A comprehensive overview about selected Ad Hoc Networking Routing Protocols

Daniel Lang*

March 14, 2003

Contents

*D. Lang works at the Department of Computer Science, Technische Universität München, Germany. E-Mail: dl@leo.org.

Abstract: A Mobile Ad Hoc Network (*MANET*) is a mobile, wireless network, that does not need preexisting infrastructure. The routing infrastructure needs to be established in a distributed, self-organized way. Many routing protocols for MANETs have been proposed, and also some evaluating work has been done. The number of the proposed routing protocols and evaluations makes it difficult to keep track of the development and to get an overview about the strengths and weaknesses of routing protocols. In this report we provide a comprehensive introduction into most currently proposed routing protocols in Mobile Ad Hoc Networks. We present each routing protocol with references and the results of any evaluations known. We also classify the routing protocols according to their key characteristics.

Keywords: MANET, Routing

1 Introduction

In Mobile Ad Hoc Networking, the communication does not rely on any existing infrastructure such as dedicated routers, transceiver base stations or even cables (of any kind)[1]. Mobile devices (e.g. notebook computers, PDAs, cell phones, etc.) with wireless radio equipment are supposed to communicate with each other, without the help of any other (fixed) devices. In order to make that work, typically each node needs to act as a router to relay packets to nodes out of direct communication range. Under these circumstances, routing is much more complex than in conventional (static) networks. Many of the possible solutions are determined by the characteristics of the media, the behavior of nodes and the data flow.

Since research in Ad Hoc Networking has resulted in a such a large amount of routing algorithms and protocols, it has become more and more difficult to decide, which algorithms are superior to others under what conditions. For a successful deployment, this is an important problem, since a wrong choice may have a severe impact on the performance, and consequently on the acceptance of the new technology. Also providing just any protocol is not feasable, due to the different requirements on hardware and lower network layers. Further, it would not make sense, since all devices in an area would need to agree on one method if they want to communicate.

In order to help with the decision, in this article we will give an overview about the advantages and weaknesses, characteristics and requirements of a variety of Ad Hoc routing protocols. We will classify the routing protocols according to their characteristics. This should help to identify which routing protocols are best suited for which situations.

This report is structured as follows. First the short section 2 explains about the selection of the considered routing protocols. Then we present our classification of the routing protocols in section 4, followed by the alphabetic ordered introduction of all considered routing protocols in section A. This also includes references to known literature about each protocol.

In appendix B we will define various terms that are used in this report.

2 Choice of Protocols examined in this report

I have chosen to present a very comprehensive overview, including most protocols in the research area of Mobile Ad Hoc Networks. This goal is very hard to achieve, not just because of the huge amount of proposed protocols, but also by the dynamic nature of this research topic; new algorithms are even developed by the time of this writing. Further there is limited information available to certain strategies, such that some protocols cannot be considered in every detail.

Also we focus on a certain type of application. We do not consider Bluetooth[2] or sensor networks. Both are special cases, which require a certain restricted class of algorithms. Instead we concentrate on methods enabling people to communicate with each other using mobile devices, ranging from a cell phone to a car-fitted computer and communications system. Multicast protocols are not examined, too.

Finally this overview will be used as a basis to select a much more reduced set of algorithms, that are most worthwhile to be investigated throuroughly.

Routing protocols considered in this work are: *ABR, ADV, AODV, CBRP, CGSR, CEDAR, DDR, DREAM, DSDV, DSR, DST, FORP, FSLS, FSR, GEDIR, GPSR, GSR, HSR, LANMAR, LAR, LMR, LRR, OLSR, SSA, STAR, Terminode Routing, TBRPF, TORA, WAR, WRP and ZRP.*

I am aware of the following protocols, that are not considered in this paper, due to an aimed deadline for this work: *DST (Dynamic Source Tracing), BEST (Bandwidth Efficient Source Tracing), NSR (Neighbourhood Aware Source Routing), SOAR (Source-Tree On-Demand Adaptive Routing) and ZHLS (Zone-based Hierarchical Link State Routing).*

The protocols can be classified and distinguished in many ways. We will also present the most common classes, and protocols which are members of that class. However, these sets are not always disjoint.

For terms used in the following sections, please check the glossary section in appendix B.

3 Status of IETF development efforts

Currently, the development efforts in the IETF are focused on four routing protocol drafts which will be submitted as experimental RFCs or internet standards. These are AODV, DSR, OLSR and TBRPF. Most other drafts have expired since their latest submission.

4 Categorization of Ad Hoc Routing Protocols

The large variety of routing protocols reflects the fact that these protocols do implement strategies very differently. We can categorize the routing protocols into different classes, that represent the key aspect of their strategy. The classes will not be disjoint, as we have several levels of strategies.

Such work was already done previously in [3], and to some extent in [4]. However at that time, just a small set of routing protocols has been classified. We will set up a set of classes which we believe are representative for the different aspects of routing and which may correspond to some of the classes suggested in [3]. Unlike [3] this work does not try to structure disjoint classes into a tree. Although some characteristics are typically dependent upon others, some are not and need to exist in parallel. Some classes will be of an opposing nature, i.e. the protocols can clearly be distinguished between two disjoint classes, e.g. *reactive* and *proactive*. Others will not have a counterpart with characteristics worth pointing out (like *hierarchical* routing protocols, since *non-hierarchical* protocols do not share distinguished characteristics, apart from not being hierarchical).

The diagram in figure 1 gives an overview about the classes we have devided to use for our characterization. There are no real relations between these classes other than those indicated by arrows:

The choice of some of these classes was inspired by [3].

4.1 Single Channel vs. Multichannel Protocols

This is essentially a layer 2 property, but several protocols may depend on a certain link layer, while others are specified link-layer independent.

Single channel protocols use just one shared channel to communicate. The IEEE 802.11 DCF medium access method is the most widely used example for such a shared channel link layer. Multichannel protocols utilize CDMA, FDMA or TDMA to form specific channels. Although communication can be much more efficient using such a method, it is difficult to be used in an Ad Hoc Network, since usually a distinguished controlling station is needed to assign the channels.

There are many protocols which do not specify the link layer, but their performance may still depend on it.

Multichannel protocols The following protocols require a multichannel link-layer either explicitly, or their performance depends heavily on it.

CGSR Clusterhead Gateway Switched Routing requires TDMA within a cluster and CDMA between clusters.

TLR/TRR [5] states that the considerations for a link layer protocol for the *Terminode* project center around CDMA.

TORA Implementations of TORA did rely on the encapsulation protocol IMEP [6] used as an underlying secure link layer protocol. IMEP, however did perform very badly together with the IEEE 802.11 Wireless LAN standards and DCF. It was suggested in [7] that other link layer techniques should be used with TORA.

Protocols that use the IEEE 802.11 or a related link layer This class includes all protocols that use a CSMA/CA, MACAW, IEEE 802.11 WLAN with DCF or related link layers. This is the great majority. We just list the protocols, by their abbreviated name:

ADV, AODV, CEDAR, DSR, GPSR, FSLS/HSLS, LANMAR, OLSR.

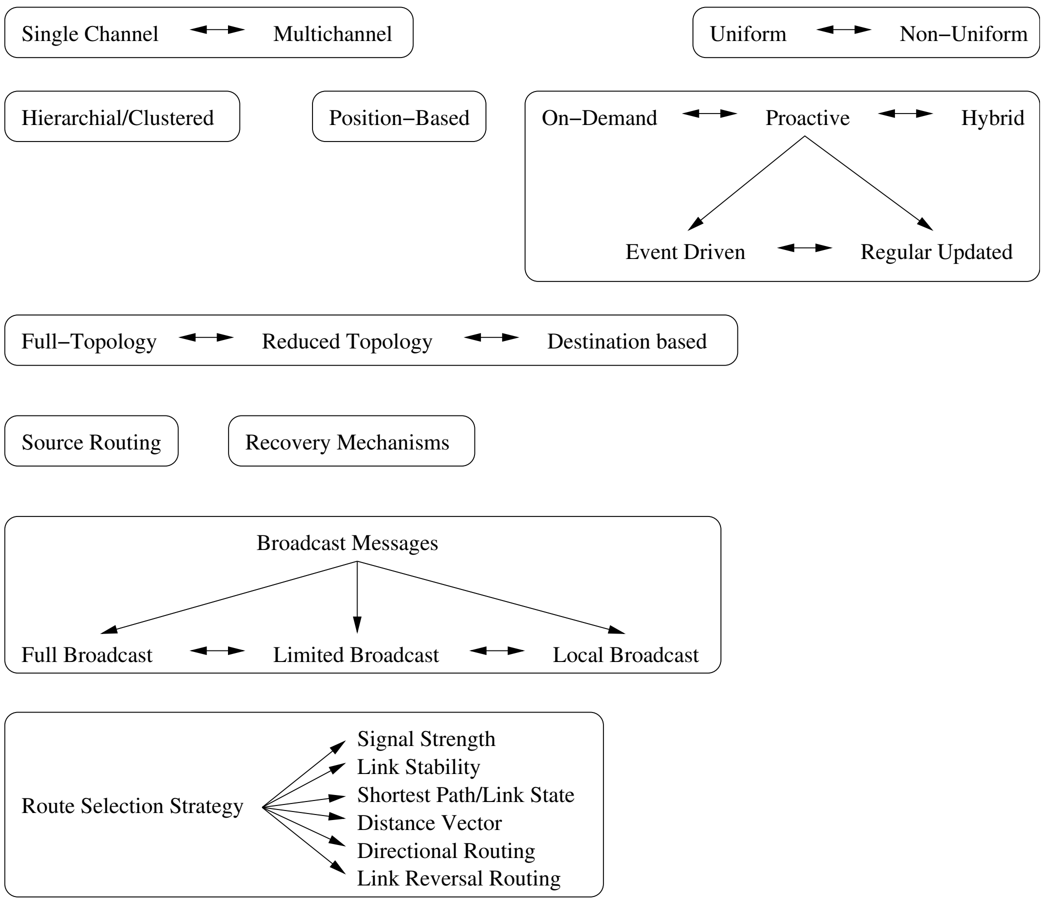

Figure 1: Protocol-Class Overview

Unspecified link layer: This remaining list contains all protocols that did not specify a link layer.

ABR, CBRP, DDR, DREAM, DSDV, DST, FORP, FSR, GEDIR, GSR, LAR, LMR, SSA, TBRPF, WAR, WRP.

4.2 Uniform vs. Non-Uniform Protocols

As defined in [3], a uniform protocol does not assign any special roles to any node. In a non-uniform protocol some nodes may be assigned a special role, which needs to be performed in a distributed fashion. Typically clustering protocols are non-uniform.

Non-Uniform Protocols Apart from the following *non-uniform* protocols, all others are *uniform*:

CBRP The cluster based routing protocol forms clusters and thus requires clusterheads, which are distinguished nodes.

CGSR The same applies to CGSR, which additionally defines gateway nodes.

CEDAR forms a "core network" (like a backbone), which requires a special role for the nodes, which are part of the core.

DST also creates a backbone on the stable regions of the network.

HSR forms clusters like CBRP and CGSR, but there are no gateway nodes, but multilevel clusters and clusterheads.

LANMAR needs landmark nodes for each group of nodes.

OLSR requires the selection of MPR (multipoint relay) nodes, which is also a special role.

4.3 Hierarchical Topology/Clustered Routing

Clustering is often discussed in the Ad Hoc Networking context. The idea is, to use clusters to introduce some structure into the (otherwise very chaotic and) dynamic nature of the network.

Clusters are usually represented by a dedicated node: the *clusterhead*. This node forms the cluster and attached nodes use the cluster head to describe the cluster they belong to. Clusters can also be formed hierarchically, such that there are multiple layers of clusters. The clusterheads are usually responsible for managing communication within a cluster and are informed about joining and leaving nodes. Additionally to clusterheads, *gateway nodes* are suggested in CGSR and in HSR. These are responsible to transmit information from one cluster to another and therefore may be part of more than one cluster.

Since cluster formation and election of clusterheads is usually a significant effort in terms of signaling traffic, as is the removal and addition of nodes from/to a cluster, cluster-stability has become one important aspect of clustering algorithms.

However, clustering in general does suffer from some drawbacks, especially with very stable clusters. Since the clusterhead and also the gateway nodes have to do the routing and managing work, they can easily become a bottleneck. The communication load will certainly be higher for a clusterhead or a gateway node than for an ordinary node, thus consuming more energy which can lead to an early outage of these nodes due to exhausted batteries.

There are also other hierarchical properties we take into account in this class. Some protocols (FSR, DREAM, FSLS) introduce a set of scopes for routing information. In any of these protocols, close, fast moving nodes receive more information more frequently than others. Further there are routing protocols, which use different routing strategies, depending where and how far a packet has traveled, like Terminode Routing or also ZRP.

Routing Protocols that use Clustering or a hierarchical Structure:

CBRP The Clustered Routing Protocol defines clusters and clusterheads.

CGSR The Clusterhead Gateway Switch Routing Protocol routes alternating between clusters and clusterheads.

FSLS The Fuzzy Sighted Link State protocol (as well as it's derivates like HSLS), define scopes for dissemination of routing information, and thus also can be considered hierarchical, but not clustered.

FSR The same applies to Fisheye State Routing.

HSR The Hierarchical State Routing protocol does physical and logical clustering. It also is capable of multilevel clustering.

