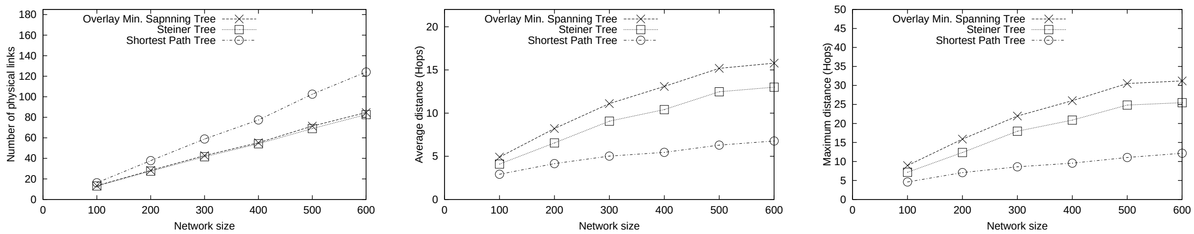
4.3 Choice of Tree Construction Technique

HRPM multicast involves the construction of a tree rooted at the source and containing at least all the multicast group members. A first-cut approach is that the tree be constructed using global knowledge of the locations of all nodes V in a MANET (both group members and non-members), i.e. the well known Steiner tree problem in graphs. The Steiner problem has been shown to be NP-Complete, and many heuristics have been proposed that provide an approximate solution in polynomial time. For example, the TM heuristic [42] provides an approximation in $O(N^2)$ time. The work in [2] (DSM) proposes such an approach in which given global knowledge of locations and group membership, a source can construct an approximate Steiner tree using heuristics to perform multicast. However such an approach requires the flooding of location and group membership information of each node to all nodes in the network in order to allow the construction of the Steiner tree at any source. Thus, this approach potentially limits the scalability of multicast.

A second approach is to construct an overlay minimum spanning tree (i.e. a tree that spans the group members without involving intermediate nodes)[1]. This approach is advantageous because: (1) it reduces the group management overhead by managing the membership and location of only the G group members, and (2) the overlay tree can be built using computationally simpler algorithm such as Prim's or Kruskal's MST algorithms. However, the overlay tree potentially can be less bandwidth-efficient than a Steiner tree constructed using both group member and non-member nodes.

To evaluate which tree construction algorithm provides the best tradeoff between bandwidth efficiency, delay and computational complexity, we performed simulation experiments comparing the performance of three tree construction algorithms: (1) An overlay minimum spanning multicast tree built using a MST algorithm, (2) A Steiner tree built using the TM heuristic, and (3) A low-delay multicast tree in which shortest paths (with lowest accumulated weight edges) are used to deliver data to each

[1] An overlay tree is in fact a heuristic for the Steiner tree problem [12]



(a) Avg. bandwidth consumed (b) Avg. distance (hops) (c) Maximum distance (hops)

Figure 2: Bandwidth consumption and delay of various tree building algorithms. Multicast group size for each network size is 10% of the network size.

group member built using Dijkstra's single-source shortest path algorithm. Each tree construction algorithm was evaluated over many randomly generated sample network topologies of different sizes. The edge weights between a pair of nodes were set to the geographic distance between the pair of nodes.

Figure 2 depicts the average bandwidth consumed (measured in number of physical links on which a transmission is required) as well as the average and maximum distance to any node in the multicast tree (measured in number of hops). The results show that as the network size is increased, the bandwidth efficiency of the overlay and Steiner trees are very close and both are much better than the shortest path tree. We also found that for a given network size, as the group size increases, there is a slight gain in the Steiner tree since there are more opportunities to use non-group members to improve the bandwidth efficiency. However, the gains observed are not significant enough to warrant the requirement that every node knows every other node's location since this incurs high continuous overhead. Since Steiner tree construction using the TM heuristic is also more computationally expensive as the network size and the group size increases, the delays associated with the computations were also not acceptable.

In summary, due to the reduced location management overhead and acceptable bandwidth efficiency observed, HRPM uses an overlay MST to construct both the $Src \rightarrow AP$ tree and each of the $AP \rightarrow Member$ trees. The Src/AP constructs a graph with edge weights being the geographic distances between group members. It then constructs an overlay MST using Prim's algorithm which has a complexity of $O(g^2)$, where g is the group members in the tree, and is bounded by $max(d^2, \frac{G}{d^2})$. The tree is encoded in each data packet originated at the tree root (the Src or an AP). As the data packet traverses down the tree, only the remaining subtree needs to be encoded in the data packet. The $AP \rightarrow Member$ tree built at each active AP has on average $\frac{G}{d^2}$ overlay edges. The $Src \rightarrow AP$ tree has in the worst case d^2 overlay edges which happens when every cell has active group members.

5 Performance Study

In this section, we first describe the methodology of our study. We then present the performance results.

