

# Ad hoc Protocol Evaluation and Experiences of Real World Ad Hoc Networking

David Lundberg  
davidl@update.uu.se

Department of Information Technology, Uppsala University

**Examiner:**

Per Gunningberg, Christian Tschudin

**Supervisor:**

Henrik Lundgren

Uppsala University  
Department of Information Technology  
Box 337  
SE-751 05 Uppsala  
Sweden

## **Abstract**

This report give an introduction to ad hoc networking and presents a list of over 60 ad hoc routing protocols that have been proposed between 199x and 2002. It also discuss the problems of performance evaluation of these protocols and the experiences gathered during the implementation of a real world evaluation testbed.

A major challenge when evaluating ad hoc routing protocol proposals is to agree on a metric under which a protocol's performance will be measured. The fact that most proposed ad hoc protocols have different goals makes it very important to find the essential properties and invent methods how to measure them. This is the main focus in this this report.

The first part discuss the methods and metrics used in simulations performed during recent years. The results show that mobility models, traffic patterns, metrics and propagation models are crucial when doing simulations in order to get valid results.

The second part of this paper describes a new metric called virtual mobility(vM) describing the mobility from a physical point of view opposed to geometrical or link-stability metrics. It also describes the APE-testbed (Ad hoc Protocol Evaluation) that we have created in order to be a able to conduct large scale experiments in an real environment. A lot of effort was put on making the testbed streamlined and as easy to use as possible.

# Contents

<b>1</b>	<b>Introduction - What is an ad hoc network?</b>	<b>2</b>
<b>2</b>	<b>Background and problem formulation</b>	<b>3</b>
2.1	Methods . . . . .	3
2.2	Results . . . . .	4
2.3	Structure of the report . . . . .	4
<b>3</b>	<b>Ad hoc routing protocols</b>	<b>5</b>
3.1	Brief description of the AODV algorithm . . . . .	5
3.2	List of protocols . . . . .	5
<b>4</b>	<b>Simulation</b>	<b>8</b>
4.1	Mobility models used in simulations . . . . .	8
4.1.1	Random waypoint model (RWM) . . . . .	8
4.1.2	Random direction model (RDM) . . . . .	8
4.1.3	Modified Random direction model (MRDM) . . . . .	8
4.1.4	Brownian model (BM) . . . . .	8
4.1.5	Column model (CM) . . . . .	8
4.1.6	Random Gauss-Markov model (RGM) . . . . .	9
4.1.7	Pursue model (PM) . . . . .	9
4.1.8	Exponential Correlated Random model (ECR) . . . . .	9
4.1.9	Reference Point Group mobility model (RPGM) . . . . .	9
4.1.10	Individual Simulated Behavioral model (ISB) . . . . .	9
4.2	Mobility metrics . . . . .	9
4.2.1	Geometric-based mobility metric . . . . .	9
4.2.2	Minimal route-change metric . . . . .	10
4.3	Discussion of simulation models and metrics . . . . .	10
4.3.1	Mobility models not accurate enough . . . . .	10
4.3.2	Mobility models not used combined . . . . .	10
4.3.3	Metrics not advanced enough to capture needed properties . . . . .	10
4.3.4	The propagation models used are too simple . . . . .	11
4.3.5	Frequency related modelling is required . . . . .	11
<b>5</b>	<b>Protocol Performance Measurement</b>	<b>12</b>
5.1	Field tests . . . . .	12
5.2	APE - testbed . . . . .	13
5.2.1	Reproducibility . . . . .	13
5.2.2	Simple installation & operation . . . . .	13
5.2.3	The adaptive solution . . . . .	14
5.2.4	Synchronisation - the relative word . . . . .	14
5.2.5	What to log and what not to log . . . . .	15
5.2.6	Experiment workflow . . . . .	16
5.3	Metric toolbox . . . . .	17
5.3.1	Traffic description . . . . .	17
5.3.2	Connectivity & relative density . . . . .	17
5.3.3	Virtual mobility . . . . .	18
5.4	Conclusion . . . . .	18
5.5	Functional scenarios . . . . .	18
5.5.1	Simple scenarios . . . . .	19
5.5.2	Virtual mobility scenarios . . . . .	19
5.6	Experiences from scaling up experiments . . . . .	21
5.6.1	Handling a lot of participants and hanging nodes . . . . .	22

5.6.2	In-lab testing with troublesome equipment is important . . . . .	22
5.6.3	Time is expensive . . . . .	22
5.6.4	Small matters take a lot of time . . . . .	22
<b>6</b>	<b>Conclusion</b>	<b>23</b>
6.1	Ad hoc protocol evaluation . . . . .	23
6.2	Discussion of protocol evaluation . . . . .	23
6.3	Acknowledgements . . . . .	25
<b>A</b>	<b>Protocol abstracts</b>	<b>26</b>
A.1	Associativity-Based Routing (ABR) [31] . . . . .	26
A.2	Adaptive Demand-Driven Multicast Routing protocol (ADMR) [47] . . . .	26
A.3	Ad hoc On Demand Distance Vector protocol (AODV) [4] . . . . .	26
A.4	Bordercast Resolution Protocol (BRP) [25] . . . . .	27
A.5	Core Extraction Distributed Ad hoc Routing (CEDAR) Specification [51] .	27
A.6	Cluster Based Routing Protocol (CBRP) [8] . . . . .	27
A.7	Differential Destination Multicast (DDM) Specification [20] . . . . .	28
A.8	Dynamic Source Routing protocol (DSR) [41] . . . . .	28
A.9	DSR Simple Multicast and Broadcast protocol (DSR-MB) [42] . . . . .	28
A.10	DSR-Flow protocol [43] . . . . .	29
A.11	Fisheye State Routing protocol (FSR) [38] . . . . .	29
A.12	Host Specific Routing (HSR) [18] . . . . .	29
A.13	Intrazone Routing Protocol (IARP) [26] . . . . .	29
A.14	Interzone Routing Protocol (IERP) [27] . . . . .	30
A.15	An Internet MANET Encapsulation Protocol (IMEP) Specification [30] . .	30
A.16	Landmark Routing Protocol for Large Scale Networks (LANMAR) [7] . . .	30
A.17	Lightweight Underlay Network Ad hoc Routing (LUNAR) [3] . . . . .	31
A.18	Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing [50] . .	31
A.19	Mobile Mesh Border Discovery Protocol (MMBDP) [63] . . . . .	31
A.20	Mobile Mesh Link Discovery Protocol (MMLDP) [64] . . . . .	31
A.21	Mobile Mesh Routing Protocol (MMRP) [65] . . . . .	31
A.22	Multicast Zone Routing protocol (MZR) [48] . . . . .	32
A.23	On-Demand Multicast Routing Protocol (ODMRP) [11] . . . . .	32
A.24	Optimized Link State Routing Protocol (OLSR) [39] . . . . .	32
A.25	Relative Distance Micro-discovery Ad Hoc Routing (RDMAR) Protocol [10]	32
A.26	Source Routing-based Multicast Protocol (SRMP) [49] . . . . .	33
A.27	Source Tree Adaptive Routing protocol (STAR) [9] . . . . .	33
A.28	Topology Broadcast Reverse-Path Forwarding protocol (TBRPF) [40] . . .	33
A.29	Temporally-Ordered Routing Algorithm protocol (TORA) [28] . . . . .	33
A.30	Zone Routing Protocol (ZRP) [24] . . . . .	34
<b>B</b>	<b>Techno stuff</b>	<b>35</b>
B.1	Software . . . . .	35
B.2	Hardware . . . . .	35

# 1 Introduction - What is an ad hoc network?

The history of wireless networks started in the 1970s and the interest has been growing ever since. During the last decade, and especially at the end, the interest has almost exploded probably because of the fast growing Internet. Today we see two kinds of wireless networks but the difference between them is not as obvious as it may seem. The first kind and most used today is a wireless network built on-top of a “wired” network and thus creates a *reliable infrastructured wireless network*.

The wireless nodes also connected to the wired network and able to act as bridges in a network of this kind are called base-stations. An example of this is the cellular-phone networks where a phone connects to the base-station with the best signal quality. When the phone move out of range of a base-station it does a “hand-off” and switch to a new base-station within reach. The “hand-off” should be fast enough to be seamless for the user of the network. Other more recent networks of this kind is wireless networks for offices, cafés etc which usually are called Wireless Local Area Networks (WLAN).

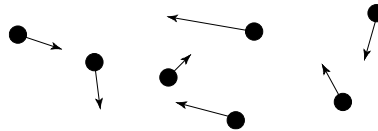


Figure 1: An ad hoc network with nodes moving in different directions and speeds.

The other kind is as it may seem the orthogonal kind. One where there is no infrastructure at all except the participating mobile nodes. This is called an *infrastructurless network* or more commonly an ad hoc network. The word “ad hoc” can be translated as “improvised” or “not organized” which often has a negative meaning, but the sense in this context is not negative but only describing the network situation, i.e. dynamic.

All or some nodes within an ad hoc are expected to be able to route data-packets for other nodes in the network who want to reach other nodes beyond their own transmission-range. This is called *peer-level multi-hopping* and is the base for ad hoc networks that constructs the interconnecting structure for the mobile nodes.

An ad hoc network is usually thought of as a network with nodes that are relatively mobile compared to a wired network. Hence the topology of the the network is much more dynamic and the changes often unpredictable oppose to the Internet which is a wired network. This fact creates many challenging research issues since the objectives of how routing should take place is often unclear because of the different resources like bandwidth, battery power and demands like latency and other types of QoS. The routing protocols used in ordinary wired networks are not well suited for this kind of dynamic environment. They are usually built on periodic updates of the routes and create a large overhead in a relative empty network and also cause slow convergence to changes in the topology.

The research interest in solving the problem with protocols for routing in an ad hoc environment has spawned a *mobile and ad hoc networking* (MANET)[2] working group within the Internet Engineering Task Force (IETF) which has the following goal: “The primary focus of the working group is to develop and evolve MANET routing specification(s) and introduce them to the Internet Standards track”

Well, what could be the possible use for this ad hoc networking? As with the development of the Arpanet now known as the Internet it was the military that was one the major contributors when wireless ad hoc networks started to be developed. The initial thought was that ad hoc would be perfect for the battlefield where there is usually no infrastructure for communication at all. Today one of the most popular scenarios are communication within groups of people with laptops and other hand-held devices. Another one is communication networks for supporting rescue personnel in disaster areas after a earthquake or similar.

## 2 Background and problem formulation

Practical work with ad hoc networks at Uppsala university have been going on for a few years. Small tests with active routing was carried out in [116] which showed the possibility to do active routing with simple software. During spring 2000 I built a 802.11b based wireless network infrastructure sponsored by Ericsson. The wireless network extends the local network and enables a user to have connectivity with the network anywhere in the covered buildings. A number of cards, also sponsored by Ericsson is lent out to students who take the course in “Advanced computer communication” and are from the Information Technology Program (ITP). All of the students in the ITP program are equipped with laptops and all this creates a unique opportunity for doing real world, large scale hands-on research.

One of the hands-on projects in the course “Advanced computer communication” was about testing to evaluate a ad hoc protocol by trying an implementation. The results of that project spawned into a project that aimed for the possibility to run large scale without the massive work encountered by the earlier project. Two students where hired for the project which later became the the APE testbed. APE is short for *Ad hoc Protocol Evaluation* which summarizes the goal with the project, i.e. to be able to evaluate an ad hoc protocol’s properties in a real life environment.

When the work was completed with [119] and I choose to continue the work I identified three different issues that I wanted to investigate further.

- Differences between simulations and the real world  
Are the mobility models used reliable? Can the metrics used in simulations be used in an real world environment? What propagation models are used and what effect do they have on the simulations?
- Metrics for real world mobile networks  
What properties are important in an ad hoc network? Is the property possible to measure? What kind of traffic are likely to be found in an ad hoc network and how does that affect the performance?
- Large scale experiments  
How do you conduct a large scale test for ad hoc networks? How to handle large amount of users and log-files and still not waste to much time on logistic problems?

### 2.1 Methods

- Study literature on simulations and comparisons that have been done and try to understand what people think is essential properties in an ad hoc protocol. MANET test RFC [67]. Also study ad hoc protocols in general and try to understand what type of environment/scenarios they are created for.
- Assist in the implementation of a testbed for ad hoc routing in the real world.
- Design suitable scenarios for testing of different properties in an ad hoc routing protocol.
- Scale up the numbers of nodes in the proposed scenarios.
- Perform experiments with one or more ad hoc routing protocol implementations.

## **2.2 Results**

A result from the studies of simulations and comparisons is a list of all the protocols which is presented in Section 3.2.

Section 4 presents an extensive analysis of mobility models and metrics used in the studied simulations. The conclusion discuss several problems about the methods used today.

A description of the testbed that was built is presented in Section 5 together with the “metric toolbox” containing several metrics that can be of use.

## **2.3 Structure of the report**

The report begins with an introduction in Section 1 and continues with background and problem formulation in Section 2. Section 3 discuss routing in ad hoc networks and which protocols that exists today.

The study of simulations and comparisons, which mobility models and metrics have been used is presented Section 4. Section 5, “Protocol performance measurement” is the section about the testbed, tested scenarios, experiences and the toolbox with metrics.

Finally there is Section 6 with conclusions and a discussion of ad hoc protocol evaluation. Also added to the report is Appendix A which consists of abstracts from protocol drafts found on the Net during the work with the thesis.

### 3 Ad hoc routing protocols

The world of ad hoc protocols is evolving very fast and new drafts pop-up all the time. Sadly, drafts are removed and not archived within six months if they are not maintained. This creates a huge problem if you try to get a view of what has done within the field during the latest years. In order to hopefully not reinvent the wheel I tried to compile a list (3.2) of the existing protocols.

This section begin with a brief description of how the AODV algorithm work to give a quick introduction to an ad hoc protocol. This is also the protocol we used in the testbed.

#### 3.1 Brief description of the AODV algorithm

As many ad hoc routing protocols an AODV-node informs its neighbours about its own existence by constantly sending “hello messages” at a defined interval. This enables all nodes to know the status about their neighbours, i.e. if they gone down or moved out of reach.

To resolve a route to another node in the network AODV floods its neighbours with a route request(RREQ). A RREQ contain the senders address, the address of the sought node and the last sequence number received from that node if there exist one. The receiving node checks if it has a route to the specified node. If a route exists and the sequence-number for this is higher than the supplied a new route is found. The node replies to the requesting by sending a route reply (RREP). If on the other hand a route does not exist the receiving node sends a RREQ itself to try to find a route for the requesting node.