LANMAR Since a landmark can be considered a representative node on a higher level, LANMAR can also be considered a hierarchical routing scheme.

ZRP The Zone Routing protocol defines a routing zone, this is also some sort of clustering.

(All other protocols can be considered as *non-hierarchical*).

4.4 Position based Protocols

[8] explains aspects of position based routing in detail. Position based routing algorithms claim that no routing tables need to be maintained and thus no overhead due to route discovery and route maintenance is imposed. But they need to obtain position data of their corresponding destinations, either by an internal discovery process, or by an independent position service, which will then impose overhead to maintain the position information (either proactively or on-demand). Several position services are discussed in [8]. Further in this paper, position based routing algorithms are compared in terms of their characteristics and forwarding strategies.

Greedy algorithms like GPSR (cf. A.16) use either a *most forward within radius* or a *nearest with forward progress* strategy. It is argued, that NFP is of great advantage if the transmission radius/power can be controlled, and additionally have the benefit of reduced channel competition. Recent studies ([9]) have shown that power control may not improve channel utilisation much, because the longer path lengths (in hops) make up for the benefit. Greedy algorithms can route to a local maximum and need a recovery strategy in this case. Among several suggestions, the planar graph traversal methods seem to be the most reasonable.

DREAM and LAR use a flooding approach, but packets are not sent to all neighbours, but only to those in the right direction of the target (i.e. the packet is forwarded to any node within the *request zone*).

Finally some protocols use a hybrid/hierarchical scheme, like Terminode Routing. For long distances a greedy directional routing scheme is used. If the packet is close enough, some non-directional mechanism will guide the packet to the destination.

We will now summarize the position based protocols:

DREAM requires an *all-for-all* position service (each node carries a location table for each other node)[8], and requests are forwarded in the right direction.

GEDIR also uses directional routing. To obtain the right direction, the locations of source and target and intermediate nodes must be known. It is not specified how the location information should be determined.

GPSR forwards always to the node closest to the destination within reach, until the target or a local minimum is reached (i.e. there is no other node within range that is closer to the target). Again, it is not specified how location information should be obtained, except for a vague reference to a location database service (cf. Section A.16).

LAR tries to predict the movement of the target node within a time interval to determine a *request zone* to which the data will be broadcast to. Location information from nodes can be piggybacked on messages, but again it is unclear, how a node is aware of it's position. Also speed and direction are important parameters, and although they could be derived from the positions to some degree, this is not made clear.

Terminode/AGPF Anchored Path Geodesic Packet Forwarding from the Terminode Project (cf. Section A.27) uses locations, too, to route packets close to their destination. Two schemes (FAPD and DRD, explained in A.27) are proposed to determine the anchored path, but still a general location service like GPS is required.

Other protocols do not use location information.

4.5 Proactive versus On-Demand Routing Protocols

A routing protocol can maintain routing information either on-demand or proactively (at all times). We characterize the protocols accordingly in this section. Further proactive protocols can be divided into protocols that update routing information in regular intervals and protocols that update on certain events. Finally, there are routing protocols that are hybrid and make use of both schemes.

4.5.1 On-Demand or Reactive Protocols

A network using an on-demand protocol will not maintain correct routing information on all nodes for all times. Instead, such routing information is obtained *on demand*. If a node wants to transmit a message, and does not have enough routing information to send the message to the destination, the required information has to be obtained. Usually the node needs at least to know the next hop (among its neighbours) for the packet. Although the node could just broadcast the packet to all neighbours this leads to serious congestion in many cases. However, such broadcasts are used in a route discovery process, since there is no other next-hop information available, yet.

Usually this consists of a broadcast message from the originating node, indicating the desired route. Nodes which have the required information will respond to the originating node, which will eventually choose a route from the replies it received. The broadcast may be limited to travel only a few hops first, before netwide broadcast will be issued (which would flood the whole network).

Of course, the route request and selection process must be finished, before the message can be sent. This leads to an initial setup delay for messages, if their route is not known to the node. To limit the impact of this delay, most protocols will use a route cache for once established routes. However, the information in this cache will time out, since in a mobile environment, the routes will be invalid after some time.

Clearly, applications that are used over an on-demand routing protocol need to be tolerant for such an intial setup delay.

The advantage of on-demand routing protocols lies in the fact that the wireless channel (a scarce ressource) does not need to carry a lot of routing overhead data for routes, that are not even used. This advantage may diminish in certain scenarios where there is a lot of traffic to a large variety of nodes. Thus the scenario will have a very significant impact on the performance. In such a scenario with lots of traffic to many nodes, the route-setup traffic can grow larger than a constant background traffic to maintain correct routing information on each node. Still, if enough capacities would be available, the reduced efficiency (increased overhead) might not affect other performance measures, like throuhgput or latency.

We also consider some location based protocols as on-demand protocols, since they determine the direction in which to send the packet on demand and some protocols may even initiate a location query of the destination nodes for their packets on demand.

Thus, examples for on-demand protocols are the following:

ABR, AODV, CEDAR, DREAM[1], DSR, FORP, GEDIR, LAR, SSR, WAR.

4.5.2 Proactive Protocols

Proactive routing protocols will try to maintain correct routing information on all nodes in the network at all times. This can be achieved in different ways, and thus divides the protocols into two subclasses: *event driven* and *regular updated* protocols.

[1]This algorithm does not intiate a routing selection process, but uses directional routing, but the direction is obtained on demand.

Event driven protocols will not send any routing update packets, if no change in topology occurs. Only if a node detects a change of the topology (usually a node moves out of reach of this node, or a new node comes close enough), this is reported to other nodes, according to the strategy of the routing protocol.

Protocols that are updated in regular intervals will always send their topology information to other nodes at regular intervals. Many link state protocols work in such a manner (but varying the maximum distance of an update message with the length of the interval). Nodes farther away get updates less frequently than close nodes, thus balancing the load imposed on the network.

Proactive protocols of either subclass impose a fixed overhead to maintain the routing tables. Even if many of the entries are not used at all. Their advantage is, that the routes can be used at once and there is no setup delay.

[10] compares "flooding protocols" [2] with "hello protocols" (those that periodically announce their neighbours and routes) in terms of overhead in an analytical way.

Event driven proactive routing protocols are the following: CBRP, CGSR, DSDV, GSR, LMR, TORA and WRP.

Regular updated protocols are: DDR, FSLS, FSR, GPSR, LANMAR, OLSR, STAR and TBRPF.

4.5.3 Hybrid Protocols

Also, there are protocols (as to say protocol sets) that utilize both proactive and on-demand routing.

These are:

ADV - Adaptive Distance Vector Routing Routes are maintained proactively, but only to certain nodes (active receivers), and the size and frequency of the updates is adapted. So the authors claim its a hybrid protocol.

Terminode Routing Terminode Routing consists of an on-demand location based component: AGPF, Anchored Path Geodesic Packet Forwarding, and a proactive local routing component, which works similar to IARP from ZRP.

[2] which is the authors' term for on-demand routing protocols, that flood route requests

ZRP - Zone Routing Protocol The Zone Routing Protocol also consists of a proactive Intra Zone Routing Protocol (IARP) and an on-demand Inter Zone Routing Protocol (IERP).

4.6 Full vs. Reduced Topology Information

Many Routing Protocols transmit topology information, but not all distribute the complete topology information they are aware of. It is difficult to classify the protocols according to this characteristic. Also even if full topology information is maintained in each node, the messages usually only carry sufficient information to reflect the changes in topology but never the whole topology information, since that would not scale.

Full topology is maintained in: DDR, GSR, OLSR, STAR (in ORA mode), TBRPF (in *full topology* mode).

Reduced Topology is maintained in: FSLS, FSR, LANMAR, STAR (in LORA mode), TBRPF (in *partial topology mode*), WRP, ZRP.

This kind of classification is either not applicable to the remaining routing protocols or their role remains uncertain.

4.7 Use of Source Routing

A few routing protocols utilize source routing. This means, forwarding depends on the source of the message. Commonly, the source puts all the routing information into the header of a packet. Forwarding nodes utilize this information. In some cases, the forwarding nodes may alter the routing information in the packet to be forwarded. They are just a few protocols using source routing: CBRP, DSR, Terminode/AGPF and WAR.

4.8 Use of broadcast messages

Broadcast can have different meanings in a wireless environment. There is a *full netwide broadcast*, which means, a message is intended for every node in the network, and needs to be retransmitted by intermediate nodes. On the other hand, there is a *local broadcast*, which is intended for any node within the senders reach, but which is not retransmitted at all. In between there are limited broadcasts, in which the maximum hop count (time to live) is limited as desired.

There is no routing protocol, that *always* issues full broadcasts, but there are some, that may use full broadcasts: ABR, ADV, AODV, CEDAR, DSDV, DSR, FORP and WAR.

Many protocols prefer a limited broadcast: AODV, FSLS, FSR, HSR, LANMAR, LAR, LMR, SSR, Terminode and ZRP.

And also there are protocols, which use only local broadcasts: DDR, GSR, GPSR, OLSR, STAR, TBRPF, TORA and WRP.

Finally, directional routing protocols do not use broadcasts by intention, but would use local multicasts (like a local broadcast, but not addressed to all neighbours), like DREAM and GEDIR.

4.9 Recovery Mechanisms

Since the routing information in each node may become stale, some protocols may need a route recovery or route conservation mechanism. It is clear, that proactive routing protocols do not need a specific recovery mechanism, since they react to topology changes anyway within a short period. On-Demand protocols however, need to fix routes which are not available any more.

The following protocols have some (explicit or implicit) recovery mechanism: ABR, AODV, CBRP, DREAM[3] DSR, FORP, WAR and ZRP.

The following protocols could utilize such a mechanism, but do not support one: ADV, GEDIR, LAR.

4.10 Route Selection Strategy

The route selection strategy is an important aspect of a routing protocol. We describe the main representatives and the protocols, which use them.

Signal Strength: Route packets along the connection with the best signal strength. This is mainly used by **ABR** and **SSR**.

Link Stability: Route packets along the connections that appear most stable over a period of time. It is is used by **DST** and **FORP**

Shortest Path/Link State: Select a shortest path according to some metric. This is used by many protocols: **CEDAR**, **DDR**, **FSR**, **GSR**, **HSR**, **LANMAR**, **OLSR**, **STAR**, **TBRPF**.

[3]The recovery mechanism is not specified, just a `Recovery()` routine is mentioned.

Distance Vector: The common distance vector method, usual by hop count, is used by **ADV**, **AODV**, **DSDV**, **DSR**, **WRP**, **ZRP**.

Directional Routing: This routes into the geographic direction of the target and is mainly used by location based protocols: **DREAM**, **GEDIR**, **GPSR**, **LAR**, **Terminode/AGPF**.

Link Reversal Routing: is a routing family which is used by **LMR** and **TORA** (cf. also section A.22). It is based on flows in a graph.

5 Performance Comparisons

In literature many performance comparisons have been done, but not as much, as one could probably expect. Of course it would not be possible or even useful to compare every single routing protocol with every other protocol in any kind of scenario.

The comparisons done so far are summarized in the following table, the literature references can be found in the corresponding section to the protocol in the first column in appendix A.

- ABR vs. DSR, DBF[4]
- ADV vs. AODV, DSDV, DSR
- DSDV vs. AODV, DSR, TORA
- DST vs. flooding
- GEDIR vs. DIR[5], MFR[6]
- GRSR vs. DSR
- GSR vs. DBF, OSPF, ILS
- HSR vs. FSR, DSDV
- FSR vs. HSR, DSDV
- LANMAR vs. AODV, DSR, FSR
- OLSR vs. AODV, DSR
- STAR vs. topology broadcast, ALP[7], DSR

[4] Distributed Bellmann-Ford
[5] Directional Routing
[6] Most Forward within Radius
[7] Adaptive Link State

- WAR vs. DSR
- WRP vs. DBF, DUAL[8], ILS[9]

A Routing Protocols for Mobile Ad Hoc Networks

This section will now discuss most of the proposed routing algorithms for mobile ad hoc networks in alphabetic order.

A.1 ABR - Associativity Based Routing

ABR [11] is an on-demand routing protocol: Routes are discovered with a *Broadcast Query* request. From these requests, the destination learns all possible routes, and replies along a selected route to the source.

If a route breaks, several route-reconstruction methods can be applied, depending if the source, the destination or an intermediate node moves out of reach.

Further, ABR maintains a “degree of associativity” in form of associativity ticks. These are not clearly defined, but from context it appears that every node maintains a tick-value for every one of his neighbours. Every time interval a link-layer hello message from that neighbour is received and the tick value is increased. If the neighbour moves out of reach, the value is reset to zero. A tick level above a certain threshold indicates a stable association between those two nodes.

On selecting a route, the destination (which does the selection) prefers most stable routes, i.e. those with the highest associativity tick value. Hence, this “degree of associativity” is used as a metric of mobility. This strategy is similar to SSA (cf. section A.24).

In [11], there are statements about the complexity of ABR, but since they lack a clear definition, they may not be very useful.

In [12] ABR was compared to DSR and DBF by a simulation study using GloMoSim in a small scenario[10]. The results are in favour of ABR, in terms of overhead, throughput and end-to-end delay, although the advantage to DSR is quite small. Other criteria, like memory requirements for the table and power consumption show disadvantages of ABR.

[8] from EIGRP

[9] Idealised Link State

[10] 30 nodes in a 20×20m area.

ABR is also described in [13], which focuses on the impact of HELLO-messages (beacons) on the battery life of nodes.