5.1 Simulation Methodology and Metrics

We implemented HRPM in the Glomosim [50] simulator. Glomosim has a comprehensive radio model and has been widely used for simulation studies of MANETs. We use a IEEE 802.11 radio with a bit rate of 2Mbps and a transmission range of 250m. The mobility scenarios were generated using the modified random way point mobility model [49]. For all simulations, the nodes move with a speed distributed uniformly at random between 1 and 20 m/s, and a pause time of 0 second is chosen. The simulation duration is 500s and the node density is 20 nodes per radio range. For multicast traffic, a source generates 512-byte packets at a constant rate of 2 packets/second. HRPM uses geographic forwarding with a beacon period of 4 seconds. Nodes send a LOCATION UPDATE after every movement of 100m.

Since HRPM is the first location-based multicast proposed for large groups, we compare it to ODMRP [27], a non-location based mesh multicast protocol well suited to operate in large groups and widely used in multicast protocol studies. We used the Glomosim implementation of ODMRP with parameters set to the values specified by authors in [26]. We also implemented flooding-based multicast (FLOOD) [33] in Glomosim for comparison. We also compare HRPM with a non-hierarchical version of HRPM (RPM) as a representative of the previously proposed location-based multicast protocols that are not hierarchical. In all the sections, unless otherwise specified, HRPM adjusts the decomposition index d to the group size based on the equations in Section 4, using $\omega = 20\% \cdot PacketSize$.

The multicast protocols are evaluated using the following metrics:

- Multicast Delivery Ratio (MDR): Fraction of multicast data packets originated by the source that are received by the receivers.
- Forwarding Cost (FC): Average number of data packet transmissions per delivered data packet to a receiver.
- Control Overhead: Number of control packets transmitted by the multicast protocol.
- Normalized Encoding Overhead (NEO): Ratio of the total number of encoding bytes[2] transmitted *at every hop* (including in the data packets finally not received) to the total number of data bytes received at the final destinations.

[2] bytes used for encoding of destinations and/or locations

- Average Delivery Latency (Delay): Packet delivery latency averaged over all of the multicast packets delivered to all receivers.

5.2 Impact of Decomposition Index d

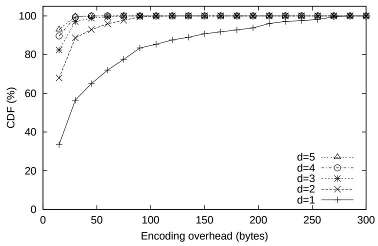
In this section, we study the impact of the decomposition index d. We use a network of 500 nodes in a terrain of 2300mx2300m with 1 multicast group and 1 source. The HRPM hierarchy has two levels as there is no need for more levels as discussed in Section 4. To evaluate the impact of different values of d on a given group size, HRPM's dynamic adjustment of d is disabled. Instead, the decomposition index d is progressively assigned values of 1, 2, 3, 4 and 5 which divide the network into 1, 4, 9, 16 and 25 cells, respectively. We first evaluate the savings in encoding overhead for a small group (25 members) and a large group (125 members). We then evaluate the impact of d on multicast performance with the group size ranging from 25 to 250 members.

Figures 3(a) and 3(b) depict the CDF of encoding overhead for all data packets transmitted. HRPM with $d = 1$ (no hierarchy) is equivalent to a small-group location-based multicast protocol (RPM) and suffers the long tailed distribution of encoding overhead. Additionally, these large packets are near the source, making the source a hot spot of congestion.

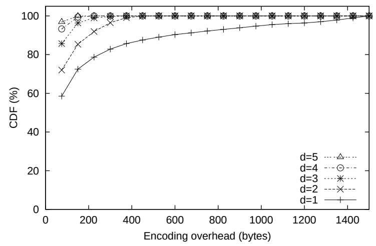
As d increases, the number of packets that have large encoding overhead decreases sharply. This occurs due to the reduction in encoding when a hierarchy is introduced. Since the largest value of d used is 5, the $Src \rightarrow AP$ tree has low encoding overhead, i.e., less than 4 bytes. The maximum encoding overhead inside each AP cell is d^2 times smaller than that at the RP in the non-hierarchical case. This explains the short tailed distribution of encoding overhead for larger values of d. The significantly reduced encoding overhead also reduces the hot spot around the source.

Figure 3(c) depicts the normalized encoding overhead (NEO) as d is increased. Note that HRPM with d=1 (RPM) cannot support more than 125 members and hence we do not show any data points for larger groups. This occurs because the packet size grows beyond the 802.11 MAC layer threshold in Glomosim (2346 bytes) beyond which MAC fragmentation is required. This fragmentation feature is not supported in Glomosim. As predicted by the analysis, for large groups, the NEO is reduced significantly as d is increased. For a group size of 125 members, the NEO is reduced from 41% to 4% as d is varied from 1 to 5, a saving of 37%. More significantly, these savings are achieved at the cost of minimal increase in forwarding cost (FC) (Figure 3(d)) and no reduction in MDR. The MDR is not depicted since it remains close to 100% with varying d for all group sizes.

