University of Southampton
Faculty of Engineering and Applied Science
Department of Electronics and Computer Science

Ad-Hoc routing protocols for large-scale wireless networks

9 Month Report

By Michael Saywell

Supervisor: Prof. David De Roure

October 1, 2003

Supported by a Doctorial Training Award as part of the Equator Interdisciplinary Research Collaboration, Engineering and Physical Sciences Research Council, Grant number GR/N15986/01

Contents

1 Introduction

Wireless networking and in particular Wi-Fi[1] or 802.11b[12] have become increasingly popular technologies of late with hundreds of commercial stand-alone hot-spots being setup in many major cities. Community wireless networks such as those affiliated with FreeNetworks.org[2] are using the same technology to create free to use wide area wireless networks, bypassing the traditional wired local loop and allowing communication on a peer to peer basis.

Such networks introduce challenges in their design at the radio layer and in their requirements for routing, this are the areas I intend to investigate with my research.

2 Background

2.1 Mesh Networks

In his paper[13] Shepard introduces the concept of a rooftop network, such networks use low power wireless equipment placed on the rooftops of many buildings to communicate with other nearby nodes as shown in figure 1. Nodes forward traffic co-operatively for each other, thus it's possible for communication to occur between any 2 connected nodes on the network even though they may not have a direct connection with one-another. This is the area in which I initially focused my research, I intended to see if such networks were practical and if so what routing protocols would be suitable for use in such a network.

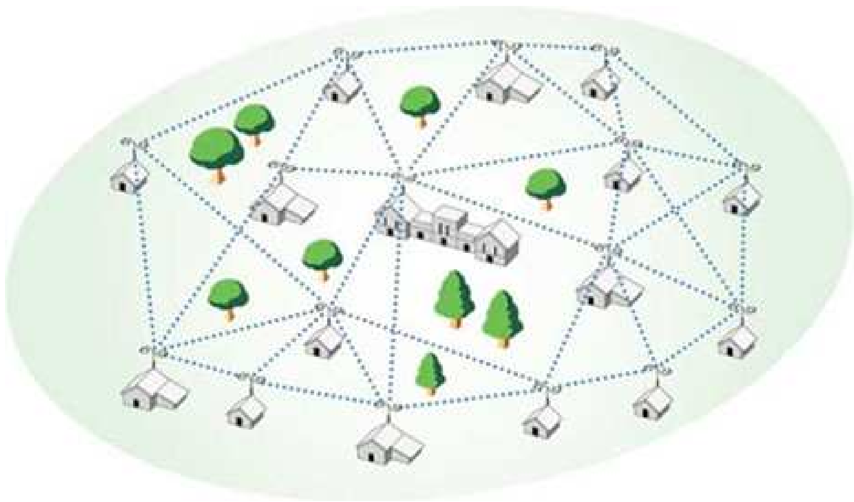

Figure 1: An example of a mesh network

[1] The Wi-Fi alliance, http://www.weca.net/

[2] FreeNetworks.org is a voluntary cooperative association dedicated to education, collaboration, and advocacy of the creation of free digital network infrastructures. `http://freenetworks.org`

2.2 802.11b

802.11b utilises the 2.4GHz spectrum for radio communication, this a public frequency in many countries and is typically limited to low power usage, in the UK the maximum transmit power is 100mW EIRP (effective transmit power after aerial gain). Naturally this severely restricts the range of such equipment, especially when there is no direct line of sight between stations, typical distances are 2km line of sight or up to 500m otherwise depending on the obstructions.

802.11b is typically deployed as one more more access points connected to a single wired backbone network, wireless clients connect to the access point with the strongest signal and can freely migrate between APs with no loss of connectivity. There is no direct communication between the access points via the wireless medium, handovers are handled via the wired backbone.

Obviously this is a radically different deployment to the rooftop network described previously, however the 802.11 specification includes 2 other approaches which may be more suitable.