If the original node does not receive an answer within a time-limit the node can deduce that the sought node are unreachable. Since the request was sent to all neighbours the node may end up with several routes but they are easily separated by the sequence numbers. nodes along the route keep their routing tables updated as long as traffic flows along the route. If not, the nodes will discard the routing entries after a specified time. To be sure that the route still exists, the sender has to keep the route alive by periodically sending packets. All nodes along the route are responsible for the upstream links which means that a broken link will be discovered by the closest node. This node signal the broken link by sending an error message (RRERR) downstream so that the using nodes can start to search for a new route.

AODV is a pretty simple routing-scheme. It has low overhead and supports multicast, but requires symmetric links. The on-demand structure makes it along with DSR very power efficient as stated by [113].

#### 3.2 List of protocols

This is an attempt to assemble a list containing most of the ad hoc protocols found on the Net. It became clear that there were many protocols but no survey over them all. Since drafts are removed from IETF when they are not continually supported I have collected abstracts from drafts I have come across in order to create a small reference. This can be found the Appendix A.

The protocols are sorted based on the type of routing they use. The classic pro-active type updates the routingtables on a regular basis compared to the reactive where they only are updated when requested. The reactive type is very popular in ad hoc networks since they adapt fast which is a key feature. Then there is the hierarchical type which usually combine two or more strategies to create several routing-layers. Routing based on geographical information have also been used but usually requires extra equipment compared to the other protocols. Last of the unicast-protocols’ are the power-aware that optimize usage of the power stored in the nodes. And last there are two sections with multicast protocols.



**Pro-active: (Table-driven)**

CGSR	Clusterhead Gateway Switch Routing protocol [23]
DBF	Distributed Bellman-Ford routing protocol [21]
DSDV	Distance Source Distance Vector routing protocol [19]
DTDV	Highly Dynamic Destination-Sequenced Distance Vector routing protocol [6]
HSLS	Hazy Sighted Link State routing protocol [62]
HSR	Hierarchical State Routing protocol
LCA	Linked Cluster Architecture [60]
MMBDP	Mobile Mesh Border Discovery Protocol [63] (A.19)
MMLDP	Mobile Mesh Link Discovery Protocol [64] (A.20)
MMRP	Mobile Mesh Routing Protocol [65] (A.21)
OLSR	Optimized Link State Routing Protocol [39] (A.24)
STAR	Source Tree Adaptive routing protocol [9] (A.27)
TBRPF	Topology Broadcast based on Reverse-Path Forwarding routing protocol [40] (A.28)
WRP	Wireless Routing Protocol [22]

**Reactive: (On-demand)**

ABR	Associativity Based Routing protocol [31] (A.1)
AODV	Ad hoc On Demand Distance Vector routing protocol [4] (A.3)
BSR	Backup Source Routing protocol [32]
DSR	Dynamic Source Routing protocol [41] (A.8)
DSRFLOW	Flow State in the Dynamic Source Routing protocol [43] (A.10)
FORP	Flow Oriented Routing Protocol [34]
LMR	Lightweight Mobile Routing protocol [29]
LUNAR	Lightweight Underlay Network Ad hoc Routing [3] (A.17)
RDMA	Relative-Distance Micro-discovery Ad hoc Routing protocol [10]
SSR	Signal Stability Routing protocol [33]
TORA	Temporally-Ordered Routing Algorithm routing protocol [28] (A.29)

**Hierarchical:**

CBRP	Cluster Based Routing Protocol [8] (A.6)
CEDAR	Core Extraction Distributed Ad hoc Routing [51] (A.5)
DDR	Distributed Dynamic Routing Algorithm [35]
GSR	Global State Routing protocol [17]
FSR	Fisheye State Routing protocol [38] (A.11)
HARP	Hybrid Ad Hoc Routing Protocol [36]
HSR	Host Specific Routing protocol [18] (A.12)
LANMAR	Landmark Routing Protocol for Large Scale Networks [7] (A.16)
ZRP	Zone Routing Protocol protocol [24] (A.30)
BRP	Bordercast Resolution Protocol [25] (A.4)
IARP	Intrazone Routing Protocol [26] (A.13)
IERP	Interzone Routing Protocol [27] (A.14)

**Geographical:**

DREAM	Distance Routing Effect Algorithm for Mobility [44]
GLS(Grid)	Geographic Location Service [46]
LAR	Location-Aided Routing protocol [45]
ZHLS	Zone-Based Hierarchical Link State Routing [61]

**Power aware:**

- ISAIHA    Infra-Structure Aodv for Infrastructured Ad Hoc networks [5]  
PARO     Power-Aware Routing Optimization Protocol [37]  
PAMAS    PAMAS-Power Aware Multi Access Protocol with Signaling Ad Hoc Networks [52]

**Multicast:**

- ABAM     On-Demand Associativity-Based Multicast [53]  
ADMR     Adaptive Demand-Driven Multicast Routing protocol [47] (A.2)  
AMRIS     Ad hoc Multicast Routing protocol utilising Increasing id-numbers [13]  
AMRoute   Ad hoc Multicast Routing Protocol [12]  
CAMP     Core-Assisted Mesh Protocol [14]  
CBM     Content Based Multicast [15]  
DDM     Differential Destination Multicast [20] (A.7)  
FGMP     Forwarding Group Multicast Protocol [54]  
LAM     Lightweight Adaptive Multicast protocol [16]  
DSR-MB   Simple Protocol for Multicast and Broadcast using DSR [42] (A.9)  
MAODV   Multicast Ad hoc On-Demand Distance Vector routing [50] (A.18)  
MCEDAR   Multicast CEDAR [55]  
MZR     Multicast Zone Routing protocol [48] (A.22)  
ODMRP   On-Demand Multicast Routing Protocol [11]  
SRMP     Source Routing-based Multicast Protocol [49] (A.26)

**Geographical Multicast (Geocasting):**

- LBM     Location Based Multicast [56]  
GeoGRID   Geographical GRID (see GLS) [57]  
GeoTORA   Geographical TORA (see TORA) [58]  
MRGR     Mesh-Based Geocast Routing [59]

## 4 Simulation

There are two main issues when doing simulations except for modeling of the physical environment. The first one is the modeling of how the nodes move in the mobile ad hoc network. This is very tricky if you don't know how the nodes will move in the real world. There are a number of suggestions how to do this and some are found in the next section.

The other issue is that in order to compare two different networks and describe their behaviour you need some kind of metrics. If you are trying to evaluate two different routing protocols you may want to know the *mobility*, i.e how hard it is to handle routing in the network. It is a popular term and in this case it merely describes the activity of link-changes caused by external physical interference.

### 4.1 Mobility models used in simulations

#### 4.1.1 Random waypoint model (RWM)

Johnson and Maltz describe in [110] the Random waypoint model. It works as follows. All nodes are uniformly distributed around the simulation area at starting time. Each node then choose a random destination and moves there with a speed uniformly distributed over  $[0, v_{max}]$ . Then there is a pause time which could be selected be 0 to give continuous motion.

#### 4.1.2 Random direction model (RDM)

In [95] Royer et al describe another random based model. This a more “stable” model than a random waypoint model. At start the nodes selects a random direction and starts to move along it. Since the area of simulation is confined the node may end up reaching one of the boundaries during the simulation. When a boundary is reached the node pause for a given time and then chooses a new direction to travel. Since the node is on a boundary the selectable angle is 180 degrees. The result of this model is a more stable distribution of the nodes than the RWM (4.1.1). The behaviour can be thought as a micro-cell of a larger area which is a useful property.

#### 4.1.3 Modified Random direction model (MRDM)

A second more advanced version described by Royer et al in [95]. To give a even more realistic simulation the Random Direction Model (4.1.2) was extended with a extra choice for the nodes when their pause time is over. The nodes don't have to travel all the way to the boundary but could stop anywhere along the path.

#### 4.1.4 Brownian model (BM)

Hu and Johnson describe in [100] another way of modelling the speed of the nodes. Changes speed and direction at discrete time intervals and at the beginning of each interval each node chooses  $r \in [0, v_{max}]$  and  $\theta \in [-\pi, \pi]$  moves with velocity vector  $(r \sin \theta, r \cos \theta)$ . This model is very similar to the random direction model except for the speed which is smooth in this model.

#### 4.1.5 Column model (CM)

A mobility model suited for experiments is described by Sanchez in [112]. Nodes are only moving along the x-axis. The initial position of node  $i$  is  $(10i, 10i)$  and the node changes the speed  $v \in [0, v_{max}]$  at the discrete intervals. This will produce a mobility pattern that is one dimension simpler than the random mobility model since the nodes only move along the x-axis.

#### 4.1.6 Random Gauss-Markov model (RGM)

Uses discrete time intervals to divide up the motion. The nodes update their velocity vectors at the beginning of each interval according to:

$$v_{x_t} = \alpha_{x_{t-1}} + (1 - \alpha) * \bar{v}_x + R\sqrt{1 - \alpha^2} \quad (1)$$

$$v_{y_t} = \alpha_{y_{t-1}} + (1 - \alpha) * \bar{v}_y + R\sqrt{1 - \alpha^2} \quad (2)$$

$R$  is a random variable with mean 0 and variance  $\sigma$ . This model is describe by Sanchez [112] and further developed by Liang and Haas [109].

#### 4.1.7 Pursue model (PM)

Another model done by Sanchez [111] in order to try to create group movement. One node in each group is moving according to the random waypoint model (4.1.1). The rest of the group is moving towards the target that the “leading” is aiming for. The speed of the pursuing nodes is chosen uniform random in the range  $[v_{pmin}, v_{pmax}]$ .

#### 4.1.8 Exponential Correlated Random model (ECR)

The ECR is able to model all possible movements of individuals and groups. This is done by changing the parameters of a motion function. A new position  $b(t+1)$  is a function of the previous position  $b$  to which a random deviation is added. The function  $b(t) = (r_t, 0_t)$  can be defined either for a single node or a group at time  $t$ .  $r$  is a random Gaussian variable with variance  $\sigma$ . The parameters are then changed to give different mobility patterns. Very hard to create a predefined motion pattern by selecting the parameters. This model is described by BBN in [115].

#### 4.1.9 Reference Point Group mobility model (RPGM)

Ho et al describes another way to simulate group behaviour in [96] where each node belong to a group where every node follow a logical centre reference point. The nodes in a group are usually randomly distributed around the reference point. The different nodes use their own mobility model and is then added to the reference point which drives them in the direction of the group.

This general description of group mobility can be used to create a variety of models for different kinds of mobility applications.

#### 4.1.10 Individual Simulated Behavioral model (ISB)

This is another new and different idea how to do more accurate and better simulations. They use a theory about an individually simulated behavioral model where all objects has their own properties. They verified their idea with DSR and proved that it generates reproducible and “realistic” mobility patterns. [121]

### 4.2 Mobility metrics

#### 4.2.1 Geometric-based mobility metric

Johansson et al [108] described a geometric mobility metric as

$$\frac{2}{N(N-1)T} \sum_{i=1}^N \sum_{j=i+1}^N \int_{t=0}^T \left| \frac{d\|P_j(t) - P_i(t)\|_2}{dt} \right| dt$$

where  $P_k(t)$  is the physical position of node  $k$  at time  $t$ ,  $T$  is the length of the test and  $N$  is the number of nodes participating in the test. The sum is calculated over all node pairs

over the scenario duration. Hu and Johnson [93] used exact this model in their simulations, while Johansson et al [108] used an approximation of this geometric mobility metric in their simulations.

#### **4.2.2 Minimal route-change metric**

This metric described by Hu & Johnson [100] basically calculates the link-breaks in a route counted by the metric. It is assumed that all link are bidirectional. The routes counted by the metric can either be the ones between the nodes that communicate over all pair of nodes regardless of the traffic during the measured time.

### **4.3 Discussion of simulation models and metrics**

#### **4.3.1 Mobility models not accurate enough**

Simulations have shown that the existing models are not accurate enough for real world simulations. In [96] results show that real ad hoc environment cannot be simulated with a “random walk” type of mobility model. The more advanced models that exists provide a much better simulation than the random walk types but have the disadvantage that they demand much more computation time. Models like the ECR have great possibilities but have shown very hard to control to get the desired motion patterns.

#### **4.3.2 Mobility models not used combined**

A common view of an ad hoc network is that nodes join/part/move in the network relatively often. The result of this is that the traffic patterns get more important since an mobile/unstable network is less suitable for some types of traffic, e.g. high-speed backbone traffic with demands on low latency. The most effective traffic pattern for this kind of network seem to be bursty low-volume traffic e.g. www/mail. The reason for this is the low demand of long-lived routes and latency.

As mentioned, the movements of the nodes in an ad hoc network is very essential for the routing protocol and therefore the mobility model is important for simulation results. However, all the simulations studied have only used one of the mobility models and that fact illuminate a problem that should be investigated. To conclude, there’s a need to look carefully at the mobility models used in simulations as these are crucial for the real-world functioning of an hoc network.

#### **4.3.3 Metrics not advanced enough to capture needed properties**

Johansson et al [108] used the protocols DSDV [19], AODV [4] and DSR [41] to simulate a conference, a event coverage and a disaster area. The conclusion was that the metric worked fine when doing simulations.

Whereas the geometric mobility metric give us a view of how mobile the nodes are physically the minimal route-change metric give us a view how stable the links in the network are. Thus should these two combined give a quite good view of how the network behave. High mobility but at the same time low route-change could describe a network where the almost all nodes move, but not enough to potentially affect the routes. If it was possible to do accurate positioning with our testbed this would be a hypothesis to verify since the propagation in the real world is very unstable.

But since we have no possibilities to do accurate positioning in large scale experiments due to the fact that it would too expensive there is a need for new metrics. These metrics can not as stated rely on expensive positioning devices and should use the data available about the ongoing traffic for best economics.

#### 4.3.4 The propagation models used are too simple

The simulations that have been done have been using a very simplified model of the real world environment. Most of the simulations have been using NS-2 [76] which has an extension [77] to simulate a wireless environment. The extension uses the Free Space Path Loss (FSPL) [97] model or Two Ray Ground Reflection (TRGR)[97] depending on the distance:

$$FreeSpacePathLoss(FSPL) \text{ in dB} = 20 * \log_{10}(\frac{4\pi R}{\lambda}) \quad (3)$$

$$TwoRayGroundReflection(TRGR) \text{ in dB} = \frac{R^4}{(ht * hr)^2} \quad (4)$$

Where  $R$  is the distance,  $\lambda$  the wavelength and  $ht, hr$  the heights of the transmitter and receiver antennas.