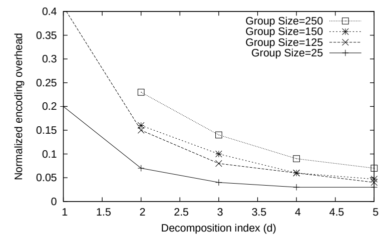
Figures 3(d) and 3(e) show that as d is increased, both the forwarding cost (FC) and the delay of HRPM increase very slowly compared to the non-hierarchical version for large groups (with 125 members or more). This is because for large groups, the detours to the APs are not as costly since a packet needs to travel many hops within the cell to reach multiple nodes anyway. For the small group (25



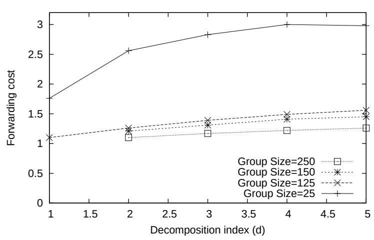
(a) Group size 25



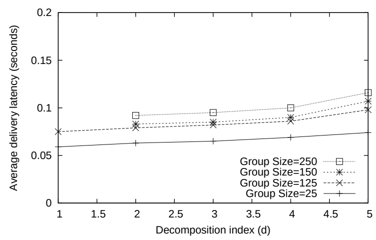
(b) Group size 125



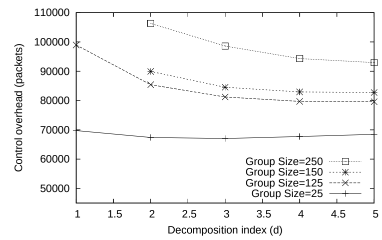
(c) Normalized encoding overhead



(d) Forwarding cost



(e) Average delivery latency (s)



(f) Control overhead (packets)

Figure 3: Impact of the decomposition index d in HRPM.

members), the increase in FC is more significant as d is increased. This is because the group members are sparser, but HRPM always sends packets to APs first which then forward the packets to the few group members in their corresponding cells. A similar effect is expected even in large groups when d is increased to an extent that the group members in each cell are sparse. Thus, the choice of d trades off NEO with FC and delay.

Figure 3(f) shows that the control overhead decreases as d is increased. This overhead is dominated by a constant number of beacon packets required for geographic forwarding. The remaining overhead are from HRPM's control packets including JOIN, LOCATION UPDATE, and HANDOFF, with LOCATION UPDATE packets dominating the others. In non-hierarchical multicast, the updates travel to the RP whereas in a hierarchy, the more frequent member-to-AP updates travel shorter distance to the nearest APs. Further, the aggregated updates from AP to RP which travel the longer distance are less frequent (as discussed in Section 3.1.3). Thus, the overhead of HRPM is lower than that of RPM. Also, as d increases, the member-to-AP updates travel shorter distances to the APs and thus the control overhead reduces further. Although HANDOFF packets happen at all APs in a hierarchy in addition to at the single RP, this overhead is overshadowed by update.

5.3 Impact of Group Size

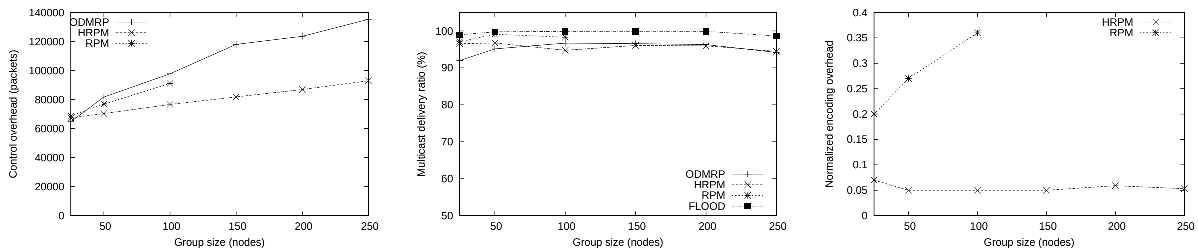
In this section, we study the impact of the group size on the protocol performance. We vary the group size from 50 to 250 members in a 500-node network with 1 multicast group and 1 source. The source sends data packets at 2 packets/second.

Figure 4(a) shows that the control overhead of HRPM is lower than those of ODMRP and RPM across all group sizes with the gap widening as the group size increases. FLOOD does not have any control overhead as the protocol directly floods the data packets. ODMRP requires the source to periodically flood JOIN REQUEST messages. Each member node sends a JOIN TABLE packet in response to form a forwarding group (mesh) for the delivery of data packets. Thus, as the number of group members increases, the JOIN TABLES increase, thereby increasing the overhead of ODMRP. Similarly, the overhead of HRPM increases with group members due to the increase in the updates. However, HRPM builds a virtual hierarchy and performs group management without incurring any flooding cost due to the use of geographic hashing and thus has a lower overhead than ODMRP. In fact, the overhead of HRPM/RPM is dominated by *beaconing* required for geographic forwarding. Beaconing incurs a constant overhead of 62,000 packets for all the group sizes depicted in the graph and the actual protocol overhead of HRPM/RPM is a smaller fraction of the total overhead (8% at 25 nodes and 33% at 250 nodes). As explained earlier, aggregation of LOCATION UPDATE at the APs in HRPM reduces its overhead compared to RPM.