The first is Ad-Hoc mode. This is a standard feature present in client adapters, it is designed to allow communication to occur directly between clients without a co-ordinating access point. The radio layer assumes that all clients are within range of each other and thus does not re-transmit packets, however it is possible to over-come this restriction by forcing hosts to re-transmit at the IP layer. Ad-Hoc mode is not ideally suited for large networks though as it's nodes attempt to form a virtual network known as an IBSS. Each IBSS has an ID taken from the MAC address of the station which created it, nodes which join the IBSS use this ID to identify it and provided that at least 1 node remains running the ID will continue to exist. Thus issues arrive with the merging of multiple IBSS networks, for example when 2 initially disconnected networks become linked by a new host then they will not merge as one might expect, instead the new host must choose one network or the other.

The alternative to ad-hoc is wireless bridging, this is present in most access points and is often used to connected 2 wired networks together via a point to point wireless bridge. 2 basic variations are supported, point to point and point to multi-point, only access points in point to point mode can connect to those in multi-point mode. However typically when an access point is in bridging mode it generally ceases to function as an access point, that is wireless clients can no longer connect to it, although it should be noted that this is a software limitation not a protocol one.

2.3 Community Wireless

Community wireless is a movement within networking akin to open source in the software industry, it involves using wireless to create open, free to use, multi-hop meshed networks. Currently most community wireless efforts are concerned with sharing broadband Internet connections, this is particularly popular in rural areas where cheap broadband is not available and wireless can be used to share a single leased line between many homes.

In cities and towns there is less motivation since broadband connectivity is already widely available, however if rooftop networks could provide an alternative wireless backbone of several MBit this would entice many more users. Potential uses of such networks include providing free telephony via VoIP, free Internet access for students via a University VPN and pervasive wireless connectivity on the street and in public areas.

2.4 Traditional IP routing

Routing protocols require a strict hierarchical architecture in order to scale efficiently, a typical example would be the aggregation between Department, Organisation, ISP and finally the global Internet. Without such aggregation the Internet could not function, as of July 2003 there are 143,033 routing prefixes announced on the Internet [3] compared to an estimated 171,638,297 hosts[4].

It should be noted that many still consider the current routing table to be un-acceptably large, consequently with IPv6 additional tiers of connectivity are enforced to try and encourage more levels of aggregation and thus less routing overhead.

3 Related work

3.1 Location Based Addressing

The use of geographic data for routing is a well established mechanism for dealing with large scale networks. It was first discussed by Gregory Finn in 1987[5] as a mechanism for dealing with Metropolitan scale networks by using latitude and longitude as components of a hosts network address. Given the wide use of IP in todays networks such a drastically different addressing mechanism would likely be un-practical, however convergence of geographic addressing with IP and in particular IPv6 is briefly discussed in [7], IPv6 addresses are sufficiently large that it's possible to encode the location data in the address itself.

In his Internet Draft[6] Tony Hain demonstrates this very principle, showing how geographic locations can be used to form a provider independent address. As an example 20 bits of address space can be used to cover a region of approx 26 square km at a resolution of 6.4m square. The mechanism detailed also makes use of bit interleaving, meaning that shorter prefixes cover large areas, as more bits are used the accuracy increases. For example 16 bits refer to an area $104km^2$ in size, 32 bits $407m^2$ and 44 bits just $6.4m^2$.

3.2 Cartesian Routing

In his work on Metropolitan area networks Finn envisaged a scenario where city blocks were linked to each other directly rather than via the Internet and an ISP as is

[3] As of July 2003, see `http://www.cidr-report.org/`

[4] Based on the number of DNS hostnames as of Jan 2003, see `http://www.isc.org/ds/WWW-200301/index.html`

commonplace today. Although Finn was considering wired links the basic principles of geographic forwarding that he outlines are still applicable to wireless medium.