1999 ABR was submitted as a draft to the IETF MANET working group under the title "Long-lived Ad-Hoc Routing based on the concept of Associativity". However, the draft has expired since[11], so one can assume the topic was no longer of interest for the working group, and now it even seems that C.K.Toh has abandoned work on ABR (possibly in favour of more promising methods).

However, in 2001 [14] was published, which proposes an enhancement to ABR. The stability property, also measured in *ticks* is now determined in a more advanced and improved way. Further an optimized threshold for associativity is introduced, such that no longer the route with the highest degree of associativity is chosen, but the one closest to the optimal threshold value. It was claimed, that this OABTR called protocol was compared to DSR (cf. section A.10) and ABR, but no details about this evaluation have been given.

A.2 ADV - Adaptive Distance Vector Routing

ADV, the Adaptive Distance Vector Routing Algorithm by Boppana and Konduru [15] is described as a combined proactive and on-demand type of protocol. The main characteristic is proactive, since routes are maintained all the time. The on-demand character is implemented by two key aspects:

- only routes to *active receivers* are maintained
- the frequency and size of routing updates is adapted to the current network conditions.

Of course *active receivers* must be announced in a broadcast-like fashion, similar to broadcast route requests. Also, if a node ceases to be a receiver, this must be announced, too. Every node keeps a *receiver flag* for each destination in its routing table, to reflect the status of this node.

To adapt the frequency and contents of routing updates to the network load and mobility, a *trigger meter* is kept by each node. This variable can be increased in certain steps, depending on the events that the nodes receive. There are two thresholds, the first is a dynamic threshold, which is computed on the recent past and the role of the node (e.g. if the node is part of an active route, etc). If this dynamic threshold is exceeded, a partial update

[11] Six months after submission, a draft must either be updated or be submitted as rfc or internet standard, otherwise it will expire.

is scheduled. The second threshold is a fixed constant TRGMETER_FULL, which will trigger a full update, if it is reached. The trigger meter is reset after each update.

In [15] ADV was compared against DSDV, AODV and DSR, and outperformed any of these in most considered performance metrics (cf. section B). In [16] ADV was compared against AODV and DSR, but explicitly with TCP traffic and various TCP modifications to improve TCP performance in mobile ad hoc networks. This paper is in favour of ADV in several terms. ADV is much better if there is significant background traffic. Since it does not benefit much from TCP enhancements like fixed RTO, SACK and delayed ACK (but performs equally well as AODV or DSR with these enhancements), it is argued, by using ADV such special TCP features may not be needed.

A.3 AODV - Ad Hoc On Demand Distance Vector Routing Protocol

This is one of the most discussed and most advanced routing protocols. It is an important part of the work of the *MANET* IETF working group. The draft has reached version 11 and will be submitted to enter the RFC track as an experimental standard. So this is probably the most mature suggestion for an ad hoc routing protocol. Its main developers are Charles E. Perkins (Nokia) and Elizabeth Belding-Royer (UCSB). AODV is discussed in lots of studies and is often used as a reference to compare other routing protocols.

AODV was derived from C. Perkins earlier work, DSDV (cf. section A.9). Compared to DSDV, AODV no longer needs to exchange periodic messages proactively, but works in an on-demand fashion, instead.

If a route to a destination is unknown, a route discovery process is initiated. This consists of broadcasting a Route Request (RREQ) packet throughout the network. To limit the impact of a netwide broadcast, these request should be sent with an expanding ring search technique: the TTL of the packets starts with a small value; if no route has been found, the TTL will be increased and the request will be resent. Each node that rebroadcasts this request, adds its address into a list in the packet. If the destination sees the request, it will reply with a unicast Route Reply (RREP) to the source. Each intermediate node may cache the learned routes.

The routing table entries consist of a destination, the next hop toward this destination and a sequence number. Routes are only updated if the sequence number of the updating message is larger than the existing one. Thus routing loops and updates with stale information are prevented. The sequence number technique was already used in DSDV (cf. section A.9) and was adopted by a variety of other routing protocol developers.

The amount of information, which needs to be present at each node, is rather limited:

- The node is aware of its neighbours (via link-layer-notification, or explicit HELLO messages).
- The node knows route destinations and the next hop.
- The node has a "precursor list" for each destination. This list consists of all nodes, which use the current node as a relay for the destination. In case of a route failure to this destionation, the node knows exactly which other nodes to notify.
- Each routing entry also has a lifetime.

The authoritative description of AODV is the current IETF draft [17].

A more easy to read description is given in [18]. A huge number of Ad Hoc related papers cite AODV as a reference (we do not list them here). However, some papers did an independent comparison between some ad hoc routing protocols including AODV, like [19], [20] and [21]. In these papers AODV and DSR compete better than other protocols and AODV shows the best results overall.

A.3.1 MAODV

There is a special form of AODV for multicast traffic, called MAODV. We did not have a close look at MAODV and just mention the main differences in a rough way:

- Routing tables have more than one next-hop.
- Route discovery is initiated on joining of a group, or on sending a message to a group with no route.
- On join requests, only a multicast router or a tree member should respond, otherwise, any node with a route will do.

A.4 CBRP - Cluster Based Routing Protocol

CBRP maintains clusters of two hops diameter, with an elected clusterhead for each cluster. Clusters may be overlapping, but each node must be part of at least one cluster. Clusterheads are not allowed to be direct neighbours, except for a short period (called "contention period"). Nodes maintain a neighbour table which also includes the link type. Also a cluster adjacency table is kept in each node. Source routing is used, with the route in the CBRP header. This allows a limited local repair mechanism and a route cache (much like DSR, see section A.10) to be used.

For clustered routing, the key argument is that with a clustered hierarchy, it is again possible to channel information (cf. Section 4.3). Thus scalability may be regained, even if broadcasts need to be used.

This routing protocol was submitted as a draft to the IETF MANET working group in 1999 [22]. This draft is now also expired, but CBRP is also described in [23].

Unfortunately there does not seem to be much more work published about it.

A.5 CGSR - Clusterhead Gateway Switch Routing

In [24], Clusterhead Gateway Switch Routing is proposed. It consists of a clustering scheme, called *Least Cluster Change* which is combined with either "lowest id", or "maximum links", to form clusters and elect clusterheads. The scheme focuses on cluster stability. CGSR explicitly specifies requirements on the link layer and medium access scheme:

- Inter-cluster communication requires a CDMA system, such that each cluster is assigned a different code (spatial reuse of codes is utilized, though).

- Within each cluster, TDMA is used. The allocation of time slots is done by a token passing method.

Gateway nodes are nodes, that are within more than one cluster, and therefore need to communicate in different codes.

The protocol uses a sequence number scheme (as developed in DSDV) to gain loop-free routes and avoid stale routing entries. In CGSR, a packet is routed alternating between clusterheads and gateways, hence the name. In the paper, several enhancements (e.g. priority token passing) are suggested, as well.

Simulation of the protocol was done by using a special simulation language called *Maisie*. A 500×500m region was used, with 100 nodes. The nodes did move according to a random strategy that was no further specified.

CGSR is mentioned in some work, like [4] and [25], but not much more. Implementations for common simulators, or even real use, don't seem to exist.

A.6 CEDAR - Core-Extraction Distributed Ad Hoc Routing

This is more a routing framework scheme for QoS requirements than a MANET routing protocol. In CEDAR a subset of the nodes is selected that will form a backbone within the network (the core). This structure is used for broadcast messages, hence no flooding is needed. The messages sent over the core network are *increase waves* (slow propagating) and *decrease waves* (fast), which notify about an increase or decrease of available bandwidth. The propagation of these waves is dynamically limited, depending on the available bandwidth. So the relevant information for QoS is disseminated in an efficient way. Within the core network, any established ad-hoc routing protocol may be used. The usage of this information, in order to establish QoS routes, works as follows:

A node contacts its "dominator" (local core node) with a route request, that contains source, destination and required bandwidth. The dominator computes a QoS route, if this is feasable and then continues to establish it. This includes possible discovery of the dominator of the destination and a core path to it.

CEDAR was presented in [26]. Its QoS focus, and also the proposal of a core/backbone network distinguishes CEDAR from most other routing strategies, and it is often mentioned and cited in other papers.

[27] picks up the idea and suggest a way how to improve performance of AODV or DSR by the use of a core infrastructure as proposed in [26].

However, it does not seem to have caught enough interest such that subsequent practical work, like implementations of CEDAR for simulators or real environments and evaluations, would have been done.

A.7 DDR - Distributed Dynamic Routing Algorithm

DDR is based on the construction of a forest to represent topology, which is constructed by using local periodic messages only. (This is a similar approach as in OLSR, see section A.23.)

DDR also forms a set of disjoint routing "zones" (cf. section A.31). There is a zone for each tree in the forest. Routing information is exchanged only with nodes, that are within a node's zone and which concerns only neighbour zones. The zone size is not fixed in DDR but will be adjusted dynamically. The algorithm in detail is given in [28], but no simulation or any other performance comparison was done so far.

A.8 DREAM - Distance Routing Effect Algorithm for Mobility

This algorithm was suggested in [29] on MobiCom 1998. It is a location based algorithm, that makes use of the *distance effect*. This means, that two nodes appear to move slower with respect to each other with increasing distance. Thus, location information for distant nodes does not need to be updated in such an accurate and frequent way, as for close nodes (see also FSR in section A.14 for a similar approach).

Each node has a routing table with location information about each other node. DREAM can be considered proactive, since location information must be disseminated (the method of location determination is not specified, so a separate location service, like GPS may be required).

On sending a message, a *direction* is determined by using the location of the destination. Then, the message is passed to all neighbour nodes in that direction. This method is more related to reactive protocols, as the route is not fixed in advance.

Distance and mobility of a node determine the frequency of location updates. A fast moving node sends location control messages much more often than a slow one. Also the messages are sent with different lifetimes (in terms of hops) and short-lived messages are sent much more frequent than long-lived. The long-lived messages will reach far away nodes, but are sent much less frequent. This leads to a bandwidth and energy efficient protocol. Although the routes are not fixed in advance, there is no setup-delay.

Basagni et. al. claim that this protocol is inherently loop-free, since the messages travel away from the node into a specific direction. This could be questioned, since in a network with very high mobility, the target direction can change, even back to a node who has sent the message already. There is also some discussion about this in the papers about GEDIR, cf. section A.15.

Another problem is, that location table entries may be stale and that no close neighbour in the required direction can be found (e.g. due to lack of connectivity). Both problems are addressed in [29], but not very detailed: an unspecified `Recovery()` routine should be called in these cases. The authors chose to use flooding in their prototype implementation.

There was not much more work on DREAM, but other routing schemes such as LAR (cf. A.20) or FSR (cf. A.14) did pick up some concepts of DREAM.

A.9 DSDV - Destination Sequenced Distance Vector Routing Protocol

This protocol is the result to adapt an existing distance vector routing algorithm (Distributed Bellman Ford, [30]), as used in RIP, to an ad hoc networking environment. This is a proactive protocol, that updates routing information on a regular basis. To avoid routing loops, destination sequence numbers have been introduced. DSDV is one of the first attempts to adapt an established routing mechanism to work with mobile ad hoc networks.

Each routing table lists all destinations with their current hop count and a sequence number. Routing information is broadcast or multicast. Each node transmits its routing table to its neighbours. Routes with more recent sequence numbers obsolete older routes. This mechanism provides loop freedom and prevents stale routes.

The routing information is transmitted every time a change in the topology has been detected (i.e. a change in the set of neighbours of a node).

DSDV works only with bidirectional links.

DSDV was presented in [31] in 1994. A more detailed description is available in [32]. DSDV was also used for many comparisons like [19], [33], [20] and [15]. The results are mixed, but especially the later papers show results, where DSDV is not performing well compared to the other protocols.

A.10 DSR - Dynamic Source Routing

DSR is an on-demand protocol, that uses source routing. In this case, this means, that each packet carries the complete route to its destination in its header (which introduces some overhead). It was first described in [34]. Since DSR works on demand, a route must be discovered through a *Route Discovery Mechanism* before use. Discovered routes may be cached and routes may be *overheard* by a node (by parsing the source route information of packets that are relayed).

If broken links are detected, a corresponding *Route Error* message is transmitted through the network and a route maintenance mechanism takes over to fix the broken routes, if possible.

To further reduce unnecessary traffic, a node may reply to a route request with a locally cached route, even if it is not the destination node. Delays in these replys with promiscous observation (*overhearing*) of other routing traffic prevent multiple nodes replying with a cached entry all at once.

The dynamic source routing protocol is also a very mature protocol. The IETF draft [35] has reached version 7 and will result very likely in some IETF standard (probably experimental RFC) as well as AODV.

DSR is described in detail in [36]. DSR is also one of the few Ad Hoc Routing Protocols, that have been implemented and evaluated in a real testbed. The results are described in [37].

[38] presents an analytical study of the probabilities of successful deliveries and the total amount of traffic generated for a successful delivery. It is argued, that an end-to-end recovery mechanism (as used in DSR) does not scale if the routing path lengths increase. Instead a local recovery mechanism is suggested, that gives much better results (according to the analysis). This algorithm is implemented in the Witness Aided Routing Protocol (WAR, cf. section A.29). Although the analysis is convincing, it was done by the authors of WAR, so the result being strongly in favour of WAR is not surprising.