Figure 4(b) shows that FLOOD achieves the highest MDR of all the protocols. HRPM, ODMRP and RPM also achieve close to 100% MDR for all the group sizes. Note that MDR for ODMRP for small group sizes is slightly lower due to a sparse forwarding mesh.

The encoding overhead (Figure 4(c)) of HPRM remains steady as the group size increases since it adjusts the d value to the varying group sizes. Note that although ω (encoding overhead constraint) is chosen to be 20% of the packet size, the average encoding overhead of HRPM is always below 7% (with an average of 5.5%). The encoding overhead of RPM significantly increases as the group size increases. Note that ODMRP and FLOOD do not encode destinations/locations in the data packet.

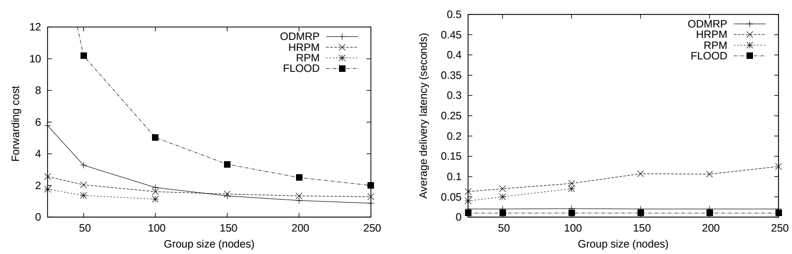
The next two performance metrics, forwarding cost (FC) (Figure 4(d)) and delay (Figure 4(e)), are affected by the tree construction algorithm used. Since HRPM constructs bandwidth-minimizing trees, it has the desirable property of a much lower FC than FLOOD and ODMRP for small groups, in which a large number of non-member nodes are part of the mesh resulting in higher FC. However, as the group size increases, more and more member nodes become part of the forwarding mesh, which lowers the FC of FLOOD and ODMRP. Note that as the group size becomes large and group members become dense, ODMRP achieves an FC lower than 1 due to multicast advantage. Despite using a virtual hierarchy, the FC of HRPM is very close to that of RPM as the group size increases which shows the HRPM hierarchy



(a) Control overhead (packets)

(b) MDR (%)

(c) Normalized encoding overhead



(d) Forwarding cost

(e) Average delivery latency (s)

Figure 4: Impact of the group size. RPM cannot support more than 125 group members due to its large incurred encoding overhead.

does not cause significant detours in routing data packets.

The delay of FLOOD is the lowest as expected (shown in Figure 4(e)). Further, due to the detour in the hierarchy approach, HRPM has a slightly higher delay than RPM. In contrast, ODMRP has a lower delay than both RPM and HRPM for the following three reasons. First, ODMRP can deliver multiple packets with a single transmission (wireless multicast advantage). Second, ODMRP uses unreliable broadcast thus avoiding the cost of RTS/CTS channel access. However, this mechanism becomes increasingly unreliable with increased network size, number of groups, number of sources, etc. RPM/HRPM use reliable unicast delivery which incurs higher delay. Third, RPM/HRPM use bandwidth-minimizing overlay minimum spanning trees for data delivery. However, since any tree construction algorithms can be used under the HRPM/RPM framework without affecting overhead, delay-minimizing trees such as LGK trees [7] could be potentially used to reduce the delay. Since FC and delay are affected by the tree construction algorithm and other factors mentioned above, and such metrics are less meaningful when one protocol achieves low delivery ratio, the rest of the performance comparison focuses on the control overhead and the delivery ratio.

Multiple sources We also performed simulations increasing the number of sources in the group to 5 for a small group (25 members) and a large group (150 members). The results are summarized in Table I. ODMRP requires each source to refresh forwarding state in the network periodically to deal with

Table 1: Impact of increasing the number of sources to 5.

Group Size	25		150	
	HRPM	ODMRP	HRPM	ODMRP
Overhead	69,103	340,477	87,728	580,970
MDR	98.66	98.46	97.78	97.80
FC	2.47	11.7	1.45	2.7
Delay	0.07	0.02	0.09	0.03

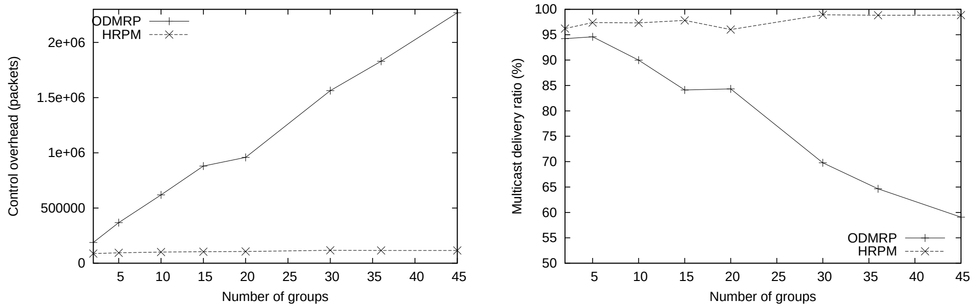
mobility and build the data delivery mesh. Thus its overhead grows significantly with the number of sources. HRPM allows each source to build a virtual tree with almost no extra cost – it just needs to hash the active APs based on the group vector retrieved from the RP. Thus the overhead of HRPM grows very slowly as the number of sources increases. Compared to Figure 4(a), the overhead of ODMRP increases by 425% for the group of size 25 and by 392% for the group of size 150 when the number of sources is increased from 1 to 5, while the overhead for HRPM only increased by 2.5% and 6.3%, respectively. Note HRPM achieves significant overhead reductions while delivering comparable numbers of packets as ODMRP for both group sizes.