His proposal was that packets are forwarded greedily, that is to a neighbour nearer to the destination than the current node, until no such node exists or the packet has reached it's destination. If no closer node exists then it is assumed that there is an obstacle and the current node floods the packet to all of it's neighbours until a node with a neighbour closer to the destination is reached and greedy forwarding can be resumed, as shown in figure 2.

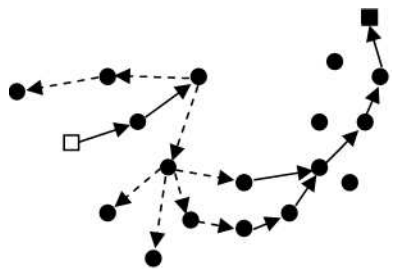

Figure 2: Flooding in Cartesian Routing

Flooding is an inefficient solution as it wastes bandwidth not only by the nature of the flood itself but also through duplication of packets when multiple routes around the obstacle are found. The flooding radius diameter is limited to help reduce this, however as a consequence thus the protocol will fail to find a link if the closer node lay outside the flood radius.

3.3 GPSR

In his thesis[10] Brad Carp outlines a more efficient solution to routing around obstacles as part of the routing protocol GPSR. GPSR actively maps outages, creating a planar graph (no crossing edges) of the nodes encircling the void. When a packet is in perimeter node it is forwarded along the edges of this graph until greedy forwarding can be resumed, see figure 3. Nodes on the edges of voids have knowledge about the size and shape of the outage, thus they can intelligently forward packets around the void in the correct direction to minimise the number of hops.

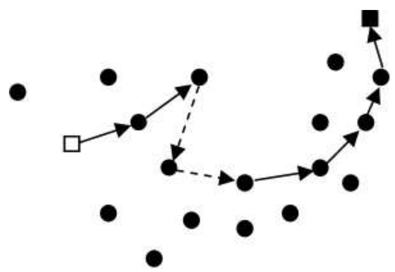

Figure 3: Perimeter forwarding in GPSR

GPSR will never fail to reach a connected node, however it's possible to demonstrate situations where non-optimal routes are chosen, this is particularly the case where networks are sparsely connected as in figure 4.

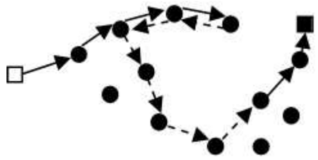

Figure 4: Extra hops incurred by greedy forwarding

GPSR assumes all link speeds are equal, therefore higher capacity links will not be utilised unless they are on the trajectory to the destination. With a wide range of wireless standards of varying speed becoming available it is likely that large scale wireless networks will use links of varying speeds, routing protocols should be able to identify and exploit such links. The same reasoning applies to longer distance links which may exist due to the use of point to point equipment and directional antenna, see figure 5.

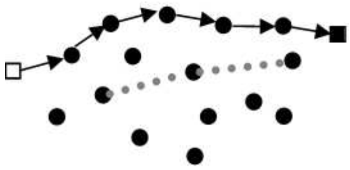

Figure 5: Failure to utilise faster/longer links

The perimeter forwarding model is also inefficient. Shorter routes could be achieved by altering the trajectory of packets before they reach an obstruction which would also avoid network congestion at void edges. Additionally the planar graph created to accurately forward around the perimeter of voids drastically reduces the number of potential links between nodes, therefore packets may incur additional hops when in perimeter mode.

3.4 Fisheye State Routing

Fisheye State Routing (FSR)[8] is based on a link state protocol, that is whenever a link is added to or removed from the network notifications are propagated outward and nodes update their routing tables. To enhance scalability the Fisheye protocol propagates updates slower as the distance from the source increases, consequently routing updates do not flood the entire network, if mobility is high then obsolete data will be dropped at outer nodes due to these forwarding delays. Although distant nodes may have incorrect routing information as packets approach their destination the accuracy of the routing tables increases, compensating for this.

Fisheye is primarily designed as a mechanism to reduce routing traffic in networks with high mobility rates which is not a large issue in rooftop networks, however the design philosophy is an elegant one which may be of use in different contexts.