DSR was used in many performance comparisons, evaluating studies, and was used as a reference for a lot of other protocols. Further, it was used as a reference protocol for investigations to find general improvements for Mobile Ad Hoc Networks (like reduced energy consumption). Papers referring to DSR include: [19, 39, 33, 3, 4, 20, 40, 21, 41, 16, 38].

[12] also compared ABR to DSR and DBF. The result is that both ABR and DSR perform much better than DBF, with a slight advantage of ABR.

In [42], a closely related protocol called *Neighborhood Aware Source Routing, NSR* is described, which is based on the DSR ideas. We do not describe NSR any further.

A.11 DST - Distributed Spanning Tree Protocol

This approach takes into consideration that in a mobile ad hoc environment, there can be regions of different stability. So this approach proposes the establishment of a backbone network in the stable regions, using a spanning tree algorithm.

For the unstable reagions a flooding or a "shuttling" approach is used to transmit the packet to the destination even through a very unstable area.

DST is described in [43] and compared against pure flooding.

There was no comparison to other approaches and this paper is also just mentioned in [28].

A.12 FORP - Flow Oriented Routing Protocol

FORP[44] is designed for real-time traffic flows (over IPv6). It works in an on-demand fashion (similar to other corresponding protocols), such that traffic flows[12] are requested first and can be used, if granted. The charac-

[12] i.e. the routes for a flow

teristic of FORP is that for each link there is a Link Expiry Time (LET), and the minium of all LETs for all links in a route gives the Route Expiry Time (RET). [44] suggest, how these expiry times can be predicted and [45] discusses route prediction in more detail. Just before a link or route expires (i.e. a critical time is reached), the destination sends a Flow-HANDOFF message, which triggers another Flow-REQUEST, thus finding a new route over which the current flow can be rerouted, without interrupting it.

A.13 FSLS - Fuzzy Sighted Link State Algorithms

This class of algorithms, as described in [46] also addresses the problem of the limited dissemination of link state information, similar to DREAM or FSR (cf. sections A.8 and A.14). LSUs (link state updates) are sent with a dynamically limited time-to-live, and in certain intervals, which depend on the number of hops, the updates can travel. Far reaching LSUs are sent much less frequent than short reaching LSUs. Also LSUs are only created if the state of a link has changed within the scope of the LSU. The length of the intervals and scope of the LSUs is the design parameter of the class of FSLS algorithms. An extreme case is the *discrete link state* algorithm DLS, in which each LSU is sent through the whole network (TTL is set to ∞). It differs from standard link state only in the fact, that the LSU is not sent immediately after a link status changes, but at the beginning of the next interval.

[46] also derives an optimal case for a FSLS algorithm, the *Hazy Sighted Link State Algorithm* (HSLS). [47] goes into more detail about both.

In [47], a more comprehensive overhead definition, which includes overhead due to non-optimal routes, is used to analyse the class of FSLS, derive the HSLS and prove its optimality.

However, for the analysis some assumptions are made that may not be the case in certain scenarios (e.g. the traffic a single node generates is independent of the network size). They are meant to apply for the average scenario, without regarding border effects.

More analytical studies about FSLS and overhead due to non-optimal routes can be found in [48] and [10].

A.14 FSR - Fisheye State Routing

Fisheye State Routing was proposed by Mario Gerla et.al. to the MANET IETF working group (see draft [49]).

Similar to DREAM [29] FSR wants to reduce unnecessary traffic by introducing a multi-level scope. By concept, FSR is a protocol that periodically updates link state information (table driven).

FSR is derived from Global State Routing (cf. section A.17). The major drawback of GSR is the large message size and the propagation latency of the link state changes. FSR now helps by introducing *scopes*, which depend on the number of hops a packet has reached from its source. Nodes within the smallest scope are considered most often in update packets; nodes, which are far away are considered much less frequent. This means, the message size can be greatly reduced, as information for most nodes can be omitted. Although routes may become inaccurate for distant destinations under increased mobility, packets will find more and more accurate routes while getting closer to the target, thus they don't suffer much from the inaccuracy. FSR is explained also in [50], while [51] reports about the implementation of FSR.

A.15 GEDIR - Geographic Distance Routing

This approach uses geographical information, like DREAM and LAR (cf. sections A.8 and A.20). While these protocols use directional routing (i.e. a message is sent to one or more neighbours *in the direction* of the target), GEDIR uses an approach based on progress (in terms of packet proximity to its destination) to select the set of neighbourgs to forward the message to. [52] describes a set of related geographic routing protocols (Directional Routing - DIR, Most Forward Within Radius - MFR and GEDIR is some variations) and their advantages and disadvantages.

The authors of [52] show by a counterexample, that loop-freedom for DREAM does not hold and they intend to show, that their methods are loop free, provided loops are not formed intentional. The situation described in the counterexample appears very artificial and may not appear in practice at all. Also no statement is made about the duration of loops. However, in a static network without any movement the formation of such loops is much more likely.

The result of [52] is, that most of them perform well under certain conditions. Their suggested algorithm (GEDIR) performs best among the discussed ones. Another interesting result is that multiple paths, as provided by many geographic routing protocols, do not improve the overall delivery ratio much.

Alas, [53], an earlier paper about GEDIR, is no longer available.

A.16 GPSR - Greedy Perimeter Stateless Routing

GPSR [54] is another location based routing protocol. A node learns about the position of its neighbours by a beacon or by information piggybacked on data packets (similar to other neighbour discovery methods).

The node forwards a message in a *greedy* way, i.e. to the neighbour which is geographically closest to the destination. If there is no such neighbour (which means, the node itself is currently closest node to the target within its transmission range) and the target is not in range, GPRS switches to *perimeter mode*, which guides the packet around this void area, using a planar-graph traversal with the right-hand rule (Chapter 3 in [54]). On entering perimeter mode, the current location is registered in the packet, such that greedy forwarding can be resumed, as soon as the void is traversed.

Of course GPSR requires that each node is aware of its own position, possibly by means of a GPS device. Also it is required that any source node knows the location of its destination. This information is registered once in the packet and never changed. Karp and Kung do not really address the problem of how to obtain the target location for a source node in [54]. There is only a reference to a location database service, which needs to be looked up. Hence GPSR depends on such a service like GLS[55], which may not be available in a Mobile Ad Hoc Network. However, GPSR was not designed just to work in a Mobile Ad Hoc Network, but also in rooftop and sensor networks. GPSR is certainly very interesting, but it has some severe prerequisites (e.g. a location service must be available), which may limit the applicability.

Performance of GPSR was compared against DSR using the NS-2 network simulator and showed better results than DSR for the environment used in the evaluation.

GPRS is described in detail in B. Karp's PhD-Thesis [56] and mentioned in a large variety of papers, as a reference. Few papers, like [57] and [58] go more into depth, but there is none, which does a detailed comparison with other protocols[13]. [60] and [25] show (as a corollary) that perimeter mode can be improved if a *Delaunay Graph* is used instead of a *Gabriel Graph.* [61] presents an improved routing algorithm based on GPRS.

A.17 GSR - Global State Routing

Global State Routing[62] was developed by Tsu-wei Chen and Mario Gerla. It is an early attempt to introduce link state routing in an Ad Hoc Networking context. The main problem of traditional link state routing, is the high amount of topology information, which is sent from each router to each other router. Since in Ad Hoc Networks, each node is also a router, this mechanism does not scale and needs to be limited. Therefore GSR adopts the information

[13] Except [59], but that focuses on energy conservation and sensor networks.

dissemination process from the Distributed Bellman-Ford algorithm[30]. In GSR topology information is exchanged periodically only among neighbours. If a topology change occurs, this change is transmitted further. Messages are sent only in such triggered cases. GSR uses sequence numbers based on timestamps, but no method of clock synchronisation of the nodes is suggested or even mentioned[14].

GSR was evaluated in simulation and compared to traditional link state and Distributed Bellman-Ford. There are no other comparisons or any further work with GSR, that we are aware of. Since the research group of Mario Gerla has developed a wide range of other routing strategies since the development of GSR, it can be assumed that further work on it has been abandoned in favour of more superior approaches, like FSR or HSR (cf. sections A.14 and A.18).

Although GSR is cited in some papers, the citations mention it only as an example.

A.18 HSR - Hierarchical State Routing

HSR[50] introduces a multilevel clustering infrastructure. Clustering is done on a physical and logical basis.

Physical clustering starts with level 0, the bottom layer. On each level, nodes can form clusters, which are represented by a clusterhead each. The clusterheads itself can form another cluster at the next higher level. On higher levels the clusterheads are connected via *virtual links*, which need to be mapped to physical links on the bottom layer. A virtual link will usually contain gateway nodes on the lowest level.

Each clusterhead collects link state information of each cluster member, regarding its neighbours, and propagates a summary to its fellow clusterheads on the higher levels, possibly using gateway nodes. On the higher levels the same happens with the link state information about the virtual links, which are computed from the lower level link states.

A special hierarchical addressing scheme is used, which is sufficient to route a message from any node to any other node[50]. A node passes a message up to the node of the highest level in its current hierarchy. This one will pass it to the destination cluster node (through a virtual link), which will pass it down the levels to the right node on the lowest level.

Additionally to physical clustering, a logical partitioning is used, which works similar to Mobile IP. The details can be looked up in [50].

HSR (as well as FSR, cf. section A.14) claims to be “QoS ready” [50],

[14] Synchronous clocks are probably silently assumed.

since QoS criteria can simply be taken into account into the link state and both protocols operate as link state protocols.

In the evaluation section in [50], a channel allocation scheme was used such that the cluster heads poll each cluster member and subsequently assign channels to them on demand. This can even be used with the WLAN standard IEEE 802.11 using the PCF instead of the DCF.

The evaluation compares HSR, FSR, DSDV and two on-demand routing schemes, which are not specified in detail. The results are not as explicit, as one could have expected. Still HSR shows its advantage as beeing the most scalable approach, FSR instead (which is also proposed in this papers) does not perform equally well.

The paper is cited by various other works, including [63], an analytical study about clustering overhead in general, [46] (cf. section A.13), [64] (cf. section A.19). In [65] an enhancement to HSR, now called EHSR is proposed, designed for the military.

A.19 LANMAR - Landmark Routing Protocol

LANMAR [66] is the result of combining FSR (cf. section A.14) with Landmark routing [67]. LANMAR combines both link state and distance vector characteristics. LANMAR utilizes the landmark routing with group mobility, i.e. groups of nodes, that are likely to move together build a subnet. In each subnet a landmark node is elected. Compared to FSR, only the information about nodes in scope (within a subnet/group) and those of the landmarks are transmitted in the link state updates. If a packet needs to get routed to a distant node, it is routed to its landmark node. As soon as it gets within scope of the destination node, it will get a more accurate route to the destination. It may not even be required to route the packet through the landmark.

The link state update process is very similar to FSR with the addition of a distance vector which is determined by the number of landmarks (logical subnets). Also there is now a fixed scope (all nodes within the scope are fixed and not determined by distance), and the update interval is now constant. Paths are kept to nodes within a subnet that travel out of scope of their landmark. The extra overhead to keep track of these drifters is shown to be relatively low. Another problem can be the existance of isolated nodes, which belong to no group, but could be their own landmark. Depending on the fraction of such isolated nodes, special handling may be required (e.g. reverting to traditional FSR).

LANMAR was evaluated per simulation and compared against plain FSR, as well as DSR and AODV. The results are in favour of LANMAR, especially in the cases with many nodes and high mobility. Also LANMAR clearly outperforms FSR in these simulations. It has to be made clear, that the scenarios given do explicitly use a group mobility model of which LANMAR can get a high benefit.

In [64] the LANMAR is extended by a landmark election process, which was not specified in [66].

[66] is cited in a few papers including [68], but no further analysis or evaluation was done (to my knowledge).

LANMAR was also submitted as an Internet Draft[69] to the MANET IETF working group.

A.20 LAR - Location Aided Routing

Location Aided Routing, as proposed by Ko and Vaidya [70], is an enhancement to flooding algorithms to reduce flooding overhead. Most on-demand schemes, including DSR and AODV (cf. sections A.10 and A.3) use flooding to obtain a route to the destination. This flooding results in significant overhead. LAR now aims to reduce the overhead to send the route requests only into a specific area, which is likely to contain the destination.

For this purpose the notions of *expected zone* and *request zone* is introduced. The *expected zone* covers the area, where the destination node is expected, according to the currently known information like:

- location at some time t (this will be the center of the expected zone)
- speed at time t
- direction at time t

Of course, this extrapolation of the state of the node at time t, does not need to be accurate at some later time t', but it provides a good start. Since the expected zone does not need to contain the source node, a larger area than the expected zone must be covered by the flooding, including all possible nodes on the way from the source to the expected zone. This expanded expected zone is called *request zone* and is used to restrict the flooding, i.e. only nodes that are part of the *request zone* forward a route request. On unsuccesful route discoveries, the request zone may need to be expanded further, possibly covering the whole network. Such subsequent route requests increase the initial latency for connections. This results in a tradeoff between reduced overhead and increased latency, and needs to be balanced carefully.

Depending on the scheme used, a sender needs to include a specification of its request zone in its route request such that nodes receiving the request, can determine, whether they are within the zone or not. A node replying with a route will include its coordinates along with the current time (and possibly other parameters like speed and direction) in the reply, so that the sender will have its coordinates (at that time) for future requests.

LAR was evaluated with MaRS [71] in a couple of scenarios, but the authors of [70] just compare two different modes of LAR, but no other routing protocols.