In summary, HRPM scales well with the group size unlike non-hierarchical protocols (e.g. RPM) which cannot function beyond a certain group size (e.g. 125 members). Due to adaptive hierarchy construction, it maintains the encoding overhead below 7% as the group size scales. It delivers comparable percentages of packets as ODMRP across a range of group sizes while incurring lower overhead. For a fixed group size, as the number of sources increase, HRPM's overhead only increases slightly while ODMRP suffers a large increase in overhead.

5.4 Impact of Number of Groups

In this section, we study the impact of the number of groups on protocol performance. We consider a 500-node network in an area of 2300mx2300m. As the number of groups is increased, the group size is adjusted to keep the total number of receivers constant. We consider several scenarios varying NxG where N is the number of groups and G is the group size. The configurations are: 2x90, 5x36, 10x18, 15x12, 20x9, 30x6, 36x5 and 45x4 with each scenario having 180 receivers, and each group having 1 source.

Figure 5(a) shows that the overhead of HRPM grows very slowly as the number of groups is increased. This is because the update overhead of HRPM does not increase with the number of groups in the network. In contrast, ODMRP's overhead increases significantly as the numbers of groups increases. Increasing the number of groups results in the sources of different groups competing to broadcast JOIN



(a) Control overhead (packets)

(b) MDR (%)

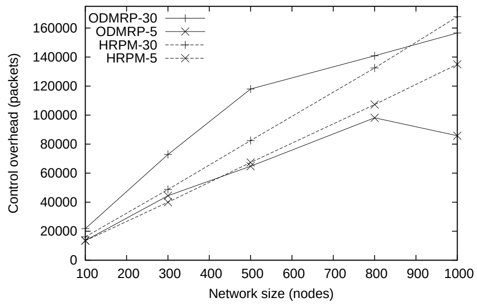
Figure 5: Impact of the number of groups.

REQUEST messages. This causes congestion and results in a drop in MDR to below 60% for ODMRP (Figure 5(b)). In contrast, HRPM consistently delivers 95% or more data packets.

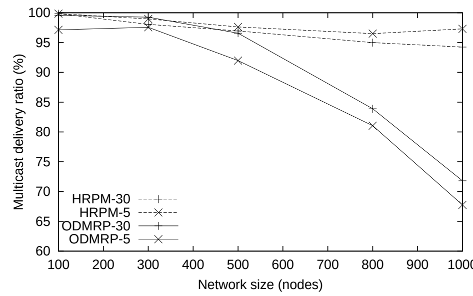
5.5 Impact of Network Size

In this section, we evaluate how the network size affects the multicast protocol performance. We vary the network size from 100 to 1000 nodes while keeping the density constant at 20 nodes per radio range as before. For each network size, we consider group sizes of 5% and 30% of nodes in the network, respectively. To isolate the effect of the network size from the effect of multiple sources or multiple groups, for each network size we consider only 1 group with 1 source.

Figure 6(b) shows that the MDR of ODMRP drops significantly as the network size increases for both small and large groups. As the network size increases, the flooding-based mesh construction of ODMRP becomes increasingly costlier and unreliable. Additionally the group members are spread further apart leading to higher probabilities of failures. In contrast, HRPM delivers more than 95% of the packets for the smaller groups and close to 95% of the packets for the larger groups across a wide range of network sizes. Additionally, as shown in Figure 6(a), HRPM incurs comparable overhead as ODMRP for small groups and lower overhead than ODMRP for large groups except for a 1000-node network. At such a large network size, the JOIN REQUEST for ODMRP do not reach all the members, thereby reducing the number of JOIN TABLES sent. This causes a reduction in overhead as well as MDR. In summary, HRPM is more scalable than ODMRP for both small and large groups as the network size increases. Note that ODMRP is expected to further degrade in comparison to HRPM if the number of sources or groups are increased in large networks as suggested by results in the previous sections.



(a) Control overhead (packets)



(b) MDR (%)

Figure 6: Impact of the network size.

6 Related Work

HRPM is closely related to previous location-based multicast protocols. In addition, it is related to stateless non-location-based multicast protocols, hierarchical non-location-based multicast protocols, and other uses of geographic hashing.

Location-based Multicast Protocols Previous location-based protocols [2, 7, 32] were proposed for small groups due to the constraint of encoding either the entire tree or the destinations in the data packet headers. In DSM [2], each node floods its location in the network. DSM constructs a physical Steiner tree using the TM heuristic [42] at the source, optimally encodes the physical multicast tree into each packet, and delivers the packet using source routing. LGT [7] requires each group member to know every other group member's location. LGT proposes two overlay multicast trees: a bandwidth-minimizing LGS tree and a delay-minimizing LGK tree. PBM [32] does not explicitly construct trees but rather relies on a multicast geographic forwarding strategy similar to the hop-by-hop forwarding proposed by SGM [3] and DDM [20].