It can not simulate objects as walls etc that have influence on the propagation. In [108] they extended NS to be able to simulate solid objects that block the signal if LOS is lost. This is still a very simple approximation of the complex real world. A more accurate model to do more accurate simulations is the Simulation of Indoor Radio Channel Impulse-Response Model (SIRCIM) [78] which simulates fading, barriers, foliage, multipath interference etc. It has been implemented by [105] in Global Mobile Simulation library GloMoSim [79] which uses PARSEC [99]. The computation time when using this model is about three magnitudes longer and it has to be tuned to each specific scenario. The model have not been used in any simulation results but when doing simulation of complex indoor environments models like the SIRCIM are required.

#### 4.3.5 Frequency related modelling is required

Even more advanced model have to be constructed in order to be able to simulate “soft” objects like people since they have great influence on the propagation, especially in the 2.4 GHz band. When a person is standing in the LOS between two nodes equipped with WaveLAN 2.4GHz cards the signal is dampened with about 15 dBm which is equal about 20 meters in distance according to [120]. This fact is critical if you try to do positioning systems using only the signal level.

This also give a hint that propagation models for objects that absorb radiation will be required to do simulation of scenarios like a conference where a lot of people will be moving around. To be able to create these models there is a need for studies of the absorption of water-rich substance around the free 2.4GHz band. For example, if there are a lot of humans in between it might help to change the channel to a higher/lower frequency.

## 5 Protocol Performance Measurement

### 5.1 Field tests

#### Quantative lessons learned

During 1998-1999 there were a lot of experiments going on at CMU reported in [101], [102]. The project involved five moving nodes and two stationary where one of the stationary was connected to the Internet. The radio-cards used was 2Mbit WaveLan which operated at 900 Mhz. The propagation should be a lot different compared to the cards working in the 2.4 GHz band today where the signal are easier to disturb. Some important experiences from the CMU testbed:

- GPS is very useful to keep track of the nodes in the network

GPS [85] was used to be able to position the nodes both in real-time and post-run analysis. During the time for the experiments the GPS-system still used encryption which resulted in an accuracy of about 100 m. Due to this they used both differential GPS(DGPS) [86] during test-runs and real time kinematic mode(RTK) [87] when available. In order to use DGPS correction information have to be received at least once per 25s and once per second in order to use RTK. The accuracy of DGPS is about 1m and RTK gives about 1cm or better.

- GPS can be used to control the network during experiments

The GPS was also used to control the nodes to try generate network without route diversity. A network with diversity is much harder to control and understand when doing post-run analysis. With the possibility the see every nodes' position during the test-runs the effects of the wireless propagation can easily be monitored. One of their main conclusions was that the wireless propagation is not as you would expect it to be. This was painfully verified during our first tests [119].

- In-lab tests crucial for success in real world environment

In order to do multihop experiments in the in-lab testing of the protocols they invented a MAC-filter which functions as a virtual world for nodes. This was very useful when debugging the implementations so the in-lab testing was found very critical for the success of the project. Another experience made was that personnel management is nontrivial and in order to solve this they made automatic scripts that controlled the nodes during the test-runs.

A couple their lessons learned when dealing with the lower-level programming was:

- Adaptive retransmission timers are a necessary complexity

The local retransmission is very important and the choice of algorithm will be very important for all multi-hop ad hoc network. If the retransmission is implemented above link-layer it must be able to adapt to congestion and high contention for the channel.

- Multi-level priority queues are worth implementing

Packets with routing information should be scheduled ahead of user-packets to ensure maximum performance in the network. They implemented a multi-level queueing scheme with one queue for each interface. The choice of scheduling algorithm could be potential for the performance.

#### Implementation, a crucial step in verification

Royer and Perkins did an implementation [94] of AODV [4] and came to the conclusion that implementations of a routing protocol is a crucial step in verification. This due to that simulation studies often assumes and simplify properties which do not hold true in a real world scenario.

```
# generic setup instructions -----
choreography.scenario.title=double_lost_n_found

# node 0 -----
node.0.ip=193.10.133.40
node.0.ipmask=255.255.255.128
node.0.action.0.msg=Test is starting...
node.0.action.0.command=start_spyd
node.0.action.0.duration=1
node.0.action.1.command=ping_node 2 660
node.0.action.1.msg=Stay at this location. You are ping node 2. (30sec).
node.0.action.1.duration=30
```

Table 1: Scenarioscript

### Basic tests are important

Toh and Chen reported [90] from a testbed experiment with four nodes running an implementation of ABR[31] on Linux [69] 2.0.30 with 2Mbit WaveLAN [74]. They measured route discovery time, throughput and end-to-end delay. The most important result was that varying beacon intervals has a very small influence on route discovery time.

## 5.2 APE - testbed

The APE testbed is designed to help to reach the long term goal to be able to tell a good ad hoc protocol from a not that good. One of the other goals is to make it possible to do large scale tests with more than a few nodes which today (year 2001) is less than 10.

Just how log the networks performance during these experiments without disturbing the traffic generated by the users is quite a problem. For example, if you don't have enough disk-space on the laptops to store everything during the experiment you need some time of the connected time to upload everything and that time has to be synchronised with all other nodes so they know that they should ignore this traffic.

In order to get to reach this goal some time in the future we needed to create a controlled way to do experiments. We decided to go for an entirely controlled way where the users follow choreographed instructions and a script generates all traffic. A script-format where we could control everything during a test-run was invented. The script is then run from the instructor-script which reads the current scenario-script and tells the user what to do. See fig. 1

### 5.2.1 Reproducibility

To test ad hoc protocols in a real life environment the best thing would be to deploy implementations for daily use. But since we never will have exactly the same circumstances when deploying two or more different protocols it will be very hard to compare results. A more thorough analysis of the work with reproducibility is done by E. Nordström in [117]. It also contain more in-depth information about the testbed than this report.

### 5.2.2 Simple installation & operation

When conducting experiments with more than just a few nodes the requirements change quickly and the required time needed for installation of the software, running the experiment and then collecting all the logged data have to be minimised.

When doing large scale tests we cannot require every participant to be very familiar with the Linux [69] environment. Therefore there is a need for the testbed to be very easy to handle for the participant to minimise the possibility for errors caused by a confused



user. It is important to think about what information you give the user to keep him/her informed about what's happening but not at the same time drowned in information about what is happening.

### 5.2.3 The adaptive solution

A solution to most of the problems is to use a small Linux [69] distribution that only contains software for running experiments. The whole distribution can then be distributed as an image-file that contains everything needed to be able to participate.

The major disadvantage that most users only have Windows [72] installed turns into a great advantage since a image-file with Linux on can be started with Loadlin. Loadlin is a small command written for DOS [73] which is able boot Linux through Windows via DOS. The only thing that a user has to do then is to download the self-extracting package, unpack and run.

### 5.2.4 Synchronisation - the relative word

Keeping track of the exact time in an ad hoc network is not a trivial task since every node have it is own view of the reality. The usual techniques to keep all synchronised don't work well in ad hoc networks and they are worthless when conducting experiments for different reasons.

#### Don't play with unlinear time

The clocks in the laptops of today are often of a very low quality. They usually have very large skews and even skews related to the load of the computer. An illustration of this is found in fig 2. This could be handled by protocols like the Network Time Protocol (NTP) [66], but the result is unlinear time which just gives us more problems when trying to analyse it afterwards. Secondly we don't have the ability to have a synchronised server that NTP requires.

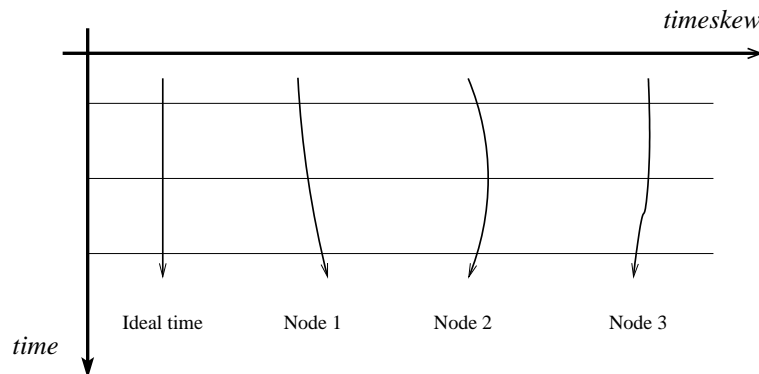


Figure 2: Example of the TSB synchronising theory

#### The broadcasting solution

A scheme for synchronising the time between the nodes had to be invented to be able to handle these clock skew problems and still give a relatively linear time. A simple solution is to write an application that regularly broadcast the local time to the other nodes who add their local time and then save it. The TSB (Time Stamp Broadcaster) is kept running during the whole experiments and the synchronisation is performed during the post-run analysis.

```

# tsb - TimeStamp Broadcaster and listener
#
# ip 193.10.133.40
# mask 255.255.255.128
# UDP broadcasting at 193.10.133.127/9332
# delay 10 sec
#
# received-at received-from sent-at
985339178.605728 193.10.133.76 985358236.400775
985339178.900611 193.10.133.78 985358045.60742
985339179.642631 193.10.133.51 985358746.425765
985339179.787813 193.10.133.45 985358484.187402

```

Table 2: Example of a log-file from the TSB

### 5.2.5 What to log and what not to log

One of the major issues when doing experiments with networks is what data that's important to log. Everything about everything would be the perfect way but since we only have a few MB of free space to play with we have to come up with something better.

In order to follow "better to be safe than sorry" we tried to log as much as possible from the software we ran during the tests. We dropped the idea of logging whole packets to the disk since we were short of disk-space and doubted we would need the data within the packets.

The WaveLAN driver was patched in order to measure the signal level for each incoming packet. The information from which host the packet came from and when it arrived is then added in the spydaemon which appends everything to a log-file, called the spylog.

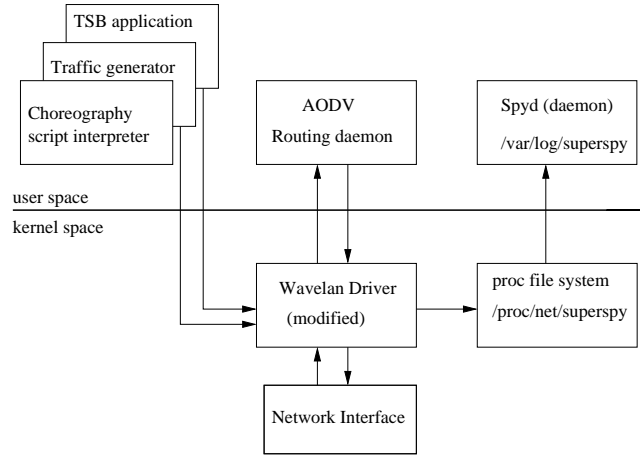


Figure 3: View of how the internal daemons and interpreter of the testbed work

```

**** Log for spy with address: 00:60:1D:F6:FB:A0, Fri Mar 23 10:21:27 2001
1 985339287.428371 00:02:2D:02:89:4A 1
2 985339287.432445 00:60:1D:F7:29:A3 30
3 985339287.450908 00:60:1D:F6:59:FA 5
4 985339287.452116 00:02:2D:02:89:4A 12
5 985339287.454274 00:60:1D:F6:FB:AF 4

```

Table 3: Example from a spylog-file

For the purpose of testing the virtual mobility metric we only used ping to create traffic in the network. When ping was active we saved each answer together with a timestamp. In order to be able to see the routing daemons view of the network and possibly find differ-

ences with did the same as with the data from the ping commands. In order to effectively use the disk-space all the logs are compressed by filtering them through gzip in real-time.

```
# Ping - logfile

Fri Mar 23 10:21:28.998176 2001
PING 193.10.133.41 (193.10.133.41) from 193.10.133.40 : 56(124) bytes of data.
# time=985339289.265846
64 bytes from 193.10.133.41: icmp_seq=0 ttl=255 time=10.6 ms
RR:      193.10.133.40
         193.10.133.41
         193.10.133.41
         193.10.133.40
# time=985339289.723344
64 bytes from 193.10.133.41: icmp_seq=1 ttl=255 time=4.5 ms      (same route)
# time=985339290.223359
64 bytes from 193.10.133.41: icmp_seq=2 ttl=255 time=3.9 ms      (same route)
# time=985339290.722748
64 bytes from 193.10.133.41: icmp_seq=3 ttl=255 time=3.9 ms      (same route)
```

Table 4: the pinglog

### 5.2.6 Experiment workflow

When you have come up with a scenario that you want to verify with an experiment you want to spend as little time as possible on the administrative parts. Still, you want to have complete control over all the crucial logged data. Doing this in a large scale and repeating tests many times you realize there is a need for some kind of “production-line” and that most of the tasks must be automated.

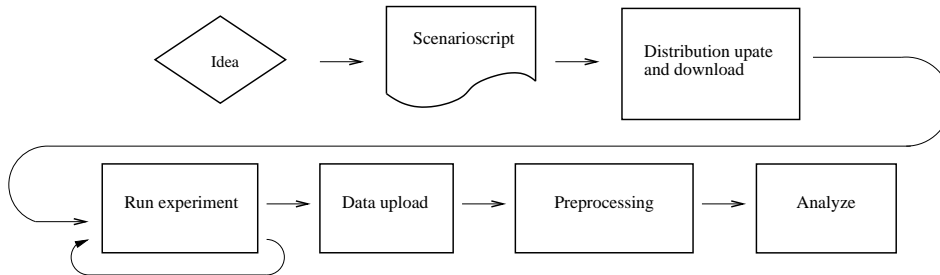


Figure 4: The experiment workflow

Figure 4 show work-flow for a typical experiment with the testbed. The first thing to do is create a scenario-script which contain instructions for all participants in the experiment. During the experiment a script run on each node that parse the scenario-script and tells the “monkey”, i.e. the participant what to do. After the experiment is over all the logs from each node are uploaded to a central node where the logs can be processed and analysed.

When uploading the logs an additional file contain nothing more than a timestamp of the nodes clock are also included. When uploaded it has a modification-time from when it was uploaded so then it is a very simple task to calculate the difference in time between the clocks and the correct the log-files so they are synchronised with the rest of the world. This simple trick which gives an accuracy of a couple a seconds enables automatic sorting of all the log-files uploaded since we now know the starting time of all test-runs.