Suggested improvements include adaption of the request zone on the fly by the intermediate nodes of the route request. More flexible forms of request zones may be used and location information can be piggy-backed to any node, to keep location information more accurate within the network.

[70] is one of the most cited papers in reasearch area of Mobile Ad Hoc Networks, but it is commonly used only for reference by related work of other authors. [72] is a subsequent paper, that shortly emphasizes on a few optimizations of LAR.

A.21 LMR - Lightweight Mobile Routing

Lightweight Mobile Routing is a link reversal routing (LRR, cf. section A.22) algorithm, that was developed to overcome the non-convergence problem in partitioned networks with the previous methods as proposed by [73]. LMR was published in [74] and also described in [7].

The scope is generally a scenario, where changes happen too frequently for link state algorithms to adapt to, but not that frequently, that flooding is the only choice. LMR focuses on low complexity instead of optimal paths, such that it can even scale in very large networks.

Like many other protocols, LMR also uses three basic messages: QRY (query), RPY (reply) and FQ (failure query). They corespond to the messages used in AODV, DSR (cf. sections A.3 and A.10) and many others.

A QRY is sent by the source node by a limited broadcast (see Section B). The source then waits for a RPY packet, which will be issued by any node, which has a route to the destination and received a QRY or FQ packet. The *directed flood* caused by the RPY messages forms a directed acyclic graph (DAG), rooted in the originator of the RPY. The route itself and the up- and downstream links formed depend on the order of the RPY transmissions.

If a node loses its last route to the destination and it has upstream neighbours (cf. precursors in AODV), a FQ is broadcasted, to erase invalid routes. On reception of a FQ, the node may either transmit a RPY (if it still has another route) or another FQ if its last link was erased by the first FQ. So instead of a direct link reversal, LMR erases the links and sets them up new.

Loop freedom in a dynamic environment is ensured by marking previous unassigned links as "downstream-blocked" if the node has already an upstream link. These markers time out after a while, but it may happen that a downstream link cannot be used, because of possible loop formation. A similar mechanism is used to avoid deadlocks.

LMR is mentioned in a large number of papers, but only as a reference. LMR became less interesting with the development of TORA (cf. section A.28) as a successor.

A.22 LRR - Link Reversal Routing

As described in [7] LRR is a certain routing approach for highly dynamic networks. Its objective is to minimize the amount of overhead, when topology changes need to be announced. The maintained topology is reduced to a directed acyclic graph (DAG), rooted in the destination.

As the graph is directed, each link is either upstream or downstream to the destination. If a node in the graph becomes a local minimum, i.e. it has no downstream, one of its links is reversed. To achieve this, a notion of *height* is introduced, thus the problem is similar to flows in a graph. The height of the minimum node is raised such that it is higher than the lowest of its neighbours, thus reversing the direction of this link. The reversal can cause another node to become a minimum and the process continues.

The drawback is that no node knows about the "distance" (in any term) of itself to the destination, so optimizing metrics, as used in distance vector or link state algorithms, cannot be used.

LRR itself may be used in a proactive or reactive way.

The first and simple approach for LRR is *Gafni-Bertsekas*' Algorithm [73].

Further development lead to LMR and TORA (cf. sections A.21 and A.28).

A.23 OLSR - Optimized Link State Routing

OLSR is another proactive link state protocol, which is claimed to work best in large dense networks.

Each node selects a set of "Multipoint Relays" (MPRs) from its neighbours. The radio range of the MPR set should cover all 2-hop neighbours. Each node knows for which node it acts as a MPR. Thus OLSR requires bidirectional links. OLSR distributes routing packets via UDP. Each routing packet contains one or more OLSR messages. Messages exist for neighbour sensing, topology declaration and MPR information, interface, host- and network declaration.

OLSR explicitly requires avoidance of synchronous packet emissions among nodes in the neighbourhood, to reduce channel competition, which is probably a unique explicit requirement for an Ad Hoc routing protocol. For this purpose, jitter is used during transmission periods.

From the topology information, a shortest path for each destination is computed.

OLSR was first introduced as an IETF draft to the MANET working group in 1998. The draft has evolved since to [75]. The first draft was cited in several papers, but none of them goes into much detail. There were few performance comparisons: [76] which does an analytic comparison of OLSR with DSR in a random graph model, and [77] did a very detailed comparison of OLSR with AODV, which is mainly in favour of OLSR (but not in all cases).

OLSR has some similarities to TBRPF (cf. section A.26) [15].

A.24 SSA - Signal Stability-Based Adaptive Routing

SSA [78] presents a totally different approach from most other routing algorithms. The focus is to use signal and location stability as main routing criteria. The routing framework behind that works like most on-demand routing algorithms, i.e. route requests are broadcast through the network, route replies are returned by the destinations, routes are set up accordingly. The stability criteria interact with the standard procedure like this:

Each packet received is first passed to a module called DRP (dynamic routing protocol). DRP interacts with the device driver of the network inferface using an API, that allows to pass signal strength information. DRP maintains a *signal stability table* and categorizes each link to its neighbours as either beeing *strong* or *weak*. This table is updated with every packet received. Beacons (HELLO packets) are not processed further, but routing and data packets are passed up to the SRP (static routing protocol) module, which performs the usual routing tasks, like reacting to route request, forwarding packets according to the routing table, etc.

A route request can state, whether it wants any kind of links or just strong links. If only strong links are requested, any node receiving a route request over a weak link will drop it. Thus only route requests over strong links will reach the destination. The destination selects the first route request received

[15]This lead to some arguments between Philippe Jacquet (author of OLSR) and Richard Ogier (author of TBRPF) on the MANET mailing list.

by the same originator as the route and sends its reply vie the reversed hop list in the received request. The strategy suggested in [78] is first to try only strong links and fall back to any link, if no route could be found.

Also two enhancements are suggested: an additional link requirement, which just *prefers* strong links over weak ones (but does not rule them out). In this case route requests are all forwarded, but each intermediate node adds the link quality into the route request packet. Also the destination does not choose the first route request it receives, but waits a while to choose the best route in terms of strong links from all the route requests it has received so far for that source. The second improvement is a gratious route reply by intermediate nodes that already know a route to the destination (as in various other proposals, like DSR).

The simulations have been done without stating the simulation software used. Comparison was against a so called "simple routing protocol", which always choses the shortest path. It is unclear if this should be regarded as an optimal routing algorithm. The result shows some advantages (fewer route repairs need to be done), but also drawbacks (longer routes on average, since not all links can be used, and a short distance between hops is encouraged due to the stability criteria). Overall performance measures like routing overhead, throughput or packet latency have not been considered. So it is very unclear, if there is any benefit at all, or if the advantage of fewer repairs and reduced broadcast is consumed by the longer path-length or multiple route requests.

Signal Stability-Adaptive Routing seems to be related to the concept of ABR (cf. section A.1), with just some minor differences.

SSA was mentioned in some other papers, but only as an example for this specific routing approach. There were no detailed comparisons or analysis of SSA performed, so far.

A.25 STAR - Source Tree Adaptive Routing

STAR is proposed as an efficient link-state protocol by J.J. Garcia-Luna-Aceves[79]. Each node maintains a source-tree, which consists of its preferred links to each other destination. The source tree is computed on the information of its own links and the source trees reported by its neighbours. Changes in its own source tree are consequently reported to the neighbours. This can be done in an incremental way. The source tree and neighbour information forms the partial topology information in each node. Based on this information a route selection algorithm is run to obtain the route table with destination and next hop.

Information is updated with link state updates (LSU). An update message can consist of several LSUs, which reflect the changes in the nodes source-tree. Sequence numbers are used to distinguish current from outdated information. The link state information does not time out, thus removing the need for a periodic update.

STAR can operate in several ways. Suggested are two modes: Optimum Routing Approach (ORA) and the Least Overhead Routing Approach (LORA). In ORA shortest path routing is the goal, while in LORA path optimality is not as important as reduced overhead. However the *total overhead*, which includes overhead due to non-optimal paths, as described in [10], [48] and [47] (cf. section A.13) is not taken into account.

It is claimed in [79] that STAR is the first table-driven protocol, that can use the LORA approach. Other such protocols would need periodic updates, as well, to prevent routing loops. This can be avoided by STAR with the use of the routing trees, which can tell any router if a loop may be formed.

STAR requires a *neighbour protocol*, which ensures that new neighbours and leaving neighbours are detected in finite time. Further, STAR requires a link layer, capable of transmitting local broadcast messages without hidden terminal interference. This requirement is not entirely clear, but it seems related to the problems, that occurred with TORA over IMEP in several simulation studies (cf. section A.28).

However, STAR can still work without this prerequisite, but it is advised to include the whole source tree in each LSU. The broadcasts should then be done in an unreliable (but much more lightweight) way.

STAR was compared against a traditional link state algorithm based on topology broadcast like OSPF, a scheme called *Adaptive Link State Protocol* (ALP)[80] [16] and DSR (cf. section A.10).

All simulations are in favour of STAR but we note, that they have been performed by the authors of STAR.

STAR is described also in [81], but the article is very similar to the original in [79].

[82] describes ALP, STAR and NSR (cf. last paragraph in section A.10) in detail including comparisons.

Some further development of STAR lead to SOAR[83][17].

[16] ALP is explained in [81] as *Account, Login, Password*. This seems to be an error and can confuse the reader.

[17] SOAR is not described in this report.

A.26 TBRPF - Topology Broadcast Based on Reverse Path Forwarding

TBRPF is a proactive link state protocol, first presented in [84]. It is based on the *Extended Reverse Path Forwarding* Algorithm [85], but does overcome the reliability problems with ERPF.

TBRPF maintains a spanning tree in each node for each other node as the source. This tree is formed by each parent of the *source* node. A list of parents is kept at each node for every other node, as well as a full topology table, including cost and sequence number for each link, the node is aware of. The topology update messages are sent along these spanning trees but in the reverse direction. Of course these updates also will result in modifications of the current spanning tree.

[84] describes only the *full topology* mode of TBRPF. It also provides a proof of correctness (under certain constraints), some complexity analysis and a simulation based performance evaluation.

TBRPF was submitted as an IETF draft to the MANET working group, which has reached version 05 now[86]. Since the original draft, there have been significant changes: A *partial topology* mode was introduced, and in the most recent draft, this is also the default operation (*full topology* mode still exists as an option).

TBRPF supports only bidirectional links. The topology updates are transmitted reliable (they are acknowledged). A HELLO message is used for neighbour detection. The HELLO messages also come with a list of router IDs and a sequence number, such that each node can maintain its neighbour table. The update information is now differential, such that only changes in the router list are transmitted.

In the most recent draft, TBRPF is described as being composed of two main components: neighbour discovery and routing. For routing (as described above) each node computes its source tree, using a modified version of Dijkstra's algorithm. Only significant parts of the source tree are communicated to neighbours. TBRPF also has abandoned the use of sequence numbers, in favour of another technique based on "believing" (trusting) only certain nodes about their topology updates. The draft [86] is very detailed, including precise terminology description, protocol message formats and packet headers and even a detailed algorithm description in pseudocode.

In [84], TBRPF is roughly compared to other ERPF based protocols, but most of them were not designed for a wireless mobile network but for static networks, instead. A simulation based evaluation in this paper compared TBRPF against two slightly different flooding algorithms, but no comparisons against other protocols for Mobile Ad Hoc Networks have been done.

[84] is cited just twice and only for example purposes, and also the IETF draft did not seem to be part of any other scientific work on the subject. Apparently there have been no detailed comparisons with TBRPF against any other routing protocol for mobile ad hoc networks.

A.27 TLR/TRR/AGPF - Terminode Routing

Terminode routing is developed at the EPFL in Switzerland. The project aim is to develop a system that is capable of wide area Ad Hoc Routing. The project did explicitly choose an independent roadmap from the IETF MANET working group efforts. Support for IP or interoperability are not very important requirements. Still, Terminode Routing will work with IP in most cases, though.

Routing between terminodes is a hybrid process. First the packets are routed based on geographic position. The target address used in this routing is called LDA (location dependent address). From the target LDA the closest *friend*-node is computed and the packet is passed to it. A *friend* is a selected node in close, but not necessarily direct communication range. If the target node for the packet is among the friends of the node holding the packet, a local routing method is used to pass the packet to its destination.

As position based routing needs some kind of position service, terminodes use the concept of a *virtual home region* (VHR), which is a *some-for-some* approach [8]. For each node, there exists such a home region, which is specified by a fixed position and a radius. The region can be determined by a hash function over the node's id. Each node within the VHR of a certain node must maintain the current position of this node, so that other nodes can obtain it.

The position-based routing method is called AGPF (anchored path geodesic packet forwarding). As a simple greedy forwarding mechanism doesn't work in many situations (i.e.running into a local maximum), the concept of *anchors* are used. To avoid running into a maximum, the route is oriented on a set of anchors along the path. An anchor is just a specific location. The anchored path is determined by the source using FAPD (friend assisted path discovery) and included into the packet (similar to source routing).

FAPD is based on small world graphs[87]. Alternatively, the path can be determined by DRD (directed random discovery), which just sends the packet to a set of neighbours whose angle is the smallest to the right direction.

The local routing method is no longer based on position information, but only on a unique node identifier, the *target id*. A two hop neighbourhood information is maintained by each node by using HELLO packets. If the neighbourhood is known and a packet can utilize local routing (i.e. the target is known to the node which received the packet), a path discovery is initiated to direct the packet to the destination.