3.5 Clustering

Clustering has been used in several protocols[3],[8] and is an extension of the clubs algorithm[11]. Clustering is the process by which individual nodes group themselves dynamically, there are several mechanisms to do this, including simply emulating the wireless topology (where clusters represent a group of directly connected nodes), or by some algorithmic method such as that described in [11].

In HSR the clustering process is repeated recursively to form multiple layers of "clusters", Level 0 is the physical layer, Level 1 clusters comprise solely of the Level 0 leaders (cluster heads), Level 2 of the Level 1 leaders and so on, 2 levels of clustering are shown in 6. Clusters are comprised of 2 basic node types, gateway nodes and internal nodes, gateway nodes are present wherever 2 clusters meet, they are responsible for forwarding traffic between the 2 cluster heads.

Addressing in HSR is directly linked to the hierarchical structure and is defined as being the sequence of MAC addresses of nodes that must be traversed to reach the destination node from the highest level.

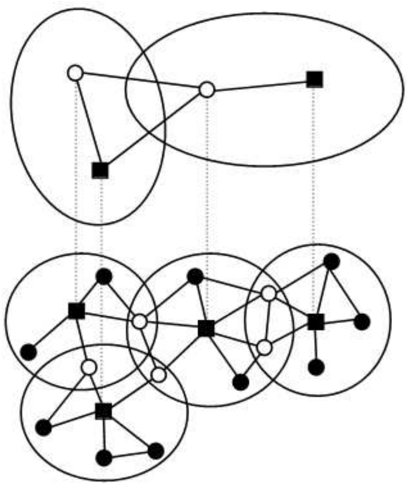

Figure 6: Clustering. Leaders are black squares, and gateways are white circles.

4 Work to date

4.1 Simulation

My initial work concentrated on ways in which GPSR routing could be improved, in particular I wanted to remove the location service through use of IPv6 and deal with voids or outages in a more intelligent way by diverting packets in advance.

In order to test and help prototype my work I would need a simulator, my initial requirements were support of IPv6, ability for nodes to be aware of their location, rapid prototyping and good visualisation. I investigated several available simulators including

HLSIM/Gunk[2, 1], NS-2[4], and GloMoSim[15] but rejected all of them as in one way or another they would have required fairly substantial enhancement to meet my criteria. Instead I choose to develop a basic simulator in Java, although this would be slower for large scale simulations this was not (yet) my main concern, it allowed me to quickly prototype different algorithms and since I was designing from scratch I could easily provide information such as location to nodes on the network.

Using this simulator I began first to try and repeat the work done in GPSR, following this I planned to extend the protocol to address the problems I identified previously. I have successfully implemented basic greedy forwarding without the use of a location service by embedding the location within the IPv6 destination address and had begun to investigate the mapping of voids in order to intelligently route around them.

4.2 Deployment

I have been involved with the deployment of SOWN[5], the Southampton Open Wireless Network, a local community wireless effort. We have developed our own infrastructure based on the rooftop network approach, using an open source Linux driver known as HostAP[6]. For routing we use OSPF [7] (a traditional routing protocol) since our network topology is both static and still relatively small this has sufficed.

4.2.1 Automatic linking

HostAP allows wireless cards based on the prism2 chipset to function as an access point whilst simultaneously linking to neighbours using wireless bridging, this process can be automated allowing new nodes to be added without additional configuration. These links appear as new virtual interfaces to the operating system referred to as WDS interfaces, WDS is the name given to the 802.11 frame type used during wireless bridging.

In order for IP layer communication to occur these interfaces need to be assigned IPs, unfortunately OSPF requires that they are unique and have proper subnet masks, consequently we were unable to simply re-use a single IP and add a host route to the neighbour via the relevant WDS interface. Individual subnets per interface were required, to automate this I implemented a small daemon which uses reverse APR to address the interfaces, /30 subnets are allocated from a pool of addresses unique to that node. The first usable IP is used for the local interface and the second for the remote, the host with the greater MAC address runs a reverse ARP server, the other runs as a client which requests an IP.