When all log-files have been sorted according to their test-run a special script then combines logs from all nodes to a single file. This combined log-file is then used to calculate virtual mobility, density, routing performance etc. Also created during the development is a small application called ApeView where the nodes are displayed an animated according to the data in the combined In [117] E. Nordström describe the analysis phase of the testbed.

### 5.3 Metric toolbox

In order to compare two or more different protocols there is a need for metrics that describe the characteristics of the topology during the experiments. The simulation-environment is a very quiet and stable world so there is a need for extending the metrics used in simulations to the unstable and very unpredictable reality.

Another problem in the real world is that positioning of the nodes is a difficult problem that earlier real world experiments [101], [102], [90] have used GPS to be able keep track of the nodes. In an indoor environment such devices don't work and since the goal also includes large scale experiments you have to multiply every small add-on with 50 in our setup. Another way to view the problem is that it is not really necessary to know your exact position according to the real world. If we don't have any extra equipment that helps us with this the only thing we know about the nodes in the surrounding world is the signal level on the incoming packets. From the nodes' inverse view of the world we may deduce a couple of simple facts that helps us a little bit.

#### 5.3.1 Traffic description

Perhaps the most important thing in an ad hoc network is the traffic. This because that's often the only source of information that tells a node something about the surrounding environment. This is handled in different ways in ad hoc protocols and in the different hardware you run this on ex WaveLAN [74], Bluetooth [82]. A few metrics for each node in range is needed to have a simple description of how the signal level and traffic behaves.

#### 5.3.2 Connectivity & relative density

When we have information of which our neighbouring nodes are and how we recognise the links to them we may deduce how our *connectivity* (5) is i.e. how many nodes we are in contact with.

$$C^t(node_i) = n \in S \mid s(n) > 0 \quad (5)$$

$S$  is the number of nodes in the network and  $s(n)$  is the signal quality of all packets received from  $node_n$  during time interval  $t$ .

Secondly we are also able to calculate how *dense* (9) the network is around us by trying to calculate the "physical distance" to the other nodes by converting the signal level.

Since the frequencies used today are very affected by the materials in the building it is not very accurate when used indoors but it gives a slight hint about the relative distance. Since we base our calculation of the "physical distance" on our *measured* signal quality the result will be our "virtual" view of the the measured node.

For the calculation we used the path loss model mentioned in [98] to relate signal strength and distance. It Is been created to fit an indoor environment and is divided into two parts depending on the distance between the nodes. We chosed the far field definition since our experiments focus on large movements and long distances with nodes going out of reception range and the difficulty of measuring a stable signal level at short range.

$$Q \text{ in dB} = \alpha - 33 * \log(\text{dist}/\beta) \quad (6)$$

which, after calibration with the signal level range of the WaveLAN card and our measurements, results in the following inverse path loss formula:

$$D_j(node_i) = 4 * 10^{\frac{40 - 0.9 * Q_j(node_i)}{33}} \quad (7)$$

where  $Q$  is the WaveLAN signal quality (0..75) for a packet received from node  $j$  at node  $i$ .  $D$  is in the range of 0.5 to 65 m. The "virtual distance" between node  $node_i$  and  $node_j$  during a time interval is calculated for  $node_i$  as follows:

During time interval  $t_k$  the link quality for all packets heard from  $node_j$  are summed and divided with the number of packets heard from  $node_j$  during this interval. We define  $D_j^k$ , the mean virtual distance from  $node_j$  for time slot  $t_k$  as

$$D_j^k(node_i) = \frac{1}{N_j^k} \sum_{a=1}^{N_j^k} D_j^a \quad (8)$$

where  $N_j^k$  is the number of packets received from  $node_j$  during  $t_k$  and  $D_j^a$  is the virtual distance between  $node_a$  to  $node_j$  calculated for the packet  $a$  received by  $node_a$  during interval  $t_k$ . It is now possible to calculate the density of nodes around the measured node mentioned before with:

$$D_{avg}^t(node_i) = \frac{1}{C^t} \sum_{a=1}^{C^t} D_a^t \quad (9)$$

### 5.3.3 Virtual mobility

By dividing the time in discrete intervals and measure the mean virtual distance for each of them it possible to capture the relative physical movement between the nodes or the nodes mobility. More formally, the virtual distance from  $node_j$  to  $node_i$  during time interval  $t_{k+1}$  is the change in mean link quality, namely

$$vM_j^{k+1}(node_i) = |D_j^{k+1}(node_i) - D_j^k(node_i)| \quad (10)$$

Thus its possible to call this metric *Virtual mobility* between  $node_j$  to  $node_i$ .

A more general description for a network with S number of nodes the average virtual mobility perceived by  $node_i$  at time  $t_k$  is calculated by

$$vM_{avg}^k(node_i) = \frac{1}{S} \sum_{l=1}^S vM_l^k(node_i) \quad (11)$$

The choice of how to define which nodes that belong to the network and thus should be included in the network is discussed in [118]. Finally, let *network* virtual mobility for time  $t_k$  be

$$vM^k = \frac{1}{N} \sum_{i=1}^N vM_{avg}^k(node_i) \quad (12)$$

where N is the number of nodes in the network. This definition refer to the *mean* network vM-value at each time interval and represents the average relative movement for a node during the defined time. A more extensive use of this metric is explored in [118].

## 5.4 Conclusion

With the metrics connectivity, density and virtual network mobility, we now have a way to describe some basic properties of the network which could be very usefull. The virtual network mobility could be called a “fingerprint” of the networks’ topology.

## 5.5 Functional scenarios

When verifying routing protocols and metrics there is a need for scenarios that bring forward the important properties. We start with very simple scenarios and then move on to more advanced for verifying the virtual mobility.

### 5.5.1 Simple scenarios

To evaluate the most basic functions we have used a few simple scenarios that measure route discovery and routing performance.

#### Basic route discovery



Figure 5: Route discovery scenario

One of the most important characteristics of an ad hoc protocol is the ability to lookup a route fast despite the mobile environment. One simple way to test this feature is to create a simple scenario like fig 5 consisting of  $n$  number of nodes. To send a simple ping to these  $n$  nodes under different circumstances will give a view of the performance of the lookup function in the protocol-implementation.

#### Routing performance

When a route is established there are a few simple tests that can be made to verify the routing-performance of the implementation of the protocol. Measuring End-To-End time and Round-Trip-Time (RTT) for a packet gives a hint of how fast the implementation is. Extending this into pinging the network with packets at different speed and with different sizes gives information about the routing-capacity along the chosen route.

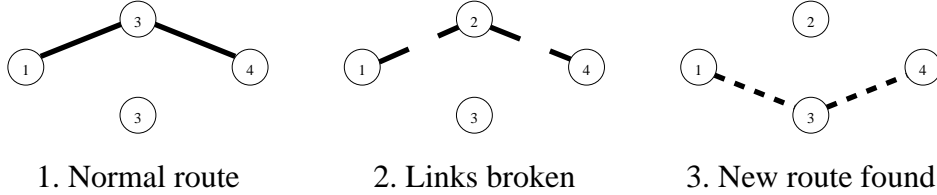


Figure 6: Basic rerouting scenario

#### Simple rerouting

Another very crucial function to test is the ability to recreate the route to the destination if one of the used links fails. This is easy to test with a simple rhomb-scenario fig 6. It consist of four nodes placed like a rhomb, where node 1 is unable to hear node 4 and node 2 and 3 hear both 1 and 4. A connection is established between node 1 and 4 trough either 2 or 3. One of the two active links can then the broken in order to force the protocol to create a new route to the destination. Usually this is done by shutting down the radio on either node 2 or 3 and that will brake both active links. This will force the protocol to re-establish the route through the only still working node in contact between node 1 and 4. If only one of the active links are broken there is a possibility that the new route may be created with the old active link included.

### 5.5.2 Virtual mobility scenarios

All the work around the new virtual mobility metric resulted in that the scenarios chosen for the experiments was to examine how the vM reacted under different circumstances.

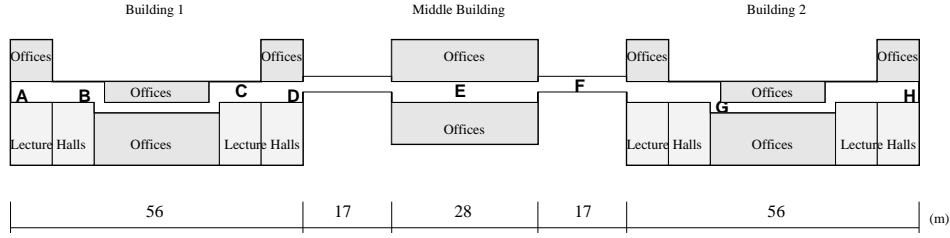


Figure 7: A schematic view of the physical environment used for testing the scenarios. The white paths are the corridors and the letters [A..H] the positions used in the scenarios.

### Global warming

This was our first basic scenario we tried to see how the vM reacted when more and more nodes were added to the network. Secondly we wanted to see what happened with vM and the throughput when the maximum capacity of the network was reached, i.e. the air is “heated”. Another goal with this scenario was to have this as an introductory where you verify that everything is working before continuing to more complex scenarios.

The test starts with the first node pinging the second for a predefined time. Then the second starts to ping the third group and so on until every group is pinging the next one. Figure 8.

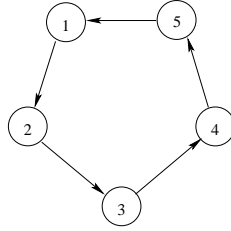


Figure 8: Global Warming

### Lost’n’Found

The purpose with this scenario was to see how the movement of the nodes in the network and the link breakages affected the vM.

In this scenario the nodes are divided into two groups standing together at start. Then one of the groups move out of range so all links break and then pause for a while. Then the groups are reunited so they end in the positions just like they started. A more detailed description is found in [118].

### Double Lost’n’Found

To extend the Lost’n’Found scenario this scenario was created to evaluate how the vM metric could be used to visualise different movement patterns in the network. It starts out like the last scenario but the two “away groups” reunite at different times. The theory behind this was that the vM during the two second group movements would yield less vM than the first one since there were less moving nodes at the same time. During the analysis of this scenario it was discovered that by computing the upper and lower quantile of the vM we were able to show how homogeneous or heterogeneous the mobility in the network is. A more detailed description of this is also found in [118].

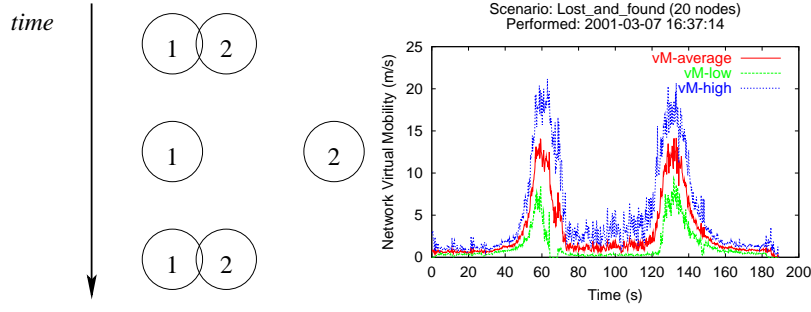


Figure 9: Lost'n'Found

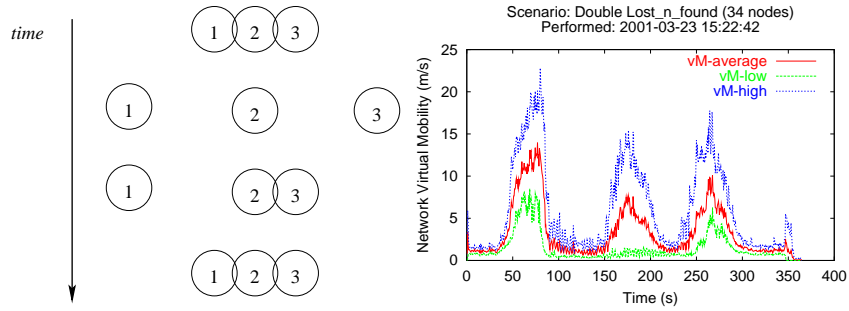


Figure 10: Double Lost'n'Found

### Double split

This scenario was created to enable a multi-hop environment and see how the routing daemon performed when the link qualities were weak and fluctuating. The two groups farthest out move away from the centre as in Figure 11 to break the existing links and force the routing daemon to go multi hopping. See [118] for more details.

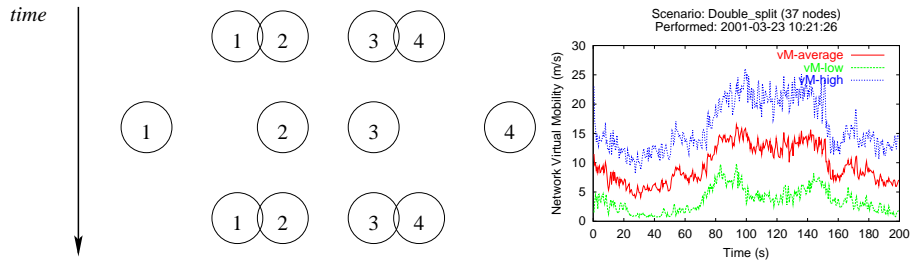


Figure 11: The *double split* scenario with a graph of the virtual mobility.

## 5.6 Experiences from scaling up experiments

When conducting larger and larger experiments a number of different non-science problems start to appear. As said we aimed for doing a test-run with 50 participants but that was literarily easier said than done. Our approach was to begin with very small tests and then move with larger and larger groups until the 50 node test.



### **5.6.1 Handling a lot of participants and hanging nodes**

To cope with the administrative problem of handle everything within and after the test-run we made a bunch of scripts to minimise the chance of errors caused by a confused monkey(participant). We realized that at some point we would have to face the problem with a node hanging during the test-run. We chosed the solution to just ignore the hang during the test-run and then filter this node during the post-run analysis. The risk with this solution is that the hanging node may have had a lot of influence on the network during its uptime. This may cause a potential error during the analysis phase if the data is not handled correctly.

Another problem that's easy to forget and may cause loss of quality of the test-runs is the need for things to make the monkeys feel that they are participating in an important test. Thus is important to tell them about the goal of the experiment and maybe show results from previous experiments. To add extra motivation for them it is very good to have a Swedish fika or something similar.