The concept of terminodes and terminode routing is described in several papers: [88, 5, 89, 90].

Terminode routing was compared against DSR in simulations, using scenarios which were designed for the usecase of terminode routing (i.e. large areas with large distances, some nodes clustered, with few roaming nodes). In these scenarios terminode routing outperformed DSR by nearly an order of magnitude [90].

The terminode project also addresses some other problems of ad hoc routing. [91] is a paper about positioning without a GPS-like device. The problem of stimulating cooperation of node operators is addressed in [92], which proposes a virtual currency, the *nuglet*: Relaying a message will benefit the relaying node with some units of cash, sending a message to a destination will require some units of cash as "payment". The paper also explains detailed precautions against undesired manipulations.

A.28 TORA - Temporally Ordered Routing Algorithm

TORA is a *link reversal routing* (LRR) algorithm (cf. section A.22) and was introduced by Park and Corson in [93]. It evolved from LMR and combines also features from Gafni-Bertzekas[73] in a unique single-pass strategie. In this context "single pass" means, that by processing a single event, all route maintenance tasks (errorneous route deletion, search and establish new routes) can be combined.

As in LRR algorithms in general, for each destination, a destination-rooted DAG is constructed. A height gets associated with each node and thus upstream and downstream links can be identified to route traffic to the destination.

The algorithm itself is rather complex, we refer to the cited literature [93, 7] for a detailed description.

TORA was used in some performance comparisons, notably [19], where it performed very bad. In [7] the authors state, that this is due to the nature of the underlying protocol (IMEP) used in the simulations, which prevents TORA from efficiently using the wireless broadcast channel. Other studies and an analytical comparison against an idealized link state algorithm (ILS) showed excellent performance. [7] also describes an extension which performs a proactive optimization, which may be of use in certain scenarios.

Although TORA can suffer from an unbounded worst-case convergence time, simulations have shown, that even for very stressful scenarios, TORA converges quickly and performs significantly better than the former mentioned ILS algorithm.

A.29 WAR - Witness Aided Routing

Witness Aided Routing by Aron and Gupta[94] is specifically designed to utilize unidirectional links.

WAR makes use of the possibility to overhear any transmission in range of a node on a wireless channel in a special way [18]. A node, which can overhear a transmission from one host to another over a relay, acts as a passive *witness* for that transmission. If the relay is not able to reach the destination or does not get an acknowledge, the witness node becomes an *active witness* and tries to deliver the packet on behalf of the relay node, thus saving the packet, even if the original route failed. Because many nodes can be witness of a certain transmission, special care is taken to avoid contention.

The goal is to perform just one single successful delivery. To achieve this, each witness host, which intends to deliver the packet, must get permission from the target host[19]. In order to get the permission, the node sends a request to the target host. If the target host did receive the packet before by the relay (but the witness hosts did not overhear this), the request will be rejected, in any other case, the set of witnesses will be polled by the target until the packet could be successfully delivered.

The route discovery is similar to DSR (cf. section A.10), with the enhancement of multiple route selection criteria. The target can be instructed to await a certain amount of route requests, or to wait for a certain time period, and then choose the route to answer the route discovery according to some specified criteria. Alternate routes can be remembered, to have them ready if the first choice breaks.

Again like DSR, WAR uses source routing to forward packets. Any forwarding node regards the delivery as successful, if it receives an acknowledgement from either the intended relay node or from any witness. If not, the route is considered broken and a route recovery process is initiated. Just like DSR the source route information in a relayed packet can be used to update local routing information.

Route recovery works by broadcasting the packet to all neighbours of the host, which failed to deliver it to the next hop, and setting a special flag. These hosts now try to deliver it, using the remaining source route information, treat it as a regular packet and clear this flag. However a packet

[18] Of course, other protocols, like DSR also make use of that fact.

[19] This is the target host of the witnessed transmission, not the final target of the packet.

can only be recovered a fixed number of times, which is set by the source. If all these attempts fail, the acknowledge for the packet will eventually time out at the source and the source will reinitiate a route discovery.

[94] gives a short analytical comparison against DSR and provides constraints under which WAR is more bandwidth efficient, than DSR. A much more detailed analytical study is presented in [38]. This study aims to prove the scalability of schemes like WAR and problems with on-demand routing protocols like DSR. [95] is the master thesis by Ionut Aron about the subject, which provides the same results in more detail.

A.30 WRP - Wireless Routing Protocol

The Wireless Routing Protocol by Shree Murthy and J.J. Garcia-Luna-Aceves is one of the first suggestions of a routing algorithm for Mobile Ad Hoc Networks. It was proposed 1996 in [96] and the only other protocol mentioned therein is DSDV (cf. section A.9).

WRP is related to the DBF[30] algorithm. Routing update messages are only sent locally to the neighbour set. They contain all the routing information the originating node knows of. Of course not the whole routing table is sent in each update. Only changes are transmitted, either by receiving an update from another node, or of a link in the neighbourhood has changed. WRP is a proactive routing protocol, since routes are maintained all the time and no special route requests by source nodes need to be performed.

The routing table consists of an entry for each destination with the next hop and a cost metric. The routes are selected by choosing the node from the neighbour set, which provides the path with the lowest cost (provided it's loopfree), as next hop. The link costs will be kept in a separate table, but it is not specified, how the cost for each link should be determined. Various possiblities exist: hop count, end-to-end delay, utilization, etc.

To keep the state of the neighbour links up to date, empty update messages (HELLO messages) are sent in a regular fashion, if no other updates would be sent anyway. Update messages which are not empty, need to be acknowledged.

[96] presents a proof of correctness and some simulation results, where WRP is compared against DBF, DUAL (the routing algorithm from EIGRP) and ILS (and idealized link state algorithm). The simulation was very simplified, a few simple static topologies have been simulated with randomly forced link failures to model movement related link breaks. The message overhead is counted for the regarded protocols. The results are clearly in favour of WRP.

WRP is referred to in a lot of papers, mainly due to the fact, that is one of the earliest proposals. The authors continued some work on WRP, which lead to WRP-lite in [97], which is later called BEST (Bandwidth Efficient Source Tracing) in [98][20].

A.31 ZRP - Zone Routing Protocol

The Zone Routing Protocol by Zygmund Haas was first introduced in [99].

It is a hybrid protocol, that combines reactive and proactive strategies. Since the advantages of either approach depend on the characteristics of the network (like the degree of mobility), it could be beneficial to combine them.

ZRP introduces the notion of a *routing zone*, which is a set of nodes within the local neighbourhood. In practice the zone is defined by the maximum number of hops, a node within the zone may be distant from the zone's center node. Each node maintains routing information actively within its zone. The scheme used is called *Intrazone Routing Protocol, IARP*. A basic link state algorithm is used for this purpose.

To discover a route outside the local routing zone, a reactive protocol, the *Interzone Routing Protocol, IERP* is used. For this purpose a *bordercast* of a request message is used. *Bordercast* means, the request is forwarded to the peripheral nodes of the zone, which in turn can check if the target is within their own zone, or continue to bordercast (cf. Section B). The bordercast process must take care, not to bordercast requests back into regions already covered. To achieve this, queries must be recorded for some time by the relaying nodes. ZRP uses a special technique for this, called *Advanced Query Detection* and *Early Termination*. Route caching and local repair is also possible.

Additionally to [99], ZRP is described more detailed in [100]. Some more investigations have been published in [101]. ZRP is also described in [102].

ZRP was also mentioned as a reference protocol that utilizes the hybrid approach. However it was not used in independent performance comparisons.

B Definitions

B.1 Terms

We will use several phrases and terms in the following sections, which will be defined for use in our context, first.

[20] Among others, BEST and DST (Distributed Source Tracing) are not further discussed in this paper.

Active Receiver: A node that is receiving data at the moment or is part of a session and likely to receive data in the near future. The term *active receiver* is only used in ADV (cd. Section A.2).

Anchor: A certain geographical position in the network's area. Anchors are used for geographical routing with *Terminodes* (cf. Section A.27).

Beacon, HELLO message: A (usually) periodic *local broadcast* message emitted from a node, destined for it's *neighbours* to announce itself in the neighbourhood. In some routing protocols, such a beacon may carry additional information.

Bordercast: A term from the Zone Routing Protocol ZRP (cf. Section A.31). A message is transmitted to one or more nodes on the border of the current routing zone, where it might be transmitted further.

Broadcast, local: A local broadcast is a broadcast message, that can be received from any node within reach of the sender. It is not intended to be retransmitted by the receivers.

Broadcast, limited: A limited broadcast can be retransmitted, but only to a subset of nodes in the network. Usually into a certain direction.

Broadcast, netwide: This broadcast ist retransmitted, until every node in the whole network has received the message.

Cluster: A group of nodes, that act together in some way. Usually a cluster is represented by a single node, the *Clusterhead*. Clustering is used in hierarchical routing.

Clusterhead: The representative node of a cluster. On a higher routing level, routing happens between the clusterheads. On the next higher level again clusters will be built but out of the clusterheads of the previous level.

Distance Effect: The *distance effect* is that two nodes appear to move slower with respect to each other if they are more distant. DREAM (and certain other protocols) make use of that fact. Routing or position information for distant node does not need to be as accurate as for short distance nodes.

Distance Vector Routing: Simple, table based routing. Each destination is entered into the routing table with the next hop and a distance metric. The topology of the network is unknown.

Expected Zone: The area in the network, where a certain node is expected to be. The expected zone is predicted from the last known movement characteristics of a node. The expected zone is used to derive a *request zone*, needed for location based protocols.

Flooding: A message is *flooded* through the whole network. This is another term for a *netwide broadcast.*

Friend Node: In *Terminode routing* a node, which is in close vicinity, but does not need to be in direct communication range. A friend helps determining an *anchored path* in AGPF (cf. section A.27).

Gateway: A node within a *Cluster*, often part of more than one clusters, which route messages from one cluster to another.

GPS: The *geographic position system*, a satellite based position service operated by the US military. It enables a GPS receiver to determine its position.

Group Mobility: Nodes can form groups and move together as a group. This is a likely event in realistic scenarios and needs to be modeled in the scenario model. Group mobility has significant impact on routing performance, depending on the protocol and it's ability to handle group mobility.

IEEE 802.11: IEEE standard family for wireless LAN communication. It defines the *distributed coordinate function* (DCF) or the *point coordinate function* (PCF) as channel allocation method. PCF can not be used easily in Ad Hoc Networks, since it would require a central instance (like an Access Point), but DCF is very common. DCF defines a RTS/CTS (request to send/clear to send) handshake to allocate a channel, thus circumventing the hidden terminal problem.

ILS: An *idealized link state* algorithm. Such an algorithm is mentioned in several papers as a reference for comparison, but is never specified in detail.

Link Layer Notification: A mechanism, that allows the routing module to be notified of local link breaks or new links (a node moves out of reach, or a node moved into reach) from the link layer.

Location Dependent Address: An address, which depends on the geographical location of a node. It must be determined by a *location* *service*. Location based routing protocols, like Terminode/AGPF (cf. Section A.27) make use of LDAs.

Location Service: See *Position Service*.

MANET: Abbreviation for *Mobile Ad Hoc Network*, also name of the corresponding IETF working group.

Multipoint Relay: A dedicated node, that relays traffic for other nodes in OLSR (cf. Section A.23).

Neighbour: Any node within direct communication range.

Node: A device, capable of communication over a wireless link and attached to some (in most cases mobile) unit, like a person or a car. Nodes are members of the network.

to overhear: A node can *overhear* messages not destined for it, by setting it's interface into *promiscious* mode. A node can benefit from routing information not explicitly for it. Additional routes may be learned and routes may be updated before the routes are needed and fewer route discovery processes may be needed. DSR makes use of this feature.

Parent Node: The node's current uplink in a route. See also *Precursor*.

Partial Topology: Several link state routing protocols do not maintain full topology information which would use far more ressources, but only *partial topology*, sufficient for efficient routing.

Position Service: A service, that can provide positional information for nodes in a mobile network. The position service needs to provide the position of any node to any node (GPS only provides the position for the node itself). Depending on how the position data is obtained, stored and distributed, there are *all-for-all*, *some-for-all* and *some-for-some* types of position services [8].

Precursor: The precursor node in the route to a destination. If a route becomes invalid, the precursors may need to be notified to update their routing entries. AODV (cf. Section A.3) explicitly uses a *precursor list*.

Proactive: Routing protocols are considered as *proactive* if they constantly maintain routing information for all routes, regardless, if in use or not. Maintenance can be event-driven (also named *table driven*) or in regular intervals.

Reactive: Routing protocols, which obtain and maintain only routes that are currently needed, are called *reactive* or *on demand.* They can cache learned routes, but if a route is unknown, a route discovery process needs to be initiated.

Request Zone: A geographical zone in the network, that covers all the nodes a *route request* should be sent to. The request zone is used in geographical routing algorithms, which results in a *limited broadcast.*

Rooftop Network: A static Ad Hoc network. Nodes are deployed on rooftops, but don't move, once deployed. They still need to organize themselves in an Ad Hoc fashion. We do not focus on rooftop networks.

Route Cache: A local cache in a node used by *reactive* routing protocols, to cache discovered routes. The routes will eventually time out from the cache, or be expunged if the route is detected to be invalid.

Routing Loop: The route forms a loop, such that packets are routed in the loop and possibly never reach their destination if the loop persists. Routing loops need to be avoided for successful routing.