The SPBM protocol proposed in [43] is closely related to HRPM as the two share the essence of improving the scalability of location-based multicast using hierarchical group management. However, HRPM is different from SPBM in several fundamental aspects: (1) SPBM uses *flooding* in hierarchical group management. In contrast, HRPM uses *mobile geographic hashing* (convergence to the rendezvous point) in hierarchical group management which does not incur any flooding cost. (2) SPBM defines a static hierarchy by dividing the network into a quad-tree with a predetermined maximum aggregation level L. In contrast, HRPM uses a per-group hierarchy that dynamically adjusts to the group membership. (3) The need to propagate the group member's *location information* up the hierarchy while exploiting the distance effect (so that flooding to remote squares in the grid is less frequent) makes SPBM more susceptible to node mobility due to the delayed propagation of information when a node's position in the hierarchy changes. For example, when a group member enters a large square that did not previously contain a group member, it will not receive any packets until the membership in the square has been spread to reach the source. In contrast, in HRPM, the locations of APs are “virtual”, and hence fixed, and the low-overhead rendezvous-based group management allows an AP to update the RP as soon as its *membership* changes. Within each leaf cell, the AP keeps track of each group member’s last known location, which allows packets to be forwarded to group members despite them moving into a new cell, since geographic forwarding is resilient to slight variations in destination locations. Experimentally, the evaluation of SPBM [43] shows that SPBM degrades in PDR and becomes similar to ODMRP as the maximum node speed increases to 15 m/s with pause time 10s. In contrast, all the simulations in our study were carried out at a higher maximum node speed of 20 m/s with pause time 0s and HRPM consistently outperforms ODMRP.

Note that our evaluation of HRPM uses a similar node density as in the SPBM evaluation [43] although with a packet size 8 times larger and twice the traffic volume. Moreover HRPM has been evaluated by scaling a much wider range of network parameters such as the group size, number of groups, number of senders, and network size. The magnitude of the parameters studied are also different. While SPBM has been shown to work well for 25 group members, HRPM has been evaluated for up to 250 group members. Similarly, while SPBM has been evaluated for 196 nodes, HRPM has been evaluated for up to 1000 nodes.

Stateless Multicast Protocols Stateless multicast protocols have been proposed to reduce state at forwarding nodes by encoding multicast destinations in headers and are typically used for small groups. The work in SGM [3] proposed this technique for the Internet. REUNITE [39] requires only branch point routers to keep state for IP multicast. DDM [20] uses similar principles to provide stateless multicast in MANETs.

Hierarchical Multicast Protocols Several hierarchical non-location-based protocols have been proposed which can be overlay or non-overlay based. Protocols such as AMRIS [46] and PAST-DM [15] propose an overlay-based approach in which the overlays are a form of hierarchies.

An example of a non-overlay hierarchical MANET multicast protocol is HDDM [16] which extends DDM to include a hierarchical structure. Similar to HDDM, HRPM also leverages the well known technique of introducing a hierarchical structure to reduce overhead. Despite this similarity, HDDM is a topology-aware approach while HRPM is a location-aware approach. Thus, the design challenges and issues in both protocols are very different. HRPM needs to provide location management and routes using locations rather than topology. The focus of our paper was to improve the scalability of location based multicast and so a comparison with HDDM is out of the scope of this work. Note that our evaluation of HRPM scales parameters (network size, group size, number of groups) to larger values than previous work on scalable multicast.

The work in [31] proposed the use of *cores* to reduce control traffic for creating multicast delivery structures. They propose that group members form a multicast group by sending join requests to a set of cores. Rendezvous points are similar in concept to core nodes. However RPs/APs in HRPM can be located without any overhead using geographic hashing and can be more resilient to mobility due to not being tied to a particular node whose movement needs to be tracked.

Location Management for Unicast Several location management protocols have been proposed for *unicast* services [23, 1, 25, 17, 41, 13, 40, 45, 28, 48, 18, 9]. However, location management for unicast is fundamentally different from that for multicast such as in HRPM since it does not need to provide locations of an entire group to the source node.

Other Uses of Geographic Hashing HRPM shares the concept of geographic hashing with GHT [36] for data-centric storage systems and consistent hashing in distributed indexing such as DIM [29] and DIFS [14] for supporting range queries in sensor networks. These protocols were proposed for use in static sensor networks and do not have to deal with mobility. Geographic hashing has also been used in location services (e.g., [13, 40, 45, 9]) in which each node's identifier is hashed to a home region consisting of one or more nodes which serve as that node's location servers.

Geographic Routing in Wireless Networks HRPM uses geographic forwarding to forward data and control packets. The first proposal for geographic forwarding was laid out in [11]. Subsequently detailed algorithms have been designed for the application of geographic forwarding in wireless networks ([5],[21],[24]). Many optimizations and modifications have been proposed for these basic algorithms to deal with real network topology and provide robustness. Work has also been done on real testbeds to identify and remove pathologies that arise in geographic routing [34]. Geographic forwarding in HRPM is not restricted to one single proposal and can potentially take advantage of new and more efficient techniques developed for geographic routing.