### **5.6.2 In-lab testing with troublesome equipment is important**

In-lab testing of how our testbed performed on different hardware proved to be very important. The PCMCIA [83] package seemed to be a bit unreliable since we had quite some trouble with the AST (B.2) machines where at least one during each test-run hanged. For our 50 node experiment we where confronted with a number of new Dell (B.2) machines that we never tried the testbed in. During the earlier experiments as in the 50 node one most of the participants had an older Dell model (B.2) which worked fine but these machines where incompatible with the used PCMCIA package and could not be used.

### **5.6.3 Time is expensive**

When doing large experiments as we did it is important to keep in mind that an hour lost during the experiment-time is roughly equal to one week of working hours.

Since time is expensive during the experiment-time there is a need for good strict organisation where everyone know what area they are responsible for, e.i. helping with installations, inform about the scenarios etc. If the monkeys get bored or confused there is a lot bigger chance for errors during the test-runs so make clear for the monkeys who is responsible for what so they know what's going on.

### **5.6.4 Small matters take a lot of time**

Apart from these "on-site" issues it is worth mentioning that getting all the people and equipment to the experiment showed to be a lot of work. Foreseeing all problems that may arise during the experiment is not an easy task. A little example is how to minimise the chance of a laptop running out of battery during a test-run. The solution is very simple at a first glance. Simply supply them with power during the breaks! On a second thought it requires some time to solve this except buying enough power outlets. If we have 50 nodes connected where each of them consumes about 50W they will need 2.5kW all together and together with an ordinary OH-projector it sums up to about 3kW. This requires two different groups of power in order not to blow a fuse or start a fire and that's not easy to find in one single room...

## 6 Conclusion

### 6.1 Ad hoc protocol evaluation

- Mobility models not accurate enough

In [96] results show that real ad hoc environment cannot be simulated with a “random walk” type of mobility model. The more advanced models that exists provide a much better simulation than the random walk types but have the disadvantage that they demand much more computation time. Models like the ECR have great possibilities but have shown very hard to control to get the desired motion patterns.

- Mobility models not used combined

The movements of the nodes in an ad hoc network is very essential for the routing protocol and thus the mobility model is important for simulation results. This is not however not recognized in the simulations studied where they only have used one of them.

- Metrics not advanced enough to capture needed properties

Whereas the geometric mobility metric give us a view of how mobile the nodes are physically the minimal route-change metric give us a view how stable the links in the network. A high mobility but at the same time low route-change could describe a network where the almost all nodes move, but not enough to potentially affect the routes.

- The propagation models used are too simple

The simulations that have been done have been using a very simplified model of the real world environment. Most of the simulations have been using FSPL [97] and TRGR [97] to simulate the wireless environment which is a simple approximation of the complex real world. A more accurate model is SIRCIM [78] which is better but the computation is heavy and it has to be tuned to each specific scenario.

- Frequency related modelling is required

When a person is standing in the LOS between two nodes equipped with WaveLAN 2.4GHz cards the signal is dampened with about 15 dBm which is equal about 20 meters in distance. This give a hint that propagation models for objects that absorb radiation will be required to do simulation of scenarios like a conference where a lot of people will be moving around.

### 6.2 Discussion of protocol evaluation

- The sum of the discussion about simulation is that neither the mobility or propagation models used in simulation are mature enough to give a accurate result that could compare with an real environment.

When an ad hoc grow larger and larger it can be assumed that the topology no longer will be homogenous, i.e. in the respect of node distribution and their mobility. This is not reflected in most of the simulations performed and it would probably yield very interesting results since it introduces new demands on the routing-protocol.

The metric virtual mobility described in this paper captures a view of an ad hoc network that reflect the nodes view of the network. This is very orthogonal to the geometric views often used.

The metrics described in the report could be used for other things than to tell a good protocol from a bad, but could also be calculated in real-time within a running network to help to decide what type of routing that fit all/some of the nodes depending on their demands/properties. An example of this is discussed later.

- A field not deeper investigated is what traffic to test in an ad hoc network. Depending on the purpose of a special protocol it is important to try to define what type of traffic patterns (between what nodes and when) and flows (bursty/flat) that will be used in the network. Networks with continuous traffic are much easier to create good performance in through the continuous feed of information in the network about the overall connectivity and mobility in the network.
- The very different demands on a ad hoc protocol makes it almost impossible to come up with single simple protocol that could fit all purposes. Very small nodes in e.g. sensor networks need to have small simple routing algorithms to be cheap to produce while self-configuring large networks for residential areas could use high output power, lot of frequencies etc. An ad hoc protocol that can be used in different environments have to be adaptive and self-configurative.

One possible future scenario is the residential area with a number of lampposts with integrated ad hoc nodes. Travelling on the streets are cars equipped with nodes and on the sidewalk pedestrians walk around with their small battery-driven PDAs.

Here we can identify three different types of roles:

- Backbone (high routing speed)
- Big mobile node (high bandwidth demand)
- Small mobile node (battery)

The nodes in the lampposts could make use of the fact that they have each-other as neighbours with very stable links. These could be used as an reliable backbone with little routing overhead among each-other.

A passenger in the car may want to watch some video-stream with the latest news. A high bandwidth requirement and relatively fast changes of links creates a higher routing overhead.

Finally we have a pedestrian with his battery driven PDA running a group-ware with email and calendar etc. The goal here is to use as little power as possible and the applications used don't require continuous bandwidth. The lower rate of link changes to other nodes also add to the possibility to save power by talk to other nodes as seldom as possible.

When a mobile node is moving through the network the need for seamless hand-overs is essential for many types of services. To be able to accomplish this ways to predict link loss is required, i.e. it's better to change route before loss of a link than after. Most of the existent protocol proposals have no built in mechanism to handle this but [91] discuss one solution.

To round off this discussion and my master's thesis report I only have one more thing to add. My personal thought is that the work with ad hoc networks will be the foundation for a new generation of protocols that are selfconfigurable and much more adaptive than the ones used today. I finish with a quote from [93] written by Elizabeth Royer and I fully agree.

"Flexibility is very important since many of the protocols have specific uses. Adaptivity and self-configuration are key features but non-trivial to combine."

### **6.3 Acknowledgements**

Per Gunningberg  
Christian Tschudin  
Henrik Lundgren  
Mats Björkman  
Josh Fennessy  
Richard Gold  
Björn Knutsson  
Erik Nordström  
Arnold Pears  
Pontus Sköld

## **A Protocol abstracts**

Since drafts are removed from IETF when they are not continually supported and therefore are very hard to find I have collected abstracts from drafts I have come across in order to create a small reference.

### **A.1 Associativity-Based Routing (ABR) [31]**

This document describes the associativity-based long-lived routing (ABR) protocol for ad hoc mobile networks. It is a simple and bandwidth-efficient distributed routing protocols which does not attempt to consistently maintain routing information in every node. In an ad hoc wireless network where mobile hosts are acting as routers and where routes are made inconsistent by mobile hosts' movement, we propose an Associativity-based routing scheme where a route is selected based on nodes having associativity states that imply periods of spatial, temporal, connection and signal stability. In this manner, the routes selected are likely to be long-lived and hence there is no need to restart frequently, resulting in higher attainable throughput. Our proposed protocol is based on source-initiated on-demand routing. Route requests are broadcast on a per-need basis. To discover shorten the route discovery time when the association property is violated, the localized- query and quick-abort mechanisms are respectively incorporated into the protocol. The association property also allows the integration of ad hoc routing into a base station oriented wireless LAN environment, providing the fault tolerance in times of base station failures. This draft will describe the protocol functions and information about packet headers and routing tables.

### **A.2 Adaptive Demand-Driven Multicast Routing protocol (ADMR) [47]**

Adaptive Demand-Driven Multicast Routing protocol (ADMR) has been designed specifically for use in the ad hoc network environment. Multicast routing state in ADMR is dynamically established and maintained only for groups that have at least one receiver and one active sender in the network. Each multicast data packet is forwarded along the shortest-delay path with multicast forwarding state, from the sender to the receivers. Senders are not required to announce their intention to start or stop sending data to the group, or to join the group to which they wish to send. Receivers dynamically adapt to the sending pattern of senders and mobility in the network in order to efficiently balance overhead and maintenance of the multicast routing state as nodes in the network move or as wireless transmission conditions in the network change. State for groups whose senders have become inactive or whose receivers have left the group is expired automatically without the need for control signaling or application-level notification at the source. ADMR also detects when mobility in the network is too high to efficiently maintain multicast routing state, and instead reverts to flooding for a short period of time it determines that the high mobility has subsided.

### **A.3 Ad hoc On Demand Distance Vector protocol (AODV) [4]**

The Ad Hoc On-Demand Distance Vector (AODV) routing protocol is intended for use by mobile nodes in an ad hoc network. It offers quick adaptation to dynamic link conditions, low processing and memory overhead, low network utilization, and determines unicast between sources and destinations. It uses destination sequence numbers to ensure loop freedom at all times (even in the face of anomalous delivery of routing control messages), solving problems (such as "counting to infinity") associated with classical distance vector protocols.

#### **A.4 Bordercast Resolution Protocol (BRP) [25]**

The Bordercast Resolution Protocol (BRP) provides the bordercasting packet delivery service used to support network querying applications. The BRP uses a map of an extended routing zone, provided by the local proactive Intrazone Routing Protocol (IARP), to construct bordercast (multicast) trees, along which query packets are directed. Within the context of the hybrid ZRP, the BRP is used to guide the route requests of the global reactive Interzone Routing Protocol (IERP). The BRP employs special query control mechanisms to steer route requests away from areas of the network that have already been covered by the query. The combination of multicasting and zone based query control makes bordercasting an efficient and tunable service that is more suitable than flood searching for network probing applications like route discovery.

#### **A.5 Core Extraction Distributed Ad hoc Routing (CEDAR) Specification [51]**

This draft presents CEDAR, a Core-Extraction Distributed Ad hoc Routing algorithm for QoS routing in ad hoc network environments. CEDAR has three key components: (a) the establishment and maintenance of a self-organizing routing infrastructure, called the "core", for performing route computations, (b) the propagation of the link-state of stable high-bandwidth links in the core through "increase/decrease" waves, and (c) a QoS route computation algorithm that is executed at the core nodes using only locally available state.

##### **1. Establishment and Maintenance of a core using Local Core Extraction**

CEDAR does core extraction in order to extract a subset of nodes in the network that would be the only ones that perform state management and route computation. The core extraction is done dynamically by approximating a minimum dominating set of the ad hoc network using only local computation and local state. The core computation and core management upon change in the network topology are purely local computations to enable the core to adapt efficiently to the dynamics of the network.

##### **2. Link State Propagation using Increase/Decrease waves**

While it is possible to execute ad hoc routing algorithms using only local topology information at the core nodes, QoS routing in CEDAR is achieved by propagating, in the core, the bandwidth availability information of stable links. The basic idea is that the information about stable high-bandwidth links can be made known to core nodes far away in the network, while information about dynamic links or low bandwidth links should remain local. The key questions to answer in link state propagation are: when should an increase/decrease wave be initiated, how far should a wave propagate, and how fast should a wave propagate.

##### **3. Route Computation**

Route computation first establishes a core path from the domain of the source to the domain of the destination. This initial phase involves probing on the core, and the resultant core path is cached for future use. The core path provides the directionality of the route from the source to the destination. Using this directional information, CEDAR iteratively tries to find a partial route from the source to the domain of the furthest possible node in the core path.

#### **A.6 Cluster Based Routing Protocol (CBRP) [8]**

Cluster Based Routing Protocol (CBRP) is a routing protocol designed for use in mobile ad hoc networks. The protocol divides the nodes of the ad hoc network into a number of overlapping or disjoint clusters in a distributed manner. A cluster head is elected for each cluster to maintain cluster membership information. Inter-cluster routes are discovered dynamically using the cluster membership information kept at each cluster head. By clustering nodes into groups, the protocol efficiently minimizes the flooding traffic during route

discovery and speeds up this process as well. Furthermore, the protocol takes into consideration of the existence of uni-directional links and uses these links for both intra-cluster and inter-cluster routing.

### **A.7 Differential Destination Multicast (DDM) Specification [20]**

This draft describes a multicast routing protocol for mobile ad hoc networks (MANETs). The protocol termed Differential Destination Multicast (DDM) differs from common approaches proposed for ad hoc multicast routing in two ways. Firstly, instead of distributing membership control throughout the network, DDM concentrates this authority at the data sources (i.e. senders) thereby giving senders knowledge of group membership. Secondly, differentially-encoded, variable-length destination headers are inserted in data packets which are used in combination with unicast routing tables to forward multicast packets towards multicast receivers. Instead of requiring that multicast forwarding state be stored in participating nodes, this approach also provides the option of stateless multicasting. Each node independently has the choice of maintaining cached forwarding state, or requesting its upstream neighbor to insert this state into self-routed data packets, or some combination thereof. The protocol is best suited for use with small multicast groups operating in dynamic networks of any size.

### **A.8 Dynamic Source Routing protocol (DSR) [41]**

The Dynamic Source Routing protocol (DSR) is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. DSR allows the network to be completely self-organizing and self-configuring, without the need for any existing network infrastructure or administration. The protocol is composed of the two mechanisms of "Route Discovery" and "Route Maintenance", which work together to allow nodes to discover and maintain source routes to arbitrary destinations in the ad hoc network.

The use of source routing allows packet routing to be trivially loop-free, avoids the need for up-to-date routing information in the intermediate nodes through which packets are forwarded, and allows nodes forwarding or overhearing packets to cache the routing information in them for their own future use. All aspects of the protocol operate entirely on-demand, allowing the routing packet overhead of DSR to scale automatically to only that needed to react to changes in the routes currently in use. This document specifies the operation of the DSR protocol for routing unicast IP packets in multi-hop wireless ad hoc networks.

### **A.9 DSR Simple Multicast and Broadcast protocol (DSR-MB) [42]**

The protocol specified in this document is designed to provide multicast and broadcast functionality in mobile ad hoc networks. It utilizes the Route Discovery mechanism defined by the Dynamic Source Routing protocol (DSR) to flood the data packets through the network. Although this protocol is derived from DSR, it can be implemented as a stand-alone protocol. In fact, it does not rely on unicast routing to operate. If DSR is already implemented on the network, very minor modifications are required to enable this protocol.