Route Request: Important part of a route discovery process. A route request is usually a netwide broadcast message destined for the target node of the required route. If the destination receives the route request, it will answer with a route reply. The route request messages (also named *broadcast query*) are abbreviated as RREQ, REQ or QRY.

Route Reply: The answer to a *route request*, destined for the rource of the request. The route will be set up during the travel of the or can be carried directly in the *route reply.* This is a unicast message. It is abbreviated as RREP, RPY or REPLY.

Route Error: This messagetype indicates a broken, stale or otherwise unusable route. It is emitted from the node, which detected the broken route and can be unicast or broadcast. It is abbreviated RERR or ERROR.

Scenario: A scenario consists of a set of nodes (not necessarily of the same type or of a fixed number), that communicate and move according to the rules of the scenario. Scenarios may be very tight specified but usually are not. Scenarios are used in simulations to evaluate routing protocols. More work about scenarios was done in [103].

Scope: A term from FSR (cf. Section A.14). A scope is defined by the distance in number of hops from a node. Within each scope different update policies for routing information applies.

Sensor Network: An Ad Hoc Network of tiny sensoric nodes, that are deployed in the target area. The measured data is transmitted in an Ad Hoc fashion to some collecting node. Sensor networks are rather static, but have only limited transmission ranges and even more limited power capacity.

Sequence Numbers: DSDV (cf. Section A.9) introduced destination sequence numbers for routes in the routing table. The sequence numbers prevent old, stale routes from beeing entered into the routing table. Many other routing protocols have adapted this method.

Source Tree: A topology graph, representing the current routes from a source to any destination. *Source Trees* are used in several link state protocols, most notably STAR (cf. Section A.25).

Terminode: A term for the combination of a *terminal* and *node*, which is the common member of Ad Hoc Networks. It was created by the founders of the *Terminode Project* [104], a long term Ad Hoc WAN project at the EPFL in Switzerland.

Virtual Home Region: The *Position Service* suggested for *Terminode routing*.

Virtual Link: In hierarchical routing methods, links on higher layers, than the bottom (physical) layer are called *virtual links*. The need to be mapped to a set of physical links on the bottom layer for the actual communication.

Zone: Some area or set of nodes, that interact in a certain way. In ZRP, a *zone* is defined by the zone radius in terms of hops, in DDR a *zone* covers a tree in the forest routing topology.

B.2 Performance Metrics

Hop count: The number of hops, a packet has to travel to reach it's destination.

End-to-End delay: The time interval between sending a packet, and the reception of the packet at the destination.

Route setup delay: The delay a packet can not be sent from a node, until the route is set up.

Overhead: In general *overhead* is the amount of data transmitted, which is no payload data. There are many different types of overhead. Overhead is usually given as a ratio of total data and payload/useful data.

Routing Protocol Overhead: The data, which is sent to maintain or build routes.

Retransmission Overhead: The additional data transmitted, due to retransmission of lost or garbled packets.

Suboptimal Route Overhead: This type of overhead was introduced in some papers, to reflect the overhead due to subobtimal routes (i.e. routes with a longer hop count, than necessary).

Total Overhead: This type of overhead should include everything, including overhead due to suboptimal routes.

Utilization: The utilization of the available network capacity.

Delivery Ratio: The amount of packets actually delivered versus those being sent. The quality of a routing strategy can be well measured against the delivery ratio.

This is just a rough overview about performance metrics to be considered and used in numrous papers. More detailed guidelines can be found in [105].

References

[1] Charles E. Perkins, *Ad Hoc Networking*, Addison-Wesley, 2001.

[2] "The bluetooth special interest group," http://www.bluetooth.com/.

[3] Laura Marie Feeney, "A taxonomy for routing protocols in mobile ad hoc networks," Tech. Rep., Swedish Institute of Computer Science, Box 1263, SE-164 29 Kista, Sweden http://www.sics.se/ lmfeeney/, October 1999.

[4] C.-K. Toh Elizabeth M. Royer, "Review of current routing protocols for ad-hoc mobile wireless networks," Tech. Rep., UCSB, 1999.

[5] J. Hubaux, J. Le Boudec, M. Vojnovi'c, S. Giordano, M. Hamdi, L. Blazevi'c, and L. Butty'an, "Toward mobile ad-hoc wans: Terminodes," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC'2000), Chicago*. Ecole Polytechnique Federale de Lausanne, September 2000, p. 5, IEEE.

[6] Tony Larsson and Nicklas Hedman, "Routing protocols in wireless ad-hoc networks - a simulation study," M.S. thesis, Lulea University of Technology, 1998.

[7] M. Scott Corson and Vincent Park, *Link Reversal Routing*, chapter 8, pp. 255–298, In Perkins [1], 2001.

[8] Martin Mauve, Jörg Widmer, and Hannes Hartenstein, "A survey on position-based routing in mobile ad-hoc networks," *IEEE Network Magazine*, vol. 15, no. 6, pp. 30–39, Nov. 2001.

[9] Felix Ko, "Kapazitätssteigerung von multi-hop ad-hoc netzwerken durch power control," Diplomarbeit, Technische Universität München, Oct. 2001.

[10] Philippe Jacquet and Laurent Viennot, "Overhead in mobile ad-hoc network protocols," Tech. Rep. 3965, Institut National de Recherche en Informatique et en Automatique, Domaine de Voluceau, Rocquencourt, BP105, 78153, LE CHESNAY Cedex France, June 2000.

[11] Chai-Keong Toh, "Associativity-based routing for ad-hoc mobile networks," Tech. Rep., University of Cambridge Computer Laboratory, Cambridge CB2 3QG, United Kingdom, 1996.

[12] C-K. Toh Sung Ju Lee, Mario Gerla, "A simulation study of table-driven and on-demand routing protocols for mobile ad hoc networks.," *IEEE Network*, vol. 13, pp. 48–54, 1999.

[13] C-K. Toh and Vasos Vassiliou, *The Effects of Beaconing on the Battery Life of Ad Hoc Mobile Computers*, chapter 9, pp. 299–321, In Perkins [1], 2001.

[14] R.K. Padmanaban-Prof. Dr. V. Ramachandran S. Manikandan, R. Naveenan, "Optimized associativity-based threshold routing for mobile adhoc networks," Tech. Rep., College of Engineering, Guindy, Anna University, 2001.

[15] Rajendra V. Boppana and Satyadeva Konduru, "An adaptive distance vector routing algorithm for mobile ad hoc networks," in *Proceedings of the Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, 2001, vol. 3, pp. 1753–1762.

[16] Thomas D. Dyer and Rajendra V. Boppana, "A comparison of tcp performance over three routing protocols for mobile ad hoc networks," in *Proceedings of the ACM Symposium on Mobile Ad Hoc Networking & Computing (Mobihoc)*. Oct. 2001, ACM.

[17] Samir R. Das Charles E. Perkins, Elizabeth M. Belding-Royer, "Ad hoc on-demand distance vector (aodv) routing," http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-10.txt, Jan. 2002, Work in progress.

[18] Charles E. Perkins and Elizabeth M. Royer, *The Ad Hoc On-Demand Distance-Vector Protcol*, chapter 6, pp. 173–219, In Perkins [1], 2001.

[19] Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking, ACM, Dallas, TX*. ACM, October 1998.

[20] Per Johansson, Tony Larsson, Nicklas Hedman, Bartosz Mielczarek, and Mikael Degermark, "Scenario-based performance analysis of routing protocols for mobile ad-hoc networks," in *Proceedings of The Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*. ACM/IEEE, August 1999, pp. 195–206.

[21] Samir R. Das, Charles E. Perkins, and Elizabeth M. Royer, "Performance comparison of two on-demand routing protocols for ad hoc networks," in *Proceedings of INFOCOM 2000 Conference, Tel-Aviv, Israel*, March 2000.

[22] Mingliang Jiang, Jinyang Li, and Y.C. Tay, "Cluster based routing protocol(cbrp)," Internet Draft ftp://ftp.leo.org/pub/comp/doc/standards/internet-drafts/manet-cbrp-spec/draft-ietf-manet-cbrp-spec-01.txt.gz, August 1999, Work in progress, draft expired.

[23] Martha Steenstrup, *Cluster-Based Networks*, chapter 4, pp. 75–138, In Perkins [1], 2001.

[24] Ching-Chuan Chiang, Hsiao-Kuang Wu, Winston Liu, and Mario Gerla, "Routing in clustered multihop, mobile wireless networks with fading channel," Tech. Rep., University of California at Los Angeles Computer Science Department, 1997.

[25] John Hershberger Li Zahng An Zhu Jie Gao, Leonidas J. Guibas, "A geometric spanner for routing on mobile networks," in *Mobihoc 2001*, Sung Ju Lee, Ed. 2001, pp. 45–55, ACM Sigmobile.

[26] Raghupathy Sivakumar, Prasun Sinha, and Vaduvur Bharghavan, "Cedar: a core-extraction distributed ad hoc routing algorithm," in *IEEE Journal on selected areas in communication*, vol. 17. IEEE, August 1999.

[27] Vaduvur Bhafghavan Prasun Sinha, Raghupathy Sivakumar, "Enhancing ad hoc routing with dynamic virtual infrastructures," in *Proceedings of the 20. annual joint conference of the IEEE Computer and Communicatin Societies (INFOCOM 2001)*, Apr. 2001.

[28] Houda Labiod Navid Nikaein and Christian Bonnet, "Ddr-distributed dynamic routing algorithm for mobile ad hoc networks," Tech. Rep., Institut Eurecom, 2229 Route des Cretes, B.P.193 06904 Sophia Antipolis, France, 2000.

[29] Stefano Basagni Imrich Chlamtac Violet R. Syrotiuk and Barry A. Woodward, "A distance routing effect algorithm for mobility (dream)," in *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, Dallas, TX*. Oct. 1998, pp. 76–84, ACM.

[30] D. Bertsekas and R. Gallager, *Data Networks*, pp. 297–333, Prentice-Hall, Englewood Cliffs, N.J., 1987.

[31] Charles E. Perkins and Pravin Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers," in *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, 1994, pp. 234–244.

[32] Charles E. Perkins and Pravin Bhagwat, *DSDV Routing over a Multihop Wireless Network of Mobile Computers*, chapter 3, pp. 53–74, In Perkins [1], 2001.

[33] Samir R. Das, Robert Castañeda, Jiangtao Yan, and Ringli Sengupta, "Comparative performance evaluation of routing protocols for mobile,

ad hoc networks," in *Proceedings of 7th Int. Conf. on Computer Communications and Networks (IC3N), Lafayette, LA*. University of Texas at San Antonio, October 1998, pp. 153–161.

[34] David B. Johnson and David A. Maltz, "Dynamic source routing in ad hoc wireless networks," Tech. Rep., Computer Science Department Carnegie Mellon University, 5000 Forbes Avenue Pittsburgh, PA 15213-3891, 1996.

[35] Yih-Chun Hu Jorjeta G. Jetcheva David B. Johnson, David A. Maltz, "The dynamic source routing protocol for mobile ad hoc networks (dsr)," http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-07.txt, Feb. 2002, Work in Progress.

[36] David B. Johnson, David A. Maltz, and Josh Broch, *DSR The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks*, chapter 5, pp. 139–172, In Perkins [1], 2001.

[37] David A. Maltz, Josh Broch, and David B. Johnson, "Experiences designing and building a multi-hop wireless ad hoc network testbed," Tech. Rep., The CMU Monarch Project Computer Science Department Carnegie Mellon University, Pittsburgh, PA 15213, 1999, http://www.monarch.cmu.edu/.

[38] Ionut D. Aron and Sandeep K.S. Gupta, "On the scalability of on-demand routing protocols for mobile ad hoc networks: An analytical study," *Journal of Interconnection Networks*, Jan. 2001.

[39] David A. Maltz, "Resource management in multi-hop ad hoc networks," Tech. Rep., School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, Nov. 1999.

[40] Samir R. Das, Robert Castañeda, and Jiangtao Yan, "Simulation based performance evaluation of mobile, ad hoc network routing protocols," In *ACM/Baltzer Mobile Networks and Applications (MONET) Journal* [33], pp. 179–189.

[41] Mattias Östergren, "Tcp performance in ad hoc networks," Tech. Rep. T2000:14, Swedish Institute of Computer Science, Box 1263, S-164 29 KISTA, SWEDEN, November 2000.

[42] Marcelo Spohn and J.J. Garcia-Luna-Aceves, "Neighborhood aware source-routing," in *Proceedings of the 2001 ACM International Symposium on Mobile Ad Hoc Networking and Computing*, 2001.

[43] Chandra N. Sekharant-N.S.V.. Rao Steven G. Batsell S. Radhakrishnan, Gopal Racherlat, "Dst - a routing protocol for ad hoc networks using distributed spanning trees," *IEEE Wireless Communications and Networking Conference*, pp. 100–104, 1999.

[44] William Su and Mario Gerla, "Ipv6 flow handoff in ad hoc wireless networks using mobility prediction," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), Rio de Janeiro, Brazil*, Department of Computer Science University of California at Los Angeles, C, December 1999, IEEE, pp. 271–275, IEEE.

[45] Mario Gerla William Su, Sung-Ju Lee, "Mobility prediction and routing in ad hoc wireless networks," Tech. Rep., Wireless Adaptive Mobility Laboratory, Computer Science Departmen, University of California, Los Angeles, Los Angeles, CA 90095-1596, 2000.