Once the addresses has been assigned OSPF becomes active on the interface and links are exchanged with the new peer.

[5]The Southampton Open Wireless Network, `http://www.sown.org.uk`

[6]HostAP, `http://hostap.epitest.fi/`

[7]RFC 2328: OSPF Version 2, `http://www.ietf.org/rfc/rfc2328.txt`

4.2.2 Bandwidth

My involvement in SOWN had gave me an opportunity to test the bandwidth available over wireless networks. At it's widest point, currently 3 hops, we were surprised to see that the throughput was only approximately 1Mbit, compared to the 3 to 4 Mbit available between each hop (note that these are actual throughput, not raw 802.11b speeds).

Further investigation into this shows that the reason for the poor throughput has to do with radio propagation issues and the mechanisms employed by 802.11 to tackle the hidden node problem. The RF spectrum by definition is a shared medium between all hosts, therefore when 1 host is transmitting all others must be silent else there will be interference. The hidden node problem occurs when you have 3 or more nodes, 2 of which are out of range of each other as shown in figure 7.

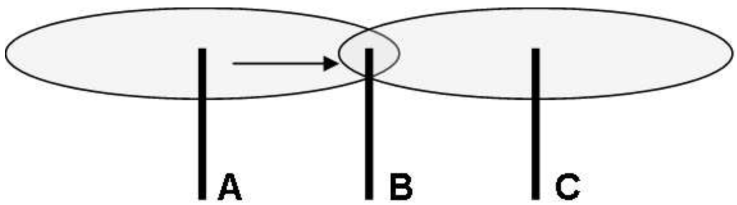

Figure 7: A and C must not transmit at the same time if either is destined for B

When host A transmits, host C must be silent in order to avoid collisions at B, to solve this 802.11 uses a virtual carrier sense mechanism known as RTS/CTS. Before transmitting data a host should send an RTS message to the destination, if the destination considers the wireless medium to be available then it responds with a CTS and the data is transmitted. Whenever a host hears an RTS or CTS message it will consider the medium to be busy, thus returning to the original example collisions are avoided.

In the context of a wireless mesh a single hop has the full link bandwidth available, the addition of a second hop requires B to re-transmit therefore capacity is halved and at the 3rd hop capacity is reduced yet again to 1/3rd. This is because before transmitting C will send an RTS to D, B will receive this message and thus consider the wireless medium to be busy, hence when A sends an RTS to B it will get no response as B knows the medium is in use, as shown in figure 8. If B allowed A to transmit then it is unlikely to receive an un-corrupted frame since there will be interference from C's transmission to D.

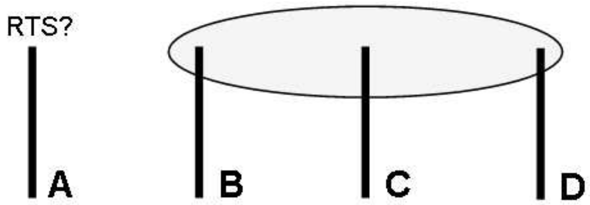

Figure 8: B knows the wireless medium is in use so will not permit A to TX

Only once we reach the 4th hop can we do 2 simultaneous transmissions. However the problem is exasperated further when the bi-directional nature of TCP is considered, figure 9 shows that when D transmits to C (as TCP ACKs require), A should not transmit to B. This is because radio layer ACKs from C will interfere at B.