This multicast and broadcast protocol utilizes controlled flooding to distribute data in the network and does not require the establishment of state in the network for data delivery. It is not intended as a general purpose multicast protocol. Its applicability is mainly in environments characterized by very high mobility or by a relatively small number of nodes. In the former case, protocols relying on the establishment of multicast state perform inadequately because they are unable to track the rapid changes in topology. In the latter case, the overhead of keeping multicast state exceeds the overhead of flooding.

### **A.10 DSR-Flow protocol [43]**

This document defines an extension to the Dynamic Source Routing protocol (DSR), a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. DSR enables the sender of a packet to determine the sequence of nodes through which the packet must be forwarded to reach the intended destination node, and to route that packet along that sequence of hops by including a source route header in the packet.

All aspects of the protocol operate entirely on-demand, allowing the routing packet overhead of DSR to scale automatically to only that needed to react to changes in the routes currently in use. The DSR extension defined in this document, known as "flow state", allows the routing of most packets without an explicit source route header in the packet, further reducing the overhead of the protocol while still preserving the fundamental properties of DSR's operation.

### **A.11 Fisheye State Routing protocol (FSR) [38]**

The Fisheye State Routing (FSR) algorithm for ad hoc networks introduces the notion of multi-level "scope" to reduce routing update overhead in large networks. A node stores the Link State for every destination in the network. It periodically broadcasts the Link State update of a destination to its neighbors with a frequency that depends on the hop distance to that destination (i.e., the "scope" relative to that destination). State updates corresponding to far away destinations are propagated with lower frequency than those for close by destinations. From state updates, nodes construct the topology map of the entire network and compute efficient routes.

The route on which the packet travels becomes progressively more accurate as the packet approaches its destination. FSR resembles Link State routing in that it propagates Link State updates. However, the updates are propagated as aggregates, periodically (with period dependent on distance) instead of being flooded individually from each source. FSR leads to major reduction in link O/H caused by routing table updates. It enhances scalability of large, mobile ad hoc networks.

### **A.12 Host Specific Routing (HSR) [18]**

This memo overviews the need for intra-domain Host specific routing (HSR) in the Internet. Host Specific Routing provides a number of benefits if route entry and look-up scalability issues can be adequately addressed. These benefits are the enabling of flat routing domains that eliminate the need for hierarchy and associated configuration, and the potential to support rapid movement of IP addresses through the routing fabric. This draft describes some of the current work in this area, including TORA and Wireless Internet Protocol (WIP), and the as yet unresolved research issues associated with large-scale host routing. This draft requires and makes no topological assumptions for HSR. Specifically it does not require a strict tree, as implied by CIP and HAWAII. These micro-mobility protocols do however share many of the scalability and inter-protocol issues associated with host specific routes.

### **A.13 Intrazone Routing Protocol (IARP) [26]**

This document describes the Intrazone Routing Protocol (IARP), a limited scope proactive routing protocol used to improve the performance of existing globally reactive routing protocols. With each node monitoring changes in its surrounding R-hop neighborhood (routing zone), global route discoveries to local destinations can be avoided. When a global route search is needed, the IARP's routing zones can be used to efficiently guide route queries outwards (via bordercasting) rather than blindly relaying queries from neighbor to neigh-



bor. The proactive maintenance of routing zones also helps improve the quality of discovered routes, by making them more robust to changes in network topology. Once routes have been discovered, IARP's routing zone offers enhanced, real-time, route maintenance. Link failures can be bypassed by multiple hop paths within the routing zone. Similarly, suboptimal route segments can be identified and traffic re-routed along shorter paths.

#### **A.14 Interzone Routing Protocol (IERP) [27]**

This document describes the Interzone Routing Protocol (IERP), the reactive routing component of the Zone Routing Protocol (ZRP). IERP adapts existing reactive routing protocol implementations to take advantage of the known topology of each node's surrounding R-hop neighborhood (routing zone), provided by the Intrazone Routing Protocol (IARP). The availability of routing zone routes allows IERP to suppress route queries for local destinations. When a global route discovery is required, the routing zone based bordercast service can be used to efficiently guide route queries outward, rather than blindly relaying queries from neighbor to neighbor. Once a route has been discovered, IERP can use routing zones to automatically redirect data around failed links. Similarly, suboptimal route segments can be identified and traffic re-routed along shorter paths.

#### **A.15 An Internet MANET Encapsulation Protocol (IMEP) Specification [30]**

This memo describes a multipurpose network-layer protocol, named the Internet MANET Encapsulation Protocol (IMEP), designed to support the operation of many routing algorithms, network control protocols or other Upper Layer Protocols (ULP) (where "upper" denotes \*any\* layer above IMEP) intended for use in Mobile Ad hoc Networks (MANET). The protocol incorporates mechanisms for supporting link status and neighbor connectivity sensing, control packet aggregation and encapsulation, one-hop neighbor broadcast (or multicast) reliability, multipoint relaying, network-layer address resolution and provides hooks for interrouter authentication procedures. Indirectly, the IMEP also puts forth a framework for MANET router and interface identification and addressing.

#### **A.16 Landmark Routing Protocol for Large Scale Networks (LANMAR) [7]**

The Landmark Routing Protocol (LANMAR) utilizes the concept of "landmark" for scalable routing in large, mobile ad hoc networks. It relies on the notion of group mobility: i.e., a logical group (for example a team of coworkers at a convention) moves in a coordinated fashion. The existence of such logical group can be efficiently reflected in the addressing scheme. It assumes that an IP like address is used consisting of a group ID (or subnet ID) and a host ID, i.e. <Group ID, Host ID>. A landmark is dynamically elected in each group. The route to a landmark is propagated throughout the network using a Distance Vector mechanism. Separately, each node in the network uses a "scoped" routing algorithm (e.g., FSR) to learn about routes within a given (max number of hops) scope. To route a packet to a destination outside its scope, a node will direct the packet to the landmark corresponding to the group ID of such destination. Once the packet is within the scope of the landmark, it will typically be routed directly to the destination.

Remote groups of nodes are "summarized" by the corresponding landmarks. The solution to drifters (i.e., nodes outside of the scope of their landmark) is also handled by LANMAR. Landmark dynamic election enables LANMAR to cope with mobile environments. Thus, by using the truncated local routing table and the "summarized" landmark distance vector, LANMAR dramatically reduces routing table size and update overhead in large nets. LANMAR is well suited to provide an efficient and scalable routing solution

in large, mobile, ad hoc environments in which group behavior applies and high mobility renders traditional routing schemes inefficient.

#### **A.17 Lightweight Underlay Network Ad hoc Routing (LUNAR) [3]**

LUNAR (Lightweight Underlay Network Ad hoc Routing) is an on-demand routing system for wireless ad hoc IP networks. It incorporates an explicit "remote state patching" (RSP) approach and is based on extending ARP to do multihop name resolution for dynamically establishing a L2.5 route to the destination. LUNAR creates one or more virtual IP subnets supporting unicast as well as broadcast communications. The LUNAR system is self-configuring both at the level of routing as well as the level of IP interfacing: it handles node address assignment and default gatewaying automatically, enabling zeroconf forming and joining of an ad hoc network. LUNAR was designed for a simplified code base and easy extensibility of its functionality.

#### **A.18 Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing [50]**

The multicast operation of the Ad hoc On-Demand Distance Vector (AODV) routing protocol (MAODV) is intended for use by mobile nodes in an ad hoc network. It offers quick adaptation to dynamic link conditions, low processing and memory overhead, and low network utilization. It creates bi-directional shared multicast trees connecting multicast sources and receivers. These multicast trees are maintained as long as group members exist within the connected portion of the network. Each multicast group has a group leader whose responsibility is maintaining the group sequence number, which is used to ensure freshness of routing information.

#### **A.19 Mobile Mesh Border Discovery Protocol (MMBDP) [63]**

The Mobile Mesh Border Discovery Protocol (MMBDP) enables a mobile adhoc network to utilize a fixed/wired network for dissemination of routing information and for forwarding of data. MMBDP is one protocol in a set of related Mobile Mesh protocols that also includes the Mobile Mesh Link Discovery Protocol (MMLDP) and the Mobile Mesh Routing Protocol (MMRP). Together, these protocols provide a flexible, extensible mobile adhoc networking capability.

#### **A.20 Mobile Mesh Link Discovery Protocol (MMLDP) [64]**

The Mobile Mesh Link Discovery Protocol (MMLDP) provides a media independent mechanism for discovering neighbors in a mobile adhoc network, and is capable of determining whether links are unidirectional or bidirectional. MMLDP is one protocol in a set of related Mobile Mesh protocols that also includes the Mobile Mesh Routing Protocol (MMRP) and the Mobile Mesh Border Discovery Protocol (MMBDP). Together, these protocols provide a flexible, extensible mobile adhoc networking capability.

#### **A.21 Mobile Mesh Routing Protocol (MMRP) [65]**

The Mobile Mesh Routing Protocol (MMRP) is a robust, scalable, efficient mobile adhoc routing protocol based upon the "link state" approach. MMRP is one protocol in a set of related Mobile Mesh protocols that also includes the Mobile Mesh Link Discovery Protocol (MMLDP) and the Mobile Mesh Border Discovery Protocol (MMBDP). Together, these protocols provide a flexible, extensible mobile adhoc

### **A.22 Multicast Zone Routing protocol (MZR) [48]**

This document proposes a multicast protocol for Mobile Ad Hoc networks, called the Multicast routing protocol based on Zone Routing (MZR). MZR is a source-initiated on-demand protocol, in which a multicast delivery tree is created using a concept based on the zone routing mechanism. It belongs to the family of source-tree-based protocols, in which a delivery tree rooted at the source is created for each active multicast session. MZR does not depend on any underlying unicast protocol for a global routing substructure. The protocol's reaction to topological changes is restricted to a node's neighborhood and is not propagated throughout the network.

### **A.23 On-Demand Multicast Routing Protocol (ODMRP) [11]**

On-Demand Multicast Routing Protocol (ODMRP) is a multicast routing protocol designed for ad hoc networks with mobile hosts. ODMRP is a mesh-based, rather than a conventional tree-based, multicast scheme and uses a forwarding group concept (only a subset of nodes forwards the multicast packets via scoped flooding). It applies on-demand procedures to dynamically build routes and maintain multicast group membership. ODMRP is well suited for ad hoc wireless networks with mobile hosts where bandwidth is limited, topology changes frequently and rapidly, and power is constrained.

### **A.24 Optimized Link State Routing Protocol (OLSR) [39]**

This document describes the Optimized Link State Routing (OLSR) protocol for mobile ad hoc networks. The protocol is an optimization of the pure link state algorithm tailored to the requirements of a mobile wireless LAN. The key concept used in the protocol is that of multipoint relays (MPRs). MPRs are selected nodes which forward broadcast messages during the flooding process. This technique substantially reduces the message overhead as compared to pure flooding mechanism where every node retransmits each message when it receives the first copy of the packet.

In OLSR, information flooded in the network "through" these MPRs is also "about" the MPRs. Thus a second optimization is achieved by minimizing the "contents" of the control messages flooded in the network. Hence, as contrary to the classic link state algorithm, only a small subset of links with the neighbor nodes are declared instead of all the links. This information is then used by the OLSR protocol for route calculation. As a consequence hereof, the routes contain only the MPRs as intermediate nodes from a Source to a Destination. OLSR provides optimal routes (in terms of number of hops). The protocol is particularly suitable for large and dense networks as the technique of MPRs works well in this context.

### **A.25 Relative Distance Micro-discovery Ad Hoc Routing (RDMAR) Protocol [10]**

This document describes the Relative Distance Micro-discovery Ad Hoc Routing (RDMAR) protocol for use in mobile ad hoc networks (MANETs). The protocol is highly adaptive, bandwidth-efficient and scaleable. A key concept in its design is that protocol reaction to link failures is typically localised to a very small region of the network near the change. This desirable behaviour is achieved through the use of a novel mechanism for route discovery, called Relative Distance Micro-discovery (RDM). The concept behind RDM is that a query flood can be localised by knowing the relative distance (RD) between two terminals. To accomplish this, every time a route search between the two terminals is triggered, an iterative algorithm calculates an estimate of their RD, given an average nodal mobility and information about the elapsed time since they last communicated and their previous RD. Based on the newly calculated RD, the query flood is then localised to

a limited region of the network centred at the source node of the route discovery and with maximum propagation radius that equals to the estimated relative distance. This ability to localise query flooding into a limited area of the network serves to increase scalability and minimise routing overhead and overall network congestion.

#### **A.26 Source Routing-based Multicast Protocol (SRMP) [49]**

The Source Routing-based Multicast Protocol (SRMP) is designed to provide multicast functionality in mobile ad hoc networks. It applies the source routing mechanism defined by the Dynamic Source Routing (DSR) [1] in a modified manner decreasing the size of the packet header. SRMP obtains multicast routes on-demand through constructing a mesh (an arbitrary subnet) to connect group members providing robustness against mobility. This protocol minimizes the flooding scope using the Forwarding Group (FG) nodes concept [2]. The criteria used for selecting FG nodes allows the choice of stable paths with higher battery life. This protocol operates in a loop-free manner, minimizing channel overhead and making efficient use of network resources. The mesh-based approach avoids drawbacks of multicast trees, where multicast packets can be delivered to the destination in case of frequent node movements and channel fading. SRMP outperforms other multicast protocols by providing available paths based on future prediction for links state. These paths also guarantee nodes stability with respect to their neighbors, strong connectivity between nodes, and higher battery life.

#### **A.27 Source Tree Adaptive Routing protocol (STAR) [9]**

Unlike most of the other table-driven ad hoc protocols it does not use periodic messages to update its neighbours. STAR is an attempt to try to create the same routing performance as the other table-driven protocols and still be equal or better on the bandwidth efficiency. To be able to do this the demand on route optimization has been put aside and the routes are allowed to be non-optimal since. This saves bandwidth but STAR depends on an underlying protocol which must reliably keep track of the neighbouring nodes. This could be implemented with periodic messages, but it is not required to. In addition to this demand the link layer must provide reliable broadcasting or STAR will require this to be built into STAR with an extra routing-rule.

#### **A.28 Topology Broadcast Reverse-Path Forwarding protocol (TBRPF) [40]**

Topology Broadcast based on Reverse-Path Forwarding (TBRPF) is a proactive, link-state routing protocol designed for use in mobile ad-hoc networks. TBRPF has two modes: full topology (FT) and partial topology (PT). TBRPF-FT uses the concept of reverse-path forwarding to reliably and efficiently broadcast each topology update in the reverse direction along the dynamically changing broadcast tree formed by the min-hop paths from all nodes to the source of the update.