[46] Ioannis Stavrakakis Cesar Santivanez, Ram Ramanathan, "Making link-state routing scale for ad hoc networks," in *Proceedings of the 2001 ACM International Symposium on Mobile Ad Hoc Networking and Computing*. Oct. 2001, pp. 22–32, ACM.

[47] Ram Ramanathan Cesar A. Santivanez, "Hazy sighted link state (hsls) routing: A scalable link state algorithm," Tech. Rep., Internetwork Research Department, BBN Technologies, 10 Moulton St., Cambridge, MA 02138, Aug. 2001.

[48] Cesar Santivanez, "Asymptotic behaviour of mobile ad hoc routing protocols with respect to traffic, mobility and size," Tech. Rep., Center for Communications and Digital Signal Processing, Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, 02115, Oct. 2000.

[49] Mario Gerla, Guangyu Pei, Xiaoyan Hong, and Tsu-Wei Chen, "Fisheye state routing protocol (fsr) for ad hoc networks," Internet Draft http://www.ietf.org/internet-drafts/draft-ietf-manet-fsr-00.txt, November 2000, Work in progress.

[50] Atsushi Iwata, Ching-Chuan Chiang, Guangyu Pei, Mario Gerla, and Tsu wei Chen, "Scalable routing strategies for ad hoc wireless networks," Tech. Rep., Department of Computer Science University of California, Los Angeles, 1999.

[51] Allen C. Sun, "The design and implementation of fisheye routing protocol for mobile ad hoc networks," M.S. thesis, Department of Electrical and Computer Science, MIT, May 2002.

[52] Xu Lin and Ivan Stojmenivoc, "Gps based distributed routing algorithms for wireless networks," Tech. Rep., Computer Science, SITE, University of Ottawa, Ottawa, Ontario K1N 6N5, Canada, 2000.

[53] X.Lin and L. Stojmenovic, "Geographic distance routing in ad hoc wireless networks," Tech. Rep. TR-98-10, University of Ottawa, SITE, December 1998.

[54] Brad Karp and H.T.Kung, "Gpsr: Greedy perimeter stateless routing for wireless networks," in *Processings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom 2000)*, Aug. 2000, pp. 243–254.

[55] Douglas S.J. DeCouto-David R. Karger Robert Morris Jinyang Li, John Jannotti, "A scalable location service for geographic ad hoc routing," in *Proceedings of the 6th Annual Internation ACM/IEEE Conference on Mobile Computing and Networking (MobiCom 2000)*, Aug. 2000.

[56] Brad Nelson Karp, *Geographic Routing for Wireless Networks*, Dissertation, Harvard University, Cambridge, Massachusetts, Oct. 2000.

[57] Linda Briesemeister, *Group Membership and Communication in Highly Mobile Ad Hoc Networks*, Dissertation, Technische Universtiaet Berlin, Nov. 2001.

[58] Rene Meier-Laurent Mazare Vinny Cahill Marc-Olivier Killijian, Raymond Cunningham, "Towards group communication for mobile participants," Tech. Rep., Trinity College of Dublin, Dublin 2, Eire, 2001.

[59] Deborah Estrin Yan Yu, Ramesh Govindan, "Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks," Tech. Rep. UCLA/CSD-TR-01-0023, UCLA Computer Science Department, May 2001.

[60] Peng-Jun Wan Xiang-Yang Li, Gruia Calinescu, "Distributed construction of planar spanner and routing for ad hoc wireless networks," Tech. Rep., Illinois Institute of Technology, Chicago, Chigago, IL, 60616, 2001.

[61] Lata Narayanan-Jaroslav Opatrny Lali Barriere, Pierre Fraigniaud, "Robust position-based routing in wireless ad hoc networks with unstable transmission ranges," in *Proceedings of 5th ACM International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM '01)*, Rome, Italy, July 2001.

[62] Mario Gerla Tsu-Wei Chen, "Global state routing: A new routing scheme for ad-hoc wireless networks," Tech. Rep., Computer Science Department, University of California, Los Angeles, 1998.

[63] John Sucec and Ivan Marsic, "Clustering overhead for hierarchical routing in mobile ad hoc networks," Tech. Rep., Rutgers University, 2002.

[64] Xiaoyan Hong Mario Gerla and Guangyu Pei, "Landmark routing for large ad hoc wireless networks," in *Proceedings of IEEE GLOBECOM 2000*, San Francisco, CA, Nov. 2000.

[65] Henry Ly Mario Gerla Daniel Lihui Gu, Guangyu Pei and Xiaoyan Hong, "Hierarchical routing for multi-layer ad-hoc wireless networks with uavs," in *Proceedings of IEEE Milcom 2000*, 2000.

[66] Guangyu Pei, Mario Gerla, and Xiaoyan Hong, "Lanmar: Landmark routing for large scale wireless ad hoc networks with group mobility," in *roceedings of the ACM/IEEE Workshop on Mobile Ad Hoc Networking and Computing (MOBIHOC), Boston, MA*. ACM/IEEE, August 2000, pp. 11–18.

[67] P.F. Tsuchiya, "The landmark hierarchy: a new hierarchy for routing in very large networks," *Computer Communication Review*, vol. 18, no. 4, pp. 35–42, Aug. 1988.

[68] Sung-Ju Lee, *Routing and Multicasting Strategies in Wireless Mobile Ad hoc Networks*, Ph.D. thesis, University of California at Los Angeles, 2000.

[69] Li Ma Mario Gerla, Xiaoyun Hong, "Landmark routing protocol (lanmar) for large scale ad hoc networks," Internet-Draft http://www.ietf.org/internet-drafts/draft-ietf-manet-lanmar-04.txt, June 2002, Work in progress.

[70] Young-Bae Ko and Nitin H. Vaidya, "Location-aided routing (lar) in mobile ad hoc networks," Tech. Rep., Department of Computer Science Texas A&M University, College Station, TX 77843-3112, June 1998.

[71] University of Maryland, "Mars maryland routing simulator," http://www.cs.umd.edu/projects/netcalliper/software.html.

[72] Young-Bae Ko and Nitin H. Vaidya, "Optimizations for location-aided routing (lar) in mobile ad hoc networks (a brief note)," Tech. Rep. 98-023, Department of Computer Science, Texas AM University, College Station, TX 77843-3112, Nov. 1998.

[73] E. Gafni and D. Bertsekas, "Distributed algorithms for generating loop-free routes in networks with frequently changing topology," *IEEE Transactions on Communications*, vol. 29, no. 1, pp. 11–15, Jan. 1981.

[74] M. Scott Corson and Anthony Ephremides, "A distributed routing algorithm for mobile wireless networks," *ACM/Balzer Wireless Networks*, vol. 1, no. 1, pp. 61–81, Feb. 1995.

[75] Philippe Jacquet, Paul Muhlethaler, Amir Qayyum, Anis Laouiti, Laurent Viennot, and Thomas Clausen, "Optimized link state routing protocol," Internet-Draft http://www.ietf.org/internet-drafts/draft-ietf-manet-olsr-04.txt, March 2001, Work in Progress.

[76] Philippe Jacquet and Anis Laouiti, "Analysis of mobile ad-hoc network routing protocols in random graph models," Tech. Rep. 3835, INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78 153 Le Chesany Cedex, France, 1999.

[77] Lars Christensen and Gitte Hansen, "The optimized link state routing protocol," M.S. thesis, Department of Computer Science, Aalborg University, Fredrik Bajers Vej 7, 9220 Aalborg Ost, 2001.

[78] Kuang-Yeh Wang Satish K. Tripathi Rohit Dube, Cynthia D. Rais, "Signal stability based adaptive routing (ssa) for ad-hoc mobile networks," Tech. Rep. CS-TR-3646, UNMIACS-TR-96-34, Institute for Advanced Computer Studies, Mobile Computing and Multimedia Laboratory, Department of Computer Science, University of Maryland, College Park, MD 20742, Aug. 1996.

[79] J.J. Garcia-Luna-Aceves and Marcelo Spohn, "Source-tree routing in wireless networks," in *IEEE International Conference on Network Protocols (ICNP '99)*, Oct. 1999.

[80] J.J. Garcia-Luna-Aceves and Marcello Spohn, "Scalable link-state internet routing," in *IEEE International Conference on Network Protocols*, Oct. 1998.

[81] J.J. Garcia-Luna-Aceves and Marcelo Spohn, *Bandwidth-Efficient Link-State Routing in Wireless Networks*, chapter 10, pp. 323–350, In Perkins [1], 2001.

[82] Marcelo Spohn, *Routing in the Internet using Partial Link State Information*, Dissertation, University of California Santa Cruz, Sept. 2001.

[83] Souma Roy and J.J. Garcia-Luna-Aceves, “Using minimal source trees for on-demand routing in ad hoc networks,” in *Proceedings if INFOCOM*, 2001, pp. 1171–1181.

[84] Bhargav Bellur and Richard G. Ogier, “A reliable, efficient topology broadcast protocol for dynamic networks,” in *Proceedings of INFOCOM 1999*, 1999.

[85] Y. Dalal and R.Metclafe, “Reverse path forwarding of broadcast packets,” *Communications of the ACM*, vol. 21, no. 12, pp. 1040–1048, Dec. 1978.

[86] Bhargav Bellur, Richard G. Ogier, and Fred L. Templin, “Topology broadcast based on reverse-path forwarding (tbrpf),” Internet-Draft http://www.ietf.org/internet-drafts/draft-ietf-manet-tbrpf-05.txt, March 2002, Work in progress.

[87] D.J.Watts, *Small Worlds: The dynamics of networks between order and randomness*, Princeton University Press, 1999.

[88] J. P. Hubaux, S. Giordano, M. Hamdi, and J. Y. Le Boudec, “The terminodes project: Towards mobile ad-hoc wans,” in *Proceedings of MOMUC’99, San Diego*. EPFL-ICA, November 1999.

[89] Ljubica Blazevic, Slivia Giordano, and Jean-Yves Le Boudec, “Self-organised wide-area routing,” Tech. Rep., EPFL - Communication Systems Department (DSC), Ecole Polytechnique Federale de Lausanne, CH-1015 Lausanne, Switzerland, 2000.

[90] Ljubica Blazevic, Slivia Giordano, and Jean-Yves Le Boudec, “Self organized terminode routing,” Tech. Rep., EPFL - Communication Systems Department (DSC), Ecole Polytechnique Federale de Lausanne, CH-1015 Lausanne, Switzerland, December 2000.

[91] Jean-Pierre Hubaux Srdan Capkun, Maher Hamdi, “Gps-free positioning in mobile ad-hoc networks,” in *Proceedings of INFOCOM 2001*, 2001.

[92] Levente Buttyán and Jean-Pierre Hubaux, "Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks," Tech. Rep., Institute for Computer Communications and Applications Department of Communication Systems Swiss Federal Institute of Technology, Lausanne, EPFL-DSC-ICA, CH-1015, Lausanne, Switzerland, January 2001.

[93] Vincent D. Park and M.Scott Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *Proceedings of INFOCOM 1997*, 1997.

[94] Ionut D. Aron and Sandeep K.S. Gupta, "A witness-aided routing protocol for mobile ad-hoc networks with unidirectional links," Tech. Rep., Department of Computer Science, Colorado State University, Fort Collins, CO, USA, 1999.

[95] Ionut Dragos Aron, "The witness aided routing protocol - a scalable, fault-tolerant solution for mobile ad hoc networks," M.S. thesis, Department of Computer Science, Colorado State University, Fort Collins, Colorado, USA, 2000.

[96] Shree Murthy and J.J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," Tech. Rep., Copmuter Engineering, University of California at Santa Cruz, Santa Cruz, CA 95064, 1996.

[97] J.J. Garcia-Luna-Aceves Jyoti Raju, "A comparison of on-demand and table driven routing for ad-hoc wireless networks," in *Proceedings of ICC 2000*, 2000.

[98] J.J. Garcia-Luna-Aceves Jyoti Raju, "Scenario-based comparison of source-tracing and dynamic source routing protocols for ad hoc networks," *ACM Computer Communication Review - Special Issue in mobile extensions to the Internet*, Oct. 2001.

[99] Zygmund J. Haas, "A new routing protocol for the reconfigurable wireless networks," Tech. Rep., School of Electrical Engineering, Cornell University, Ithaca, NY, 14853, 1997.

[100] Zygmund J. Haas and Marc R. Pearlman, "The performance of query control schemes for the zone routing protocol," *IEEE/ACM Transactions on Networking*, vol. 9, no. 4, pp. 427–438, Aug. 2001.

[101] Zygmund J. Haas and Marc R. Pearlman, "Determining the optimal configuration for the zone routing protocol," *IEEE Journal on Selected Areas Communications*, vol. 17, no. 8, Aug. 1999.

[102] Zygmunt J. Haas and Marc R. Pearlman, *ZRP A Hybrid Framework for Routing in Ad Hoc Networks*, chapter 7, pp. 221–253, In Perkins [1], 2001.

[103] Daniel Lang, "On scenarios to evaluate mobile ad hoc networking routing protocols," Tech. Rep. TUM-I0702, Technische Universität München, July 2002.

[104] "Terminodes / mobile information and communication systems," http://www.terminodes.org/.

[105] Scott Corson, "Routing protocol performance issues and evaluation considerations," ftp://ftp.isi.edu/in-notes/rfc2501.txt, 1999.