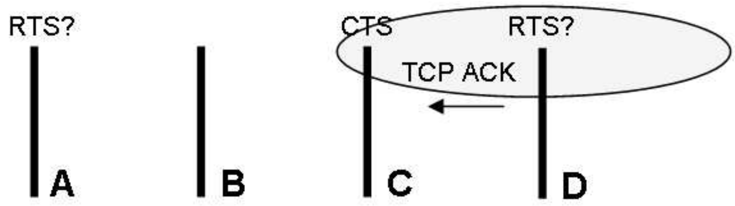

Figure 9: A and C must not transmit at the same time if either is destined for B

Given these results I began to question how practical a single radio 802.11b wireless mesh network is. An obvious solution to increase bandwidth would be to use higher base speeds as provided by 802.11g and 802.11a, however both of these require high signal strengths to achieve the faster bit-rates which are un-realistic over distances of several hundred meters or more.

It is interesting to compare these figures with the results achieved by Gerla et al[9] who simulated massive falloffs in throughput when using the FAMA MAC (which they cite as being similar to 802.11). With a channel data rate of 2Mbps they simulated throughput of just 355Kbps over 3 hops with a TCP window size of 1, when this was increased to 32KB the throughput dropped to just 71Kbps. All of our experiments were conducted with window sizes of 64KB, I have yet to investigate the reason for this large discrepancy.

There is no straight forward answer to this problem, I'm currently investigating a more centralised/cellular approach where a single node is placed on a tall building to which lots of clients can connect to using directional antenna. These central nodes are linked with high speed backbone connections, either wired or point to point wireless.

4.3 Related work

I have been involved in 2 other projects related to my research, the Ambient Wood and Floodnet. Ambient Wood is a sub-project of the Equator IRC [8], about creating playful learning experiences. The infrastructure requires the deployment of a battery powered ad-hoc wireless network infrastructure which supports mobility of clients. Floodnet [9] is concerned with the deployment of an ad-hoc wireless sensor network for monitoring water depth in rivers with the goal of predicting flood events.

The ambient wood network is identical to that deployed in SOWN, client mobility is handled by transparent mobile IP software[10], the network provided a marked improvement over the previous design which used bridging code and suffered brief outages whenever a client moved (mobility was detected as topology changes).

[8]The EQUATOR Interdisciplinary Research Collaboration. `http://www.equator.ac.uk/`

[9]FloodNet, `http://envisense.org/floodnet/`

[10]TMip: Transparent Mobile IP, `http://www.slyware.com/projects_tmip.shtml`

The FloodNet network is still in development, although the basic approach of single radio repeater nodes is the same, as is the hardware used. A FloodNet could in theory be a very large network spanning many km of a river and comprising of hundreds of nodes, the main requirement in FloodNet are rapid convergence as the nodes may only be in an active state for brief periods of time, and a small memory/CPU footprint as it needs to run on embedded PDA like devices.

5 Future work

5.1 Improving Geographic Routing

It is becoming clear to me that in wireless networks containing voids nodes require knowledge of the topology to optimise routing, at a local scale intimate knowledge is required but as distance to the destination increases only very large voids would need to be known. This is in line with the principles behind FSR, where local topology is better known than distant ones.

5.1.1 Geo-Hierarchy

Node clustering as employed by HSR seems to meet these needs whilst providing an excellent mechanism for scaling, I have begun to investigate how the addressing mechanism could be IP based rather than using a series of MAC addresses. In IPv4 one could envisage a Level 0 cluster representing a /24 subnet, Level 1 is /16 and so on. The main challenge here would be allocating clusters non-conflicting address ranges, especially in the context of a rooftop network where sites may initially be in a disconnected state and thus unable to detect address conflicts.

I believe that IPv6 geographic addressing could be used to tackle this, the geographic addressing scheme presented previously could address square areas at set intervals of 6.4m using 44 bits of network address. 4 such areas in a square can be aggregated into a 40 bit prefix, this process repeats reducing the prefix length by 4 bits with each iteration.

I propose that clusters should aim to form on these prefix boundaries, which boundaries are used depends on node density and radio range, based on my experiences with 802.11b a 1.6km^2 area (28 bit prefix) may be a appropriate for Level 0 cluster formation. Consecutive cluster levels would be located every 8 bits meaning they cover areas of 26, 417 and 6667km^2 respectively.