TBRPF-PT achieves a further reduction in control traffic, especially in large, dense networks, by providing each node with the state of only a relatively small subset of the network links, sufficient to compute minimum-hop paths to all other nodes. In both the FT and PT modes, a node forwards an update only if the node is not a leaf of the broadcast tree rooted at the source of the update. In addition, in the PT mode, a node forwards an update only if it results in a change to the node's source tree. As a result, each node reports only changes to a relatively small portion of its source tree.

#### **A.29 Temporally-Ordered Routing Algorithm protocol (TORA) [28]**

This document provides both a functional description and a detailed specification of the Temporally-Ordered Routing Algorithm (TORA)—a distributed routing protocol for mul-

tihop networks. A key concept in the protocol's design is an attempt to de-couple the generation of far-reaching control message propagation from the dynamics of the network topology. The basic, underlying algorithm is neither distance-vector nor link-state; it is a member of a class referred to as link-reversal algorithms. The protocol builds a loop-free, multipath routing structure that is used as the basis for forwarding traffic to a given destination. The protocol can simultaneously support both source-initiated, on-demand routing for some destinations and destination-initiated, proactive routing for other destinations.

### **A.30 Zone Routing Protocol (ZRP) [24]**

This document describes the Zone Routing Protocol (ZRP) framework, a hybrid routing framework suitable for a wide variety of mobile ad-hoc networks, especially those with large network spans and diverse mobility patterns. Each node proactively maintains routes within a local region (referred to as the routing zone). Knowledge of the routing zone topology is leveraged by the ZRP to improve the efficiency of a globally reactive route query/reply mechanism. The proactive maintenance of routing zones also helps improve the quality of discovered routes, by making them more robust to changes in network topology. The ZRP can be configured for a particular network by proper selection of a single parameter, the routing zone radius.

## **B Techno stuff**

This is a short summary of the hardware and software used when creating the APE-testbed.

### **B.1 Software**

- Linux RedHat [68] distribution version 6.2
- PCMCIA package [83] version 3.1.25
- Linux Kernel [70] version 2.2.18
- Lucent WaveLAN driver [74] version 6.06

### **B.2 Hardware**

- AST Ascentia P series Intel Pentium 133 MHz  
24 Mb RAM  
2 GB harddrive
- Dell Latitude CPi PentiumII 233 MHz  
64 MB RAM  
3 GB harddrive
- Dell Latitude PentiumIII 500 MHz  
128 MB RAM  
6 GB harddrive
- WiFi equipment  
Lucent Orinoco (WaveLAN) Silver [74]  
802.11b [81], 2.4 GHz, 15 dBm, 11 MBit 56 bit WEP

## References

- [1] *The Ad hoc Protocol Evaluation testbed from Uppsala University*,  
URL: <http://apetestbed.sourceforge.net>
- [2] *The Official IETF working group Manet webpage*,  
URL: <http://www.ietf.org/html.charters/manet-charter.html>
- [3] CHRISTIAN TSCHUDIN AND RICHARD GOLD *Lightweight Underlay Network Ad hoc Routing (LUNAR)*, URL: <http://www.docs.uu.se/selnet/lunar/>
- [4] C. PERKINS, E. ROYER AND S. DAS *Ad hoc On-demand Distance Vector (AODV) Routing*, Internet Draft, draft-ietf-manet-aodv-11.txt, *work in progress*, Aug 2002.
- [5] ANDERS LINDGREN AND OLOV SCHELÉN *Infrastructured ad hoc networks* In Proceedings of the 2002 International Conference on Parallel Processing Workshops (International Workshop on Ad Hoc Networking (IWAHN 2002)). pages 64-70. August 2002.
- [6] C. E. PERKINS, P. BHAGWAT *Highly Dynamic Destination-Sequenced Distance Vector (DTDV) for Mobile Computers* Proc. of the SIGCOMM 1994 Conference on Communications Architectures, Protocols and Applications, Aug 1994, pp 234-244.
- [7] MARIO GERLA, XIAOYAN HONG, LI MA, GUANGYU PEI *Landmark Routing Protocol (LANMAR)* Internet Draft, draft-ietf-manet-lanmar-01.txt, *work in progress*, June 2001.
- [8] M. JIANG, J. LI, Y. C. TAY *Cluster Based Routing Protocol (CBRP) Functional Specification* Internet Draft, draft-ietf-manet-cbrp.txt, *work in progress*, June 1999.
- [9] J.J. GARCIA-LUNA, M. SPOHN *Source Tree Adaptive Routing* Internet Draft, draft-ietf-manet-star-00.txt, *work in progress*, October 1999.
- [10] G. AGGELOU, R. TAFAZOLLI *Relative Distance Micro-discovery Ad Hoc Routing (RDMAR) protocol* Internet Draft, draft-ietf-manet-rdmar-00.txt, *work in progress*, September 1999.
- [11] SUNG-JU LEE, WILLIAM SU, MARIO GERLA *On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks* On-Demand Multicast in Mobile Wireless Networks, Proceedings of IEEE ICNP'98
- [12] MINGYAN LIU, RAJESH R. TALPADE, ANTHONY MCAULEY, ETHENDRANATH BOMMAIAH *Ad hoc Multicast Routing Protocol (AMroute)*, UMD TechReport 99-8
- [13] C.W. WU AND Y.C. TAY *Ad hoc Multicast Routing protocol utilizing Increasing id-numbers (AMRIS)* In Proceedings of IEEE MILCOM'99, Atlantic City, NJ, Nov. 1999.
- [14] J.J. GARCIA-LUNA-ACEVES AND E.L. MADRUGA *The Core Assisted Mesh Protocol (CAMP)*, IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, Vol. 17, No. 8, pp. 1380-1394, August 1999.
- [15] H. ZHOU AND S. SINGH *Content Based Multicast (CBM) in Ad Hoc Networks*, Proceedings of the ACM/IEEE Workshop on Mobile Ad Hoc Networking and Computing (MOBIHOC), Boston, MA, August 2000, pp. 5160. 260
- [16] L. JI AND M.S. CORSON *Lightweight Adaptive Multicast protocol (LAM)*, In Proceedings of IEEE GLOBECOM'98, Sydney, Australia, Nov. 1998

- [17] *Global State Routing protocol (GSR)*  
 [Iwata99] A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T.-W. Chen, "Scalable Routing Strategies for Ad Hoc Wireless Networks" IEEE Journal on Selected Areas in Communications, Special Issue on Ad-Hoc Networks, Aug. 1999, pp.1369-79. <http://www.cs.ucla.edu/NRL/wireless/PAPER/jsac99.ps.gz>
- [18] ALAN O'NEILL HONGYI LI *Hierarchical State Routing protocol* Internet Draft, draft-oneill-li-hsr-00.txt  
 URL: <http://alternic.net/drafts/drafts-o-p/draft-oneill-li-hsr-00.txt>
- [19] *Distance Source Distance Vector routing protocol (DSDV)*
- [20] L. JI, M. S. CORSON *Differential Destination Multicast (DDM) Specification* , Internet Draft, draft-ietf-manet-ddm-00.txt
- [21] SHREE MURTHY, J.J. GARCIA-LUNA-AVECES *Distributed Bellman-Ford routing protocol (DBF), A Routing Protocol for Packet Radio Networks*, Proc. ACM International Conference on Mobile Computing and Networking, pp. 86-95, November, 1995. URL: <http://citeseer.nj.nec.com/murthy95routing.html>
- [22] *Wireless Routing Protocol (WRP)*  
 [Chen98] Tsu-Wei Chen and Mario Gerla, "Global State Routing: A New Routing Scheme for Ad-hoc Wireless Networks" Proc. IEEE ICC'98, 5 pages. <http://www.ics.uci.edu/atm/adhoc/paper-collection/gerla-gsr-icc98.pdf>
- [23] *Clusterhead Gateway Switch Routing protocol (CGSR)* [Murthy96] S. Murthy and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks", ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks, Oct. 1996, pp. 183-97. <http://www.ics.uci.edu/atm/adhoc/paper-collection/aceves-routing-winet.pdf>
- [24] ZYGMUNT J. HAAS, MARC R. PEARLMAN, PRINCE SAMAR *The Bordercast Resolution Protocol (BRP)* Internet Draft, draft-ietf-manet-zone-zrp-04.txt, *work in progress*, July 2002.
- [25] ZYGMUNT J. HAAS, MARC R. PEARLMAN, PRINCE SAMAR *The Bordercast Resolution Protocol (BRP)* Internet Draft, draft-ietf-manet-zone-brp-02.txt, *work in progress*, July 2002.
- [26] ZYGMUNT J. HAAS, MARC R. PEARLMAN, PRINCE SAMAR *The Intrazone Routing Protocol (IARP)* Internet Draft, draft-ietf-manet-zone-iarp-02.txt, *work in progress*, July 2002.
- [27] ZYGMUNT J. HAAS, MARC R. PEARLMAN, PRINCE SAMAR *The Interzone Routing Protocol (IERP)* Internet Draft, draft-ietf-manet-zone-ierp-02.txt, *work in progress*, July 2002.
- [28] V. PARK, S. CORSON *Temporally-Ordered Routing Algorithm (TORA) Version 1* Internet Draft, draft-ietf-manet-tora-spec-03.txt, *work in progress*, June 2001.
- [29] M.S. CORSON AND A. EPHREMIDES *Lightweight Mobile Routing protocol (LMR)* ,A distributed routing algorithm for mobile wireless networks, Wireless Networks 1 (1995).
- [30] M. S. CORSON, S. PAPADEMETRIOU, P. PAPADOPOULOS, V. PARK, A. QAYYUM *Internet MANET Encapsulation Protocol (IMEP) Specification* , Internet Draft draft-ietf-manet-imep-spec-01.txt



- [31] C.-K. TOH *Associativity-based long-lived routing (ABR) protocol* , Internet Draft,  
URL: <http://www.watersprings.org/pub/id/index-wgm.html>
- [32] SONG GUO, OLIVER W. YANG *Performance of Backup Source Routing (BSR) in mobile ad hoc networks* p 440-444, *Proc. 2002 IEEE Wireless Networking Conference*
- [33] R. DUBE, C. D. RAIS, K. WANG, AND S. K. TRIPATHI *Signal Stability based adaptive routing (SSR alt SSA) for ad hoc mobile networks*, IEEE Personal Communication, Feb. 1997.  
URL: <http://citeseer.nj.nec.com/dube97signal.html>
- [34] *Flow Oriented Routing Protocol (FORP)*
- [35] NAVID NIKAEIN, HOUDA LABIOD, CHRISTIAN BONNET *Distributed Dynamic Routing Algorithm (DDR) for Mobile Ad Hoc Networks*, in *proceedings of the MobiHOC 2000 : First Annual Workshop on Mobile Ad Hoc Networking & Computing*  
URL: <http://www.eurecom.fr/~nikaeinn/ddr.ps>
- [36] NAVID NIKAEIN, CHRISTIAN BONNET, NEDA NIKAEIN “*Hybrid Ad Hoc Routing Protocol - HARP*”, in *proceeding of IST 2001: International Symposium on Telecommunications*  
URL: <http://www.eurecom.fr/~nikaeinn/harp.ps>
- [37] J. GOMEZ, A. T. CAMPBELL, M. NAGHSHINEH, C. BISDIKIAN, T.J. WATSON *Power-Aware Routing Optimization Protocol (PARO)* Internet Draft, draft-gomez-paro-manet-00.txt, *work in progress*, June 2001.  
URL: <http://comet.ctr.columbia.edu/~javierng/paro/draft-gomez-paro-manet-00.txt>
- [38] MARIO GERLA, GUANGYU PEI, XIAOYAN HONG, TSU-WEI CHEN *Fisheye State Routing Protocol (FSR) for Ad Hoc Networks* Internet Draft, draft-ietf-manet-fsr-00.txt, *work in progress*, June 2001.
- [39] PHILIPPE JACQUET, PAUL MUHLETHALER, AMIR QAYYUM, ANIS LAOUITI, LAURENT VIENNOT, THOMAS CLAUSEN *Optimized Link State Routing Protocol* Internet Draft, draft-ietf-manet-olsr-04.txt, *work in progress*, June 2001.
- [40] BHARGAV BELLUR, RICHARD G. OGIER, FRED L. TEMPLIN *Topology Broadcast Based on Reverse-Path Forwarding (TBRPF)* Internet Draft, draft-ietf-manet-tbrpf-01.txt, *work in progress*, June 2001.
- [41] D. JOHNSON, D. MALTZ, Y-C. HU AND J. JETCHEVA: *The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks*, Internet Draft, draft-ietf-manet-dsr-05.txt, *work in progress*, June 2001.
- [42] JORJETA G. JETCHEVA, YIH-CHUN HU, DAVID A. MALTZ, DAVID B. JOHNSON *A Simple Protocol for Multicast and Broadcast in Mobile Ad Hoc Networks* Internet Draft, draft-ietf-manet-simple-mbcst-00.txt, *work in progress*, June 2001.
- [43] YIH-CHUN HU, DAVID B. JOHNSON, DAVID A. MALTZ *Flow State in the Dynamic Source Routing Protocol* Internet Draft, draft-ietf-manet-dsrflow-00.txt, *work in progress*, June 2001.
- [44] S. BASAGNI, I. CHLAMTAC, V. R. SYROTIUK, B. A. WOODWARD *A Distance Routing Effect Algorithm for Mobility (DREAM)* In *Proc. ACM/IEEE Mobicom*, pages 76-84, October 1998