Another body of work proposes to use geographic routing without actual node location information [35], typically achieved by using localization algorithms to assign nodes virtual coordinates and then route geographically using these virtual coordinates. Such schemes can be used in HRPM.

7 Conclusions

In this paper, we propose the Hierarchical Rendezvous Point Multicast protocol (HRPM) which leverages two techniques: distributed mobile geographic hashing and hierarchical decomposition of large multicast groups to improve the scalability of location-based multicast. Together, the two techniques enable lightweight hierarchical membership management which reduces per-packet encoding overhead without incurring the high cost associated with maintaining distributed state at any particular mobile nodes.

Our simulation results show that HRPM significantly improves the scalability of location-based multicast in terms of the group size. Coupled with its leverage of stateless geographic forwarding, HRPM scales well in terms of the group size, the number of groups, as well as the size of the network. In particular, HRPM maintains close to 95% multicast delivery ratio while incurring on average 5.5% per packet tree-encoding overhead for up to 250 group members in a 500-node network. Furthermore, it achieves a steady 95% delivery ratio while incurring nearly constant overhead as the number of groups increases from 2 to 45, while keeping the total number of receivers constant at 180, in a 500-node network. Lastly, it steadily achieves above 90% delivery ratio as the network scales up to 1000 nodes with up to 30% group members.

For future work, we plan to study the impact of new tree construction algorithms, the use of broadcast which does not require RTS/CTS, and exploiting wireless multicast advantage on different data delivery performance metrics, such as delay, forwarding cost, and delivery ratio. We are also interested in evaluating manycast and anycast services using HRPM.

Acknowledgments

This work was supported in part by NSF grant ANI-0338856.

References

[1] S. Basagni. et al., A distance routing effect algorithm for mobility (DREAM). In *Proc. of ACM MobiCom*, October 1998.

[2] S. Basagni, I. Chlamtac, and V. Syrotiuk. Location aware, dependable multicast for mobile ad hoc networks. *Computer Networks*, 36:659–670, August 2001.

[3] R. Boivie, N. Feldman, and C. Metz. Small group multicast: A new solution for multicasting on the internet. *IEEE Internet Computing*, 4(3):75–79, 2000.

[4] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. In *Proc. of ACM DialM Workshop*, August 1999.

[5] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wirel. Netw.*, 7(6), 2001.

[6] C. Carter, S. Yi, P. Ratanchandani, and R. Kravets. Manycast: exploring the space between anycast and multicast in ad hoc networks. In *Proc. of ACM MobiCom*, 2003.

[7] K. Chen and K. Nahrstedt. Effective Location-Guided Tree Construction Algorithms for Small Group Multicast in MANET. In *Proc. of IEEE INFOCOM*, June 2002.

[8] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. The MIT Press, Cambridge, MA, 1990.

[9] S. M. Das, H. Pucha, and Y. C. Hu. Performance comparison of scalable location services for geographic ad hoc routing. In *Proc. of IEEE INFOCOM*, March 2005.

[10] C. de Morais Cordeiro, H. Gossain, and D. P. Agrawal. Multicast over wireless mobile ad hoc networks: Present and future directions. *IEEE Network*, 17(1), January/February 2003.

[11] G. Finn. Routing and addressing problems in large metropolitan-scale internetworks. Technical report, Technical Report ISI/RR-87-180, USC/ISI, March 1987.

[12] E. N. Gilbert and H. O. Pollak. Steiner minimal trees. *SIAM Journal of Applied Math.*, 16:1–20, 1968.

[13] S. Giordano and M. Hami. Mobility management: The virtual home region. Technical Report SSC/037, EPFL, October 1999.

[14] B. Greenstein. et al., DIFS: A distributed index for features in sensor networks. In *Proc. of IEEE SNPA*, May 2003.

[15] C. Gui and P. Mohapatra. Effi cient overlay multicast for mobile ad hoc networks. In *Proc. of IEEE WCNC*, March 2003.

[16] C. Gui and P. Mohapatra. Scalable multicasting for mobile ad hoc networks. In *Proc. of IEEE INFOCOM*, March 2004.

[17] Z. J. Haas and B. Liang. Ad Hoc Mobility Management with Uniform Quorum Systems. *IEEE/ACM Trans. Net.*, 7(2):228–240, April 1999.

[18] P. Hsiao. Geographical Region Summary Service for Geographical Routing. *ACM MC2R*, 5(4):25–39, January 2002.

[19] J. G. Jetcheva and D. B. Johnson. Adaptive Demand-Driven Multicast Routing in Multi-Hop Wireless Ad Hoc Networks. In *Proc. of ACM MobiHoc*, October 2001.

[20] L. Ji and S. Corson. Differential Destination Multicast–A MANET Multicast Routing Protocol for Small Groups. In *Proc. of IEEE INFOCOM*, April 2001.

[21] B. Karp and H. Kung. GPSR: Greedy perimeter stateless routing for wireless networks. In *Proc. of ACM MobiCom*, August 2000.