These represent the ideal (maximum) size for a cluster to reach, it may only achieve this if it has internal connectivity with every node inside the area defined by it's prefix.

If a cluster cannot reach it's optimal size then it should still exchange routing with neighbours of an equivalent level, however it should only advertise a route for it's actual area. As more nodes appear connectivity will improve and it should be possible for the cluster to reach it's optimal size aggregating with it's neighbours, hence denser networks will actually have smaller routing tables than sparse ones.

5.1.2 Geographic with Void Identification

Whilst the geo-hierarchy approach looks promising it requires nodes to hold some degree of state, especially if the networks are large but very sparse (with many areas lacking complete internal connectivity). Over the past 4-5 years there has been a lot of interest in Smart Dust[14] these are tiny computing devices typically with some sensing and networking capability. Such devices would require a routing protocol with minimal overheads in terms of state.

I intend to investigate a modification to the basic geographic, GPSR like approach and pro-actively identify voids in the network coverage, such voids could be mapped and their size calculated. This data could then be proporgated through the network until it was no longer needed. One such test for determining this information might be "If I have a packet destined to the opposite side of the void, does this information affect which node I would select as the next hop", if yes then the node stores the void data and forwards it, if not then the data can be safely discarded.

References

[1] Stephen Adams. A high level simulator for gunk. Technical report, MIT Department of Electrical Engineering and Computer Science, December 1998.

[2] Stephen Adams and Dave DeRoure. A simulator for an amorphous computer. In *12th European Simulation Multiconference (ESM 98)*, June 1998.

[3] C. Chiang, H. Wu, W. Liu, and M. Gerla. Routing in clustered multihop, mobile wireless networks with fading channel. In *IEEE Singapore International Conference on Networks*, pages 197–211, 1997.

[4] Kevin Fall and Kannan Varadhan. The network simulator - ns-2. Technical report. `http://www.isi.edu/nsnam/ns/`.

[5] Gregory G. Finn. Routing and addressing problems in large metropolitan-scale internetworks. Technical report, Information Sciences Institute, March 1987.

[6] Tony Hain. An ipv6 provider-independent global unicast address format. Internet draft, Internet Engineering Task Force, April 2003. `http://www.ietf.org/internet-drafts/draft-hain-ipv6-pi-addr-04.txt`.

[7] T. Imielinski and J. Navas. Geographic addressing, routing, and resource discovery with the global positioning system, 1997.

[8] A. Iwata, C. Chiang, G. Pei, M. Gerla, and T. Chen. Scalable routing strategies for ad-hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1369–1379, August 1999.

[9] Sang Bae Kaixin Xu, Mario Gerla. How effective is the ieee 802.11 rts/cts handshake in ad hoc networks? In *IEEE Globecom 2002*.

[10] Brad N. Karp. *Geographic Routing for Wireless Networks*. PhD thesis, Havard University, Cambridge, Massachusetts, October 2000.

[11] Radhika Nagpal and Daniel Coore. An algorithm for group formation in an amorphous computer. In *10th International Conference on Parallel and Distributed Computing Systems (PDCS'98)*, October 1998.

[12] The Institute of Electrical and Inc. Electronics Engineers. Wireless lan medium access control (mac) and physical layer (phy) specifications. `http://standards.ieee.org/getieee802/download/802.11b-1999.pdf`.

[13] Timothy J. Shepard. A channel access scheme for large dense packet radio networks. In *SIGCOMM*, pages 219–230, 1996.

[14] Brett Warneke, Matt Last, Brian Liebowitz, and Kristofer S. J. Pister. Smart dust: Communicating with a cubic-millimeter computer. *Computer*, 34(1):44–51, 2001.

[15] Rajive Bagrodia Xiang Zeng and Mario Gerla. Glomosim: A library for parallel simulation of large-scale wireless networks. In *12th Workshop on Parallel and Distributed Simulation (PADS 98)*, pages 154–161, Banff, Alberta, Canada, May 1998.