- [45] Y.-B. KO, V. N. H. *Location-Aided Routing in mobile Ad hoc networks* In Proc. ACM/IEEE Mobicom, pages 66-75, October 1998
- [46] JINYANG LI, JOHN JANOTTI, DOUGLAS S. J. DE COUTU, DAVID R. KARGER, ROBERT MORRIS *A Scalable Location Service for Geographic Ad Hoc Routing* M.I.T. Laboratory for Computer Science
- [47] JORJETA G. JETCHEVA, DAVID B. JOHNSON *Adaptive Demand-Driven Multicast Routing protocol (ADMR)* , Internet Draft, draft-jetcheva-manet-admr-00.txt, *work in progress*, June 2001.
- [48] VIJAY DEVARAPALLI, ALI A. SELCUK, DEEPINDER SIDHU *Multicast Zone Routing protocol (MZR)* , Internet Draft, draft-vijay-manet-mzr-01.txt, *work in progress*, June 2001.
- [49] HOUDA LABOUD, HASNAA MOUSTAFA *Source Routing-based Multicast Protocol (SRMP)* , Internet Draft, draft-laboud-manet-srmp-00.txt, *work in progress*, June 2001.
- [50] ELIZABETH M. ROYER, CHARLES E. PERKINS *Multicast Ad hoc On-Demand Distance Vector routing (MAODV)* , Internet Draft, draft-ietf-manet-maodv-00.txt
- [51] RAGHUPATHY SIVAKUMAR, PRASUN SINHA, VADUVUR BHARGHAVAN *Core Extraction Distributed Ad hoc Routing (CEDAR) Specification*, Internet Draft, draft-ietf-manet-cedar-spec-00.txt
- [52] S. SINGH, C.S. RAGHAVENDRA *PAMAS & PAMAS-Power Aware Multi Access Protocol with Signaling Ad Hoc Networks*
- [53] C.-K TOH, G. GUICHAL, AND S. BUNCHUA *On-demand associativity-based multicast routing for ad hoc mobile networks (ABAM)*, Vehicular Technology Conference, 2000. IEEE VTS Fall VTC 2000. 52nd , Volume: 3 , 2000, Page(s): 987 -993 vol.3
- [54] CHIANG, GERLA, ZHANG 1998 *Forwarding Group Multicast Protocol (FGMP)*
- [55] SINHA AND BHARGHAVAN *Multicast core extraction distributed ad-hoc routing (MCEDAR)*. In Proc. of the Wireless Communications and Networking Conference, 1999.
- [56] Y.KO AND VAIDYA 1998 *Location Based Multicast (LBM)*
- [57] WEN-HWA LIAO AND JANG-PING SHEU AND YU-CHEE TSENG *GeoGRID & Geographical GRID: A Fully Location-Aware Routing Protocol for Mobile Ad Hoc Networks*, Telecommunication Systems 2001,volume 18, number 1-3, pages 37-60
- [58] Y. KO AND N. VAIDYA *A protocol for geocasting in mobile ad hoc networks (GeoTORA)* Tech. Rep. 00-010, Dept. of Computer Science, Texas A&M University, March 2000. 23
- [59] BOLENG, CAMP AND TOLETY 2001 *Mesh-Based Geocast Routing (MRGR)*
- [60] M. GERLA, J. T. TSAI *Multicluster, Mobile, Multimedia Radio Network* ACM Wireless Networks, Vol 1, No.3, 1995, pp. 255-265
- [61] JOA NG, I-TAI LU *Zone-Based Hierchical Link State Routing (ZHLS). An abstract routing protocol and medium access protocol for mobile ad hoc networks* Submitted for partial fulfillment of the requirements for the degree of doctor of philosphy (Electrical engineering) in January 1999.
- [62] CESAR SANTIVANEZ AND RAM RAMANATHAN *Hazy Sighted Link State routing protocol (HSLS)*,BBN Technical Memorandum No. 1301, August 31, 2001.

- [63] K. GRACE (MITRE CORPORATION) *Mobile Mesh Border Discovery Protocol (MMBDP)* , Internet Draft, draft-grace-manet-mmdbdp-00.txt, *work in progress*, September 2000 URL: [http://www.mitre.org/tech\\_transfer/mobilemesh](http://www.mitre.org/tech_transfer/mobilemesh)
- [64] K. GRACE (MITRE CORPORATION) *Mobile Mesh Link Discovery Protocol (MMLDP)* , Internet Draft, draft-grace-manet-mml dp-00.txt, *work in progress*, September 2000 URL: [http://www.mitre.org/tech\\_transfer/mobilemesh](http://www.mitre.org/tech_transfer/mobilemesh)
- [65] K. GRACE (MITRE CORPORATION) *Mobile Mesh Routing Protocol (MMRP)* , Internet Draft, draft-grace-manet-mm rp-00.txt, *work in progress*, September 2000 URL: [http://www.mitre.org/tech\\_transfer/mobilemesh](http://www.mitre.org/tech_transfer/mobilemesh)
- [66] *Network Time Protocol (NTP) over the OSI Remote Operations Service*,  
URL: <http://www.ietf.org/rfc/rfc1165.txt>
- [67] *Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations*,  
URL: <http://www.ietf.org/rfc/rfc2501.txt>
- [68] *Red Hat Linux distribution*,  
URL: <http://www.redhat.com>
- [69] *The Linux Homepage*,  
URL: <http://www.linux.org>
- [70] *The Linux Kernel Archives*,  
URL: <http://www.kernel.org>
- [71] *Practical Extraction and Report Language (PERL)*,  
URL: <http://www.gnu.org/software/perl/perl.html>
- [72] *Microsoft Windows*,  
URL: <http://www.microsoft.com>
- [73] *A Short History of MS-DOS*,  
URL: <http://patersonstech.com/Dos/Byte/History.html>
- [74] *Orinoco WaveLan euipment*,  
URL: <http://www.orinocowireless.com>
- [75] *Wireless Tools for Linux*,  
URL: [http://www.hpl.hp.com/personal/Jean\\_Tourrilhes/Linux/Tools.html](http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html)
- [76] KVEIN FALL, KANNAN VARADHAN *ns notes and doceumentation*,  
URL: <http://www.mash-cs.berkeley.edu/ns>
- [77] *Wireless and Mobility Extensions to ns-2*,  
URL: <http://www.monarch.cs.cmu.edu/cmu-ns.html>
- [78] T.S. RAPPAPORT, S.Y. SEIDEL, K TAKAMIZAWA Statistical Channel Impulse Response Models for Factory and Open Plan Building Radio Communication System Design *IEEE Transactions on Communications*, vol COM-39, no.5, may 1991, pp 794-807

- [79] UCLA COMPUTER SCIENCE DEPARTMENT COMPUTING LABORATORY AND WIRELESS ADAPTIVE MOBILITY LABORATORY *GloMoSim A scalable Simulation Environment for Wireless and Wired Network Systems*  
URL: <http://pcl.cs.ucla.edu/projects/domains/glomosim.html>
- [80] *Wireless Fidelity standardization organization*,  
URL: <http://www.wirelessethernet.org>
- [81] *Institute of Electrical and Electronics Engineers (IEEE)*  
URL: <http://standards.ieee.org>
- [82] *The Official Bluetooth website*,  
URL: <http://www.bluetooth.com>
- [83] *Linux PCMCIA Card Services webpage*,  
URL: <http://sourceforge.net/projects/pcmcia-cs>
- [84] ALBERTO ESCUDERO, BJÖRN PERHSON, ENRICO PELLETTA, JON-OLOV VATN, PAWEŁ WIATR *Wireless access in the Kista - IT University* Department of Microelectronics and Information Technology IMIT, Royal Institute of Technology KTH
- [85] *Global Positioning System Joint Program Office*,  
URL: <http://gps.losangeles.af.mil>
- [86] *Differential GPS (DGPS)*,  
URL: <http://www.cnde.iastate.edu/staff/swormley/gps/gps.html>
- [87] RICHARD B. LANGLEY *RTK GPS*,  
URL: <http://www.gpsworld.com/0199/0199innov.html>
- [88] FREDRIK LILIEBLAD, OSKAR MATTSSON, PETRA NYLUND, DAN OUCHTERLONY, ANDERS ROXENHAG *AODV daemon, MAD-HOC Technical Documentation*  
URL: <http://mad-hoc.flyinglinux.net/techdoc.ps>
- [89] A. IWATA, C.-C. CHIANG, G. PEI, M. GERLA, T. CHEN *Scalable Routing Strategies for Ad hoc Wireless Networks* UCLA
- [90] C.-K. TOH, RICHARD CHEN, MINAR DELWAR, DONALD ALLEN *Eperimenting with an Ad Hoc Wireless Network on Campus: Insights and Experiences* School of Electrical and Computer Engineering, Georgia Institute of Technology
- [91] WILLIAM SU, SUNG-JU LEE, M. GERLA *Mobility Prediction and Routing in Ad Hoc Networks* UCLA
- [92] KEVIN SHEA, ROBERT W. IVES, MURALI TUMMALA *Mobile Ad Hoc Routing Protocol Analysis and Its Application to a Programmable Modular Communication System* Naval Postgraduate School
- [93] ELIZABETH M. ROYER, C.-K. TOH *A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks* IEEE Personal Communications Magazine, April 1999, pp. 46-55.
- [94] ELIZABETH M. ROYER, CHARLES E. PERKINS *An Implementation Study of the AODV Routing Protocol* Proceedings of the IEEE Wireless Communications and Networking Conference, Chicago, IL, September 2000.

- [95] ELIZABETH M. ROYER, P. M. MELLIAH-SMITH, LOUISE E. MOSER *An Analysis of the Optimum Node Density for Ad hoc Mobile Networks* Proceedings of the IEEE International Conference on Communications, Helsinki, Finland, June 2001.
- [96] XIAOYAN HONG, MARIO GERLA, GUANGYU PEI, CHING-CHUAN CHIANG *A Group Mobility Model for Ad Hoc Wireless Networks* University of California
- [97] T.S. RAPPAPORT *Wireless Communications: Principles and Practice* Prentice Hall, Upper saddle river, NJ, 1995.
- [98] A. Kamerman, *Coexistence between Bluetooth and IEEE 802.11 CCK Solutions to Avoid Mutual Interference*, Lucent Technologies Bell Laboratories, Jan. 1999, also available as IEEE 802.11-00/162, July 2000.
- [99] R. BAGRODIA ET AL. PARSEC "A Parallel Simulation Environment for Complex Systems" IEEE Comp. vol 31, no 10, Oct 1998, p 77-85
- [100] YIH-CHUN HU, DAVID B. JOHNSON *Caching Strategies in On-Demand Routing Protocols for Wireless Ad Hoc Networks*, proceedings of The Sixth Annual International Conference on Mobile Computing and Networking (MobiCom 2000), U.S. August 2000.
- [101] DAVID A. MALTZ, JOSH BROCH, DAVID B. JOHNSON *Quantative Lessons From a Full-Scale Multi-Hop Wireless Ad Hoc Network testbed* Carnegie Mellon University
- [102] DAVID A. MALTZ, JOSH BROCH, DAVID B. JOHNSON *Experiences Designing and Building a Multi-Hop Wireless Ad Hoc Network testbed* Carnegie Mellon University
- [103] RATISH J. PUNNOSE, PAVEL V. NIKITIN, JOSH BROCH, DANIEL D. STANCIL *Optimizing Wireless Network Protocols Using Real-Time Predictive Propagation Modeling* Carnegie Mellon University
- [104] DIANE TANG, MARY BAKER *Analysis of a Local-Area Wireless Network* Stanford University
- [105] SUNG-JU LEE, WILLIAN SU, JULIAN HSU, MARIO GERLA, RAJIVE BAGRODIA *A performance Comparison Study of Ad Hoc Wireless Multicast Protocols* University of California
- [106] DAVID B JOHNSON *Validation of Wireless and Mobile Network Models and Simulation* Carnegie Mellon University
- [107] DAVID A. MALTZ, JOSH BROCH, JORJETA JETCHEVA, DAVID B. JOHNSON *The Effects of On-Demand Behaviour in Routing Protocols for Multi-Hop Wireless Ad Hoc Networks* Carnegie Mellon University
- [108] P. JOHANSSON, T. LARSSON, N. HEDMAN, B. MIELCZAREK AND M. DEGERMARK: *Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks*, proceedings of the Fifth Annual International Conference on Mobile Computing and Networking, U.S. August 1999.
- [109] B LIANG, Z. HAAS In *Personal communication* February 4, 2000
- [110] DAVID B. JOHANSSON, DAVID A. MALTZ "Dynamic Source Routing in Ad hoc Wireless Networks" in *Mobile computing* pages 153-181. Kluwer Academic Publishers, 1996.
- [111] MIGUEL SANCHEZ RE: *Mobility pattern in a MANET* June 25, 1998. IETF MANET Mailing List, Message-ID: <000a01bda055\$d84f9380\$11352a9e@msanchez.disca.upv.es>

- [112] MIGUEL SANCHEZ *Re: Node Movement Models in Ad hoc*, July 15, 1999. IETF MANET Mailing List, Message-ID: <378DC8F6.B01CF351@discaupv.se>
- [113] JOHN HEIDEMANN, NIRUPANA BULUSU, JEREMY ELSON, CHALERMEK INTANAGONWIWAT, KUN-CHAN LAN, YA XU, WEI YE, DEBORAH ESTRIN, RAMESH GOVINDAN *Effects of Detail in Wireless Network Simulation* USC/ISI TR-2000-523b Informations Sciences Institute
- [114] IVAN STOJMENOVIC, ZU LIN *Power-aware localized routing in wireless networks* Computer Science, SITE, University of Ottawa
- [115] M. BERGAMO ET AL. *System design specification for mobile multimedia wireless network(MMWN) (draft)* Technical report, DARPA project DAAB07-95-C-D156 Oct. 1996
- [116] C. TSCHUDIN, H. GULBRANDSEN, H. LUNDGREN *Active Routing for Ad-Hoc Networks* Uppsala University, Ericsson Research - Switchlab
- [117] ERIK NORDSTRÖM *APE - A Large Scale Multihop Network for Reproducible Tests* Department of Computer Systems (DoCS), Uppsala University
- [118] H. LUNDGREN, D. LUNDBERG, J. NIELSEN, E. NORDSTRÖM, C. TSCHUDIN: *A Large-scale Testbed for Reproducible Ad hoc Protocol Evaluations*. Proc. IEEE Wireless Communications and Networking Conference 2002 (WCNC'02), March 2002.
- [119] ERIK NORDSTRÖM, JONAS EKMAN, MATTIS FJÄLLSTRÖM, ERIK HAGSTRÖM, JESPER HOLMBERG, MARIA LANDBERG, DAVID LUNDBERG *MANET project - Testing of Ad hoc Routing Protocols* Report from the course "Computer Networks II", June 2000
- [120] DANIEL ANDERSSON, JOHAN BACKSTRÖM, MARC-AURÈLE BROTHIER *Positioning with WaveLAN 802.11b*, Report from the course "Computer Networks II", May 2001
- [121] TAN, D.S., ZHOU, S., HO, J., MEHTA, J.S., TANABE, H. *Design and Evaluation of an Individually Simulated Mobility Model in Wireless Ad Hoc Networks* Communication Networks and Distributed Systems Modeling and Simulation Conference 2002, San Antonio, TX.