[22] L. Kleinrock and F. Kamoun. Hierarchical routing for large networks: performance evaluation and optimization. *Computer Networks*, 1:155–174, 1977.

[23] Y.-B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. In *Proc. of ACM MobiCom*, October 1998.

[24] F. Kuhn, R. Wattenhofer, and A. Zollinger. Worst-case optimal and average-case effi cient geometric ad-hoc routing. In *Proc. of ACM MobiHoc*, June 2003.

[25] V. Kumar and S. R.Das. Performance of dead reckoning-based location service for mobile ad hoc networks. *Wireless Communications and Mobile Computing Journal*, December 2003.

[26] S.-J. Lee. et al., A Performance Comparison Study of Ad Hoc Wireless Multicast Protocols. In *Proc. of IEEE INFOCOM*, March 2000.

[27] S.-J. Lee, M. Gerla, and C.-C. Chiang. On-Demand Multicast Routing Protocol. In *Proc. of IEEE WCNC*, September 1999.

[28] J. Li. et al., A scalable location service for geographic ad hoc routing. In *Proc. of ACM MobiCom*, August 2000.

[29] X. Li. et al., Multi-dimensional range queries in sensor networks. In *Proc. of ACM SenSys*, November 2003.

[30] J. Luo, P. T. Eugster, and J.-P. Hubaux. Route Driven Gossip: Probabilistic Reliable Multicast in Ad Hoc Networks. In *Proc. of IEEE INFOCOM*, March 2003.

[31] E. Madruga and J. Garcia-Luna-Aceves. Scalable Multicasting: The Core Assisted Mesh Protocol. *ACM/Baltzer Mobile Networks and Applications, Special Issue on Management of Mobility in Distributed Systems*, 6(1), 2001.

[32] M. Mauve, H. Fler, J. Widmer, and T. Lang. Position-based multicast routing for mobile ad-hoc networks. Technical Report CS TR-03-004, University of Mannheim, 2003.

[33] K. Obraczka, G. Tsudik, and K. Viswanath. Pushing the limits of multicast in ad hoc networks. In *Proc. of ICDCS*, April 2001.

[34] G. R. M. Practical. Young-jin kim and ramesh govindan and brad karp and scott shenker. In *Proc. of USENIX NSDI*, May 2005.

[35] A. Rao, C. Papadimitriou, S. Shenker, and I. Stoica. Geographic routing without location information. In *Proc. of ACM MobiCom*, September 2003.

[36] S. Ratnasamy. et al., GHT: A geographic hash table for data-centric storage in sensornets. In *Proc. of ACM WSNA*, September 2002.

[37] E. M. Royer and C. E. Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. In *Proc. of MobiCom*, August 1999.

[38] P. Sinha, R. Sivakumar, and V. Bharghavan. MCEDAR: Multicast Core-Extraction Distributed Ad hoc Routing. In *Proc. of IEEE WCNC*, September 1999.

[39] I. Stoica, T. S. E. Ng, and H. Zhang. REUNITE: A recursive unicast approach to multicast. In *Proc. of IEEE INFOCOM*, March 2000.

[40] I. Stojmenovic. Home region based location updates and destination search schemes in ad hoc wireless networks. Technical Report TR-99-10, SITE, University of Ottawa, September 1999.

[41] I. Stojmenovic. A routing strategy and quorum based location updte scheme for ad hoc wireless networks. Technical Report TR-99-09, SITE, University of Ottawa, September 1999.

[42] H. Takahashi and A. Matsuyama. An approximate solution for the steiner problem in graphs. *Mathematica Japonica*, 24:573–577, 1980.

[43] M. Transier, H. Fuler, J. Widmer, M. Mauve, and W. Effelsberg. Scalable Position-Based Multicast for Mobile Ad-hoc Networks. In *Proc. of the First International Workshop on Broadband Wireless Multimedia:Algorithms, Architectures and Applications (BroadWim)*, October 2004.

[44] USCG Navigation Conter GPS page. `http://www.navcen.uscg.nil/gps/default.html`, January 2000.

[45] S.-C. M. Woo and S. Singh. Scalable routing protocol for ad hoc networks. *Wireless Networks*, 7(5):513–529, 2001.

[46] C. Wu and Y. Tay. AMRIS: A Multicast Protocol for Ad hoc Wireless Networks. In *Proc. of MILCOM*, November 1999.

[47] J. Xie, R. R. Talpade, A. Mcauley, and M. Liu. AMRoute: ad hoc multicast routing protocol. *Mob. Netw. Appl.*, 7(6):429–439, 2002.

[48] Y. Xue, B. Li, and K. Nahrstedt. A scalable location management scheme in mobile ad-hoc networks. In *Proc. of IEEE LCN*, November 2001.

[49] J. Yoon, M. Liu, and B. Noble. Random waypoint considered harmful. In *Proc. of IEEE INFOCOM*, April 2003.

[50] X. Zeng, R. Bagrodia, and M. Gerla. Glomosim: A library for parallel simulation of large-scale wireless networks. In *Proc. of PADS Workshop*, May 1998.